Enhancing Dynamic Pickup and Delivery Systems with Advanced Task Switching: An Auction-Based Approach in Industry 4.0

Rosa Abate*, Guido Guizzi*

**Università degli Studi di Napoli "Federico II", Dipartimento di Ingegneria Chimica, dei Materiali e della Produzione Industriale, P.le Tecchio, 80, 80125, Napoli, ITALY*

(e-mail: rosa.abate@unina.it, g.guizzi@unina.it)

Abstract: In this paper, we address the challenges of dynamic pickup and delivery problems (PDPs) within the framework of Industry 4.0 by proposing an approach centered around dynamic auctions. This method employs auctions to assign tasks efficiently among Automated Guided Vehicles (AGVs), ensuring optimal workload distribution and enhanced system performance. The model introduces a detailed exploration of task switching logic, allowing for real-time task insertion and adjustment during ongoing auctions. This approach assesses the impact of dynamic task reallocation on the AGVs' existing work schedules, aiming to minimize the overall make-span and distance travelled while maximizing operational efficiency. Initial findings suggest that this auction-oriented and multi-agent simulation method improves system performance, providing a viable solution to the dynamic and intricate PDP issues in the rapidly changing environment of Industry 4.0. Our study contributes to the field by demonstrating how advanced task management and optimization techniques can lead to smarter, more efficient industrial operations.

Keywords: Pickup and Delivery Problem, Auction, Task Switching, Internal Logistics, Routing Problem

1. Introduction

In the contemporary production context, there is a growing need to respond quickly to changes, operate efficiently and accurately, and ensure the flexibility of the system. Adopting innovative solutions for material handling and dynamic task planning becomes a critical strategic lever to meet these new challenges of the contemporary market *(Pereira et al., 2017)*. In realizing an industrial environment that reflects the Industry 4.0 vision, material handling operations can be managed and optimized using various innovative technologies *(De Martino et al., 2023)*. Among these technologies autonomous guided vehicles (AGVs) are the most popular. These vehicles, as described by *Rocha et al. (2010)*, operate autonomously and use V2V (vehicle-to-vehicle) and V2I (vehicle-to-infrastructure) communication technologies to coordinate and operate safely. This not only minimizes errors made during the movement of materials but also ensures high levels of performance in terms of precision, safety, and speed *(Barreto et al., 2017; Holubčík et al., 2021)*. In areas such as the dynamic management of AGVs, the traditional problem-solving approach might not always be the most effective. Instead of continually seeking an optimal solution through optimization algorithms every time new challenges or changes arise, it is often more practical and responsive to adopt strategies that provide "good enough" solutions quickly. This concept is reflected in the "satisficing" theory of *Simon (1955)*. Similarly, *Assunta et al. (2017)*, demonstrate how the integration between humans and Cyber-Physical Systems, human-CPS interactions, can affect operational decisions in dynamic environments, suggesting that the adaptive responses of human operators can significantly enhance system flexibility and responsiveness, a crucial aspect in the decentralized management of AGV systems.

In this paper, the focus will be on the efficient implementation of AGVs, and on auction theory as a tool to decentralize operational decisions. The goal is to develop a flexible and scalable AGV fleet management model, capable of adapting to dynamic internal logistics changes and managing specific pickup and delivery requests, each of which has a unique origin and destination, improving main KPIs such as throughput, makespan, and distributed workload on AGVs, in line with current trends in Industry 4.0 *(Pfeiffer, 2019; Chen et al., 2019)*.

The remainder of the paper is set out as follows: Section 2 presents a comprehensive literature review of the state of the art; Section 3 outlines the proposed approach; and finally, Section 4 concludes the work.

2. State Of The Art

The Pickup and Delivery Problems (PDPs) represent a specific category of Vehicle Routing Problems (VRP). In this type of problems, a fleet of vehicles is tasked with picking up and delivering goods or people to specific locations. There are several versions of the problem, anyways they can be divided into two main categories *(Parragh et al., 2008)*. The first is where pickup and delivery points are "unpaired". In this scenario, goods collected from one point can be delivered to any other delivery point. The second category is that of "paired" problems. In this context, each transportation request has a specific origin and destination. This implies a direct route between the pickup point and the delivery point, without interruptions. In the case of paired pickup and delivery locations, no other client can be visited between a pickup and its associated delivery location. The PDP is a class of complex systems whose complexity is NP Hard, being extensions of the Traveling Salesman Problem (TSP).

Most work on the PDP assumes that a plan is made in advance and is then executed. There are several solution methods to address static PDPs; these can be classified into exact methods, heuristic methods, and metaheuristic methods, used to obtain acceptable solutions. Some of these methods are described in more detail in *Cai et al. (2023)* and *Parragh et al. (2008)*. PDPs, indeed, are always full of dynamics and uncertainty, particularly because delays and errors occurred frequently in the process of information transmission *(Converso et al., 2023)*. All of these can have serious impact on the choice of optimal solution of PDPs *(Zang et al., 2022)*. In real life, often, there is a need of altering plans and make decisions on the spot. This is known as reactive planning. In this sense we can distinguish two kind of task assignment:

- **Static Task Assignment**: the tasks once assigned are completed by the robot to which they were originally given.
- **Dynamic Task Assignment**: tasks can be reassigned if there's another robot better suited for the job, this introduces the problem of task switching.

Task switching refers to the act of exchanging or transferring tasks between different agents or vehicles, in order to optimize the overall efficiency and cost-effectiveness of the system. This process involves reassigning tasks based on various criteria such as distance, time, and resource utilization. The concept of task switching is evident in the context of dynamic pickup and delivery problems, where the routes need to be modified to accommodate disruptions and new customer demands.

Dynamic Pickup and Delivery Problems (DPDPs) have gained popularity due to their real-world applications in logistics and supply chain management. In this version of the problem, pickup and delivery tasks are not known in advance but arrive dynamically over time. The objective is to handle tasks in real-time to minimize the total distance travelled by vehicles. Due to the extreme difficulty of synchronously addressing uncertainty and dynamics, the dynamic variant of PDP has not received the same attention as its static counterpart, and only a few latest literatures made preliminary investigation on it *(Zang et al., 2022; Parragh, 2008)*.

In a dynamic problem, unlike in a static problem, the time horizon might be unlimited. Therefore, a solution to a dynamic problem cannot be a static output but rather a solution strategy that, using the revealed information, outlines actions to be taken over time *(Cai et al., 2023)*. A basic and commonly used strategy to solve a dynamic pickup and delivery problem is to adapt an algorithm that solves the static version of the problem. Two approaches can be distinguished: the first involves solving a static PDP every time new information is revealed. However, this strategy suffers from high computational complexity since it requires a complete re-optimization every time new information is revealed, potentially taking too long, thus being unsuitable for a real-time context. The second approach, which is generally used, is as follows: the static algorithm is applied only once at the beginning of the time horizon to obtain an initial solution with the available information. When new information is revealed, the current solution is updated with heuristic methods such as insertion heuristics, deletion heuristics, and swap moves, sometimes coupled with a local search algorithm *(Berbeglia et al., 2010)*. As the volume of data to process increases, the time required to provide an optimal solution becomes prohibitive, rendering these methods incompatible with real-time response needs.

Moving from a static to a dynamic context, one of the main issues is finding a balance between the need to respond quickly to new requests and the need to maintain an optimized routing plan *(Mes et al., 2007)*. Hence, there's an imperative to find more effective and efficient methods that require the implementation of algorithms and techniques capable of handling the arrival of new tasks and adapting schedules in real-time. Various approaches, such as Ant Colony Optimization algorithm *(Geiser, T. et al., 2020)*, multi-agent architecture *(Guerram, T., 2017)*, dynamic programming *(Liu et al., 2018)*, *(Ferrucci et Al., 2014)*, simulation, and agent-based approaches, have been proposed to tackle dynamic PDPs. The "Agent-based" approach is extensively discussed in the scientific literature, in particular *Barbati et al. (2012)* provides a comprehensive overview of the use of autonomous agents in solving complex optimization problems. According to the definition by *Wooldridge et al. (1995)*, to develop an ABM, is needed a complete description of some characteristics, in particular:

The Agent Interaction Paradigm: There are various interaction paradigms which can be primarily classified into:

- **Cooperative paradigms**: agents work together to achieve common objectives.
- **Competitive paradigms**: each agent is self-interested, the final solution might be best for the individual agent involved but not for the group as a whole.

A possible method to address the coordination of AGVs in a dynamic context is the use of auctions, which allows a good compromise between the number of tasks performed and the time taken to perform them *(Lagoudakis et al., 2005)*. In these scenarios, AGVs act as bidders or auctioneers, submitting bids based on their evaluation of the task. Several types of auctions have been studied and developed: first-price auctions, English auctions, Dutch auctions and second-price auctions (or Vickrey auctions), which assign the task to the AGV with the best offer, but at a price equal to the second-best bidder's offer. This approach ensures the truthfulness of the bid. Telling the "truth" is a dominant equilibrium strategy and, therefore, also a Nash equilibrium, meaning no player has an incentive to unilaterally change their strategy as they're already achieving the best possible outcome, considering the strategies of other players *(Klemperer, 1999).*

Some market-based algorithms also add a consensus phase to the auction process to improve the quality of assignments. Consensus, in this sense, is an additional transfer of tasks between robots after the tasks are assigned by the auction. Thus, there is a constant reassignment of tasks during the operation of the AGVs. Some demonstrations of the benefits and of the increase in the solution quality of task trading are presented in *Sung et al. (2013)* and *Fanti et al. (2013).*

It is also crucial to specify that auctions can be divided into two main types: centralized and decentralized. Centralized auctions are managed by a single centralized entity that collect bids from all agents and solves a combinatorial problem to optimally assign tasks to agents *(Dahlquist et al., 2023).* This approach provides a comprehensive overview of the bidding process and is useful for problems with a simpler or static structure. However, centralized management can lead to delays and inefficiencies in dynamic systems. In contrast, decentralized auctions are managed by multiple entities, in our case the AGVs, each of which can: make bids, accept bids, and determine winners. In this type of auction, participants can interact directly with each other, allowing the auction to proceed more quickly, speeding up the bidding and decision-making process. For this reason, it becomes particularly useful in dynamic or real-time situations, such as internal logistics *(DeRyck et al., 2020).* In this regard, *Meissner et al. (2017)* offers a comparison between the current centralized and decentralized control architectures in production and clarifies that centralized and hierarchical architectures aren't compatible with the needs of future systems. However, the main disadvantage in decentralized control is the increased effort required to coordinate all independent entities. When adopting a decentralized control architecture in a production process, there will always be a trade-off between optimality and flexibility.

The proposed method stands out from the existing literature because it combines the use of AGVs, the agent-based approach in simulation, the adoption of an auction mechanism, and the implementation of a cooperative logic based on task switching for the dynamic assignment of tasks in a AGV fleet management model to propose a solution to a dynamic PDPs.

3. Proposed Approach

Building on the recognized need for high production flexibility to adapt to personalized market demands promptly and cost-effectively, as explored by *Marchesano et al. (2022)*, and for dynamic control strategies evidenced by *Gebennini et al. (2013)*, our proposed logic of cooperative exchange introduces a novel application of auction-based dynamics. By focusing on real-time adjustments, we aim to overcome the limitations noted in static systems and enhance operational efficiency and to provide a robust response to unforeseen situations or deadlocks.

The framework is based on an auction model, where each AGV can offer its unsuitable tasks to a common pool. In this proposed cooperation strategy, AGVs commit to autonomously exchanging tasks that are inadequate, so that they better fit the capabilities and current positions of other vehicles in the system. Specifically, the primary goal is to optimize the distribution of tasks, in order to minimize disruptions in each AGV's operational schedule while simultaneously maximizing the efficiency and productivity of the system. The introduction of task switching acts as a regulatory mechanism to balance two seemingly opposing objectives: optimizing the overall efficiency of the system, which reflects a cooperative vision, and maximizing the number of tasks completed by each AGV, which reflects a more individual and autonomous perspective.

The cooperative exchange logic, embedded within the broader framework of solving the Pickup and Delivery Problem (PDP), is explained in Figure1, and more detailed in the following sentences:

- **Initialization**

The algorithm begins with an initialization phase, where each machine in the system is defined with a specific position (x, y). Each AGV is initialized with a starting position and a "wallet" that can be used to track transactions or economic interactions within the system.

- **Management of Incoming Tasks**

In managing incoming tasks, the algorithm enters a simulation loop that continues as long as the simulation is ongoing. If a new task arrives, a Vickrey auction is initiated to assign it. In this auction, each AGV assesses the impact of inserting the new task into its initial itinerary and calculates a bid price for the assignment of available tasks. Bids can be calculated using only the cost for the robot to perform the specific task, but they can also be calculated using the marginal cost for the robot to execute the task considering other tasks on its list *(DeRyck et al., 2020).* To achieve good performance in task allocation, bid calculation based on marginal cost is the most advantageous. In the proposed approach every AGV has a local list of tasks to perform. The goal is to assign the incoming task to the AGV that can integrate it into its work plan with the least impact. The impact of inserting a new task into an existing schedule is calculated as the difference between the total distance the AGV would travel after inserting the task in the previous schedule and the distance the AGV would travel without the task. The AGV evaluates all possible insertions of the new task within its initial planning. Among all the potential combinations the AGV could make, it selects the lowest one. The Bid Function of each AGV, for a task, is then calculated as a fraction (Task Impact\Total Distance) of the wallet associated with each AGV, proportional to the task's value. A high value of the Bid indicates that the task is very rewarding relative to the effort required to complete it. AGV bids are assessed and compared, and the task is assigned to the AGV that has submitted the second highest bid, and the AGV's route is updated accordingly.

```
Algorithm 1: AGV Simulation for Pickup and Delivery
Input  : Set of machines {M_i}, Set of AGVs {A_j}
Output: Tasks completed and system KPIs updated
Function Main():
    /* Initialization:                                              */
    foreach machine M_i in set of machines do
        Define position P_i = (x_i, y_i)
    foreach AGV A_j in set of AGVs do
        Define initial position P_j = (x_j, y_j)
        Initialize AGV wallet
    /* Management of Incoming Tasks:                                */
    while there are incoming tasks do
        foreach new task T_k do
            Start a Vickrey auction to assign T_k
            foreach AGV A_j do
                Calculate D_{orig,j} and D_{new,j}
                ΔD_j = D_{new,j} − D_{orig,j}
                Calculate the bid Bid_j = (ΔD_j / Total Distance_j) × W_j
            Assign T_k to the AGV with the second highest bid
            Update AGV's A_j route
    /* Management of Inactive Tasks:                                */
    while simulation is ongoing do
        foreach AGV do
            Identify inactive tasks that exceed T_max
            Insert the selected task into a collection of inactive tasks
            foreach inactive task T_k do
                Start a Vickrey auction to transfer T_k
                Let j be the original AGV holding T_k
                foreach other AGV i do
                    Calculate ΔD_i = D_{new,i} − D_{orig,i}
                select(i) = { 1  if ΔD_i = min{ΔD_k : ΔD_k < ΔD_j, ∀k}
                            { 0  otherwise
                If i exists: Assign the task T_k to AGV i
                Update the task lists and wallets of the involved AGVs
    /* Execution of Tasks for each AGV:                             */
    while there are incomplete tasks in the assigned task list do
        foreach AGV A_j do
            Identify task T_k with the closest origin machine
            move AGV A_j to machine origin, then to destination
            update AGV's position to machine destination
            remove completed task T_k from the list
    /* Evaluation of KPIs:                                          */
    foreach AGV A_j do
        Calculate AGV Utilization
        Calculate Average Task Completion Time
        Calculate Number of Task Completed
```

Figure 1: Pseudocode of the proposed method

- **Management of Inactive Tasks**

Similarly, the algorithm manages tasks that have become inactive. During the execution time the AGV evaluates its task queue and identifies, through a temporal variable, those that exceed a certain time threshold Tmax without being completed. These tasks are placed in a collection of inactive tasks and a Vickrey auction is used to transfer the task to another AGV. In this phase it is not certain that the task will be transferred. An activation function is needed to evaluate if the offers received to carry out the switch are actually advantageous. In this exchange of activities between AGVs, the transferring AGV pays a certain amount from its portfolio to the receiving one. Once the switch is confirmed, the task is transferred to the AGV with the second highest bid, and the list of assigned tasks and the AGV's wallet are updated.

- **Task Execution**

For each AGV, the assigned tasks are executed one after the other. The AGV moves from its current position to the machine of the task's origin, then to the destination of the task, updating its position upon completion of the task. This process continues until all assigned tasks have been completed.

- **Performance Indicator Evaluation (KPIs)**

Finally, the algorithm evaluates various KPIs, such as:

- **AGV Utilization**: through which we measure how much time AGVs are actively engaged in completing tasks.
- **Average Task Completion Time**: through which we measure the average time needed to complete a task.
- **Number of Tasks Completed**: through which we measure the total number of tasks completed within a given timeframe.

These indicators help measure the efficiency of the AGV system and the effectiveness of the Vickrey auction in task assignment.

The model was tested in Anylogic©, within an hybrid Agent Based – Discrete Event Simulation (AB-DES) simulation environment, illustrated in Figure2, with two different scenarios. By using Discrete Event Simulation (DES), it's possible to model the system as a sequence of distinct events. This allows us to take into account temporal variations and dynamic events, such as the arrival of new tasks. In our context, DES is used to model and simulate the workflow of the AGVs and task execution. Agent-Based Modelling (ABM) is used to model each AGV as an autonomous agent, which can make decisions, interact with other agents by participating in an auction, make bids based on its own task evaluation, and exchange problematic tasks over time.

In this model, leveraging the capabilities of DES blocks, we incorporated a fleet management component that allows the configuration of AGV properties to either follow path-guided or free-path settings. For our analysis, we opted for the free-path setting, thereby enhancing the operational dynamics of the AGVs. This approach aligns closely with the autonomous mobile robots (AMRs) functionality, as extensively discussed in the work by *Fragapane et al., 2021*.

In the first experimental setup, only competitive logic, without switch function, is active, while in the second experimental setup, the cooperative logic is implemented, varying the number of machines and the number of AGVs. Figure3, Figure4 and Figure5, detail the combinations of AGVs and machines involved in each simulation and the results of the KPIs are compared between the use of competitive logic and the proposed cooperative logic.

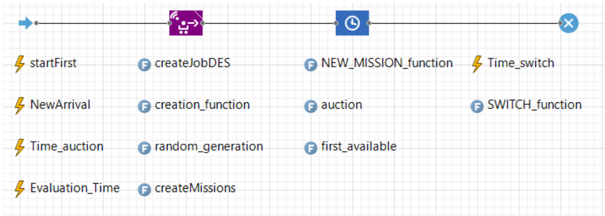

Figure 2: Anylogic Main environment detail

From this experimental campaign, it emerged that the introduction of cooperative logic led to an increase in AGV utilization. With 2 AGVs and 5 machines, AGV Utilization went from 46% to 50%, corresponding to an improvement of 8.7%. With 3 AGVs and 6 machines, the increase was from 53% to 57%, an increment of 7.5%. These improvements indicate that the AGVs were employed more intensively, reducing idle times. The Average Task Completion Time also saw a reduction, indicating increased efficiency. For AGVs with 5 machines, the time dropped from 2.12 to 2.00 (a reduction of 5.7%) and from 2.23 to 2.20 (a reduction of 1.3%) with 6 machines. This demonstrates that the cooperative logic enabled tasks to be completed more quickly, optimizing routes or improving task assignment. Finally, there was an increase in the Number of Tasks completed. For 2 AGVs with 5 machines, the number rose from 15.8 to 16.5, showing an improvement of 4.4%. Even with 3 AGVs and 5 machines, the number went from 15.08 to 16.0 (an increase of 6.1%). This rise in the number of tasks completed reflects a direct improvement in productivity due to cooperation.

It is also crucial to acknowledge that, as this study is still in its preliminary simulation phase, a comprehensive comparison with other existing works, particularly non-auction-based algorithms, has yet to be undertaken. This choice reflects the current focus on developing and testing individual functionalities rather than comparative analysis at this stage. Future research will aim to address this gap by incorporating a broader range of comparisons to fully assess the algorithm's effectiveness and versatility in more complex scenarios.

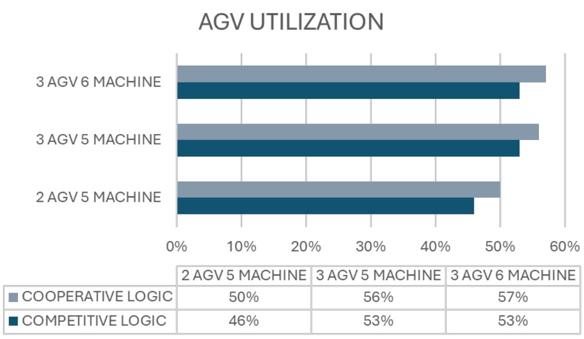

Figure 3: AGV Utilization simulation results

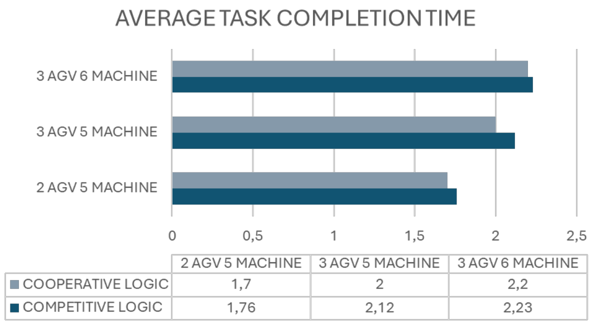

Figure 4: Average Task Completion Time simulation results

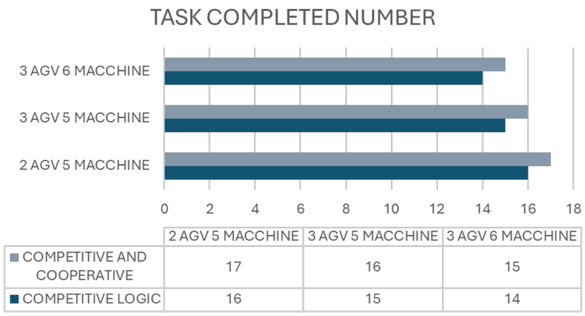

Figure 5: Task Completed Number simulation results

Testing this algorithm in a simulated environment allows us to monitor its behaviour under controlled conditions. However, extending these results to real-world environments introduces several challenges:

- Exposure to real-world complexity for refining the algorithm to handle unpredictability, such as equipment malfunctions and human interactions;
- Integration with existing logistics and management for advancing interoperability and ensuring smooth operation across different technological ecosystems.

To address these challenges, conducting pilot tests in real environments is essential to validate and refine the algorithm with actual performance data. Finally, iterating the design based on feedback received during these tests is crucial to further improve the robustness and adaptability of the system.

4. Conclusions

In an increasingly dynamic production scenario, it becomes essential to acquire the skills and ability to allocate productive resources more effectively in order to adequately respond to new market challenges. Therefore, leveraging the technological innovations made available by the Fourth Industrial Revolution, this work focuses on the efficient implementation of autonomous guided vehicles (AGVs) for Material Handling operations, aiming to exploit the potential of AGVs, and on auction theory as a tool for decentralizing operational decisions. The ultimate goal is to develop a flexible and scalable AGV fleet management model capable of adapting to the dynamic changes of internal logistics and managing specific Pickup and Delivery Problem (PDP), both static and dynamic, each with a unique and irreplaceable origin and destination. To this end, the analysis began with a study of Auction Theory and the numerous methods proposed in the literature that have previously addressed the same problem. Using the software Anylogic©, a simulation tool was developed that allowed the evaluation of the system's performance through an experimental campaign, consisting of two different experimental plans, each aimed at analysing different scenarios. By comparing the results achieved in competitive logic, which aims to maximize individual interests, and the results obtained in cooperative logic, which, despite achieving worse personal outcomes, strives for a better global solution, the task switching process is enhanced with increased efficiency. Cooperation among AGVs allows for a more strategic and flexible use of mobile resources, optimizing both workload and time management. This approach is particularly effective in complex operational environments where tasks can change rapidly and where operational efficiency is crucial. In subsequent studies, an attempt will be made to also consider the aspect related to the machines, specifically their production speed, thereby aiming to manage and limit the starvation of machines without compromising the optimal use of AGVs. This will involve developing advanced algorithms that can dynamically adjust AGV tasks based on real-time production metrics and machine availability. The goal is to create a more integrated and responsive system that not only enhances the efficiency of the AGVs but also ensures that the entire production line operates smoothly and without unnecessary delays. By further aligning the operational time of AGVs with the production demands of the machines, we aim to achieve a balanced system that optimizes resource utilization across the board.

References

Assunta C., Guido G., Silvestro V., Giusy V. (2017). Man-CPS interaction: An experimental assessment of the human behavior evolution. RTSI 2017 - IEEE 3rd International Forum on Research and Technologies for Society and Industry, Conference Proceedings.

Barbati, M., Bruno, G., Genovese, A. (2012). Applications of agent-based models for optimization problems: A literature review. Expert Systems with Applications, 6020-6028, 39(5).

Barreto, L., Amaral, A., & Pereira, T. (2017). Industry 4.0 implications in logistics: An overview. Procedia Manufacturing, 13, 1245-1252.

Berbeglia, G., Cordeau, J., & Laporte, G. (2010). Dynamic pickup and delivery problems. European Journal of Operational Research, 202, 8-15.

Cai, J., Zhu, Q., Lin, Q., Ma, L., Li, J., & Ming, Z. (2023). A survey of dynamic pickup and delivery problems. Neurocomputing, (2023), 554.

Chen, B., Wan, J., Shu, L., Li, P., Mukherjee, M., & Yin, B. (2019). Smart Factory of Industry 4.0: Key Technologies, Application Case, and Challenges. IEEE Access, 6505-6519, 6.

Converso, G., Gallo, M., Murino, T., & Vespoli, S. (2023). Predicting Failure Probability in Industry 4.0 Production Systems: A Workload-Based Prognostic Model for Maintenance Planning. Applied Sciences, 13(1938).

Dahlquist, N., Lindqvist, B., Saradagi, A. and Nikolakopoulos, G., 2023, August. Reactive Multi-agent Coordination using Auction-based Task Allocation and Behavior Trees. In 2023 IEEE Conference on Control Technology and Applications (CCTA) (pp. 829-834). IEEE.

De Martino, M., Marchesano, M. G., Guizzi, G., Salatiello, E. (2023). Deep Reinforcement Learning-Based Controller for Autonomous Guided Vehicles (AGVs) in a Multi-Department Production Plant. Proceedings of the XXVIII Summer School "Francesco Turco".

De Ryck, M., Versteyhe, M., & Debrouwere, F. (2020). Automated guided vehicle systems, state-of-the-art control algorithms and techniques. Journal of Manufacturing Systems, 54: 152-173.

Fanti, M. P., Franceschelli, M., Mangini, A. M., Pedroncelli, G., & Ukovich, W. (2013). Discrete Consensus in Networks with Constrained Capacity. 52nd IEEE Conference on Decision and Control, (2013), 2012-2017.

Ferrucci, F., & Bock, S. (2014). Real-time control of express pickup and delivery processes in a dynamic environment. Transportation Research Part B: Methodological, 1-14, 63.

Fragapane, G., de Koster, R., Sgarbossa, F., Strandhagen, J.O. (2021). Planning and control of autonomous mobile robots for intralogistics: Literature review and research agenda. European Journal of Operational Research , 294(2), pp. 405–426.

Gebennini, E., Grassi, A., Fantuzzi, C., Gershwin, S.B., & Schick, I.C. (2013). Discrete time model for two-machine one-buffer transfer lines with restart policy. Annals of Operations Research, 209(1), 41-65.

Geiser, T., Hanne, T., Dornberger, R. (2020). Best-match in a set of single-vehicle dynamic pickup and delivery problem using ant colony optimization. Proceedings of the 2020 the 3rd International Conference on Computers in Management and Business

Guerram, T. (2017). A Multi Agent based Organizational Architecture for Dynamic Pickup and Delivery Problem. Journal of Computing and Information Technology, Vol. 25, No. 4, 259–277.

Holubčík, M., Koman, G., & Soviar, J. (2021). Industry 4.0 in Logistics Operations. Procedia Engineering, 53, 282-288.

Klemperer, P. (1999). Auction Theory: A Guide to the Literature. 227-286.

Lagoudakis, M. G., Markakis, E., Kempe, D., Keskinocak, P., Kleywegt, A., Koenig, S., Tovey, C., Meyerson, A., & Jain, S. (2005). Auction-Based Multi-Robot Routing. Robotics: Science and Systems, 343-350.

Liu, S., Tan, P. H., Kurniawan, E., Zhang, P., & Sun, S. (2018). Dynamic scheduling for pickup and delivery with time windows 4th IEEE World Forum on Internet of Things WF-IoT.

Marchesano, M.G., Salatiello, E., Guizzi, G., & Santillo, L.C. (2022). A Reinforcement Learning approach in Industry 4.0 enabled production system. In Proceedings of the 27th Summer School Francesco Turco. Rome: AIDI - Italian Association of Industrial Operations Professors.

Meissner, H., Ilsen, R., & Aurich, J. C. Analysis of control architectures in the context of Industry 4.0. Production & Manufacturing Research, 244-272, 8.

Mes, M., van der Heijden, M., & van Harten, A. (2007). Comparison of agent-based scheduling to look-ahead heuristics for real-time transportation problems. European Journal of Operational Research, 181, 59-75.

Parragh, S. N., Doerner, K. F., & Hartl, R. F. (2008). A survey on pickup and delivery problems. Part I: Transportation between customers and depot. Journal für Betriebswirtschaft, 58: 21-51.

Parragh, S. N., Doerner, K. F., & Hartl, R. F. (2008). A survey on pickup and delivery problems. Part II: Transportation between pickup and delivery locations. Journal für Betriebswirtschaft, 58: 81-117.

Pereira, A.C., & Romero, F. (2017). A review of the meanings and the implications of the Industry 4.0 concept. Procedia Manufacturing, 1206-1214, 13.

Pfeiffer, S. (2019). The Vision of "Industrie 4.0" in the Making—a Case of Future Told, Tamed, and Traded. NanoEthics, 107-121, 11.

Rocha, L. F., Moreira, A. P., & Azevedo, A. (2010). Flexible Internal Logistics Based on AGV System's: A Case Study. IFAC Proceedings Volumes, 43(17): 248-255.

Simon, H. A. (1955). A Behavioral Model of Rational Choice. The Quarterly Journal of Economics, 99-118.

Sung, C., Ayanian, N., & Rus, D. (2013). Improving the performance of multi-robot systems by task switching. IEEE International Conference on Robotics and Automation, 2999-3006.

Wooldridge, M., & Jennings, N. R. (1995). Intelligent Agents: Theory and Practice. The Knowledge Engineering Review, 10, 115-152.

Zang, X., Zhu, Y., Zhong, Y., & Chu, T., 2022. CiteSpace-Based bibliometric review of pickup and delivery problem from 1995 to 2021. Applied Sciences. mdpi.com