

A MILP-Based Optimization Framework for Autonomous Mobile Robot Task Scheduling, Routing, and Charging in Manufacturing Environments

Haboul Lida, Al-Attar Bayan, Abu Al Hummos Lujain and Al-Naimat Muna

Students, Industrial Engineering Department

Al Hussein Technical University

Amman, Jordan

23110057@htu.edu.jo, 23110694@htu.edu.jo, 23110185@htu.edu.jo, 23110069@htu.edu.jo

Abstract

Autonomous Mobile Robots (AMRs) play a critical role in flexible material handling within modern manufacturing systems, enabling increased productivity and reduced operational waste. However, inefficiencies in task scheduling, routing, and battery management remain a significant operational challenge, as existing approaches typically rely on fixed charging schedules and fragmented decision-making, often resulting in unplanned downtime and missed deadlines. This paper proposes a Mixed Integer Linear Programming (MILP) optimization model that integrates multiple decision layers including task sequencing, routing, and charging detour selection into a unified mathematical framework. The model operates under interdependent constraints incorporating battery State of Charge (SOC), task durations, charging station availability, route feasibility, and task precedence. SOC transitions are tracked by accounting for energy consumption during both travel and task execution, as well as energy recovery during charging in the Constant Current (CC) phase. Moreover, routing feasibility is ensured through flow conservation and subtour elimination constraints. A weighted multi-objective function is formulated to minimize total task tardiness and total charging time, enabling data-driven routing and charging decisions without relying on predefined schedules. The model is validated through implementing it in diverse manufacturing scenarios, including multi-AMR fleets. Results demonstrate measurable improvements in charging stops, task tardiness, and production efficiency compared to standard baseline cases, confirming the proposed framework as a scalable and adaptive solution for modern manufacturing environments.

Keywords

Autonomous Mobile Robots (AMRs), Mixed Integer Linear Programming (MILP), Task Scheduling and Routing, Tardiness, Battery Management.

1. Introduction

Manufacturing systems are increasingly adopting automation, highlighting the need for efficient, effective, and flexible material handling solutions that can adapt to various conditions. Autonomous Mobile Robots (AMRs) enable reactive material transport without fixed paths and infrastructure, bridging the gap between operational efficiency and scalable layouts across diverse environments, making them suitable for complex manufacturing settings. However, their effective management is still a challenge due to the interdependence between multiple decisions, including task assignment, route determination, scheduling, and energy management. Existing approaches do not handle these decisions as a connected system. Instead, they separate them and rely on rule-based strategies, such as fixed charging times or predefined routes, leading to inefficiencies like delays, unnecessary charging, and reduced system performance. This study proposes a mathematical optimization framework that integrates and considers task scheduling, routing, and AMR charging under battery and operational constraints. The objective is to minimize tardiness and total charging times while ensuring efficient system operations.

1.1 Objectives

This study develops an integrated optimization framework for Autonomous Mobile Robots (AMRs), by proposing a MILP formulation that addresses scheduling, routing, and battery management. The model incorporates SOC dynamics and optimizes charging and scheduling decisions through a weighted objective minimizing task tardiness and charging time, while ensuring routing feasibility. Validation is conducted using Python implementation with the PuLP library to simulate manufacturing scenarios including multi-AMR fleets.

2. Literature Review

2.1 Introduction

The industrialization landscape is undergoing a paradigm shift towards "Smart Manufacturing" and "Smart Warehousing", in response to growing demands for higher degrees of flexibility and efficiency. At the heart of this transformation stand Autonomous Mobile Robots (AMRs). Unlike their predecessors, Automated Guided Vehicles (AGVs), which follow fixed magnetic strips or wires, AMRs are capable of autonomous navigation in complex environments by utilizing advanced sensors and computer vision (Ze-kun 2024; Fourie et al. 2025). Still, the problem of managing these systems becomes increasingly complex as fleet sizes increase, and the environment gets more dynamic.

2.2 The Modelling and Simulation Environments

Simulation still plays an important role, due to the risks from unoptimized algorithms being rolled out in real manufacturing units. Specialized environments like the Warehouse Autonomous Mobile Robot (WAMR) Simulator have been designed to simulate task allocation models (Hu et al. 2024). Simulation-based optimization is often utilized to handle AMR fleets for internal transport, concentrating on the efficiency of the application, without the need for a centralized task planner (Dobrzańska et al. 2025). Other work applies discrete-event simulations to forecast KPIs such as utilization rates and constraint satisfaction in decentralized multi-robot task allocation cases (Godeke et al. 2023). Optimization problems in AMR applications are most often proposed as mathematical models in high dimensions. Common formulations are Mixed-Integer Linear Programming (MILP), which is employed for centralizing optimal operation plans and trajectory planning with task space constraints (Huo 2025), or the Traveling Salesman Problem (TSP) variant: AMR path optimization for collection of ordered products can be widely considered as a dynamic TSP for total travel distance reduction (Nikolić et al. 2025).

2.3 Current Applications and limitations

AMRs are being used in a wide variety of industrial sectors, each of which has heterogeneous operational needs. Its logistics work in manufacturing studios is the logistics backbone carrying raw materials out to the work line and transferring the finished product to storage (Lyu et al. 2025; Huo 2025). Sets of application such as Automotive Industry to supply assembly lines with the parts and other components (Čech et al. 2020) or performing tasks in dedicated factory where AMRs are only used with single or multiple AMRs (Dang et al. 2013).

2.4 Deployment and Workflow Challenges

Factories and warehouses are inherently dynamic environments therefore AMRs are subject to moving objects like forklifts, human operators, and shifting floor configurations. Traffic congestion at denser operational times is also common and, if not dealt with advanced traffic-aware procedures (Kim 2025; Barnhart et al. 2024), can cause system performance to decrease. AMRs are not long-lasting when it comes to battery life. Tasks that are scheduled (without compromising their charge), whilst also maintaining reasonable trajectories, are an optimal optimization task (Wu et al. 2024; Chu et al. 2024). Also, algorithms that perform well in simulation environments encounter unexpected challenges when extended to the real world, with sensor noise and erratic mechanical behavior (Garcia et al. 2025).

2.5 Optimization Approaches

A variety of algorithm optimization strategies to optimize AMR operations have been proposed, organized into path planning, coordination, and task allocation. Modified A* algorithms for single-robot pathfinding are popular to some extent (Enright et al. 2011). Genetic Algorithms (GA) are also used to compute an independent path and optimize a path with respect to optimizing for path length and smoothness (Nikolić et al. 2025). DRL and MAPF are two classes of such models which are widely used for path planning in sophisticated high-dimensional scenarios and can aid in rapid path planning processes, effective management of computational complexity (Shen et al. 2023). To schedule a fleet, multiple autonomous agents' interactions are managed integrated, and in this respect integrated approaches such

as the Job-initiated Traffic-aware Protocol (JTAP) combine centralized strategy with decentralized deployment to reduce congestion (Kim 2025).

2.6 Limitations of Existing Evidence

Despite a strong basis in literature on AMR optimization, there are significant shortcomings. Many proposed models are tested only on small to medium-sized fleets. The high computational requirements of centralized models such as MILP may become too much in many of the case-reports for very large-scale application (Wu et al. 2025). A large part of the literature relies on homogeneous robot fleets. Factories rely on an assortment between robots of varied size and steering configuration in practice, which hampers optimization (Msala et al. 2025).

2.7 Research Gaps

There is no research dedicated to improving resilience of coordination algorithms of industrial mm Wave networks against communication delays or interference (Ren and Friderikos 2022). As well as building models that optimize makespan, energy consumption, battery health and performance in a single framework (Huo 2025; Chu et al. 2024).

Table 1. Comparison of Existing AMR Optimization Approaches and the Proposed Framework

Study	Main Features	Key Limitation	Contribution Relative to This Study
Kim (2025)	Traffic-aware fleet coordination using JTAP	Does not incorporate battery SOC or charging decisions	Highlights the importance of integrating operational coordination with energy management
Huo (2025)	Integrated scheduling and path planning using MILP	Charging behavior and battery dynamics are not considered	Demonstrates the need for battery-aware optimization
Chu et al. (2024)	Energy-efficient trajectory planning under configuration constraints	Focuses on energy consumption without task scheduling integration	Shows the value of combining energy and scheduling decisions
Wu et al. (2025)	Large-scale multi-robot coordination and conflict-free planning	Limited focus on charging optimization and task deadlines	Emphasizes coordination but not integrated charging decisions
Msala et al. (2025)	Heterogeneous multi-robot warehouse operations	Battery management and charging strategies are not explicitly modeled	Highlights challenges in practical AMR fleet operation
Proposed Study	Unified MILP framework integrating task scheduling, routing, SOC tracking, and charging decisions	Assumes known distances and charging station availability	Simultaneously optimizes scheduling, routing, and charging decisions while minimizing tardiness and charging time

Table 1 demonstrates that existing studies typically address scheduling, routing, coordination, or energy management separately, whereas the proposed framework integrates task scheduling, routing, SOC dynamics, and charging decisions within a single optimization model.

2.8 Summary

The literature discusses the emergence of Autonomous Mobile Robots (AMRs) as important enablers of smart manufacturing and smart warehousing due to their flexibility, reduced manual effort in transportation activities, and better material flow. As a challenge, their deployment relies on strong optimization models to perform path planning, task distribution, fleet coordination, and energy use control with robust optimization.

As simulation devices and mathematical approaches including MILP, TSP-based models, genetic algorithms, and reinforcement learning have laid the foundation for the studies, few serious challenges can still be identified in the real-world applications, particularly regarding congestion, battery constraints, safety, communication reliability, and heterogeneous fleet coordination. This suggests that future research should create scalable, resilient, and integrated optimization frameworks for more industrial realizations out of theoretical models.

3. Methods

3.1 Problem Description

A fleet of AMRs is considered for executing a set of tasks distributed across several locations in a manufacturing facility. Each AMR is required to visit task locations and complete the assigned tasks within specified time frames and manage battery charging through available charging stations.

3.2 Model Assumptions

The following assumptions are adopted to define the scope of the proposed model: the manufacturing environment layout and travel distances between nodes are known and fixed; task durations, energy consumption rates, and charging rates are known in advance; battery charging is linear, following the Constant Current (CC) phase model; each task is assigned to exactly one AMR at a time and can only be performed once; AMRs begin operations from a defined depot node; charging stations have sufficient capacity and are always available when needed; the initial SOC of each AMR is always above the minimum allowable SOC; and charging only occurs between tasks.

3.3 Model Notation

To formally define the proposed MILP model, the necessary sets, indices, parameters, and decision variables are introduced in this section. The notation is organized in Table 2 for Sets and Indices, Parameters, and Decision Variables respectively. This notation is used consistently throughout the mathematical formulation given in Section 3.4.

Table 2. Notation

	Symbol	Description
<i>Sets</i>	$N = \{1, \dots, N \}$	Set of nodes
	$R = \{1, \dots, R \}$	Set of tasks
	$R_0 = \{0\} \cup R$	Set of tasks including the depot node 0
	$K = \{1, \dots, K \}$	Set of autonomous mobile robots (AMRs)
	$C \subseteq N$	Set of charging stations
<i>Indices</i>	$k \in K$	Index of AMRs
	$r, s \in R$	Indices of tasks
	$i, j \in N$	Indices of nodes
	$c \in C$	Index of charging stations
<i>Parameters</i>	$o_r, d_r \in N$	Origin and destination nodes of task r
	$u(i, j)$	Travel time between nodes i and j
	p_r	Service time
	D_r	Due time of task r
	e_{xons}	Energy consumption rate per unit time
	e_{main}	Charging rate per unit time
	$\text{SOC}_{\min}$	Minimum allowable battery level
	$\text{SOC}_{\max}$	Maximum allowable battery level
	$\text{SOC}_{\text{init},k}$	Initial state of charge of AMR k

Decision Variables	$d_r^{\min} = \min_{c \in C} u(d_r, c)$	Minimum travel time from task r to the nearest charging station
	M	Large constant (Big-M)
	w_1, w_2	Weighting coefficients for tardiness and charging time
	a_{krs}	Equals 1 if AMR k travels from task r to task s , and 0 otherwise
	z_{kr}	Equals 1 if AMR k is assigned to execute task r , and 0 otherwise
	$y_{kr}c$	Equals 1 if AMR k visits charging station c after completing task r , and 0 otherwise
	S_{kr}	Start time of task r executed by AMR k
	C_{kr}	Completion time of task r executed by AMR k
	C_r	Actual completion time of task r
	$h_{kr}c$	Charging duration of AMR k at station c after task r
	SOC_{kr}	State of charge of AMR k after completing task r
	T_r	Tardiness of task r
	u_{kr}	Order of task r in the route of AMR k

3.4 MILP Formulation

This section presents the mathematical formulation based on the problem description, assumptions, and notation. The model is formulated as a Mixed Integer Linear Program (MILP) comprising an objective function and constraints governing task assignment, routing, battery dynamics, charging decisions, and tardiness computation. The Big-M method, where (M) is a sufficiently large constant, is used to activate or relax constraints according to binary variable values. The objective function incorporates weighting coefficients w_1 and w_2 to balance the trade-off between minimizing total task tardiness and total charging time, allowing decision-makers to prioritize objectives according to operational requirements.

3.4.1 Objective Function

The objective is to minimize the total tardiness of tasks along with the total charging time across all AMRs:

$$\min Z = w_1 \sum_{r \in R} T_r + w_2 \sum_{k \in K} \sum_{r \in R} \sum_{c \in C} h_{kr}c$$

3.4.2 Constraints

3.4.2.1 Task Assignment

Each task must be assigned to exactly one AMR:

$$\sum_{k \in K} z_{kr} = 1, \quad \forall r \in R$$

3.4.2.2 Flow Conservation

To ensure routing consistency by enforcing one successor and one predecessor per task:

$$\sum_{\substack{s \in R_0 \\ s \neq r}} a_{krs} = z_{kr}, \quad \forall k \in K, r \in R$$

$$\sum_{\substack{s \in R_0 \\ s \neq r}} a_{ksr} = z_{kr}, \quad \forall k \in K, r \in R$$

3.4.2.3 Depot Constraints

These constraints ensure that each AMR starts from and returns to the depot at most once:

$$\sum_{s \in R} a_{k0s} \leq 1, \quad \forall k \in K$$

$$\sum_{r \in R} a_{kr0} \leq 1, \quad \forall k \in K$$

$$\sum_{s \in R} a_{k0s} = \sum_{r \in R} a_{kr0} \leq 1, \quad \forall k \in K$$

3.4.2.4 Subtour Elimination

The following constraints eliminate subtours by enforcing a consistent ordering of tasks within each AMR route.

$$u_{kr} - u_{ks} + |R|a_{krs} \leq |R| - 1, \quad \forall k \in K, r, s \in R, r \neq s$$

$$1 \leq u_{kr} \leq |R|, \quad \forall k \in K, r \in R$$

3.4.2.5 Time Definition

The completion time of each task is defined based on its start time and processing duration:

$$C_{kr} \geq S_{kr} + u(o_r, d_r) + p_r - M(1 - z_{kr}), \quad \forall k \in K, r \in R$$

$$C_{kr} \leq S_{kr} + u(o_r, d_r) + p_r M(1 - z_{kr}), \quad \forall k \in K, r \in R$$

3.4.2.6 Sequencing with Charging

Successive task timing is modeled using constraints that account for direct travel and optional charging detours:

$$u_{rs}^{dir} = u(o_r, d_s), \quad \forall r, s \in R, r \neq s$$

$$\Delta_{rsc} = u(d_r, c) + u(c, o_s) - u(d_r, o_s), \quad \forall r, s \in R, c \in C, r \neq s$$

$$S_{ks} \geq C_{kr} + u_{rs}^{dir} + \sum_{c \in C} \Delta_{rsc} y_{krc} + \sum_{c \in C} h_{rsc} - M(1 - a_{krs}), \quad \forall r, s \in R, c \in C, k \in K, r \neq s$$

3.4.2.7 Charging Constraints

These constraints ensure that charging decisions are consistent with task assignments:

$$\sum_{c \in C} y_{krc} \leq z_{kr}, \quad \forall k \in K, r \in R$$

$$y_{krc} \leq z_{kr}, \quad \forall k \in K, r \in R, c \in C$$

$$h_{krc} \leq M y_{krc}, \quad \forall k \in K, r \in R, c \in C$$

3.4.2.8 Battery Dynamics

The state of charge evolves based on energy consumption during travel and energy gain during charging:

$$SOC_{ks} \leq SOC_{kr} - e_{cons} \left(\sum_{c \in C} \Delta_{rsc} y_{krc} \right) + e_{gain} \sum_{c \in C} h_{krc} + M(1 - a_{krs}), \quad \forall k \in K, r \in R$$

$$SOC_{ks} \geq SOC_{kr} - e_{cons} \left(u_{rs}^{dir} + \sum_{c \in C} \Delta_{rsc} y_{krc} \right) + e_{gain} \sum_{c \in C} h_{krc} - M(1 - a_{krs}), \quad \forall k \in K, r \in R$$

$$SOC_{k0} = SOC_{init,k}, \quad \forall k \in K$$

3.4.2.9 Completion Linking

These constraints link the task completion time to the AMR that performs the task:

$$C_r \geq C_{kr} - M(1 - z_{kr}), \quad \forall k \in K, r \in R$$

$$C_r \leq C_{kr} + M(1 - z_{kr}), \quad \forall k \in K, r \in R$$

3.4.2.10 Tardiness Definition

$$T_r \geq C_r - D_r, \quad \forall r \in R$$

$$T_r \geq 0, \quad \forall r \in R$$

3.4.2.11 Binary Variables

$$a_{krs} \in \{0, 1\}, \quad \forall k \in K, r, s \in R_0, r \neq s$$

$$z_{kr} \in \{0, 1\}, \quad \forall k \in K, r \in R$$

$$y_{krc} \in \{0, 1\}, \quad \forall k \in K, r \in R, c \in C$$

3.4.2.12 Non-Negativity

$$S_{kr}, C_{kr}, C_r, h_{krc}, SOC_{kr}, u_{kr} \geq 0, \quad \forall k \in K, r \in R, c \in C$$

4. Data Collection

In this section we present the dataset used to formulate and solve the AMRs battery constrained task scheduling and routing model. The data is derived from a research paper (Dobrzańska and Dobrzański, 2025) including a test of AMR

LD90 using a DES simulator. All parameter values were selected to represent a realistic operational environment. (Dobrzańska and Dobrzański,2025)

4.1 Factory layout and nodes distances

The factory environment is shown as a network of 7 discrete nodes: a charging Station node (B) and 6 tasks nodes (N1 to N6). This design choice reflects the common real-world practice in which AMRs are stored and recharged at the same base location within the warehouse. (Dobrzańska and Dobrzański,2025) If an AMR requires recharging during the schedule, it returns to B between tasks, charges for the required duration, and then continues to the next assigned task. Charging is never triggered unnecessarily; the model only activates it when the SOC level makes it operationally required. The distances were derived from the table of the number of times the task was done in one work cycle and the total number of kilometers traveled in that task per work cycle. Then by dividing the total distance travelled for each task over the total times the task was done, we obtained the distance between each two nodes in kilometers, then we converted the values to meters, as shown in Table 3:

Table 3. Distances Between Nodes

Nodes	1•2	2•1	2•3	2•5	3•4	4•1	4•3	4•5	5•6	6•1	6•3	6•5
Distance in km	0.032	0.032	0.004	0.023	0.013	0.049	0.013	0.006	0.028	0.083	0.047	0.028
Distance in m	3.2	3.2	0.4	2.3	1.3	4.9	1.3	0.6	2.8	8.3	4.7	2.8

4.2 Derived Travel Time Matrix

Since AMRs operate at a constant speed of 1.35 m/s, travel times between nodes are derived directly from the distance matrix in Table 3. (Dobrzańska and Dobrzański,2025). Calculated travel times using the following equation: $Time = Distance / Speed$, while ensuring consistent units are shown in Table 4 :

Table 4. Travel Times

Nodes	1•2	2•1	2•3	2•5	3•4	4•1	4•3	4•5	5•6	6•1	6•3	6•5
Travel Time (s)	2.370	2.370	0.296	1.703	0.963	3.63	0.963	0.444	2.074	6.148	3.482	2.073

4.3 Task Specifications

Nine tasks are defined in the study; each one is located between two nodes within the factory layout. (Dobrzańska and Dobrzański,2025) Processing duration represents the total time an AMR must spend between traveling between two nodes and the service time needed, which was 5 seconds in the case, specifically for loading and unloading. Due time defines the deadline by which the task must be completed; any completion beyond this point contributes directly to total tardiness, which forms the primary component of the objective function. Table 5 presents the task parameters, showing the due times that are varied to produce a realistic scheduling conflict by multiplying processing duration shown by an allowable buffer, which we considered to be 20%. For on-site battery consumption, energy loss is considered during stationary task execution and derived from travel-related energy consumption (Singh et al., 2024), which is calculated through the distance and speed parameters defined in Section 4.2.

Table 5.Tasks Specification

Task ID	Origin	Destination	Processing Duration (sec)	Due Time (sec)
T1	N2	N1	7.3703	8.84436
T2	N2	N5	6.7037	8.04444
T3	N4	N3	5.963	7.1556

T4	N6	N1	11.1481	13.37772
T5	N6	N5	7.0741	8.48892
T6	N2	N3	5.2963	6.35556
T7	N4	N1	8.6296	10.35552
T8	N4	N5	5.4444	6.53328
T9	N6	N3	8.4815	10.1778

4.4 System Parameters

Charging system and AMR operational parameters are consistent with the datasheet of the AMR LD90 used in the research (Dobrzańska and Dobrzański,2025) (Table 6 and Table 7).

Table 6. Charging System Parameters

Parameter	Value	Unit
Charging Rate	0.4615	% SOC per minute
Minimum SOC Threshold	20	% SOC
Initial SOC (all AMRs)	Scenario-dependent	% SOC
Maximum SOC Capacity	80	% SOC
Charging Location	B node only	—

The charging rate of 0.4615% SOC per minute according to AMR LD-90 datasheet. The minimum SOC threshold of 20% is enforced as a hard constraint throughout the schedule to prevent unsafe robot operation. The charging process follows the constant current (CC) phase of a standard lithium-ion battery cycle that works when the battery is charging up to approximately 80%. This phase produces a linear relationship between charging time and SOC gain, which is compatible with the linear programming structure of the MILP formulation.

Table 7. AMR Operational Parameters

Parameter	Value	Unit
Number of AMRs, and Initial Location	Depending on the case	—
Speed	1.35	m/s
Load Capacity	90	kg
Availability at t = 0	Yes	—

5. Results and Discussion

The model was tested using Python/ PuLP library, using the data elaborated in section 4.

5.1 Numerical Results

The analysis begins with testing the 9 tasks elaborated in section 4, with one AMR and an initial SOC of 21%, the results demonstrated as shown in Table 8, with total tardiness of 71.58 seconds and one charging detour. Table 8 shows the output for the single-AMR scenario. The chosen arcs which represent how the AMR moves between the task nodes, while the full optimized route: 0→6→8→3→2→5→9→1→7→CHG@1→4→0, and here the "CHG@1"

means that the model added a charging detour after Task 7. with a charging duration of $h = 2.0854503$ time units. Also it reports the start time (S), completion time (C), SOC after each task, and tardiness (T) for the nine tasks, and it shows that the SOC became very close to the 20% minimum limit during the route, so the charging stop was needed, while the total tardiness reached 71.58 seconds. Then the impact of fleet size on system performance was evaluated, where two scenarios are examined using the same task set and SOC parameters.

Table 8. Run 1 Results: Single AMR (SOC = 21%), Task Route and Charging Detour

Task	Arc	S	C	SOC	T
1	0 → 6	46.1112	53.4815	20.384566	44.63714
2	6 → 8	19.4074	26.1111	20.7965	18.06666
3	8 → 3	13.1481	19.1111	20.8631	15.11955
4	3 → 2	73.810449	84.958549	20.0	71.580829
5	2 → 5	28.1852	35.2593	20.658365	26.77038
6	5 → 9	1.0	6.2963	20.956833	0.0
7	9 → 1	57.1111	65.7407	20.142835	55.38518
8	1 → 7	7.2593	12.7037	20.909967	6.17042
9	7 → 4	37.3334	45.8149	20.473364	35.6371

Scenario 2 was tested, where four AMRs were used with the same SOC limits, but instead of putting all tasks on one long route, the model distributed the tasks across the four robots, and the results were clearly better, as shown in Table 9 that shows the solver output for the four-AMR scenario, and it is arranged in the same way as Table 8 so the comparison becomes easier. The nine tasks were divided more evenly across the fleet. The charging detours became zero charging stops for all of the four robots, because the workload was distributed in a way that keeps each robot's SOC within the feasible range during the whole schedule. Also it shows that the highest tardiness was only 15.3 seconds, compared with 71.58 seconds in Table 8, which is around an 81.8% reduction. Meaning that each AMR had a more manageable workload, and we did not need any charging detours because the battery stayed within the allowed range. Overall, this comparison shows clearly that increasing the number of AMRs improves both feasibility and performance since it reduces delays, avoids interruptions, and makes the system more balanced instead of having one overloaded robot doing everything (Table 9).

Table 9. Run 2 Results: Four-AMR Fleet (SOC = 21%), Distributed Task Execution Without Charging Detours

AMR	Route	Task	S	C	SOC	T
1	0→6→3→1→0	1	13.5586	20.8889	20.804	12.0445
		3	7.2593	13.2323	20.893	6.0667
		6	1.0	6.2963	20.957	0.0
		7	8.1481	16.7777	20.724	6.4222
2	0→8→7→0	2	0.0	0.0	20.0	0.0
		8	0.0	0.0	20.0	1.1704
		7	0.0	0.0	20.0	6.4222
3	0→9→5→0	9	4.7778	13.2593	20.725	3.0815
		5	2.2580	7.7037	20.910	1.1704
		5	16.7406	23.8109	20.540	15.3260
4	0→2→4→0	2	0.0	0.0	20.0	0.0
		4	4.7778	20.9259	20.636	7.5482
		9	0.0	0.0	20.0	3.0815

5.2 Graphical Results

This section shows the graphical results for the two scenarios to analyze the system performance in a clearer way, and the comparison here is based on task tardiness under different fleet sizes while keeping the same task set and SOC parameters. Proving that using more AMR reduces tardiness as elaborated earlier. As shown in Figure 1 that presents a grouped bar chart with Task ID on the x-axis and tardiness in seconds on The y-axis. the green bars represent the 1-AMR scenario and the orange bars represent the 4-AMR scenario, the one AMR scenario has much higher tardiness

values almost for all tasks, especially in tasks like Task 4 and Task 7 where the delay is clearly very large, and this kind of behavior mainly reflects that the system is overloaded since one robot is trying to handle everything sequentially without any support. On the other hand, moving the second scenario, tardiness values drop across all tasks, and the bars become much lower, and in Task 2 the tardiness disappears completely because tasks are parallelly distributed, and tardiness is not noted for task 6 in both cases as it is the first task to be performed. Moreover, from the overall shape of the chart, we can notice that scenario one has many significant fluctuations compared to scenario two, because tasks and workload is distributed with balance, avoiding any bottleneck. So, in general the idea of increasing the number of AMRs improves the system's performance.

5.3 Proposed Improvements

5.3.1 Comparison with Rule-Based Benchmark.

To make the proposed math model work in a real-world situation, we used data from (Dobrzańska and Dobrzański, 2025). In the reference study, Discrete Event Simulation (DES) approach applied a FIFO allocation rule and a fixed charging threshold of 25%, resulting in eight charging stops making 27.79% from AMR 1 of total operating time over a 120-hour period. In contrast, running the same scenario in our proposed MILP model determines charging decisions endogenously while optimizing task sequencing, routing, and energy management, it resulted in the need for a single charging stop, representing 4.88% of total operational time, making a reduction of 22.91 percentage points compared with the DES scenario. These results highlight the limitations of fixed charging strategies, which can trigger unnecessary charging and disrupt operations. The MILP approach reduces such inefficiencies by enabling adaptive, demand-driven charging decisions.

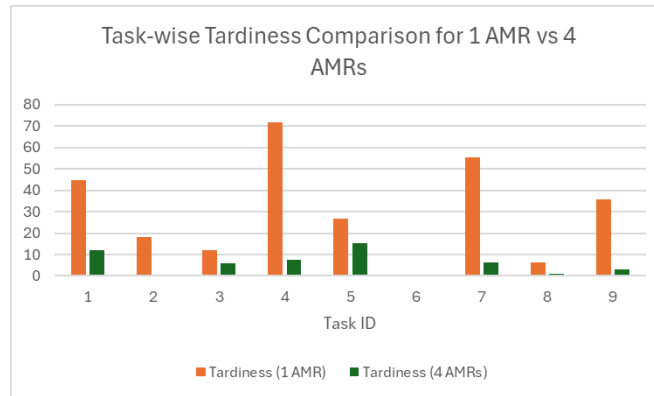


Figure 1. Tardiness Comparison: Task-wise Tardiness for 1 AMR vs 4 AMRs

5.3.2 Numerical Analysis

Charging time ratio was computed based on this equation: $Charging \% = \frac{Total\ Charging\ Time}{Makespan} * 100\%$

Results are summarized in Table 10:

Table 10. Charging Occurrence

Metric	Reference (DES)	Proposed (MILP)
Charging time (%)	27.79%	4.88%
Reduction	—	22.91 pp
Charging efficiency gain	—	5.7×

5.3.3 Graphical Analysis

Figure 2 shows a stacked bar chart titled “Operational Time Breakdown Comparison”, where the two systems, DES and MILP, are placed on the x-axis, while the y-axis shows the percentage of time distribution from 0 to 100%. Each bar is divided into two parts, the blue part showing productive time and the orange part showing charging time. For the DES system, the orange charging part takes 27.79% of the total operating time, which means only 72.21% of the time is used for productive work. On the other hand, the MILP bar shows that the charging time decreased to only

4.88%, while the productive time increased to more than 95%. So, this difference in the chart clearly shows how the fixed-threshold charging strategy used in DES is less efficient, while the proposed MILP model performs better because its charging decisions are more adaptive and based on the actual demand, as explained in section 5.3.2.

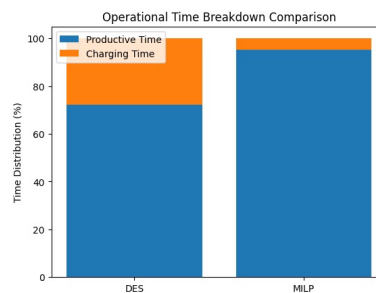


Figure 2. Productive vs Charging Time: Operational Time Breakdown (DES vs MILP)

5.5 Limitations

Although the proposed MILP framework shows a promising result, several limitations should be acknowledged. The model assumes predetermined fixed operating conditions such as task lengths, travel times, and energy usage rates, and they stay unchanged during execution. It is also assumed that charging stations are always accessible, which ignores possible congestion from several AMRs that need to charge at the same time. The model also accounts for a uniform fleet of robots with the same operational features, while actual manufacturing systems typically utilize diverse fleets. Additionally, the models computational increase in the number of tasks and robots. These suggested frameworks offer viable basis for combining scheduling, routing, and charging enhancement in manufacturing settings.

6. Conclusion

This study demonstrates that integrating scheduling, routing, and charging decisions rather than treating them independently, significantly improves AMRs operational performance. The proposed unified optimization framework incorporates battery state of charge (SOC) to enable energy-aware decisions for task scheduling and charging detours while ensuring routing feasibility using flow conservation and subtour elimination constraints.

Computational results using the Python/PuLP implementation, supported by the comparison with industrial benchmarks and case studies, demonstrated that treating the problem as an integrated system leads to more efficient and stable outcomes, "the model achieves a 22.91 percentage point reduction in charging time relative to the DES benchmark". Furthermore, deploying four AMRs instead of one reduces overall tardiness by 81.8% by adding three extra AMRs. Overall, the approach provides a more efficient and realistic basis for AMR management and supports the transition toward data-driven operations.

Acknowledgment

The authors would like to express their sincere gratitude to Dr. Hamza Adeinat for his valuable guidance, continuous support, and insightful feedback throughout this research. His mentorship and expertise were instrumental in the successful completion of this work.

References

- Barnhart, C., et al., Robotic warehousing operations: a learn-then-optimize approach to large-scale neighborhood search, *arXiv preprint*, 2024, Available: <https://doi.org/10.48550/arxiv.2408.16890>
- Čech, P., et al., Autonomous mobile robot technology for supplying assembly lines in the automotive industry, 2020, Available: <https://doi.org/10.22306/AL.V7I2.164>
- Chu, X., et al., Optimizing energy efficiency with configuration constraints for AMR trajectory planning, *Proceedings of ACC*, 2024, Available: <https://doi.org/10.23919/acc60939.2024.10644786>
- Dang, Q., et al., Modelling and scheduling autonomous mobile robot for a real-world industrial application, *IFAC Proceedings Volumes*, 2013, Available: <https://doi.org/10.3182/20130619-3-RU-3018.00261>
- Dobrzańska, M. and Dobrzański, P., *Simulation Model as an Element of Sustainable Autonomous Mobile Robot Fleet Management*, **Energies**, vol. 18, no. 8, p. 1894, 2025, <https://doi.org/10.3390/en18081894>

- Enright, J. and Wurman, P., Optimization and coordinated autonomy in mobile fulfillment systems, *Proceedings of the National Conference on Artificial Intelligence*, 2011
- Fourie, C., et al., Development of a dynamic path planning system for autonomous mobile robots using a multi-agent system approach, *Sensors*, 2025, Available: <https://doi.org/10.3390/s25175317>
- Garcia, R., et al., Autonomous robotic systems in e-commerce warehousing: a machine-learning optimization approach, 2025, Available: <https://doi.org/10.64206/ymk0e764>
- Godeke, D., et al., A simulative approach to AMR fleet sizing in decentralized multi-robot task allocation, *Proceedings of IEEE ETFA*, 2023, Available: <https://doi.org/10.1109/etfa54631.2023.10275717>
- Hu, X., et al., Task allocation optimization for warehouse autonomous mobile robot, *Proceedings of ACM Conference*, 2024, Available: <https://doi.org/10.1145/3655497.3655526>
- Huo, Y., Coordinated optimization of scheduling and path planning for delivery robots in intelligent manufacturing workshops, *Proceedings of ACM Conference*, 2025, Available: <https://doi.org/10.1145/3708597.3708601>
- Kim, B., Advancing material handling systems with autonomous mobile robots through machine learning and integrated centralized-decentralized control, 2025. Available: <https://doi.org/10.25394/pgs.28819490.v1>
- Lyu, Z., et al., Optimized multi-robot task allocation for efficient raw material supply in smart factories, *Proceedings of IEEE Conference*, 2025, Available: <https://doi.org/10.1109/raie65740.2025.11140095>
- Msala, Y., et al., A novel method for enhancing warehouse operations using heterogeneous robotic systems for autonomous pick-and-deliver tasks, *EAI Endorsed Transactions on Artificial Intelligence and Robotics*, 2025, Available: <https://doi.org/10.4108/airo.9913>
- Nikolić, M., et al., Use of a genetic algorithm to determine the optimal path for the autonomous mobile robot, 2025, Available: <https://doi.org/10.69994/12ic25038>
- Omron Corporation, *LD-60/90 Autonomous Mobile Robots*, Omron Robotics and Safety Technologies, Available: [LD-60/90 Product Page](#), Accessed: March 27, 2026.
- Ren, Y. and Friderikos, V., Interference aware path planning for mobile robots in mmWave multi cell networks, arXiv preprint, 2022. Available: <https://doi.org/10.48550/arxiv.2207.02275>
- Shen, Y., et al., Deep reinforcement learning-based multi-AMR path planning algorithm, *Proceedings of CCC*, 2023, Available: <https://doi.org/10.23919/ccc58697.2023.10240541>
- Singh, N., Akcay, A., Dang, Q.-V., Martagan, T. and Adan, I., “Dispatching AGVs with battery constraints using deep reinforcement learning,” *Computers & Industrial Engineering*, vol. 187, 109678, 2024, Available: <https://doi.org/10.1016/j.cie.2023.109678>
- Wu, D., et al., Innovative synergistic genetic algorithm and particle swarm optimization for scheduling optimization in mobile robots, *Jurnal ElektriKa*, vol. 23, no. 3, 2024, Available: <https://doi.org/10.11113/elektriKa.v23n3.573>
- Wu, H., et al., Conflict-free planning and data-driven control of large-scale nonlinear multi-robot systems, *IET Cyber-Systems and Robotics*, 2025, Available: <https://doi.org/10.1049/csy2.70027>
- Ze-kun, L., Intralogistics control and fleet management of autonomous mobile robots, 2024, Available: <https://doi.org/10.25394/pgs.25676667>

Biographies

Haboul Lida is an Industrial Engineering student at Al Hussein Technical University with interests in quality engineering, sustainable manufacturing, and cost engineering. She has contributed to multidisciplinary projects including the Formula Student electric vehicle project, PTFE extraction process optimization. Her research interests include statistical process control (SPC), life cycle assessment (LCA), carbon footprint evaluation, Lean Six Sigma, and manufacturing control, with a focus on applying data-driven approaches to Industry 4.0 environments.

Al-Attar Bayan is an Industrial Engineering student at Al Hussein Technical University. She is interested in optimization, quality management, AI, and robotics. Her work mainly focuses on applying mathematical modeling and operations research to real world challenges, particularly in logistics and manufacturing systems. She is also involved in professional and technical initiatives through IEEE robotics and Automation Society activities, where she contributes to promoting innovation and hands-on engineering experience.

Abu Al Hummos Lujain is a student of Industrial Engineering at Al Hussein Technical University in Amman, Jordan. She is interested in operations management, quality improvement, process optimization through data-driven decision-making and specializes with an emphasis on raising efficiency and better quality in industrial and service sectors. Her research interests include risk management and quality improvement, and she is interested in exploring the root causes of inefficiencies and implementing sustainable solutions.

Al Naimat Muna is an Industrial Engineering student at Al Hussein Technical University. Her interests include operations research, data analytics, manufacturing systems, and project management. She has worked on projects involving predictive maintenance, smart safety solutions, and automated product design. Her research focuses on optimization, mathematical modeling, and Industry 4.0 applications for improving industrial system performance and decision-making.