Deep Reinforcement Learning-Based Controller for Autonomous Guided Vehicles (AGVs) in a Multi-Department Production Plant

Maria De Martino*, Maria Grazia Marchesano*, Guido Guizzi*, Emma Salatiello*

**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: maria.demartino13@studenti.unina.it, mariagrazia.marchesano@unina.it , g.guizzi@unina.it, emma.salatiello@unina.it)

Abstract: With the advent of advanced industrial facilities, there arises a pressing demand for swifter and smarter production methods. This is precisely why material handling plays a pivotal role in shaping modern manufacturing practices. The swift and efficient material handling is achieved by implementing Automated Guide Vehicles (AGVs). In our study, we delve into the examination of routing and scheduling predicaments encountered by AGVs, employing reinforcement learning (RL) techniques with PPO algorithm, using AnyLogic and ALPyne. The findings we have obtained exhibit intriguing prospects for resolving the real-case study we have presented, while also demonstrating reliability in addressing various alterations and scaling operations associated with the problem.

Keywords: Material handling, AGV, Reinforcement Learning, PPO.

I. INTRODUCTION

The birth and development of Industry 4.0 in recent years has necessitated a new production paradigm that encompasses all aspects of the production process (Guizzi et al., 2020). The aim of Industry 4.0 is to resolve the long-standing conflict between the individuality of on-demand output and the savings realized through economies of scale (Vespoli et al., 2021). Despite significant advances in the field of Industry 4.0, there is still an open gap in the literature regarding advanced methodologies for production planning and control (Vespoli et al., 2023). Materials handling isan essential activity in any production process and its efficiency has severe impacts on the production costs (Vivaldini et al., 2015). The reduction of makespan in a production line is the objective of many studies, in fact Grassi et al., 2023 introduced two metrics to evaluate the scheduling decisions and optimize the scheduling process, with the competitive goal of maximizing tool utilization and minimizing production makespan. In recent years, with the need to make production ever faster and more intelligent (Vespoli et al., 2022), the introduction ofAGVs has become necessary. AGVs are automatedguided vehicles that allow the transport of material from one station to another, from one department toanother. The use of AGVs inevitably makes every material movement faster. AGVs have many advantages, but also problems. Among the problems related to AGVs, those of interest for thiswork are those related to logistics. Logistics problems are defined in three categories: dispatching, scheduling, and routing (Vivaldini et al., 2015). In this work the attention is placed on theconcepts of routing and scheduling since they influence each other. In general, routing and scheduling problems are interrelated since the scheduling strategy must ensure that the given routing algorithm's conditions are met scheduling. In the literature there are several proposals which tryto solve, at least separately, either the routing problem or the scheduling problem. Marchesano et al., 2022 propose Reinforcement Learning (RL) for resolving production scheduling difficulties of varying complexity. In this way, human intervention in production scheduling can bereduced, while planning and decision-making capabilities are improved at the same time. Salatiello et al., 2022 propose a new dispatching rule able to assign the jobs to the available resource by considering the processing time, the machine's utilisation and the due dates, evaluating the advantages of an Industr

4.0 enabled Job Shop production system. Marchesano, Guizzi, et al., 2022 present, in an application environment, a dispatching rule based on a deep reinforcement learning (DRL) algorithm. The overall objective of the research is to provide a general scheduling tool that may be used in a variety of situations, including unexpected ones. Zeng et al., 2014 proposed a two-stage heuristic algorithm that combines an improved timing method and a local search to fix it of AGV scheduling. Miyamoto & Inoue, 2016 have proposed several local/random search methods to solve the problem of conflict-free sending and routing of AGV systems with capacity constraints. To solve the problem of integrated dispatching scheduling and conflict-free routing of AGVs, Umar et al., 2015 formulated three objectives to consider makespan, AGV travel time and cost of the fine and developed a multi-target hybrid genetic algorithm. Taghaboni and Tanchoco (1995) proposed a dynamic routing technique, namely incremental route planning, which can route AGVs relatively quickly compared with some static algorithms. Gabel & Riedmiller, n.d. proposed a tabular multi- agent QL approach for addressing dynamic scheduling problems in which unexpected events may occur, such as the arrival of new tasks or the breakdown of equipment, which would require frequent re-planning. Qu et al., 2016 developed a two-agent Markov gameapproach based on QL to realize real-time cooperation between machines (scheduling) and the workforce (human resource management agents). Zhang & Dietterich, n.d. described a neural networkbased job-shop scheduling approach which demonstrated superior performance and reduced costs for manual system design. To cope with the complexities and to reduce human-based decisions, Lin et al., 2019 proposed a multi- class DQN approach that feeds local information toschedule job shops in semiconductor manufacturing. To meet the requirements in wafer fabrication dispatching, (Altenmüller et al., 2020 implementeda single-agent DQN that processed 210 data points as a single state input (such as machine loading status or machine setup). This enabled the DQN to meet strict time constraints better than competitive heuristics (TC, FIFO) while reaching predefined work-in-progress (WIP) targets as a secondary goal. Stricker et al., 2018 and Kuhnle et al., 2021 proposed a single-agent adaptive production controlsystem that maximised machine utilisation and reduced lead and throughput times compared to conventional methods that struggle partially known environments. Hu et al., 2020 implemented a mixed rule dispatching approach that determines the dispatching rule (*FCFS, STD, EDD, LWT, NV*) for an automated guided vehicle (AGV) depending on its observed state which reduced the makespan anddelay ratio by approximately 10% compared to the benchmarks.

The objective of this work is to define a solution to the problems through reinforcement learning (RL) techniques. Starting from a real case and defining the working hypotheses according to the definition constraints of a reinforcement learning problem.

II. REINFORCEMENT LEARNING

Reinforcement Learning is typically represented bya Markov Decision Process (MDP), which providesa mathematical framework for describing the environment in which reinforcement learning takes place. It consists of two main components: the agent and the environment. The agent learns through repeated interactions with the environment with the goal of making optimal decisions to achieve desired outcomes.

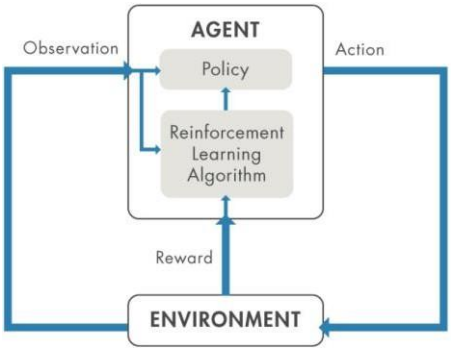

Figure 1 General diagram of Reinforcement Learning

Several parameters are defined to facilitate this process, including observations, actions, and rewards. Observations refer to the information the agent receives from the environment at each time step. Actions are the decisions that the agent can take in response to the observed state of the environment. Rewards are used to evaluate the agent's performance and provide feedback on the quality of its decisions. A Reinforcement Learning process involves defining both the agent and the environment. The agent represents the learning entity that interacts with the environment, while the environment encompasses the problem domain in which the agent operates. Additionally, a reward function needs to be formulated to guide the agent's learning process and reinforce desired behaviour. This reward function assigns numeric values to different states and actions, allowing the agent to learn from the received feedback. The choice of policy is crucial in reinforcement learning as it determines the desired outcome. A policy defines the mapping from states to actions and governs the

behaviour ofthe agent. Once the policy is defined, an appropriate algorithm needs to be selected to optimize the learning process. In the specific context of scheduling and routing problems, the Proximal Policy Optimization (PPO) algorithm has gained attention and popularity (Vanvuchelen et al., 2020). PPO has been chosen due to its numerous advantages, including stability, computational efficiency, versatility, and good performance. The stability of PPO ensures faster convergence speed. The PPO algorithm offers severaladvantages and is well-suited for this purpose and, in general, for dynamic scheduling issues (Marchesano, Staiano, et al., 2022).

III. PROPOSAL DEFINITION

The proposal presented stems from a real case study involving an aero-engine company that aims to implement an Automated Guided Vehicle (AGV) for transporting parts between different machines. The problem at hand can be summarized as follows: there are 10 available jobs, characterized by three operations, each belonging to one of three distinct types. These types differ in terms of size and work cycle. The industry plant is divided into three departments,which can be simplified as machines. All machines have the same processing time, defined by a triangular distribution with parameters minimum value, maximum value and mode equal to 5, 10, and 15. All jobs need to go through three machines to be considered *completed* but the order in which they visit these machines varies depending on the jobtype.
The jobs have the following work cycles:
Job1: Machine 1, Machine 2, Machine 3.
Job 2: Machine 2, Machine 1, Machine 3.
Job 3: Machine 3, Machine 2, Machine 1.

$$J_1(O_1, O_2, O_3)$$
$$J_2(O_2, O_1, O_3)$$
$$J_3(O_3, O_2, O_1)$$

The objective is to determine the optimal job processing order that minimizes the makespan.

To evaluate the problem, a simulation model will be employed, specifically utilizing the AnyLogic multi-method tool, integrated with ALPyne. ALPyne is an AnyLogic-Python connector. In particular, it is a Python library for interactively running models exported from the RL experiment, which can be used with any edition of AnyLogic (Personal Learning Edition, University, or Professional). The need to use this tool stems from the fact that, unlike other experiments, the RL experiment cannot be directly executed by end-users within AnyLogic. To characterize the problem proposed this study proposed an Agent-Based and Discrete Event Simulation model, involving the following agents: Job, Machine, and AGV. The model operates with *missions* that specify whether the action of the AGV is for loading or unloading. During the simulation, the jobs are processed in the order they are created. However, during the training in the Reinforcement Learning experiment, the processing order is determined by the action that the agent must take. The agent has two actions to undertake: selecting the "typeJob" and the "typeAction". So, the agent must choose the Job to process based on the type of Job, whether it's type 1, 2, or 3, and based on the action, either loading or unloading for the AGV. The agent makes decisions based on its interactions with the observation space. In the presented work, several observations are considered:

1. n_x: the number of operations completed, where x represents the number of jobs.
2. CompletedJob: the number of completed jobs.
3. UT: The utilization of the AGV (Automated Guided Vehicle) resource.
4. timeSum: The sum of processing and travel times for each individual job.

Based on the combination of actions, the target machine for the job is determined. The agent's actions aim to maximize the reward function, which is defined as the sum ofthe AGV's travel time and the processing times at the three machines, divided by the travel time.

$$R = \frac{ProcessingTime + tripTime}{tripTime}$$

Thus, the agent's goal is to take actions that maximize the reward function in order to minimizethe makespan value.

IV. RESULTS

The results obtained from the pre-experiments align with the expectations and offer valuable insights into the effectiveness of the agent in minimizing the makespan by maximizing the reward function through its actions. The observed trend indicates that as the number of timesteps increases, the reward function consistently grows, implying a positive correlation between the agent's actions and the achieved reward. Examining the reward curve depicted in Figure 2, itis noteworthy that the initial section shows a linear progression, suggesting a learning phase where the agent is exploring different actions to determine the optimal ones. As the experiment progresses, thecurve converges towards a reward value of approximately 185, indicating that the agent has successfully learned to navigate the system andoptimize the makespan. To further assess the model's performance, a comparative analysis was

conducted between the makespan values obtained from simulation using the AnyLogic multi-method tool and those derived from the experiment. Throughout the agent's learning period, a noticeable trend emerges— initially, the makespan values are higher but gradually stabilize at a lower value compared to thesimulation, as shown in Figure 3 in which the makespan function exhibits a decreasing trend as the number of time-steps increases. Eventually, it reaches a point on the graph where it stabilizes at anaverage value that is lower than the one obtained during the simulation. This finding underscores theagent's ability to learn and improve its decision- making process over time, leading to more efficientmakespan reduction compared to the traditional simulation approach. Moreover, a quantitative evaluation was performedby comparing the makespan value obtained from simulation, evaluated in 10 replications, with the average makespan achieved through reinforcement learning. The results demonstrate that the makespanachieved through reinforcement learning consistently outperforms the simulation-based approach, as shown in TABLE 1. This comparison provides concrete evidence of the agent's proficiency in optimizing the reward function and achieving substantial makespan reduction, surpassing the capabilities of traditional simulation techniques.

TABLE I. MAKESPAN COMPARISON

Makespan Simulation	**Makespan RL**
122,49	120,45

Taken together, these findings offer robust evidencesupporting the successful functioning of the proposed model. The experiment showcases theagent's remarkable ability to adapt and learn, ultimately leading to effective makespan minimization. This conclusion is bolstered by both the analysis of the reward curve, which exhibits a clear progression towards higher rewards, and the comparative evaluation that highlights the agent's superior performance when compared to simulation-based methods. In summary, the experiment's results solidify the notion that the agent can efficiently optimize the reward function to minimize the makespan. This research serves as a testament to the potential of reinforcement learning in improving system efficiency and underscores its practicality in real world applications where makespan reduction is a critical objective.

V. CONCLUSIONS

Starting from a well-known problem in the literature, this study proposes a novel and relativelyunexplored solution. The focus is on addressing thechallenging scheduling and routing problem of an Automated Guided Vehicle (AGV) through the application of reinforcement learning techniques. By leveraging the power of reinforcement learning,this approach aims to optimize the AGV's decision- making process, leading to improved efficiency and reduced makespan. The preliminary results obtained during the pre- experimental phase of the study are encouraging and demonstrate the potential of the proposed model. These early findings indicate that the Agent,guided by reinforcement learning, is capable ofmaking effective decisions to maximize the reward function and minimize the makespan. However, the scope of this project extends beyond the initial results. The ultimate goal is to implementand validate the proposed model in various real-world scenarios. This flexibility allows for adaptations such as adjusting the number ofmachines, accommodating different job types, or varying the availability of jobs. By conducting thorough analyses and experiments, the aim is to ensure the reliability and effectiveness of the model across different settings. In summary, this study presents a promising approach to the scheduling and routing problem of AGVs using Reinforcement Learning techniques. The initial results indicate the model's potential for achieving significant improvements in efficiency. Future work will focus on refining and validating the model through comprehensive analyses, ensuring its reliability and applicability in practical scenarios.

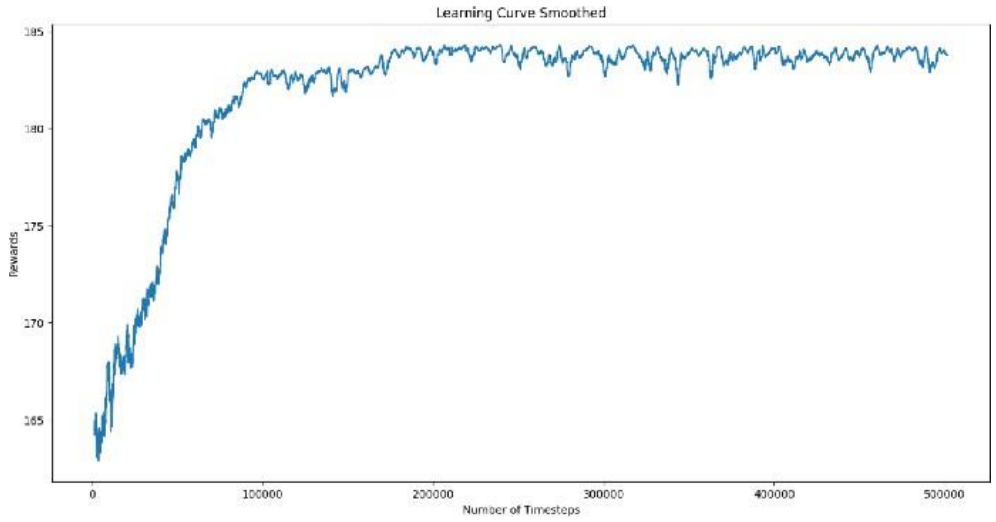

Figure 2 Reward Function

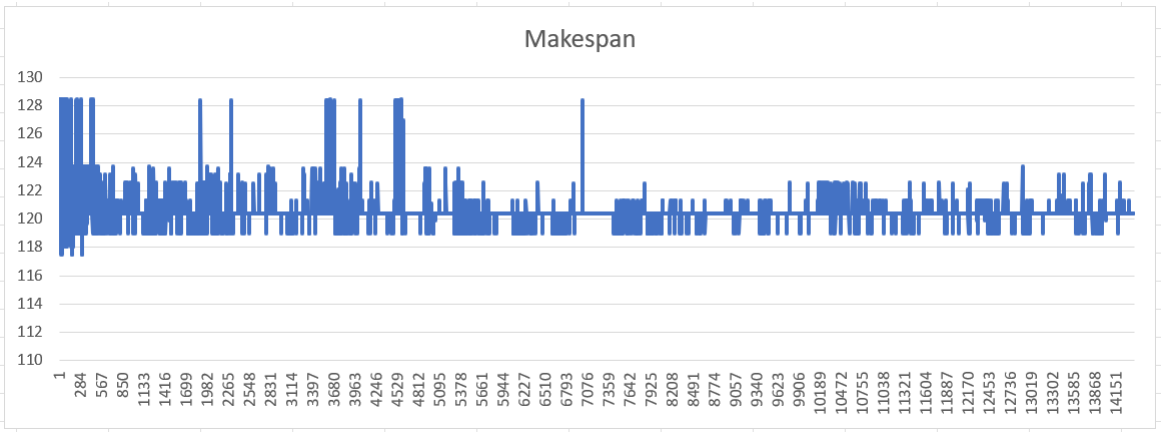

Figure 3 The Makespan behavoiur over the training time-step

.

I. REFERENCES

Altenmüller, T., Stüker, T., Waschneck, B., Kuhnle, A., & Lanza, G. (2020). Reinforcement learning for an intelligent and autonomous production control of complex job-shops under time constraints. *Production Engineering*, *14*(3), 319-328. https://doi.org/10.1007/s11740-020-00967-8

Gabel, T., & Riedmiller, M. (n.d.). ADAPTIVE REACTIVE JOB-SHOP SCHEDULING WITH REINFORCEMENT LEARNING AGENTS. In *International Journal of Information Technology and Intelligent Computing*.

Guizzi, G., Vespoli, S., Grassi, A., & Carmela Santillo, L. (2020). Simulation-Based Performance Assessment of a New Job-Shop Dispatching Rule for the Semi-Heterarchical Industry 4.0 Architecture. *Proceedings - Winter Simulation Conference*, *2020-Decem*, 1664-1675. https://doi.org/10.1109/WSC48552.2020.9383981

Hu, H., Jia, X., He, Q., Fu, S., & Liu, K. (2020). Deep reinforcement learning based AGVs real-time scheduling with mixed rule for flexible shop floor in industry 4.0. *Computers and Industrial Engineering*, *149*. https://doi.org/10.1016/j.cie.2020.106749

Kuhnle, A., Kaiser, J. P., Theiß, F., Stricker, N., & Lanza, G. (2021). Designing an adaptive production control system using reinforcement learning. *Journal of Intelligent Manufacturing*, *32*(3), 855-876. https://doi.org/10.1007/s10845-020-01612-y

Lin, C. C., Deng, D. J., Chih, Y. L., & Chiu, H. T. (2019). Smart Manufacturing Scheduling with Edge Computing Using Multiclass Deep Q Network. *IEEE Transactions on Industrial Informatics*, *15*(7), 4276-4284. https://doi.org/10.1109/TII.2019.2908210

Marchesano, M. G., Guizzi, G., Popolo, V., & Converso, G. (2022). Dynamic scheduling of a due date constrained flow shop with Deep Reinforcement Learning. *IFAC-PapersOnLine*, *55*(10), 2932-2937. https://doi.org/10.1016/j.ifacol.2022.10.177

Marchesano, M. G., Salatiello, E., Guizzi, G., & Santillo, L. C. (2022). A Reinforcement Learning approach in Industry 4.0 enabled production system. *Proceedings of the Summer School Francesco Turco*.

Marchesano, M. G., Staiano, L., Guizzi, G., Castellano, D., & Popolo, V. (2022). Deep Reinforcement Learning Approach for Maintenance Planning in a Flow-Shop Scheduling Problem. In *Frontiers in Artificial Intelligence and Applications* (Vol. 355). https://doi.org/10.3233/FAIA220268

Miyamoto, T., & Inoue, K. (2016). Local and random searches for dispatch and conflict-free routing problem of capacitated AGV systems. *Computers & Industrial Engineering*, *91*, 1-9. https://doi.org/10.1016/J.CIE.2015.10.017

Qu, S., Wang, J., Govil, S., & Leckie, J. O. (2016). Optimized Adaptive Scheduling of a Manufacturing Process System with Multi-skill Workforce and Multiple Machine Types: An Ontology-based, Multi-agent Reinforcement Learning Approach. *Procedia CIRP*, *57*, 55-60. https://doi.org/10.1016/j.procir.2016.11.011

Salatiello, E., Guizzi, G., Marchesano, M. G., & Santillo, L. C. (2022). Assessment of performance in Industry 4.0 enabled Job-Shop with a due-date based dispatching rule. *IFAC-PapersOnLine*, *55*(10), 2635-2640. https://doi.org/10.1016/j.ifacol.2022.10.107

Stricker, N., Kuhnle, A., Sturm, R., & Friess, S. (2018). Reinforcement learning for adaptive order dispatching in the semiconductor industry. *CIRP Annals*, *67*(1), 511-514. https://doi.org/10.1016/j.cirp.2018.04.041

TAGHABONI-DUTTA, F., & TANCHOCO, J. M. A. (1995). Comparison of dynamic routeing techniques for automated guided vehicle system. *International Journal of Production Research*, *33*(10), 2653-2669. https://doi.org/10.1080/00207549508945352

Umar, U. A., Ariffin, M. K. A., Ismail, N., & Tang, S. H. (2015). Hybrid multiobjective genetic algorithms for integrated dynamic scheduling and routing of jobs and automated-guided vehicle (AGV) in flexible manufacturing systems (FMS) environment. *International Journal of Advanced Manufacturing Technology*, *81*(9-12), 2123-2141. https://doi.org/10.1007/s00170-015-7329-2

Vanvuchelen, N., Gijsbrechts, J., & Boute, R. (2020). Use of Proximal Policy Optimization for the Joint Replenishment Problem. *Computers in Industry*, *119*. https://doi.org/10.1016/j.compind.2020.103239

Vespoli, S., Guizzi, G., Gebennini, E., & Grassi, A. (2022). A novel throughput control algorithm for semi-heterarchical industry 4.0 architecture. *Annals of Operations Research*, *310*(1), 201-221. https://doi.org/10.1007/s10479-021-04184-z

Vivaldini, K. C. T., Rocha, L. F., Becker, M., & Moreira, A. P. (2015). Comprehensive review of the dispatching, scheduling and routing of AGVs. *Lecture Notes in Electrical Engineering*, *321 LNEE*, 505–514. https://doi.org/10.1007/978-3-319-10380-8_48

Zeng, C., Tang, J., & Yan, C. (2014). Scheduling of no buffer job shop cells with blocking constraints and automated guided vehicles. *Applied Soft Computing*, *24*, 1033-1046. https://doi.org/10.1016/J.ASOC.2014.08.028

Zhang, W., & Dietterich, T. G. (n.d.). *A Reinforcement Learning Approach to Job-shop Scheduling*.

Grassi, A., Guizzi, G., Popolo, V., & Vespoli, S. (2023). A Genetic-Algorithm-Based Approach for Optimizing Tool Utilization and Makespan in FMS Scheduling. Journal of Manufacturing and Materials Processing, 7(2). https://doi.org/10.3390/jmmp7020075

Vespoli, S., Grassi, A., Guizzi, G., & Popolo, V. (2021). A Deep Learning Algorithm for the Throughput Estimation of a CONWIP Line. IFIP Advances in Information and Communication Technology, 630 IFIP, 143–151. https://doi.org/10.1007/978-3-030-85874-2_15

Vespoli, S., Grassi, A., Guizzi, G., & Popolo, V. (2023). Generalised Performance Estimation in Novel Hybrid

MPC Architectures: Modeling the CONWIP Flow-Shop System. Applied Sciences (Switzerland), 13(8). https://doi.org/10.3390/app13084808