






Letter

A Multi-AGV Routing Planning Method Based on Deep Reinforcement Learning and Recurrent Neural Network

Yishuai Lin , Gang Hu , Liang Wang ,
Qingshan Li , and Jiawei Zhu 

Dear Editor,

This letter presents a multi-automated guided vehicles (AGV) routing planning method based on deep reinforcement learning (DRL) and recurrent neural network (RNN), specifically utilizing proximal policy optimization (PPO) and long short-term memory (LSTM). Compared to traditional AGV pathing planning methods using genetic algorithm, ant colony optimization algorithm, etc., our proposed method has a higher degree of adaptability to deal with temporary changes of tasks or sudden failures of AGVs. Furthermore, our novel routing method, which uses LSTM to take into account temporal step information, provides a more optimized performance in terms of rewards and convergence speed as compared to existing PPO-based routing methods for AGVs.

In the highly competitive intelligent manufacturing industry, the AGVs based automated storage and retrieval system (AS/RS) has emerged as a competitive, efficient and reliable solution [1], [2], in which, AGVs are used to execute storage and retrieval missions due to their end-to-end capabilities, more efficient performance of storage and retrieval, reduced errors and labor costs, etc. [3]. Particularly, in the special period to fight an all-out global battle against COVID-19, using AGVs instead of people to transport items in semi-enclosed spaces, such as factories, workshops and warehouses, means reducing the risk of viral infection caused by the transport of freight [4].

Considering an AS/RS can automatically complete storage and retrieval tasks with AGVs, the primary and significant problem is to provide a route for each AGV so that it reaches its target point and does not encounter static obstacles or dynamic conflicts due to other AGVs. In the meantime, the AS/RS stakeholders anticipate that the related indicators of the method can be optimized.

In response to this problem, there have been a number of studies on path planning for AGVs based AS/RSs. An analysis of literature indicates that a number of solutions employ intelligent methods as the basis for global path planning and combine the time window method, game theory and priority setting to avoid obstacles, such as the path planning method based on Dijkstra combined with the dynamic priority [5], A* algorithm with the queuing mechanism [6], ant colony optimization algorithm with the elastic time window [7], genetic algorithm with the time window [8], ant colony optimization algorithm with the game theory [9], the hybrid genetic particle swarm algorithm with the dynamic priority [10], etc. These solutions above, however, are centralized, in which there is a central processing node that takes into account the position of each AGV in time, and estimates conflicts between multiple AGVs following their intended

Corresponding author: Yishuai Lin.

Citation: Y. Lin, G. Hu, L. Wang, Q. Li, and J. Zhu, "A multi-AGV routing planning method based on deep reinforcement learning and recurrent neural network," *IEEE/CAA J. Autom. Sinica*, vol. 11, no. 7, pp. 1720–1722, Jul. 2024.

Y. Lin, G. Hu, and Q. Li are with the School of Computer Science and Technology, Xidian University, Xi'an 710000, China (e-mail: yslin@xidian.edu.cn; gangH@stu.xidian.edu.cn; qshli@mail.xidian.edu.cn).

L. Wang is with Suzhou Mingyi Intelligence Warehousing Information Technology Co., Ltd., Kunshan 215300, China (e-mail: william_wang@emyiw.com).

J. Zhu is with the School of Information Engineering, Chang'an University, Xi'an 710000, China (e-mail: jiawei.zhu@chd.edu.cn).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/JAS.2023.123300

routes[11]. In the other words, if a sudden change occurs, such as a temporary addition of a task or an AGV departing from its intended path, these centralized solutions may need to reroute all AGVs [12].

Therefore, several recent publications have utilized reinforcement learning algorithms for AGVs routing planning [13]–[15], in order to increase the adaptability of pathing planning methods, in which each AGV acts as an Agent following a unique strategy and searching for its routes in accordance with the real-time environment of the AS/RS. Based on our review of the literature on reinforcement learning methods, it can be found that existing methods for planning routes of AGVs do not take into account the temporal step information. However, referring to the routing problem in other scenarios, for example, the routing planning for unmanned aerial vehicles [16] and the pathing planning for unmanned vehicles [17], it can be demonstrated that information about timing relationships is crucial for avoiding conflicts in advance.

Consequently, in the letter, with firstly considering the temporal step information of AGVs for the multiple AGVs routing planning in the multi-AGV based AS/RS, we propose a method of multi-AGV routing planning based on DRL and RNN, specially utilizing PPO and LSTM, in order to emphasize the importance of timing relationships in multi-AGV path planning, enhance the influence of past actions on current actions, and assist the AGV in planning an optimal path that is conflict-free as efficiently as possible.

System specification: An AS/RS using AGVs consists of several basic components, namely storage racks/shelves, input/output work centers, which are areas where AGVs can load and unload items, as well as charging stations for AGVs, and obstacles such as pillars and conveyor belts. In order to deliver items, the driving directions of the AGV include straight, backward, left, right, and wait in place.

As shown in Fig. 1, it is an AS/RS with 180 shelves, in which, there are 12 rows of shelves and each row contains 15 shelves. Precisely, 6 shelves (in 3 rows and 2 columns) are in a group, which are tightly packed without any gaps between them. There is a one-way channel between different groups of shelves for one AGV to pass. Additionally, at four corners of the AS/RS system, there are multiple input/output points, which can be dynamically opened or closed in response to changing business requirements. During the experiments, we design all four points opening as output points.

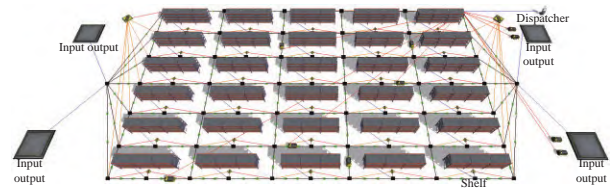


Fig. 1. A multi-AGV based AS/RS.

Meanwhile, we describe the process of AGVs executing outbound tasks. In response to the arrival of a batch of output tasks, following the scheduling results, one assigned AGV is required to start from its present location, travel to the target shelf to pick up the item, and then deliver it to the output work center. The AGV then moves to an unoccupied location to await its next assignment.

Routing planning method: To complete the inbound/outbound tasks with multiple AGVs, each AGV must have a feasible routing to avoid collisions with other AGVs and static obstacles. A routing planning method for AGVs based on DRL and RNN is presented in this section, in which each Agent simulates an AGV and searches its path based on the PPO with LSTM network.

1) State design: According to characteristics of AGVs running in the AS/RS environment, it is our designed state of the AGV that at the time t , the state of the AGV s_t includes four information, position information ($info_p$), obstacle information ($info_o$), target distance information ($info_d$) and AGV spacing information ($info_s$).

Precisely, $info_p$ is the information concerning the AGV's current

location and the target location and presented by the coordinate points of two locations, which is normalized to facilitate the AGV to establish the connection between the current position and the target position through the neural network. $info_o$ is the information that is mainly aimed at the perception of the surrounding environment during the AGV's travel, including the perception of static obstacles and dynamic obstacles. It is presented by 0 or 1 to detect whether there are obstacles at the positions of all the next possible actions. Moreover, $info_d$ represents the Euclidean distance between the next possible position of the AGV and the target point. $info_s$ is the information indicating the spacing distance (Euclidean distance) between the AGV and other AGVs, which is useful for preventing conflicts between them. Accordingly, the state information of multiple AGVs is combined to form joint state information s_{all} presented as (1). Meanwhile, we normalize and process the data separately due to the fact that the data returned by the system is in a variety of measurement formats, and then merge them into time series.

$$s_{all} = \{info_{p_1}, info_{o_1}, info_{d_1}, info_{s_1}, \dots, info_{p_n}, info_{o_n}, info_{d_n}, info_{s_n}\}. \quad (1)$$

2) Reward shaping design: The reward function is one of the keys to reinforcement learning convergence. In our proposed method, we design dense rewards for every action taken by the AGV. Specifically, the corresponding reward r is determined by the sum of five parts as shown in (2).

$$r = r_d + r_g + r_w + r_t + r_o. \quad (2)$$

To be precise, r_d represents the reward associated with reaching the target point. When executing the outbound task, two target points appear sequentially, including the location of the shelf storing the item and the input/output points. If the action can enable the AGV to reach the target point, the reward value is 5, otherwise it is 0. Moreover, r_g refers to the reward for making the AGV travel close to the target point as a result of performing the action. In this case, the reward is valued at 5. Conversely, a negative reward of -5 is provided. Furthermore, r_w is the waiting reward for making the AGV explore more and reducing unnecessary waiting actions. The value of the reward is defined as -1 if the action of the AGV is waiting, otherwise it is 0. Additionally, r_t is the step reward of the AGV, and for every action performed by the AGV, a reward of -1 is returned. r_t aims to encourage the AGV to reach the target point in as few steps as possible. Lastly, r_o is the reward focusing on whether the AGV encounters the static or dynamic obstacles. A negative reward of -2 appears for the AGV encountering obstacles, which encourages the AGV to search for a feasible path that does not collide with static obstacles or conflict with dynamic obstacles.

3) Model architecture: Based on the state design and the reward definition, we explain how to get the result of actions possibilities and the state value using PPO with LSTM.

In our designed model, generally, each Agent is provided with its own actor network and critic network. The actor network guides the agent's actions. The critic network evaluates the quality of the current state and guides the actor network to make appropriate updates. As shown in Fig. 2, we use a sequence step length of five. That is to say, at the time t , the original state of one AGV with four different information in the four previous moments ($s_{t-4}, s_{t-3}, s_{t-2}, s_{t-1}$) and

its state of the present moment (s_t) can be obtained.

In particular, based on the obtained original states, the actor network firstly classifies these states by four types of information, as position information, obstacle information, target distance information and AGV spacing information, which are defined in the subsection State Design. Additionally, feature extraction is performed on the segmented features, which are processed through two fully connected layers containing 32 and 16 neurons in the hidden layer, respectively. Following the extraction of features, they are fused and recombined into time series data, which is used as the input for the LSTM. Specially, only the last hidden state is selected as the result of feature extraction, which is as the output of LSTM transmitting to the fully connected layer network, so that the probability of each action under the current state s_t can be calculated separately. Each AGV's action at this time is selected at random from all probable actions it is likely to take at this time. Meanwhile, the action mask is used to help the AGV filter out the actions that will conflict with static obstacles and dynamic obstacles at the state s_t , as well as to enhance the efficiency of the AGV to explore the environment of the AS/RS.

In the other side, the structure of critic network is generally similar to the structure of actor network described above. It is the difference that after the LSTM outputs the feature structure of the current state of the AGV, it is calculated that the value of the current state s_t .

Experiments: A series of experiments are designed and analyzed for confirming the effect of our work on the method performance.

1) Experimental settings: The data from our experiment has been taken from an actual AGVs-based AS/RS in the Jiangsu province in China. We generate outbound tasks by randomly selecting tasks per week, and simulate a real AGVs based AS/RS environment with 180 shelves as described in the above section. Additionally, two different numbers of available AGVs (5 AGVs and 10 AGVs) are considered in this environment. All available AGVs are randomly assigned to delivery tasks, and the speed of each AGV is set to be 1.0 m/s.

Moreover, we choose a method [11] as a comparison method, which is a AGVs pathing planning method using the distributed PPO algorithm for training, but without LSTM, so that the timing relationships in AGVs path planning are not concerned. As a compared method, we derived the major idea of the method, implemented it, and applied it to our AS/RS system environment.

Furthermore, the maximum number of training episodes is set at 800 with 50 steps per episode. When either the maximum number of episodes has been reached, or when the sum of rewards returned by the episode is stable and reaches 400 in experiments of 5 AGVs and 800 in experiments of 10 AGVs, the algorithm will stop. The related parameters of the experiments are listed in the Table 1.

Table 1. Experimental Parameters

Parameter	Value
Learning rate	0.0001
Length of timing data for LSTM input	5
The number of LSTM hidden neurons	32
Reward decay	0.99
PPO cutting factor	0.2
The maximum number of steps of AGV	50

2) Experiments analysis: AGV path planning methods are generally evaluated using two types of metrics. The first are metrics related to the planned routes. During the experiments, we record the path length, the number of waits, the number of turns, and calculate the estimated delivery time of AGVs using the above data. The results indicate that our proposed method and the comparison method perform similarly in this regard. Other metrics are related to algorithm performance indicators, such as algorithm convergence speed and reward value. The following are primarily analyses of these indicators.

For each group experiment where different outbound tasks are assigned randomly, we calculate separately the sum of rewards returned by every action during routing pathing processes of all AGVs.

Fig. 3(a) is drawn based on the results of the system with 5 AGVs.

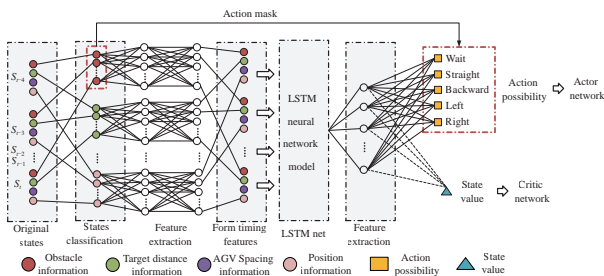


Fig. 2. PPO with LSTM network structure.

In this figure, the blue line represents the average value of all the sums of rewards returned in all related experiments of our proposed method, as well as the red line presents the average value of all the sums of rewards returned in all related experiments of the compared method respectively. Additionally, the blue shaded area behind the blue curve and the red shaded area behind the red curve show the range of all values of the sum of rewards obtained by our proposed method and the compared method respectively. Similarly, Fig. 3(b) illustrates the results of the system with 10 AGVs using lines and shaded areas in a similar manner as Fig. 3(a).

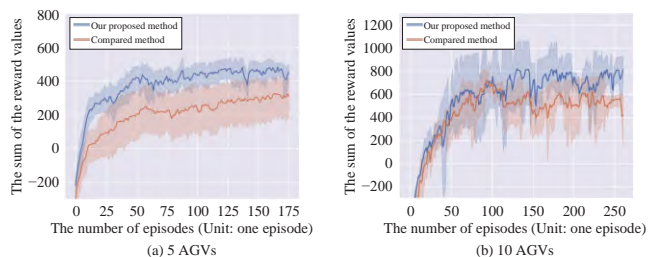


Fig. 3. Experimental results of the sum of rewards in the experiments of the system with 5 AGVs and 10 AGVs.

Accordingly, the comparison of curves and shaped areas in two figures indicates that, generally, the blue curve is higher than the red curve, and the blue shaded area is above the red shaded area. In other words, regardless of whether the system consists of 5 AGVs or 10 AGVs, the proposed method with PPO and LSTM can achieve better rewards than the compared method without LSTM.

A further concern is the comparison and analysis of the convergence speeds of two methods. We record the number of steps required for the method to converge in each group experiment where different outbound tasks are randomly assigned.

In the case of 5 AGVs in different group experiments, Fig. 4(a) shows the results of the number of steps required to reach convergence for the method. The blue pillars illustrate the average number of steps required for our proposed method to converge in each group experiment. Meanwhile, the red pillars represent the related results obtained using the compared method.

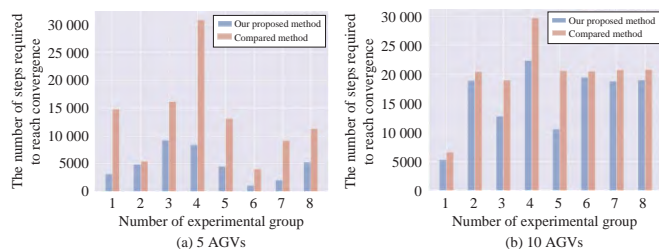


Fig. 4. Experimental results of the number of steps required to reach convergence in experiments with 5 AGVs and 10 AGVs.

It can be seen from Fig. 4(a) that every blue pillar is lower than its related red pillar. At the same time, there appears to be a similar pattern of results observed in the experiments of the system with 10 AGVs as shown in Fig. 4(b). In light of our experimental results, we conclude that our proposed method converges significantly faster than the compared method due to the fact that our proposed approach based on PPO-LSTM can improve the training speed.

Conclusions: Considering the timing relationships among multiple AGVs in pathing planning, we developed a multi-AGV routing planning method based on PPO and LSTM. Due to the AGV finding its way as it travels, this method can carry out temporary changes in tasks or sudden failures of AGVs with a greater degree of adaptability. The experiments were conducted while considering the AS/RS with different numbers of AGVs. According to our analysis of the experimental results, our work can provide better rewards and convergence speed than existing PPO-based routing method for AGVs.

Future research will focus on two directions. The first step is to

continue optimizing the method in order to reduce the computational time. Further, we devote to improving the method by evaluating the impact of two dynamic conflict avoidance strategies, such as AGV waiting and detours, in order to complete path planning by considering the smooth driving of AGVs as well as the routing time.

Acknowledgments: This work was supported by the National Natural Science Foundation of China (62202352, 61902039, 61972300), the Basic and Applied Basic Research Program of Guangdong Province (2021A1515110518), and the Key Research and Development Program of Shaanxi Province (2020ZDLGY09-04).

References

- [1] E. H. Grosse, C. H. Glock, and W. P. Neumann, "Human factors in order picking: A content analysis of the literature," *Int. J. Production Research*, vol. 55, no. 5, pp. 1260–1276, May 2017.
- [2] Y. Lin, Y. Xu, J. Zhu, *et al.*, "MLATSO: A method for task scheduling optimization in multi-load AGVs-based systems," *Robotics and Computer-Integrated Manufacturing*, vol. 79, p. 102397, Feb. 2023.
- [3] H. Yoshitake, R. Kamoshida, and Y. Nagashima, "New automated guided vehicle system using real-time holonic scheduling for warehouse picking," *IEEE Robotics and Auto. Letters*, vol. 4, no. 2, pp. 1045–1052, Apr. 2019.
- [4] M. Cardona, A. Palma, and J. Manzanares, "COVID-19 pandemic impact on mobile robotics market," in *Proc. IEEE ANDESCON*, Dec. 2020, pp. 1–4.
- [5] K. Guo, J. Zhu, and L. Shen, "An improved acceleration method based on multi-agent system for AGVs conflict-free path planning in automated terminals," *IEEE Access*, vol. 9, pp. 3326–3338, Dec. 2021.
- [6] Y. Lian, W. Xie, and L. Zhang, "A probabilistic time-constrained based heuristic path planning algorithm in warehouse multi-AGV systems," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 2538–2543, Jan. 2020.
- [7] Y. Yang, J. Zhang, Y. Liu, and X. Song, "Multi-AGV collision avoidance path optimization for unmanned warehouse based on improved ant colony algorithm," *Communications in Computer and Information Science*, vol. 1159, pp. 527–537, Apr. 2020.
- [8] T. Lu, Z. Sun, S. Qiu, and X. Ming, "Time window based genetic algorithm for multi-AGVs conflict-free path planning in automated container terminals," in *Proc. IEEE Int. Conf. Industrial Engineering and Engineering Management*, Jan. 2021, pp. 603–607.
- [9] Y. Zheng, L. Wang, P. Xi, *et al.*, "Multi-agent path planning algorithm based on ant colony algorithm and game theory," *J. Computer Applications*, vol. 39, no. 3, pp. 681–687, May 2019.
- [10] L. Z. Du, S. Ke, Z. Wang, *et al.*, "Research on multi-load AGV path planning of weaving workshop based on time priority," *Math. Biosci. Eng.*, vol. 16, no. 4, pp. 2277–2292, Mar. 2019.
- [11] S. Li, J. Zhang, and B. Zheng, "Research on obstacle avoidance strategy of grid workspace based on deep reinforcement learning," in *Proc. 2nd Asia-Pacific Conf. Communications Technology and Computer Science*, Feb. 2022, vol. 2022, pp. 11–15.
- [12] Y. Yang, J. Li, and L. Peng, "Multi-robot path planning based on a deep reinforcement learning DQN algorithm," *CAAI Trans. Intelligence Technology*, vol. 5, no. 3, pp. 177–183, Jun. 2020.
- [13] G. Shen, R. Ma, Z. Tang, *et al.*, "A deep reinforcement learning algorithm for warehousing multi-AGV path planning," in *Proc. Int. Conf. Networking, Communications and Information Technology*, Dec. 2021, pp. 421–429.
- [14] P. A. Corrales and F. A. Gregori, "Swarm AGV optimization using deep reinforcement learning," in *Proc. 3rd Int. Conf. Machine Learning and Machine Intelligence*, Dec. 2020, pp. 65–69.
- [15] L. Jiang, H. Huang, and Z. Ding, "Path planning for intelligent robots based on deep Q-learning with experience replay and heuristic knowledge," *IEEE/CAA J. Autom. Sinica*, vol. 7, no. 4, pp. 1179–1189, Jul. 2020.
- [16] N. Thumiger and M. Deghat, "A multi-agent deep reinforcement learning approach for practical decentralized UAV collision avoidance," *IEEE Control Systems Letters*, vol. 6, pp. 2174–2179, Dec. 2022.
- [17] C. Piao and C. Liu, "Energy-efficient mobile crowdsensing by unmanned vehicles: A sequential deep reinforcement learning approach," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6312–6324, Jul. 2020.