

# Dynamic task allocation strategies for AGVs in smart warehouse environments

Nathania Rachael<sup>1</sup>, Allen Reji<sup>1</sup>, Nidhish Balasubramanya<sup>1</sup>, and Prem Sankar<sup>2\*</sup>

<sup>1</sup>School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India

<sup>2</sup>Centre for Healthcare Advancement, Innovation and Research, Vellore Institute of Technology, Chennai, India

**Abstract.** The implementation of dynamic task assignment between AGVs is an important step in enhancing the overall efficiency of AGVs in a smart warehouse environment. However, existing zone-based systems limit flexibility in terms of AGVs assigned to zones, thus resulting in workload imbalance and increased AGV idle time. To mitigate these drawbacks of existing zone-based systems, a centralized proximity-based dynamic task allocation framework is proposed using CoppeliaSim for dynamic assignment of tasks to AGVs based on the Euclidean distance between each AGV and task location. Such a system avoids the assignment of AGVs to zones and provides adaptability in terms of redistribution of workload between AGVs. The experimental results show an increase in overall average utilization of AGVs by 118%, a reduction in overall idle time by 78.7%, and a reduction in variance of task completion time by 59.8% when compared to static allocation systems. Additionally, overall task completion time is improved by 3.5% when using dynamic allocation compared to static allocation systems. Hence, overall consistency in terms of system stability is achieved. The results of the paper prove that using a lightweight proximity-based system for AGV allocation in a warehouse is not only effective but also scalable.

## 1 Introduction

The intelligent robotic systems have the ability to revolutionize the field of warehouse automation. In addition, in the contemporary setting of the warehouse, the AGVs have the ability to play an essential role in the execution of numerous different tasks that are considered tedious and require a lot of labour, such as material handling, order picking, and internal transport operations. This is the case when the traditional AGV scheduling and task allocation strategies have usually relied on static area-based or zone-based allocation, where each AGV was considered capable of operating within the allocated regions. However, the static allocation mechanism was not considered capable of coping with the dynamic changes in the warehouse, such as the varying positions of the packages, the priorities of the tasks, or the operation constraints.

---

\* Corresponding author: [premsankar.n@vit.ac.in](mailto:premsankar.n@vit.ac.in)

To deal with these disadvantages, an adaptive dynamic task allocation framework for AGVs in Coppelia Sim has been proposed, wherein the dynamic allocation of tasks for AGVs is performed on the basis of the calculation of Euclidean proximities between each AGV and the available packages in real-time. This framework monitors the locations of the AGVs and the changes in the tasks in real-time and allocates the AGVs to optimize the travel time, workload, and responsiveness of the AGVs to the changing warehouse environment. As such, flexibility, efficiency, and scalability, which are essential ingredients for next-generation warehouse operations, are greatly enhanced [1,2].

The need for the development of such an adaptive approach can be justified by the inflexibility of the traditional approach in scheduling the control of AGVs. In a dynamic industrial setting, the tasks are often changing with respect to their location and urgency, and the availability of the AGVs may also be changing due to charging and maintenance downtimes [7]. The traditional approach is not capable of handling such changes in real time, and therefore the efficiency of the system suffers. Such difficulties have also been identified in the recent literature, where the coordination of multiple AGVs and dynamic scheduling were found to be important for the efficiency of logistics systems [9].

In this regard, it is particularly unique that the proposed framework focuses on real-time proximity-based adaptive task allocation in Coppelia Sim. The system determines and dynamically computes the distances between the available idle AGVs and the available packages, and the model maximizes the utilization of the AGV fleet. This ensures a more responsive and flexible AGV task allocation system compared to the traditional static systems. The proposed model developed in this work provides a good foundation for adaptive AGV scheduling in simulated warehouse environments and is in line with the current research focus on intelligent real-time coordination of automated logistics systems [3,9,11].

## 2 Literature Review

With the increasing demand for higher throughput and lower labour-intensity, Automated Guided Vehicles (AGVs) have become essential to modern warehousing and factory operations, as they are able to navigate independently to deliver products. AGV task delegation has traditionally been performed through static or zone-based allocation techniques, where the AGVs are limited to a particular zone or a fixed path of movement. It is easier to manage and control the AGVs through conventional techniques, but such techniques do not lend themselves well to adapting to change or to dynamic workloads. Moreover, static allocation techniques do not always lend themselves well to balancing the workload among the AGVs, thus causing some to be underutilized and others to be overutilized. If the workload changes dynamically, it also creates delay due to congestion. At best, static allocation introduces a 40% cost in travel time and resource usage in comparative dynamic models.

In multi-AGV systems, the limitations of traditional systems are intensified because coordination of tasks and avoidance of collisions are critical. Centralized control methods are better for achieving global knowledge of task priorities, but they lack flexibility and tolerance to faults. Decentralized methods are adaptable to varying operational conditions, but have increased communication and synchronization-related complications [4]. Moreover, static scheduling systems require configuring paths for travel, delivery locations, and task priorities ahead of time, thus compromising the systems' ability to react to real-time changes in the environment to include additional packages being dropped off, operational AGVs breaking down, or paths being blocked [10]. Thus, these issues demonstrate the need for adaptive, dynamic systems for task allocation that can respond thoughtfully to variable operational conditions.

Recently, the focus of research has expanded to adaptive and intelligent strategies for task assignment, in order to address their shortcomings. Reinforcement learning (RL)-based scheduling approaches support autonomous AGVs to learn the best task choice and routing decisions as they interact with the environment on an ongoing basis [2]. Models that combine RL with heuristic or graph-based algorithms, such as Dijkstra's algorithm and Monte Carlo Tree Search, have improved responsiveness and decision quality [4]. These are good solutions but may require a lot of computational power and training time, which could limit their use for real-time functionality in a large-scale industrial setting.

However, optimization-based scheduling techniques have also been explored in terms of their trade-off between computation time and accuracy. Heuristic and meta-heuristic techniques such as genetic algorithms, tabu search, and simulated annealing have been applied to achieve optimal solution values in a small amount of computation time [2,8]. In addition, efforts have been made to combine heuristic-based dynamic scheduling techniques with discrete event simulation to generate a realistic task sequence for better performance in a stochastic warehouse environment [8]. The advent of digital twin technology has also led to plans, such as AMODS (AGVs Multi-Objective Dynamic Scheduling), that make use of real-time updates to dynamically change the number of active AGVs, the priority of their tasks, and their travel speeds, resulting in increased responsiveness and throughput of the system [9].

Adaptive coordination models have reached human-robot collaborative environments as well, such as with Neural Approximate Dynamic Programming (NeurADP) frameworks which enable AGVs to jointly perform order picking, battery management and task sequencing with human workers [6]. These joint models augment the depth of decision-making and minimize the overall time it takes to fulfil orders in such dynamic warehouse environments.

In the most recent studies, dynamic task allocation has been examined as it relates to multi-AGV warehouses that can have an excess of tasks in relation to the available AGV units. Xin et al. examined the dynamic allocation of unbalanced transportation task load using a rolling horizon strategy by modelling the task assignment problem to be akin to a dynamic vehicle routing problem with pickup and delivery (VRPPD). In their research, they have proposed a hybrid metaheuristic solution that uses the adaptive large neighbourhood search (ALNS) to create a composite/task sequence that matches to an AGV based on the Kuhn-Munkres (KM) algorithm. The ALNS component produces the candidate task sequences while the KM component produces the optimal match for task pairs to AGVs. Experimentation of this composite methodology/approach suggests it will outperform conventional heuristic solutions and will provide for scalable, real-time task allocation among large multi-AGV systems [12].

While learning and optimization methods are advanced, distance and proximity are still among the most practical and effective strategies for a real-world deployment in real time. The overall idea is that the AGV that is nearest to the task will perform the task, hence reducing the idle time in the process and the distance travelled. The old nearest-neighbour algorithms were not very capable of global optimization, but more recent versions have introduced dynamic reassignment to the neighbourhood, hybrid optimization, and multi-objective heuristics that have resulted in improvements in path responsiveness and efficiency. Simulation studies using simulation platforms such as Coppelia Sim and ROS have shown that proximity-driven task assignments can result in substantial performance improvements through the integration of path optimization and time-tagging techniques [3,5].

In addition, some simulation studies have also incorporated the computation of proximity with other optimization techniques, such as the Hungarian algorithm and the

Traveling Salesman Problem (TSP), with the aim of obtaining trade-offs between computation time and global optimality. These techniques also account for the heterogeneity of the AGVs, provide information regarding unassigned task distribution, as well as congestion avoidance, resulting in significantly lower travel distance and time for the completion of tasks. Simulation platforms that utilize centralized algorithms for the resolution of conflicts have also validated the effectiveness of the use of proximity-based task assignment with the inclusion of logic for the avoidance of collisions in the resolution of deadlocks.

Apart from the scheduling algorithms, the simulation studies carried out today for the validation of the proposed framework for the allocation of tasks for remote applications are of critical importance. Technologies such as SLAM, Adaptive Monte Carlo Localization, etc., enable the accuracy of the location of the AGV with the aim of simulating the dynamic scheduling policies. In this context, the simulated environment can also provide an efficient means for experimentation with the parameters that are critical for the development of an efficient AGV coordination system.

The literature reviewed indicates a clear transition from static and zoned-based systems of scheduling to an adaptive, proximity-based, and optimal-support system of scheduling allowing flexible operations and effective use of resources and real-time adaptation. Proximity-based dynamic allocation is a suitable compromise between computational complexity and efficiency, and is an ideal foundation for the framework developed in this paper which supports real-time allocation of tasks based on Euclidean distance in Coppelia Sim to reduce idle time and improve throughput.

## **3 Problem Statement and Methodology**

### **3.1 Limitations of Existing Methods**

The traditional operational systems have a number of well-documented inefficiencies in terms of operational responsiveness, particularly for those systems that are based on static zone assignments. These include:

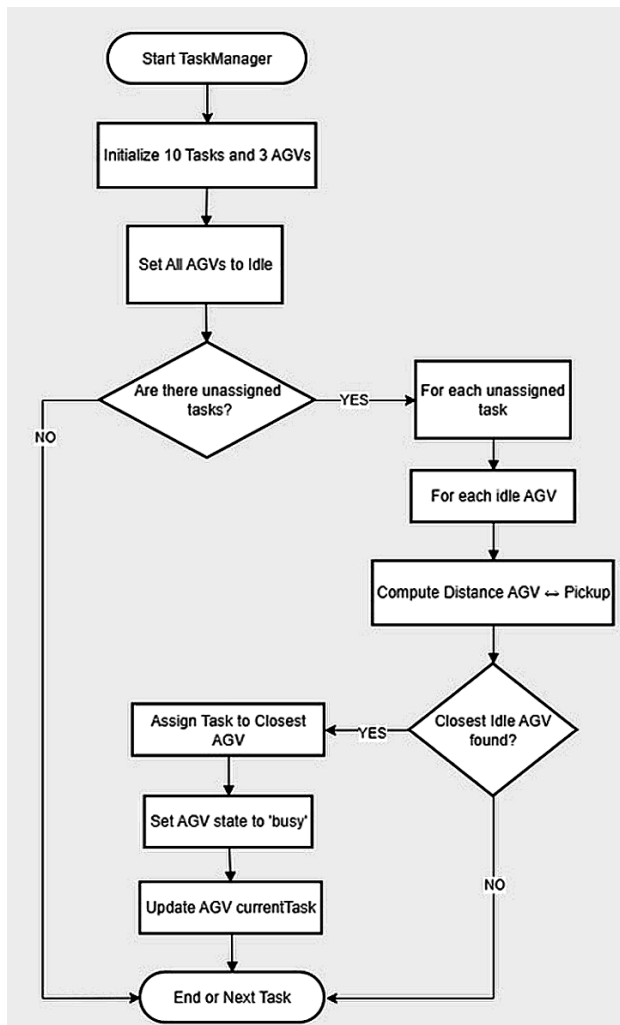
- **Workload Imbalance:** The static nature of the zones leads to a situation where the workload is not balanced. This leads to a critical operational bottleneck in the high-demand zones, while at the same time, there are zones that are not operating at their full potential.
- **Operational Inflexibility:** The nature of these systems is such that they are not adaptable to the dynamic nature of the operational environment. They are not capable of responding to real-time factors such as sudden spikes in demand, injections of priority messages, or even sudden equipment failures.
- **Suboptimal travel paths:** Rigid zone boundaries introduce artificial constraints to movement that are often associated with inefficient travel patterns. The effect is increased nonproductive travel time since operators or automated systems are not allowed to service proximal tasks that fall just outside their boundaries.
- **Constraints to Scalability:** Many of the inefficiencies in static zoning models become magnified as a system grows.

## 3.2 Proposed New Dynamic Allocation Framework

### 3.2.1 System Architecture

The overall system consists of a centralized controller that has real-time knowledge of the position of all the AGVs, as well as what available tasks are present, and consists of:

- A task status monitor that listens for the detection of new packages that require transport
- AGV status tracker to keep a record of position and available AGVs
- Distance calculator to calculate all of the Euclidean distances
- Assignment engine that will optimize the assignments based on proximity
- Coordination module that will allow for conflict-free assignments of the tasks to AGVs



**Fig. 1.** Centralized Dynamic Task Allocation Workflow showing task detection, real-time AGV status monitoring, Euclidean distance computation, optimal assignment engine, and coordination module ensuring conflict-free execution.

The centralized decision workflow is depicted in Fig. 1, which shows the process of creating tasks from task discovery through task reassignment of an AGV and updating the state of the system. Continuous monitoring, distance calculations in real time, and locking of tasks are methods that prevent two AGVs from working on the same task. A modular design allows the system to be scalable with fleet size and efficient in computing.

### 3.2.2 Euclidean Distance Algorithm

The basic core of the algorithm will minimize the total travel distance as best as possible, paired AGVs to assessed tasks:

- a) Task detection – define the transport needs
- b) Distance Matrix computation – calculate the vector distance to all AGVs to all tasks
- c) Optimal assignment – use the AGV-task pairs with the minimum distance
- d) Lock the task into the AGV to prevent any assignments
- e) Generate the route to the assigned task
- f) Monitor completion and update the system's state to complete issues

### 3.2.3 Problem Formulation

The task allocation problem is formulated as a distance-minimization optimization problem under operational constraints. Let  $A = \{a_1, a_2, \dots, a_n\}$  represent the set of available AGVs and  $T = \{t_1, t_2, \dots, t_m\}$  represent the set of transport tasks.

The objective function is defined as:

$$\min \sum_{i=1}^n \sum_{j=1}^m x_{ij} d_{ij}$$

where  $d_{ij}$  denotes the Euclidean distance between AGV  $a_i$  and task  $t_j$  and  $x_{ij} \in \{0,1\}$  represents the assignment decision variable.

Subject to:

1. Each task is assigned to exactly one AGV.
2. Each AGV can execute only one task at a time.
3. Assignments occur only for idle AGVs.

Evaluation metrics include AGV utilization percentage, idle time reduction, average task completion time, and task time variance. These metrics are used to compare static and dynamic scheduling performance.

## 3.3 Operational Sequence of AGVs

Every AGV has a set operational cycle that must follow a specific sequence of steps for it to operate correctly. Initially, all AGVs will remain in the idle or docking state until a task has been created. Once an AGV has been requested to transport something, the central controller uses a distance matrix to calculate the Euclidean distances between all idle AGVs and available tasks. The assignment engine will use this information to select which AGV and task combination has the least distance and will then lock the task to prevent duplicates.

Once an AGV has been assigned to do a task, it will go from being idle to being active,

creating a navigation path to the first pick-up location. Once at the pick-up location, the AGV will simulate that it has picked up an object based on a load confirmation and will then transport this item to the delivery site. Once it has confirmed the item was delivered, the AGV will update its status and will then be eligible for re-assignment. The closed-loop process allows for real-time responsiveness while also ensuring that no duplicate tasks are assigned.

The risk of mid-task switching is avoided when re-assigning AGVs because each AGV will only be re-assigned once they have completed the current task they are doing and a task cannot be assigned to an AGV until it is no longer assigned to another task. The coordination of assignments through the coordination module also prevents two AGVs from being assigned to the same task and allows for continuous synchronization to occur within the whole fleet of AGVs.

## **4 Implementation in CoppeliaSim**

### **4.1 Simulation Environment**

Coppelia Sim was used to create the warehouse environment to closely resemble a real-world industry context. The virtual design of the warehouse contains multi-tiered storage racks, AGV pathways, and loading/unloading areas placed in a designated manner. Each AGV is virtually implemented as a mobile robot with proximity sensors, odometry, and navigation systems to facilitate movement and location in the warehouse space.

To simulate dynamic activity in the optimal environment, a location for package delivery is randomly established throughout the warehouse floor, revealing uncertainty for actual order and dispatch tasks. There is a docking station for the AGVs so they can charge or park idle, allowing time to deliver packages throughout the simulation.

### **4.2 AGV Types and Operational Characteristics**

All AGVs used in this study are homogeneous differential-drive mobile robots modelled in CoppeliaSim. Each AGV contains a proximity sensor (for detecting obstacles) and simulated odometry (for localizing). Each AGV follows waypoint-based navigation to move from the pickup location to the drop-off location. However, the identical speed, acceleration, and capacity imposed on the AGVs was to eliminate any effect of hardware differences on the outcomes of experiments.

Using homogeneous AGVs ensures any differences in performance is due to scheduling logic only, and not mechanical heterogeneity. AGVs with a differential-drive design have the capability to perform in narrow aisles of warehouses and allow for smooth planning of trajectories. The use of AGVs that operate similarly to actual industrial AGVs in a structured warehouse, provides a more realistic and scalable simulation for evaluating outcomes.

### **4.3 Software**

The system proposed uses Coppelia Sim as the main simulation environment, coupled with external scripting via Lua for hosting control logic and task assignment algorithms. The software tracks the positions and states of all AGVs in the simulated world at all times and calculates instantaneous distance metrics between robots and target locations. It then assigns and optimizes tasks dynamically based on these inputs to reduce idle time and travel distance.

## 4.4 Implementation

Fig. 2 and Fig. 3 given below depict the AGV warehouse simulation done on Coppelia Sim



**Fig. 2.** CoppeliaSim warehouse simulation environment illustrating multi-tier storage racks, structured AGV pathways, docking stations, and dynamically generated package locations used for evaluating static and dynamic task allocation strategies.



**Fig. 3.** Real-time execution snapshot of AGVs operating under the proposed dynamic allocation framework, demonstrating task reassignment, navigation trajectories, and proximity-based coordination within the warehouse layout.

## 5 Experimental Evaluation

### 5.1 Results and Analysis

The simulation experimental findings from Coppelia Sim demonstrate the performance enhancement of the proposed Dynamic Task Allocation Framework, compared to the static assignment. In order to provide an appropriate fair comparison of performance, both configurations were tested on the same warehouse layout, the same AGV fleet, and comparable task loads for both AGV systems.

#### 5.1.1 AGV Utilization Evaluation

Table. 1 below and Fig. 4 below show the utilization metrics of the AGVs, both for static allocation and dynamic allocation. Utilization is calculated as the busy time divided by the total time the AGV is operational, while idle time is the time when the AGV is not assigned to a delivery, or waiting time when it is assigned to a delivery but not in motion.

Under static allocation, the AGVs have shown uneven workload distribution where AGV\_2 remains moderately active with a utilization of 55.46%, while AGV\_1 and AGV\_3 have considerably lower utilizations of 20.89% and 15.30%, respectively. This imbalance characterizes zone- based or pre-defined routing systems, whereby some vehicles remain underutilized and others are overloaded.

**Table 1.** Comparison of AGV utilization between static and dynamic task allocation.

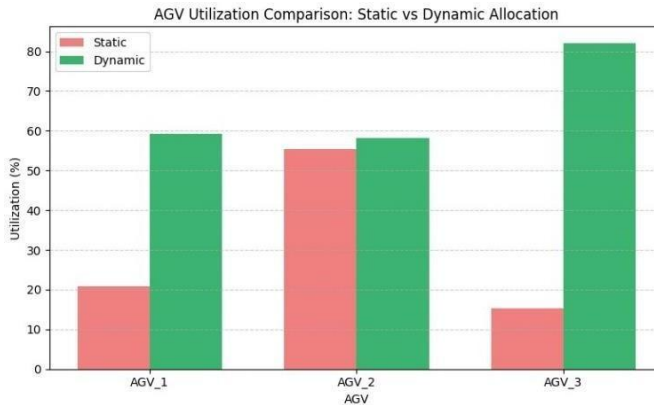
AGV	Idle Time (s)	Busy Time (s)	Utilization (%)	Idle Time (s)	Busy Time (s)	Utilization (%)	Improvement
	Static			Dynamic			$\Delta$ Utilization
AGV_1	817	215.75	20.89	186.5	271	59.23	183.60%
AGV_2	460	572.75	55.46	191	266.5	58.25	5.00%
AGV_3	874.75	158	15.3	81.75	375.75	82.13	437.00%
Average	717.25	315.5	30.55	153.08	304.4	66.54	+118% overall

In contrast, dynamic task allocation significantly improved resource utilization across the fleet. The proposed framework enabled near real-time reassignment based on Euclidean proximity, resulting in utilization gains across all AGVs. The proposed framework allows for almost real-time reassignment according to Euclidean proximity, which in turn brings about utilization gains for all AGVs. AGV\_1 increased from 20.89%  $\rightarrow$  59.23% (+183.6%). AGV\_2 had a slight increase, 55.46%  $\rightarrow$  58.25% (+5%). AGV\_3 saw the highest growth from 15.3%  $\rightarrow$  82.13% (+437%).

The improvement in efficiency on AGV\_1 and AGV\_3 is largely due to removing the rigid zone requirements found in traditional static allocations. AGV\_1 and AGV\_3 suffered from task starvation because of the use of static scheduling and the unequal distribution of tasks across the different zones. The new method of dynamically revising proximity as a basis for assigning idle AGV's to available tasks have made a huge difference in the use of AGV\_1 and AGV\_3. The overall magnitude of improvement of AGV\_3 (+437%) was because of its very low starting utilization in the static case. The difference in utilization between static and dynamic allocation provided a clear method to eliminate the imbalance that created the resource wastage.

The average utilization rose from 30.55% for the static assignment to 66.54% for the dynamic one, which shows an increase of approximately 118%. A more even distribution of

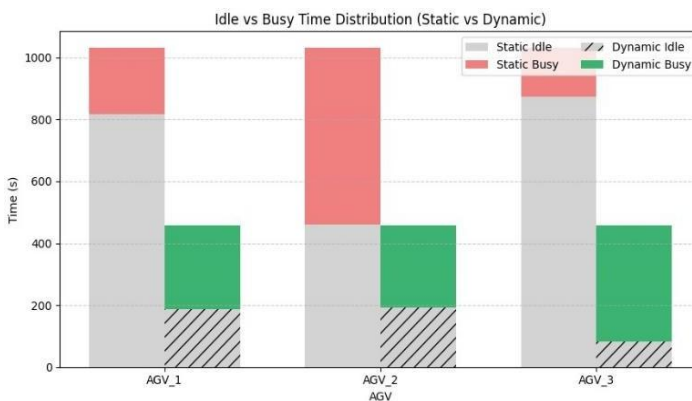
workload among the AGVs is visually confirmed by the bar plot of Fig. 4 for the dynamic case.



**Fig. 4.** Comparative AGV utilization percentages under static zone-based allocation and dynamic Euclidean proximity-based allocation, highlighting workload balancing improvements across AGV\_1, AGV\_2, and AGV\_3.

### 5.1.2 Idle vs Busy Time Distribution

The distribution of idle and busy time in Fig. 5 can assist in quantifying AGV utilization. In static allocation group, most AGVs had a majority of their operational cycle in idle time, particularly AGV\_1 and AGV\_3, which indicates poor load balancing and service. However, the idle time of AGVs in the dynamic system reduced considerably, while their busy time was distributed in a balanced manner. The incorporation of proximity-based assignment of tasks in real-time prevented AGVs in idle state from staying in idle state for a long time.



**Fig. 5.** Idle versus busy time distribution for each AGV under static and dynamic scheduling, illustrating significant idle time reduction and improved operational consistency in the proposed framework.

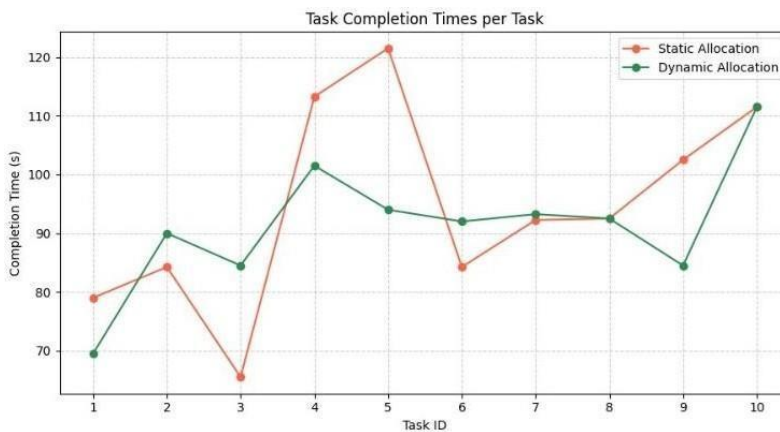
### 5.1.3 Comparison of Task Completion Times

The impact of dynamic scheduling in this study is summarized in Table. 2 and visualized in the chart shown in Fig. 6. Both conditions executed an equal number of 10 transport tasks, making them directly comparable.

**Table 2.** Task completion time comparison

Metric	Static Allocation	Dynamic Allocation	% Improvement
Total Tasks Completed	10	10	–
Average Task Completion Time (s)	94.65	91.33	↓ 3.5%
Fastest Task (s)	65.5	69.5	–6.1%
Slowest Task (s)	121.5	111.5	↓ 8.2%
Task Time Variance	32.8	13.2	↓ 59.8%

While a 3.5% average improvement in completion time seems like a small number, the remaining 59.8% reduction in task time variability is an important performance improvement. Less variability means that there will be less congestion, better coordination of AGVs and fewer delays caused by localized imbalances in workload. In real-world warehouses, consistency and predictability are often much more valuable than small improvements to the mean completion times since they have a direct impact on the ability to meet service level agreements and reliably achieve throughput.



**Fig. 6.** Task-wise completion times for ten transport operations under static and dynamic allocation strategies, showing reduced variance and improved consistency in dynamic scheduling.

The trend line for the dynamic condition shown in Fig. 6 is much more continuous with the performance compared to the static allocation condition, which is more impacted by fluctuations or peaks in delays/residence time, representing evident congestion or wait time in the system.

This is an indication of improved synchronization of the AGVs, reduced bottlenecks, and more consistent task cycles, which is essential in an industrial real-time logistics environment.

### 5.1.4 General Performance of the System

In conclusion, the results confirm that the Dynamic Euclidean Distance-Based Allocation:

- Increased system utilization by 118%
- Reduced task completion variability by 59.8%
- Averaged a reduction of idle time among the AGV system by 78.7%

The improved results above portend even better results for a proximity-based adaptive scheduling approach over a purely static or regionally circumscribed structure. The proximity-based scheduling approach provides for further responsiveness to achieve improved throughput and flexibility, particularly in the case of non-synchronous arrival tasks and/or dynamic changes in the warehouse environment.

## 5.2 Summary of Findings

Dynamic allocation helps in improved collaboration, adaptability of task allocation, and equitable distribution of loads to the AGV system. The system also leads to reduced idle time, seamless transitions between tasks, and timely utilization of time, making it a scalable and computationally viable method for real-time warehouse automation.

## 5.3 Applications

The proposed dynamic task allocation framework can be applied in different aspects of industry.

- **Smart Warehouses:** Offers the possibility of real-time allocation of tasks; enabling better coordinated product movement with each AGV spending less time waiting (idle) which enhances efficiency.
- **Distribution Centers:** Apply this framework to adjust package handling and dispatching activities in real-time to suit changing demand patterns.
- **Manufacturing Plants:** Can use this framework regarding the transport of materials dynamically between workstations; responding seamlessly for just-in-time production to occur.
- **E-commerce Facilities:** Aiding to make order-picking and deliveries faster through efficient task allocation of the different number of orders and potentially varying load conditions.

## 6 Discussion

A combination of three major components are responsible for the observed performance improvements in the dynamic allocation mechanism of the AGV system; they include: (1) removing spatial constraints, (2) redistributing real-time workloads, and (3) reducing idle dispersion. During a static zone-based system, AGVs (automated guided vehicles) are artificially restricted from servicing the nearest tasks they reach, thereby creating congestion and clustering of AGVs around their respective zones; when the proposed Euclidean proximity-based mechanism was implemented, it eliminated these additional spatial restrictions and permitted the AGVs to efficiently capture and service the tasks closest to them based on their spatial locations.

In addition, the reassignment of AGVs improved the fleet-based entropy reduction by distributing the transport workload evenly among AGVs given their relative locations and the transportation needs. This equalized distribution of transport responsibilities helped in

eliminating bottlenecks due to resource deprivation and distributing the transport workload to all available AGVs based on proximity.

The time variance between tasks completed by AGVs significantly decreased following the implementation of the proposed framework due to a more synchronized operation amongst AGVs. This synchronization is critical in producing predictable throughput in the industrial logistics environment. The performance improvements that resulted from the proposed computationally efficient distance-based calculations are also attributable to the fact that the proposed framework could be deployed in real-time and was relatively easy to implement compared to the existing excessive computational optimization of the previous optimization-heavy alternative systems; as such, the proposed framework is scalable.

## 6.1 Benefits

- Improved Responsiveness: Immediate adjustment to changing conditions.
- Flexibility Increase: Removal of zone constraints allows for full resource utilization.
- Improved Computational Efficiency: Use of a lightweight technique enables real time application.
- Scalability: Linear scalability with fleet size.

## 6.2 Limitations and Future Enhancements

There are limitations to this research such as limitations in planning paths, potential for congestion/traffic, no consideration for battery, and therefore no consideration for emergency handling. For future improvements there should be an incorporation of advanced path planning techniques, avoidance of traffic, usage of battery aware scheduling, usage of priority-based assignment, and incorporation of predicted analysis.

## 7 Conclusion

The effectiveness of the dynamic proximity task allocation in the warehouse AGV operation has been successfully proved in the current research. The implementation of the Coppelia Sim indicates a considerable advantage while proving its efficiency compared to the static approach, which results in a more efficient use of human and mechanical resources. The computational trialability of the system allows possible implementation in the real world, and the specificities of the industry are an important step in the development of more complex optimization algorithms.

Future research will focus on addressing the current limitations with updates to the path planning algorithms, machine learning-based predictions of AGV idle times, and multi-objective optimization frameworks quantifying efficiencies, reliability and operational constraints.

## References

1. Z. Zhang, Q. Guo, J. Chen and P. Yuan, *Collision-Free Route Planning for Multiple AGVs in an Automated Warehouse Based on Collision Classification*, IEEE Access, **6**, 26022–26035 (2018). <https://doi.org/10.1109/ACCESS.2018.2819199>
2. R. Xu, H. Feng, J. Liu and W. Hong, *Dynamic Spare Point Application Based Coordination Strategy for Multi-AGV Systems in a WIP Warehouse Environment*,

- IEEE Access, **10**, 80249–80263 (2022).  
<https://doi.org/10.1109/ACCESS.2022.3195173>
3. P. Padalkar, P. Kadam, S. Mirajgave and A. Mohite, *Industrial Warehouse Robot Simulation Using ROS*, in: *Advances in Intelligent Systems and Computing* (Springer), 2023. [https://doi.org/10.1007/978-3-031-22485-0\\_9](https://doi.org/10.1007/978-3-031-22485-0_9)
  4. Z. Zhang, J. Chen and Q. Guo, *Application of Automated Guided Vehicles in Smart Automated Warehouse Systems: A Survey*, *Computer Modeling in Engineering & Sciences*, **134**, 1–35 (2022). <https://doi.org/10.32604/cmes.2022.021451>
  5. J. López, E. Zalama and J. Gómez-García-Bermejo, *A Simulation and Control Framework for AGV-Based Transport Systems*, *Simulation Modelling Practice and Theory*, **116**, 102430 (2021). <https://doi.org/10.1016/j.simpat.2021.102430>
  6. A. Dehghan, M. Cevik and M. Bodur, *Dynamic Task Allocation in Intelligent Warehouses with Hybrid Workforce of Automated Guided Vehicles and Human Pickers*, *ACM Trans. Evolutionary Learning and Optimization* (2025).  
<https://doi.org/10.1145/3716862>
  7. W. Y. Akele and G. Huang-Yang, *Study on Motion Tracking Control System for AGV*, *International Journal of Engineering Research & Technology (IJERT)*, **9**(9) (2020).
  8. J. F. Leon, M. Peyman, X. A. Martin and J. A. Juan, *Simulation of Heuristics for Automated Guided Vehicle Task Sequencing with Resource Sharing and Dynamic Queues*, *Mathematics*, **12**, 271 (2024). <https://doi.org/10.3390/math12020271>
  9. S. Wu, W. Xiang, W. Li, L. Chen and C. Wu, *Dynamic Scheduling and Optimization of AGV in Factory Logistics Systems Based on Digital Twin*, *Applied Sciences*, **13**, 1762 (2023). <https://doi.org/10.3390/app13031762>
  10. N. Correia, L. Teixeira and A. L. Ramos, *Implementing an AGV System to Transport Finished Goods to the Warehouse*, *Advances in Science, Technology and Engineering Systems Journal*, **5**, 241–247 (2020).
  11. *A Novel Method for Enhancing Warehouse Operations Using Heterogeneous Robotic Systems for Autonomous Pick-and-Deliver Tasks*, *EAI Endorsed Transactions on AI and Robotics*, **4** (2025). <https://doi.org/10.4108/airo.9913>
  12. J. Xin, Q. Yuan, A. D’Ariano, G. Guo, Y. Liu and Y. Zhou, *Dynamic Unbalanced Task Allocation of Warehouse AGVs Using Integrated Adaptive Large Neighborhood Search and Kuhn–Munkres Algorithm*, *Computers & Industrial Engineering*, **195**, 110410 (2024). <https://doi.org/10.1016/j.cie.2024.110410>