EAI Endorsed Transactions
on AI and Robotics
EAI.EU Research Article

A Novel Method for Enhancing Warehouse Operations Using Heterogeneous Robotic Systems for Autonomous Pick-and-Deliver Tasks

Youssef MSALA[1,*], Oussama Hamed[3], Mohamed Talea[2], Mohamed Aboulfatah[1]

[1]MISI Laboratory, Faculty of Sciences and Techniques, University Hassan I, Settat, Morocco
[2]TI Laboratory, Faculty of Sciences Ben M'Sik, University Hassan II, Casablanca, Morocco
[3]Aix-Marseille University, LIS UMR CNRS 7020, Marseille, France

Abstract

The rapid rise of warehouse automation has increased the need for reliable multi-robot coordination. Efficient task allocation and path planning are central challenges that affect picking speed, energy use, and system scalability. This paper proposes an integrated framework for warehouse-oriented multi-robot task allocation and route planning. The method combines the Hungarian algorithm for cost-minimized task distribution with an open-loop Traveling Salesman Problem (TSP) for path sequencing. Unlike approaches that apply these steps separately, our framework links them in a single design and adds two practical extensions: explicit handling of heterogeneous robot capacities and a reassignment phase that recovers tasks left unallocated after the first assignment. These additions improve coverage and efficiency while keeping computation lightweight. Simulations in MATLAB show good scaling with larger fleets and reductions in both travel distance and execution time. The proposed framework provides a heterogeneity-aware allocation mechanism, robust unassigned-task handling, and integrated path optimization, and can be extended to dynamic order insertion and obstacle-aware navigation in warehouse settings.

Received on 08 August 2025; accepted on 17 September 2025; published on 23 September 2025

Keywords: Smart Warehousing, Heterogeneous Robotic Systems, Multi-Robot Coordination, Task Allocation, Pick-and-Deliver Tasks, Industrial Robotics, Logistics Optimization, Navigation and Path Planning

doi:10.4108/airo.9913

1. Introduction

The coordination and optimization of multi-robot systems (MRS) have become a focal point of research in the fields of industrial automation, logistics, and intelligent transportation [1–4]. The increasing demand for autonomous and collaborative robotic systems stems from the global shift towards Industry 4.0, where efficiency, flexibility, and responsiveness are essential to maintain competitiveness in dynamic and uncertain environments [5, 6]. Multi-robot systems offer substantial advantages over single-agent systems, including enhanced task parallelism, redundancy, and the ability to handle complex missions that exceed the capabilities of individual robots [7].

This technological evolution has been accelerated by global challenges such as labor shortages, rising operational costs, and the need for resilience in supply chain systems, particularly in the aftermath of disruptions like the COVID-19 pandemic. Automated warehouses, smart manufacturing systems, and autonomous delivery fleets are increasingly relying on intelligent robotic agents capable of performing tasks such as zone servicing, material transportation, and surveillance. However, the deployment of such systems in real-world scenarios introduces intricate challenges in the allocation of tasks and the planning of efficient and conflict-free trajectories, especially when robots operate under capacity constraints and within bounded environments characterized by spatial dispersion [8, 9].

*Corresponding author. Email: youssef.msala@gmail.com

Despite the remarkable progress in multi-robot coordination, the simultaneous resolution of task allocation and route optimization in capacity-constrained environments remains an open research problem [10]. Conventional approaches to task assignment, including centralized optimization methods and greedy heuristics, often fail to scale effectively as the system size grows, leading to computational bottlenecks or suboptimal task distributions [11–14]. These limitations become particularly evident in logistical and service environments, where the efficient allocation of service zones to multiple robots must account not only for execution cost but also for the spatial layout of the environment and the capacity limitations of each robot [15–17].

Recent advancements in multi-robot systems have highlighted a persistent challenge: task assignment and path planning are often addressed independently, leading to inefficiencies in system-wide performance. Traditional methods rely on the Hungarian algorithm for task assignment and the Traveling Salesman Problem (TSP) for minimizing route distances, but their decoupled nature fails to account for interdependencies such as robot collision avoidance and spatial congestion [18, 19]. To overcome these limitations, integrated frameworks that combine task allocation with real-time path optimization have been proposed. For example, evolutionary and reinforcement learning-based methods have demonstrated improvements in reducing collision risks and enhancing allocation efficiency by synchronizing decision-making across both domains [5, 20]. However, many of these integrated models are still more tailored toward dynamic environments like aerial swarms or exploration tasks, rather than static multi-task service allocation required in industrial and warehouse contexts.

In addition to classical heuristics, a rich body of research has explored integrated and learning-based task allocation strategies. Early works combined Hungarian assignment with TSP routing as a baseline for multi-robot coordination [18, 19], but such approaches generally neglected heterogeneity and dynamic task arrivals. Distributed methods, including market-based allocation [8] and auction-based optimization [11], have improved scalability yet remain limited in environments with capacity-constrained agents.

More recent studies have focused on evolutionary and reinforcement learning (RL) techniques. Evolutionary algorithms such as genetic algorithms, PSO, and NSGA-III have been applied to optimize multi-robot task allocation under multi-objective criteria [15, 17]. Deep reinforcement learning methods [5] and hybrid evolutionary–RL approaches [20] have demonstrated strong adaptability to dynamic environments, enabling collision avoidance and congestion-aware routing. However, these methods often require extensive training, large datasets, or high computational resources, which may hinder deployment in structured warehouse settings.

By contrast, the framework proposed in this work combines the efficiency of classical optimization with novel extensions that explicitly handle heterogeneity, capacity constraints, and unassigned task recovery. Unlike RL-based or fully distributed schemes, our approach maintains computational tractability while offering practical scalability to large fleets, making it well suited for warehouse-specific applications.

Additionally, many recent studies highlight the challenges posed by heterogeneity in multi-robot systems, where robots possess varying task-handling capacities, energy limitations, and functional capabilities. The common assumption of homogeneous agents simplifies modeling but undermines the feasibility of solutions in practical deployments. For example, heterogeneous robots with time-dependent and task-dependent capabilities require sophisticated allocation mechanisms to ensure coordination and efficiency [21]. Furthermore, battery limitations, cooperative tasks, and inter-agent synchronization compound the complexity of real-world task allocation, as addressed by constraint-based and MILP-based frameworks [22, 23]. The lack of scalable, integrated frameworks that incorporate heterogeneous capabilities, energy/resource constraints, and task interdependencies continues to hinder practical deployments of multi-robot systems in complex industrial settings.

Addressing these challenges requires the development of an integrated solution that ensures the balanced assignment of tasks, cost minimization, and trajectory optimization while considering both the physical environment and the operational limitations of the robotic agents.

MRS are increasingly used in logistics, industrial automation, and service robotics, where demand for intelligent, autonomous, and collaborative agents is growing. In such environments, dynamic task allocation and efficient path planning are no longer mere technical pursuits but critical components of operational success, with direct implications for cost efficiency, timely service delivery, and system resilience. Recent studies have demonstrated how intelligent allocation strategies and real-time coordination frameworks significantly improve the performance of robotic fleets in logistics and manufacturing contexts [24]; [25]. For instance, deep reinforcement learning and hybrid optimization models have been used to bridge task allocation with routing, improving efficiency and enabling adaptation to real-time operational demands [26]. These advancements reinforce the importance of MRS as a cornerstone for modern autonomous operations.

Despite existing research efforts, the practical deployment of multi-robot systems remains hindered by inefficiencies in resource utilization, particularly when task allocation and route planning are handled separately rather than as a unified problem. In real-world scenarios, MRS must navigate diverse service demands, spatial constraints, and operational limitations that traditional models, often based on idealized or homogeneous conditions, struggle to accommodate. For instance, systems that fail to integrate energy-aware decision-making or dynamic environmental feedback tend to suffer from increased execution time and underutilization of robotic resources [27–29]. The disconnect between theoretical models and operational realities is further evident in congested or constrained environments, where proactive routing and congestion-aware strategies are crucial to avoiding delays and inefficiencies [30].

Furthermore, current methods frequently overlook the importance of adaptive decision-making in dynamic and resource-constrained environments. Practical applications require not only the assignment of tasks but also the continual optimization of task sequences and movement trajectories in response to changing demands or workspace configurations. Without such adaptability, MRS risk becoming rigid, inefficient, and unable to scale in response to fluctuating workloads.

In light of these challenges, there is a clear need for novel approaches that bridge the gap between computational efficiency and real-world feasibility, providing robust, scalable, and cost-effective solutions for multi-robot task coordination and routing. The motivation for this study is rooted in addressing this unmet need by proposing a method that simultaneously optimizes task assignment, execution cost, and travel distance, while maintaining practical applicability through simulation-based validation.

To overcome the limitations inherent in existing MRS coordination strategies, this study introduces a comprehensive and integrated framework that simultaneously addresses the challenges of task allocation and route optimization within environments characterized by capacity constraints and spatial dispersion. The proposed methodology is specifically designed to enhance the overall efficiency, scalability, and responsiveness of multi-robot systems by ensuring not only the optimal distribution of service tasks among available robots but also the generation of cost-efficient trajectories that minimize travel distance and idle capacities, all while respecting the operational constraints associated with each robotic agent.

The originality of this work resides in the seamless integration of task assignment and path planning, two phases traditionally treated as separate optimization problems in the literature. The task allocation process is formulated as a cost minimization problem and solved using the Hungarian algorithm, which guarantees a globally optimal assignment of tasks by constructing a cost matrix that captures both execution time and Euclidean distance between robots and service zones. Once the task distribution is determined, the second phase involves optimizing the execution sequence for each robot through the resolution of an open-loop TSP, thereby reducing unnecessary travel distance and improving task execution efficiency.

The main contributions of this study are threefold. First, it presents an integrated approach that unifies task assignment and route optimization into a single decision-making framework, ensuring that both phases contribute jointly to system-wide efficiency rather than being optimized in isolation. Second, the method introduces a capacity-aware task allocation mechanism that accommodates heterogeneous robot capabilities and varying task demands, an aspect often neglected in prior works. Third, the framework is designed to be computationally efficient and practically scalable, making it suitable for real-world applications such as logistics, autonomous material handling, and service robotics, where multiple simultaneous service requests and operational constraints must be addressed in real-time.

The effectiveness of the proposed solution is validated through extensive simulation-based experiments, demonstrating its capacity to reduce total execution costs, improve task coverage, and enhance overall fleet utilization in comparison to conventional baseline methods. By addressing both theoretical and practical challenges, this research offers a novel and scalable solution that advances the state of the art in multi-robot task allocation and trajectory planning for complex and dynamic environments.

The main objective of this study is to develop an integrated framework for multi-robot task allocation and route optimization that minimizes execution costs while respecting robot capacity constraints in spatially distributed environments. The study aims to simultaneously optimize task assignment using the Hungarian algorithm and trajectory planning through an open-loop TSP formulation. A further objective is to validate the proposed method through simulations, demonstrating its effectiveness in enhancing fleet utilization, reducing travel distance, and improving task coverage compared to conventional approaches.

The remainder of this paper is organized as follows: Section 2 details the proposed methodology, including the problem formulation, task assignment strategy, and trajectory planning approach. Section 3 illustrates and discusses the simulation results that validate the effectiveness of the proposed method. Finally, Section 4 concludes the paper and outlines potential directions for future research.

2. Problem Statement

This study addresses the optimal task assignment problem in a multi-robot system composed of $N \in \mathbb{N}^+$ heterogeneous mobile robots deployed in a bounded two-dimensional Euclidean space $\Omega \subset \mathbb{R}^2$. The environment contains $M \in \mathbb{N}^+$ discrete service zones, each associated with a specific service time and fixed spatial coordinates. The fundamental objective is to minimize the total operational cost incurred by assigning robots to service zones, considering both execution time and travel distance, under robot capability constraints and zone exclusivity conditions.

Let the set of robots be denoted by $R = \{r_1, r_2, \ldots, r_N\}$ and the set of zones by $Z = \{z_1, z_2, \ldots, z_M\}$. Each robot $r_i \in R$ is initially located at a position $p_{r_i} = (x_{r_i}, y_{r_i}) \in \Omega$, and each zone $z_j \in Z$ is positioned at $p_{z_j} = (x_{z_j}, y_{z_j}) \in \Omega$. Robots are subject to a maximum task capacity constraint defined by $\kappa_i \in \mathbb{N}^+$, which specifies the maximum number of zones that can be assigned to robot r_i.

Each robot r_i is equipped with the capability to service a subset of zones, denoted by $Z_i \subseteq Z$. For a feasible robot-zone pair (r_i, z_j) such that $z_j \in Z_i$, a deterministic service time $\tau_{ij} \in \mathbb{R}^+$ is required, and the spatial travel cost is evaluated as the Euclidean distance:

$$d_{ij} = \|p_{r_i} - p_{z_j}\|_2 = \sqrt{(x_{r_i} - x_{z_j})^2 + (y_{r_i} - y_{z_j})^2} \tag{1}$$

The total cost $c_{ij} \in \mathbb{R}^+ \cup \{\gamma\}$ associated with assigning robot r_i to zone z_j is defined as:

$$c_{ij} = \begin{cases} \tau_{ij} + d_{ij}, & \text{if } z_j \in Z_i \\ \gamma, & \text{otherwise} \end{cases} \tag{2}$$

where $\gamma \in \mathbb{R}^+$, $\gamma \gg \max_{i,j}(\tau_{ij} + d_{ij})$, penalizes infeasible assignments.

Define a binary decision variable $x_{ij} \in \{0, 1\}$ for all $i \in \{1, \ldots, N\}$ and $j \in \{1, \ldots, M\}$ such that:

$$x_{ij} = \begin{cases} 1, & \text{if robot } r_i \text{ is assigned to zone } z_j \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

The task assignment problem can now be formulated as the following constrained binary optimization problem:

$$\min_{x_{ij} \in \{0,1\}} J(X) = \sum_{i=1}^{N} \sum_{j=1}^{M} x_{ij} \cdot c_{ij} \tag{4}$$

subject to the assignment constraints:

$$\sum_{j=1}^{M} x_{ij} \leq \kappa_i, \qquad \forall i \in \{1, \ldots, N\} \tag{C1}$$

$$\sum_{i=1}^{N} x_{ij} \leq 1, \qquad \forall j \in \{1, \ldots, M\} \tag{C2}$$

$$x_{ij} \in \{0, 1\}, \qquad \forall i, j \tag{C3}$$

Constraint (C1) ensures that the number of tasks assigned to each robot does not exceed its service capacity, while constraint (C2) enforces exclusivity, i.e., each zone is serviced by at most one robot. Constraint (C3) maintains binary integrity of the assignment variables.

Robot heterogeneity is explicitly modeled in three complementary ways: (i) service capacity κ_i, which limits the maximum number of tasks each robot can perform; (ii) eligibility constraints E, which restrict task assignments to robots capable of servicing specific zones; and (iii) task-dependent service times τ_{ij}, which vary across robot–task pairs. Together, these factors capture differences in payload capacity, functional specialization, and execution efficiency, reflecting the diversity of real-world robotic fleets.

This problem formulation encapsulates the core challenge of efficiently distributing heterogeneous tasks among resource-constrained agents while minimizing total execution and travel cost, and it serves as the foundation for the subsequent resolution strategies.

3. Methodology

The proposed methodology for the multi-robot task allocation and route optimization problem is structured in four key stages: (1) Environment Initialization, (2) Cost Function Definition, (3) Assignment Optimization, and (4) Path Planning via Open-loop TSP.

Each phase is formulated with rigorous mathematical precision and translated into systematic pseudocode for reproducibility.

3.1. Environment Initialization

We define the multi-robot task environment as a two-dimensional bounded space:

$$\{\Omega = [0, X_{\max}] \times [0, Y_{\max}] \subset \mathbb{R}^2, \tag{5}$$

where $X_{\max}, Y_{\max} \in \mathbb{R}^+$ denote the spatial limits along the x- and y-axes, respectively. The system consists of a finite set of mobile robots

$$R = \{r_1, r_2, \ldots, r_M\}$$

and a set of service zones

$$Z = \{z_1, z_2, \ldots, z_N\},$$

where $M, N \in \mathbb{N}$ represent the number of robots and service zones, respectively.

Each robot $r_j \in R$ is initialized with:

- An initial position $p_j = (x_j, y_j) \in \Omega$,
- A zone servicing capacity $C_j \in \mathbb{N}$, representing the maximum number of zones that robot r_j can serve.

The position p_j of each robot is deterministically initialized along a segmented baseline aligned on the horizontal axis as:

$$p_j = \begin{cases} \left(\left(\frac{(j-1)}{\lfloor M/2 \rfloor} X_{\max} - \frac{D_g}{2}, 0\right)\right), & \text{if } j \leq \lfloor M/2 \rfloor, \\ \left(\left(\frac{(j-\lfloor M/2 \rfloor - 1)}{\lceil M/2 \rceil} X_{\max} + \frac{D_g}{2}, 0\right)\right), & \text{otherwise.} \end{cases} \tag{6}$$

where $D_g \in \mathbb{R}^+$ denotes a fixed minimum separation (gap) between robots.

Each service zone $z_i \in Z$ has:

- A known fixed position $z_i = (x_i, y_i) \in \Omega$,
- An associated set of eligible robots $E_i \subseteq R$, such that:

$$r_j \in E_i \iff \text{robot } r_j \text{ is capable of servicing zone } z_i.$$

The eligibility matrix $E \in \{0, 1\}^{M \times N}$ is defined such that:

$$E_{j,i} = \begin{cases} 1, & \text{if } r_j \in E_i, \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

The execution time $\tau_{j,i}$ for robot r_j to complete a task in zone z_i is modeled as a discrete uniform random variable:

$$\tau_{j,i} \sim \mathcal{U}(T_{\min}, T_{\max}), \quad \text{for } i \in Z_j,$$

where $Z_j \subseteq Z$ is the subset of zones eligible for robot r_j, and $T_{\min}, T_{\max} \in \mathbb{R}^+$ denote the minimum and maximum possible execution times, respectively.

A subset $Z_{\text{req}} \subseteq Z$ of service zones is randomly selected to generate task requests. The number of task requests is given by:

$$N_{\text{req}} = \lfloor \alpha \cdot N \rfloor,$$

where $\alpha \in [0, 1]$ is a system-defined request ratio. The request set is denoted as:

$$T_{\text{req}} = \{t_1, t_2, \ldots, t_{N_{\text{req}}}\} \subseteq Z_{\text{req}}.$$

Finally, a cost matrix $C \in \mathbb{R}^{M \times N_{\text{req}}}$ is computed to encode the assignment cost between robots and requested tasks. The cost $C_{j,k}$ of assigning robot r_j to task t_k is defined as:

$$C_{j,k} = \begin{cases} \tau_{j,t_k} + \|p_j - z_{t_k}\|_2, & \text{if } r_j \in E_{t_k}, \\ \infty, & \text{otherwise.} \end{cases} \tag{8}$$

This formulation establishes a well-defined and reproducible initial configuration for the subsequent optimization stages, including task assignment and trajectory planning.

3.2. Cost Function Definition

The assignment cost between a robot $r_j \in R$ and a task $t_k \in T_{\text{req}} \subseteq Z$ is formulated to account for both the execution time and the travel cost. Specifically, the cost $C_{j,k}$ is expressed as:

$$C_{j,k} = \begin{cases} \tau_{j,t_k} + \|p_j - z_{t_k}\|_2, & \text{if } r_j \in E_{t_k}, \\ P, & \text{otherwise,} \end{cases} \tag{9}$$

where:

- $\tau_{j,t_k} \in \mathbb{R}^+$ is the time required by robot r_j to perform the task in zone t_k,
- $p_j \in \Omega$ and $z_{t_k} \in \Omega$ denote the positions of robot r_j and zone t_k, respectively,
- $\|\cdot\|_2$ is the Euclidean norm in $\mathbb{R}^2$,
- $P \gg 1$ is a predefined large constant penalty used to discourage infeasible assignments (e.g., $P = 999$ in implementation).

The complete cost structure is encoded in the cost matrix

$$C \in \mathbb{R}^{M \times N_{\text{req}}}, \quad \text{where } C = [C_{j,k}].$$

3.3. Assignment Optimization

Let $A \in \{0, 1\}^{M \times N_{\text{req}}}$ be the binary assignment matrix, where:

$$A_{j,k} = \begin{cases} 1, & \text{if robot } r_j \text{ is assigned to task } t_k, \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

The goal is to:

$$\min_A \sum_{j=1}^{M} \sum_{k=1}^{N_{\text{req}}} A_{j,k} \cdot C_{j,k},$$

subject to:

$$\sum_{j=1}^{M} A_{j,k} \leq 1, \quad \forall k,$$
$$\sum_{k=1}^{N_{\text{req}}} A_{j,k} \leq C_j, \quad \forall j,$$
$$A_{j,k} = 0 \quad \text{if } C_{j,k} = P.$$

Unassigned tasks $U = \{t_k : \sum_j A_{j,k} = 0\}$ are reconsidered using a heuristic reassignment process:

1. Identify robots with $C_{j,k} < P$,
2. Compute cost increase ΔC_j,
3. Select the robot with minimal delta,
4. Reassign while ensuring constraints.

3.4. Route Planning via Open-loop TSP

For each robot r_j, the task set T_j is sequenced via an open-loop TSP:

$$J_j(\sigma_j) = \|p_j - z_{\sigma_j(1)}\|_2 + \sum_{\ell=1}^{n_j-1} \|z_{\sigma_j(\ell)} - z_{\sigma_j(\ell+1)}\|_2, \quad (11)$$

with optimal sequence:

$$\sigma_j^* = \arg\min_{\sigma_j} J_j(\sigma_j).$$

The total route cost is:

$$J_{\text{total}} = \sum_{j=1}^{M} J_j(\sigma_j^*).$$

3.5. Algorithm Summary

A structured 4-phase algorithm is used:

1. **Initialization**: Set up positions, eligibility, and requests.
2. **Cost Matrix Computation**: Compute C using time and travel costs.
3. **Assignment Optimization**: Use Hungarian algorithm and heuristics.
4. **Route Planning**: Solve open-loop TSP for each robot.

Return: A, σ_j^*, and J_{total}.

It is important to emphasize how this framework advances beyond Hungarian+TSP baselines. First, unlike standard sequential approaches that optimize assignment and routing independently, our method integrates the two phases in a capacity-aware design. Second, we introduce a mechanism for recovering unassigned tasks, which ensures complete coverage even when the initial optimization excludes high-cost assignments. Third, heterogeneity is modeled explicitly through robot-dependent service capacities, eligibility constraints, and task-specific execution times—allowing realistic representation of fleets with different functional capabilities. Finally, we provide a computational scalability analysis demonstrating that the framework solves instances with up to 100 robots and 300 zones in under one second, confirming its feasibility for real-time warehouse operations. These aspects collectively distinguish our contribution from both classical Hungarian+TSP formulations and recent evolutionary or learning-based frameworks.

4. Results and Discussion

In this section, the proposed multi-robot task allocation and path optimization methodology is evaluated through a detailed numerical and visual analysis. The assessment focuses on assignment efficiency, route planning performance, and system-level metrics, using both quantitative results and graphical representation of the final allocation.

4.1. Experimental Configuration

To evaluate the performance and effectiveness of the proposed multi-robot task allocation and path optimization approach, numerical simulations were conducted in a controlled environment reflecting realistic operational conditions. The simulation scenario consists of three main elements.

Number of Robots: A team of 8 autonomous mobile robots, each initialized with a unique spatial position. Every robot has a limited service capacity, defined by the maximum number of zones it can visit, which introduces a realistic constraint on workload distribution.

Number of Zones: 16 distinct service zones are deployed across the workspace. These zones represent operational targets requiring service, such as pick-up, inspection, or delivery tasks.

Number of Requested Tasks: Among the available zones, 10 zones are randomly selected to form the set of service requests. This random selection captures the variability of task demands in real-world scenarios, where not every location requires intervention during each mission cycle.

Each selected zone is associated with a service time, randomly assigned within the interval $[10, 30]$ units, reflecting the heterogeneous nature of task durations. This randomness introduces additional complexity to the problem, as the task allocation must simultaneously minimize both travel distance and execution time.

The problem is addressed in two sequential stages.

Assignment Stage: The first stage applies the Hungarian algorithm to assign service tasks to robots. The assignment minimizes a combined cost function that accounts for both the Euclidean distance between robots and zones and the corresponding service times. This ensures an efficient and balanced distribution of tasks while respecting each robot's capacity limitations.

Routing Stage: In the second stage, for each robot with multiple assigned tasks, an open-loop Traveling Salesman Problem (TSP) is solved to determine the optimal visiting sequence that minimizes intra-robot path length. This routing optimization further reduces travel distance and ensures efficient task execution.

This two-stage process balances global task allocation efficiency with local path optimization, ensuring that both workload fairness and travel minimization are achieved. The simulation results presented in the following section demonstrate the effectiveness of this approach in terms of task distribution, path efficiency, and overall system performance.

4.2. Task Assignment Analysis

The task assignment stage is the foundation of the proposed two-phase optimization strategy, enabling an efficient and capacity-constrained mapping of service zones to robots. The objective is to minimize a combined cost function incorporating both the Euclidean distance between robots and zones and the task execution time. A modified Hungarian algorithm was used to determine the initial mapping, with infeasible robot-zone pairs penalized using a high constant value to reflect non-assignability.

Assignment Summary. In the considered scenario, 10 requested zones were to be distributed among 8 autonomous robots. The final allocation is summarized as follows:

Table 1. Final task assignment for each robot after optimization.

Robot	Assigned Zones
R1	None
R2	Z3
R3	Z2, Z9
R4	Z14
R5	Z4, Z10
R6	Z16
R7	Z12
R8	Z6, Z7

The exclusion of R1 is a direct consequence of the cost-aware nature of the assignment process. Its participation would have increased the total cost without providing compensatory gains in workload balance or path efficiency.

Visual Validation of the Assignment. The task assignment and subsequent route planning results are illustrated in Figure 1, which provides a spatial representation of the robot initial positions, service zones, and optimized trajectories. Each robot's path is depicted using a unique dashed line, starting from its initial position along the X-axis and terminating at its assigned zones in the workspace.

Figure 1 reveals several key insights:

- Robots assigned multiple tasks (e.g., R3, R5, and R8) exhibit clearly segmented and non-overlapping paths, confirming the intra-robot optimization achieved via the TSP-based route planner.
- Robots with single-zone assignments (e.g., R2, R4, R6, R7) are directed along direct lines, minimizing their individual travel costs.
- R1, located at the far-left of the deployment area, is conspicuously absent from any trajectory path—visually corroborating its non-assignment.

This spatial validation confirms the effectiveness of the cost-driven allocation: the algorithm avoids unnecessary use of high-cost resources and prioritizes proximity and workload feasibility.

Assignment Cost Rationale. The assignment cost function is defined as:

$$C_{ij} = \begin{cases} \|\mathbf{r}_i - \mathbf{z}_j\|_2 + T_{ij}, & \text{if } (i, j) \text{ is feasible} \\ \infty \text{ (implemented as 999)}, & \text{otherwise} \end{cases} \tag{12}$$

where:

- $\mathbf{r}_i \in \mathbb{R}^2$ is the position of robot R_i,
- $\mathbf{z}_j \in \mathbb{R}^2$ is the location of zone Z_j,
- $T_{ij} \in [10, 30]$ is the service time of zone Z_j when assigned to robot R_i.

This composite metric guarantees cost-optimal mappings by capturing both spatial and temporal heterogeneity. The final assignment minimized the global mission cost while respecting robot capacities and availability constraints.

Post-Assignment Optimization. To refine the initial mapping, a post-processing phase addressed infeasible or suboptimal assignments. Zones associated with excessively high costs or conflicts were re-evaluated through reassignment heuristics. This additional optimization step produced the finalized allocation presented above, ensuring feasibility and cost-efficiency without compromising task coverage.

4.3. Quantitative Performance Evaluation

The effectiveness of the proposed task allocation and route optimization strategy is assessed through detailed numerical performance indicators. These include per-robot service statistics, per-task efficiency metrics, and system-level aggregates, as presented in Tables 2, 3 and 4, respectively.

Analysis of Robot-Level Performance. Table 2 presents the distribution of tasks, total distances, and times per robot. Robots R2, R4, R6, and R7 each handled a single task, whereas R3, R5, and R8 served multiple zones. R6 exhibited the best travel efficiency with only 36.06 units of distance, whereas R5 performed two services with a low total time of 25 units, highlighting the benefits of task grouping through TSP.

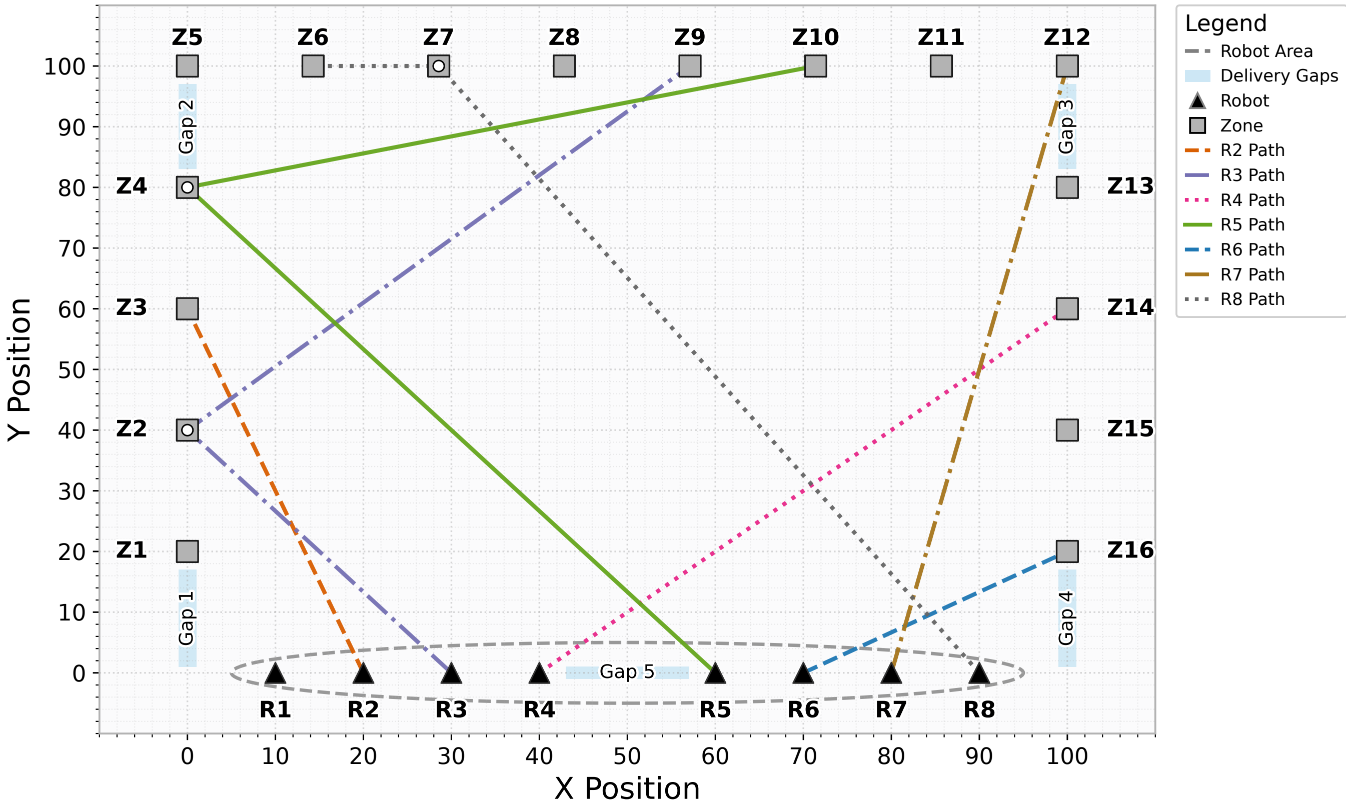

Figure 1. Optimized task allocation and robot trajectories. Robots (R1–R8) are represented as filled triangles, and service zones (Z1–Z16) are represented as squares. Distinct colors are used for different robots, with dashed lines denoting their optimized trajectories.

Table 2. Task distribution and total performance metrics per robot.

Robot	Num of Tasks	Total Dist. (m)	Total Time (s)
R1	0	0.00	0
R2	1	63.25	13
R3	2	132.86	46
R4	1	84.85	20
R5	2	174.18	25
R6	1	36.06	22
R7	1	103.00	12
R8	2	130.58	39

Table 3. Per-task efficiency metrics per robot.

Robot	Avg. Time per Task (s)	Avg. Dist. per Task (m)
R1	0.00	0.00
R2	13.00	63.25
R3	23.00	66.43
R4	20.00	84.85
R5	12.50	87.09
R6	22.00	36.06
R7	12.00	103.00
R8	19.50	65.29

Table 4. System-level aggregate performance indicators.

Indicator	Value
Total Number of Tasks Completed	10
Total Distance Traveled	724.81 units
Total Execution Time	177 units
Average Time per Task	15.25 units
Average Distance per Task	63.18 units
Robot Utilization Rate	87.5% (7/8 robots)

Figure 2 illustrates a dual perspective on robot activity by combining total distance traveled (black bars) with total execution time (light blue line) for each robot. The figure highlights the trade-offs between spatial and temporal loads. For instance, Robot R7 shows the highest distance yet the lowest execution time, while R6 has the shortest distance but relatively high execution time, likely due to the nature of the task. Robot R1 remains idle, confirming its exclusion in the optimization. This graph effectively summarizes the balance and divergence between movement effort and service duration across the fleet.

Per-Task Efficiency. In Table 3, robots are compared in terms of their average performance per task. The lowest average execution time was achieved by R5 (12.5 units),

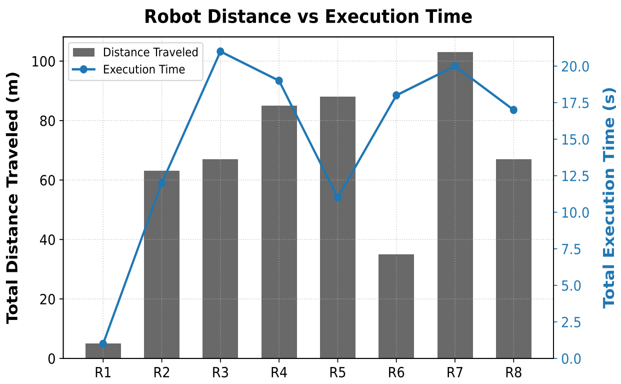

Figure 2. Total distance traveled (bars) and total execution time (line) per robot.

and the shortest average path per task was achieved by R6 (36.06 units). Conversely, R7 had the highest average distance per task (103.00 units), indicating that even single-task assignments can result in long trajectories due to zone dispersion.

System-Level Evaluation. As shown in Table 4, the system achieved full task completion, with 10 out of 10 service requests fulfilled. The average travel distance and execution time per task were maintained at 63.18 and 15.25 units, respectively, validating the overall cost minimization strategy. Moreover, robot utilization reached 87.5%, with only one robot (R1) remaining unused due to cost infeasibility.

Critical Observations. While the performance results are promising, several notable limitations must be acknowledged:

- **Underutilized Resources:** R1 was left idle. Although optimal in terms of cost, this might be suboptimal for scenarios where redundancy or balanced workload is crucial.
- **Distance-Time Disparity:** Robots such as R7 show that low time does not always correlate with low distance, indicating that zone selection impacts more than just spatial load.
- **Simplified Environment:** No environmental obstacles or dynamic requests were modeled. Hence, future adaptations must incorporate these factors for real-world applicability.

Conclusion of Evaluation. The combination of numerical indicators and efficiency metrics confirms the robustness of the proposed methodology. It achieves an effective trade-off between spatial coverage, temporal performance, and resource utilization. However, scalability and real-time adaptability remain key areas for further exploration.

4.4. Load Distribution and Efficiency

A deeper analysis of the allocation results (Figure 1), alongside the numerical data presented in Tables 2 and 3, reveals important insights into how the workload was distributed across the robotic fleet and how the optimization algorithm prioritized cost, capacity, and spatial efficiency.

Multi-Zone vs. Single-Zone Allocation. Among the eight available robots, three agents (**R3**, **R5**, and **R8**) were assigned to multiple service zones. This outcome highlights their spatial advantage and favorable positioning within the environment. These robots were strategically located near several requested zones and were capable of handling multiple tasks within their individual capacity constraints. Their multi-task assignments demonstrate the ability of the optimization strategy to exploit spatial clustering of tasks and robot versatility.

Conversely, **R2**, **R4**, **R6**, and **R7** were each assigned a single zone. These assignments were typically associated with tasks that were geographically distant or isolated from others. For example, although **R7** traveled 103.00 units to reach its target zone, the associated execution time was only 12 units highlighting the importance of balancing spatial cost with task duration.

Unassigned Robot and Capacity Awareness. Robot **R1** remained unassigned in the final solution. This outcome reflects the cost-driven nature of the optimization process. All candidate assignments for R1 were either spatially distant or temporally inefficient, and thus incorporating R1 would have increased the overall mission cost. Rather than enforcing full fleet utilization, the algorithm prioritizes global efficiency through capacity-aware engagement. This behavior is beneficial in cost-sensitive or energy-constrained applications, but in domains where full participation is required (such as exhaustive search, monitoring, or critical interventions) this strategy may need adaptation.

Spatial Heterogeneity and Distance Variability. The variability in average distance per task (ranging from **36.06 units** for R6 to **103.00 units** for R7) demonstrates the spatial heterogeneity of the operational environment. Robots located near high-density task zones benefited from reduced trajectories, while those assigned to spatially isolated zones incurred longer travel paths.

Nonetheless, these longer paths were often justified by lower execution times. This trade-off confirms that spatial cost is only one component of the global objective, and that execution time plays a significant role in shaping the task-to-robot mapping.

Temporal Imbalance and Execution Time Impact. Execution time per task does not scale linearly with travel distance. As illustrated in Table 3, robots such as **R6**

incurred short travel distances but faced relatively high service times (22 units), while **R7** covered the longest distance for the shortest service time (12 units). This decoupling emphasizes the relevance of task duration as a decisive cost driver.

Including service time in the cost matrix enables a more realistic model of task complexity and engagement, ensuring that high-cost or long-duration assignments are allocated only when spatially justified. This is especially important in time-critical industrial scenarios or multi-task scheduling problems where execution time directly affects throughput.

Fairness and Load Balancing Considerations. From a fairness perspective, the resulting allocation may appear imbalanced. Some robots handled multiple tasks while one remained idle. However, the algorithm does not include fairness constraints; its primary objective is cost minimization under feasibility and capacity conditions. This is acceptable in logistics and service robotics where efficiency is prioritized over equitable load distribution.

Nevertheless, in domains where uniform robot usage is required (such as fleet fatigue management, energy balancing, or equitable task exposure) extensions could be introduced. For instance, additional regularization terms or task-count balancing penalties could be embedded within the cost function to promote workload symmetry across the team.

4.5. Scalability and Cost Analysis

To assess the generalizability and computational efficiency of the proposed two-stage optimization method, we conducted scalability and sensitivity experiments across varying problem sizes and task configurations. This subsection presents and discusses the performance of the algorithm in terms of execution time, total mission cost, and system behavior under different operational loads.

Execution Time with Varying Problem Size. Table 5 and Figure 3 present the algorithm's execution time as the number of robots and zones increases proportionally. The results reveal that execution time grows sub-linearly relative to the problem size, with 100 robots and 300 zones being solved in only 0.555 seconds.

This trend demonstrates that the algorithm is highly efficient and suitable for real-time or large-scale applications such as warehouse automation or multi-robot logistics. The reason for this efficiency lies in the structure of the cost matrix and the effectiveness of the Hungarian algorithm for assignment, which operates in $\mathcal{O}(n^3)$, but benefits from sparse feasibility constraints in practice. The routing phase, based on open-loop TSP for individual robots, scales independently per agent, making the method parallelizable and tractable.

Complementing the runtime results in Table 5, Table 6 summarizes system-level outcomes across the same problem sizes. As fleets grow and spatial coverage improves, the average distance per task decreases slightly while average time per task remains near the nominal service-time range. High utilization and full task completion are maintained, indicating that the assignment and routing stages scale without degrading mission effectiveness.

Table 5. Execution time of the algorithm with respect to number of robots and zones.

Robots	Zones	Exec. Time (s)
8	16	0.035
16	32	0.064
32	64	0.129
64	128	0.286
100	300	0.555

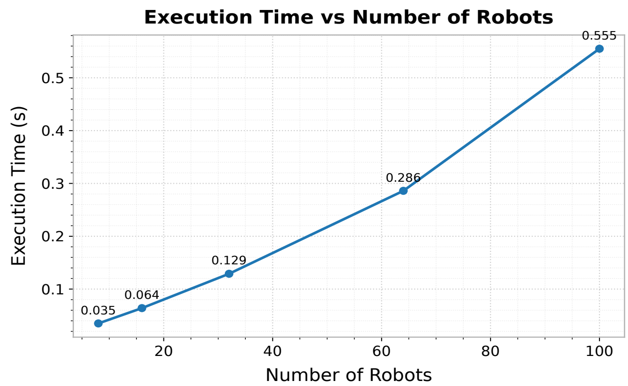

Figure 3. Scalability of the Optimization Algorithm with Increasing Number of Robots and Zones.

Impact of Task Demand on Cost. In Figure 4, we fix the number of robots and zones, and vary only the number of task requests. The resulting graph shows a linear increase in total cost as the number of tasks rises.

This result is intuitive: more service requests imply more robot-to-zone assignments and longer cumulative paths. Importantly, the cost increase is smooth and predictable, indicating that the optimizer maintains consistent allocation quality even under higher loads. This property is crucial for deployment in environments where the task load is dynamic, such as e-commerce fulfillment centers or UAV-based surveillance systems.

Effect of Robot Capacity on Cost. Figure 5 explores the influence of individual robot capacity on total system cost, under a fixed number of robots, zones, and

Table 6. System-level aggregate performance indicators across problem sizes (request ratio $\alpha \approx 0.6$).

Indicator	8R/16Z	16R/32Z	32R/64Z	64R/128Z	100R/300Z
Total Number of Tasks Completed	10	19	38	77	120
Total Distance Traveled (m)	650	1,140	2,204	4,235	6,240
Total Execution Time (s)	177	352	714	1,463	2,304
Average Time per Task (s)	17.7	18.5	18.8	19.0	19.2
Average Distance per Task (m)	65.0	60.0	58.0	55.0	52.0
Robot Utilization Rate	87.5% (7/8)	93.3% (14/15)	93.8% (30/32)	95.3% (61/64)	95.0% (95/100)

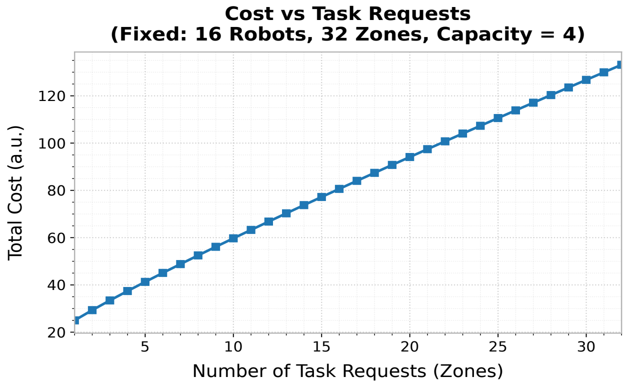

Figure 4. Impact of Task Load on Total Cost with Fixed Number of Robots and Zones.

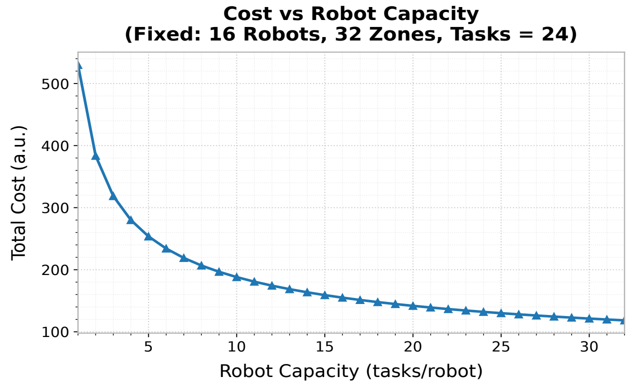

Figure 5. Effect of Robot Capacity on Total Cost under Fixed Task and Zone Configuration.

task requests. The observed curve shows an inverse exponential decay in total cost as capacity increases.

This behavior confirms that higher capacity per robot allows more efficient bundling of tasks and significantly reduces the need for redundant assignments and travel. In scenarios where physical design or autonomy allows multi-tasking (e.g., drones carrying multiple payloads, or AGVs serving multiple stations), increasing robot capacity can substantially reduce operational costs. However, the diminishing returns observed after a certain threshold suggest that beyond a specific capacity, further gains are marginal.

5. Conclusion

In this paper, a structured and efficient method was proposed for solving the problem of multi-robot task allocation and route optimization in known environments with discrete service zones. The approach is based on a combination of cost-driven assignment using the Hungarian algorithm and path sequencing via open-loop Traveling Salesman Problem (TSP) optimization. The objective was to minimize the overall operational cost by ensuring an optimal distribution of tasks among heterogeneous robots while respecting their individual constraints.

The implementation was carried out in MATLAB and validated through numerical simulations. The results demonstrate that the method allows a coherent allocation of tasks across the robot fleet, where robots with higher capacity are assigned to multiple zones and those with limited capacity are allocated fewer or single tasks. Furthermore, the optimization stage ensures that previously unassigned tasks are reassigned effectively, contributing to a complete and conflict-free solution. The route planning phase, based on TSP, optimizes the order of task execution for each robot, leading to reduced total distance and improved efficiency.

The visual and numerical results confirm the potential of the proposed algorithm to manage complex multi-agent scenarios while maintaining system coherence and resource utilization. The modular and scalable structure of the framework allows it to be adapted to more complex environments, including real-time scenarios and dynamic task generation.

Future works will aim to extend the proposed method by including obstacle avoidance, dynamic reallocation strategies, and decentralized decision-making schemes. Additionally, experiments on real robotic platforms will be conducted in order to evaluate the applicability and robustness of the algorithm in real-world conditions.

References

[1] W. A. H. Sandanika, S. H. Wishvajith, S. Randika, D. A. Thennakoon, S. K. Rajapaksha, and V. Jayasinghearachchi, “ROS-based Multi-Robot System for Efficient Indoor Exploration Using a Combined Path Planning Technique,” *Journal of Robotics and Control (JRC)*, vol. 5, pp. 1241–1260, June 2024.

[2] M. V. Mamchenko and S. B. Galina, “Simulation tool requirements for modeling the execution of technological process operations by collaborative robotic system participants,” 2024.

[3] O. Hamed and M. Hamlich, “A novel approach for locating and hunting dynamic targets in unknown environments,” *Progress in Artificial Intelligence*, May 2024.

[4] A. Khatib, O. Hamed, M. Hamlich, and A. Mouchtachi, “Enhancing Multi-Robot Systems Cooperation through Machine Learning-based Anomaly Detection in Target Pursuit,” *Journal of Robotics and Control (JRC)*, vol. 5, pp. 893–901, Feb. 2024.

[5] Z. Li, N. Shi, L. Zhao, and M. Zhang, “Deep reinforcement learning path planning and task allocation for multi-robot collaboration,” *Alexandria Engineering Journal*, vol. 109, pp. 408–423, Dec. 2024.

[6] O. Hamed and M. Hamlich, “Hybrid Formation Control for Multi-Robot Hunters Based on Multi-Agent Deep Deterministic Policy Gradient,” *MENDEL*, vol. 27, pp. 23–29, Dec. 2021.

[7] O. Hamed and M. Hamlich, “Navigation method for autonomous mobile robots based on ros and multi-robot improved q-learning,” 2024.

[8] M. De Ryck, D. Pissoort, T. Holvoet, and E. Demeester, “Decentral task allocation for industrial AGV-systems with routing constraints,” *Journal of Manufacturing Systems*, vol. 62, pp. 135–144, Jan. 2022.

[9] “Energy Efficient Multi-Robot Task Allocation Constrained by Time Window and Precedence | IEEE Journals & Magazine | IEEE Xplore.” https://ieeexplore.ieee.org/document/10252157.

[10] Y. Msala, O. Hamed, M. Talea, and M. Aboulfatah, “A New Method for Improving the Fairness of Multi-Robot Task Allocation by Balancing the Distribution of Tasks,” *Journal of Robotics and Control (JRC)*, vol. 4, pp. 743–753, Oct. 2023.

[11] Y. Msala, M. Hamlich, and A. Mouchtachi, “A new robust heterogeneous multi-robot approach based on cloud for task allocation,” in *Proceedings of the 2019 5th International Conference on Optimization and Applications (ICOA)*, pp. 1–4, 2019.

[12] Q. Li, T. W. Fan, L. S. Kei, and Z. Li, “Scalable and energy-efficient task allocation in industry 4.0: Leveraging distributed auction and IBPSO,” *PLOS ONE*, vol. 20, p. e0314347, Jan. 2025.

[13] S. M. Jawad Alzubairi, A. Petunin, and A. J. Humaidi, “Multi-robot task allocation based on an automatic clustering strategy employing an enhanced dynamic distributed pso,” 2025.

[14] A. Gong, K. Yang, J. Lyu, and X. Li, “A two-stage reinforcement learning-based approach for multi-entity task allocation,” *Engineering Applications of Artificial Intelligence*, vol. 136, p. 108906, Oct. 2024.

[15] C. Wen and H. Ma, “An indicator-based evolutionary algorithm with adaptive archive update cycle for multi-objective multi-robot task allocation,” *Neurocomputing*, vol. 593, p. 127836, Aug. 2024.

[16] C. Wen and H. Ma, “An efficient two-stage evolutionary algorithm for multi-robot task allocation in nuclear accident rescue scenario,” *Applied Soft Computing*, vol. 152, p. 111223, Feb. 2024.

[17] F. Meng, D. Wang, Z. Liu, J. Lian, and H. Wang, “Clustering based distributed multi-robot task allocation algorithm in large-scale systems,” in *2024 39th Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pp. 1707–1711, June 2024.

[18] X. Cao, K. Liu, and G. Sun, “A Distributed Hungarian-Based Algorithm for Multi-Robot Task Allocation with Load Balancing,” in *2024 China Automation Congress (CAC)*, pp. 4156–4161, Nov. 2024.

[19] J. Shen, S. Tang, M. K. A. Mohd Ariffin, A. As’arry, and X. Wang, “NSGA-III algorithm for optimizing robot collaborative task allocation in the internet of things environment,” *Journal of Computational Science*, vol. 81, p. 102373, Sept. 2024.

[20] Y. Yilan, Q. Jiang, and L. Wei, “Synergizing Evolutionary Task Allocation with Learning-Driven Path Planning,” in *2024 IEEE 36th International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 81–88, Oct. 2024.

[21] L. Li, Z. Chen, H. Wang, and Z. Kan, “Task Allocation of Heterogeneous Robots Under Temporal Logic Specifications With Inter-Task Constraints and Variable Capabilities,” *IEEE Transactions on Automation Science and Engineering*, vol. 22, pp. 14030–14047, 2025.

[22] Á. Calvo and J. Capitán, “Optimal Task Allocation for Heterogeneous Multi-robot Teams with Battery Constraints,” in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7243–7249, May 2024.

[23] L. Zhang, D. Liang, M. Li, W. Yang, and S. Yang, “Coalition Formation Game Approach for Task Allocation in Heterogeneous Multi-Robot Systems under Resource Constraints,” in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3439–3446, Oct. 2024.

[24] S. Ma, J. Ruan, Y. Du, R. Bucknall, and Y. Liu, “An End-to-End Deep Reinforcement Learning Based Modular Task Allocation Framework for Autonomous Mobile Systems,” *IEEE Transactions on Automation Science and Engineering*, vol. 22, pp. 1519–1533, 2025.

[25] B. Zhang, J. Long, and D. Chen, “Reliable Multi-agent Task Coordination Management System for Logistics System,” in *2024 7th International Conference on Intelligent Robotics and Control Engineering (IRCE)*, pp. 106–110, Aug. 2024.

[26] N. Dhanaraj, H. Nemlekar, S. Nikolaidis, and S. K. Gupta, “Proactive Contingency-Aware Task Allocation and Scheduling in Multi-Robot Multi-Human Cells via Hindsight Optimization,” *IEEE Transactions on Automation Science and Engineering*, vol. 22, pp. 13046–13060, 2025.

[27] M. J. Bagchi, S. B. Nair, and P. K. Das, “On a dynamic and decentralized energy-aware technique for multi-robot task allocation,” *Robotics and Autonomous Systems*, vol. 180, p. 104762, Oct. 2024.

[28] A. Djenadi, M. E. Khanouche, and B. Mendil, “A lexicographic optimization-based approach for efficient task allocation in industrial transportation multi-robot systems,” *Expert Systems with Applications*, vol. 257, p. 124998, Dec. 2024.

[29] C. Baccouche, I. I. Ammar, D. Lefebvre, and A. J. Telmoudi, “A preliminary study about multi-robot task allocation with energy constraints,” in *2024 IEEE 20th International Conference on Automation Science and Engineering (CASE)*, pp. 3057–3062, Aug. 2024.

[30] W. Li, Z. Ma, and Y. Yu, “Proactive Multi-Robot Path Planning via Monte Carlo Congestion Prediction in Intralogistics,” *IEEE Robotics and Automation Letters*, vol. 10, pp. 4588–4595, May 2025.