Exploring the Potential of Reinforcement Learning in Pipe Spool Scheduling in Industrial Projects

Mohamed ElMenshawy[1], Lingzi Wu[2], Brian Gue[3] and Simaan AbouRizk[1]

[1]Department of Civil and Environmental Engineering, University of Alberta, Canada
[2]Construction Management, University of Washington, USA
[3]PCL Industrial, Canada
elmensha@ualberta.ca, lingwu@uw.edu, bmgue@pcl.com, abourizk@ualberta.ca

Abstract –

Pipe spools are key components in industrial projects. Usually, they are built off-site in a fabrication shop and then shipped to the project location for installation. The fabrication shop deals with numerous spools, each designed to specific requirements according to shop drawings. The nature of pipe spools being engineered to order, together with production constraints such as lead time of materials, different processing times, and availability of resources, render the scheduling process within the shop challenging and time-consuming. As such, this research aims to automate the scheduling process by developing a reinforcement learning model that includes an agent that is capable of handling the scheduling process. The proposed model is applied to an illustrative example to investigate the concept of automating the scheduling process. The construction professionals highlight the great potential of the proposed model in the fabrication scheduling process, and its ability to minimize manual intervention.

Keywords –

Pipe spools; Reinforcement learning; Industrial projects

1 Introduction

Industrial projects include facilities like power plants, nuclear plants, and oil/gas production sites. Pipe spools—a main element in these facilities—are often fabricated off-site in fabrication shops that receive numerous orders accompanied by shipping schedules to construction sites [1,2]. A pipe spool comprises components like pipes, flanges, and elbows, each having varying lead times. The pipe spool is considered a unique product, being custom-made based on the shop drawings, engineered-to-order, and subject to frequent modifications due to design alterations or priority changes, availability of resources, processing times, and due dates [3]. All these features and constraints render the scheduling process challenging, laborious, and time-consuming. As a result, previous research has attempted to address the challenges of the scheduling process.

Metaheuristic algorithms have been explored as solutions for pipe spool scheduling [4,5]. Methods applied include genetic algorithms, artificial bee colonies, and ant colony optimizations [6-12]. Yet, their static nature limits their applicability in the dynamic and complex shop environment [13]. Simulation-based approaches have also been investigated as potential solutions including simulation models of pipe spool fabrication to study the fabrication process [14-20]. These models were used to study cycle time, bottlenecks, and resource utilization, which provides a level of support to construction professionals but does not offer a direct solution to the time-consuming nature or need for manual intervention in the scheduling process. Based on the aforementioned research efforts, we concluded that research related to automating the scheduling process and minimizing manual intervention remains relatively unexplored.

This research aims to bridge this gap using a reinforcement learning model to automate the scheduling process and reduce human manipulation. The reinforcement model is comprised of an agent, actions, environment, states, and rewards. The agent employs a dueling deep Q-network and experience replay where the agent stores past experiences while interacting with the environment. The agent observes the states within the environment, takes an action from the available actions, and then learns through a reward system.

This study advances the body of knowledge by exploring the potential of reinforcement learning models in the domain of construction management, specifically scheduling.

The remainder of this paper presents background information on pipe spool fabrication and reinforcement learning. We then provide a brief review of the state-of-the-art, identify research gaps, and discuss the potential of reinforcement learning in the scheduling process. The methodology section elaborates on the techniques and algorithms used, and then provides an illustrative example demonstrating our research's practicality.

2 Research Background

This section provides a brief review of the fabrication of pipe spools, and focuses on three main processes: fitting, welding, and inspection. Also, it presents an overview of reinforcement learning.

2.1 Pipe Spool Fabrication Process

The process of pipe spool fabrication encompasses multiple phases. This research focuses on the main fabrication processes: fitting, welding, and inspection. Pipes are the foundational element of spool assembly, being cut into specified sizes and shapes based on the shop drawings [20]. Once cut, the pipe moves to a fitting station where it is temporarily fixed with multiple components such as elbows, flanges, and reducers. Following this assembly stage, the semi-completed pipe spool moves to the welding station, either manually or lifted by machinery, based upon its weight and other handling requirements. The welding process ensures a permanent fixation of all components together [21,22]. Finally, the welded spool undergoes inspection, which represents the final stage in the process [3].

2.2 Reinforcement Learning

Reinforcement learning (RL) is a trending area of machine learning now used in many fields [23]. In RL, there are key components: agent, actions, environment, states, and rewards. An agent observes states in the environment, takes an action, and learns from interacting with its environment by receiving rewards or penalties, aiming to get more rewards and fewer penalties [23]. This learning process helps the agent make better decisions [24]. We can explain this using a Markov decision process (MDP), which includes states, actions, state transition probability matrix, reward functions, and discount factor [25,26]. This will be discussed in more detail in the methodology section.

3 Literature Review

This section explores two primary research areas: the application of RL in construction, and RL applications in job shop scheduling.

3.1 Reinforcement Learning in Construction

Scholars have explored the use of RL across multiple construction domains. Akanmu et al. [27] have presented a digital platform to train construction professionals encompassing wearable devices, RL, labor engagement, and monitoring tools. Mullapudi et al. [28] also developed a control strategy for stormwater systems that relies on an RL model to control the operations of valves, gates, and pumps.

3.2 Reinforcement Learning in Job Shop Scheduling

RL has been used to enhance dynamic scheduling in job-shop environments. Several researchers applied the Q-learning algorithm, where agents are taught to take action by choosing a dispatching rule to reduce tardiness in the process [29-32]. Yet, due to the large number of states in real-world production scenarios, Q-learning fails in practicality, as maintaining an extensive Q-table for such states becomes unfeasible.

Accordingly, there was a leap in estimating the Q-values by shifting towards deep reinforcement learning (DRL), which has demonstrated significant promise in job shop scheduling [33-35]. DRL, incorporating deep neural networks (DNNs), revolutionized the estimation of Q-functions [36].

A literature analysis reveals that there have not been any previous studies exploring the use of RL for scheduling in pipe spool fabrication. This finding aligns with the insights of Xu et al. [23], who assessed the current advancements in RL within construction engineering and management. The authors emphasize the limited number of studies incorporating RL in this domain, suggesting a need for increased focus [23]. Additionally, they highlight project scheduling and resource allocation as promising areas for future research.

The capabilities of RL make it a promising solution for pipe spool fabrication scheduling due to its strengths in the following areas: (1) navigating complex problems in changing environments with high dimensions; (2) adjusting to varied scenarios and states; (3) engaging in independent learning; and (4) predicting future action outcomes.

Building on this, our study introduces a pioneering RL-driven scheduling model tailored for pipe spool fabrication shops. The intention behind this model is to navigate the evolving landscape of such environments while addressing several inherent challenges in the scheduling area like resource allocation and adhering to due dates.

4 Model Development

Our model focuses on the fabrication process of pipe spools, specifically addressing the fitting stage. To develop the proposed model, the following information is required: start date, fitting processing time, required resources, and due date. As depicted in Figure 1, the model development involves two main stages: data preparation and RL. Each stage will be described in the coming subsections. The proposed model development process is designed to allow the integration of data related to pipe spools with an RL model. Once the RL model is trained on a training dataset, it can then be applied to an unseen dataset of pipe spools that need to be scheduled.

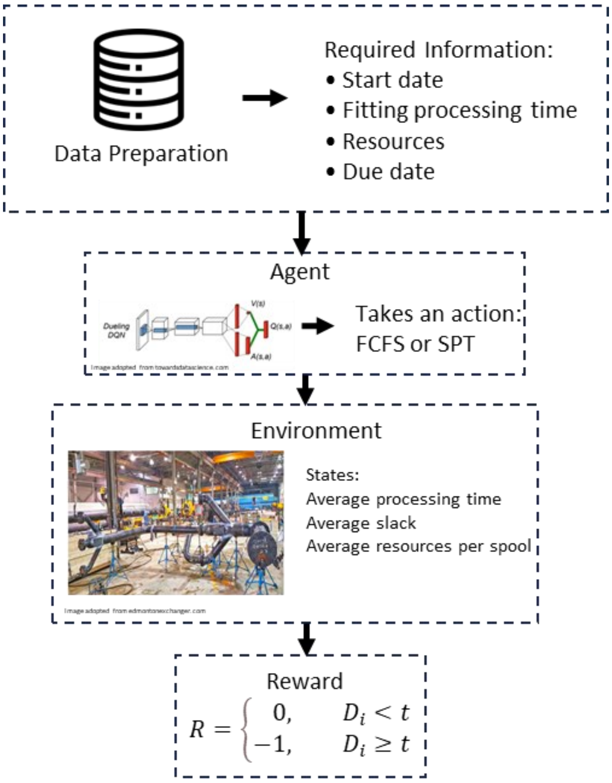

Figure 1. Methodology

4.1 Data Preparation

In this research, synthetic data is generated to mimic the spool information required to build the model. The generated dataset includes start date, fitting processing time, resources, and due date. The dataset includes approximately 1,900 records, and each record represents a spool. The dataset is then split into a training set with 80% of the records, and a testing set with the remaining 20% of the records.

4.2 Reinforcement Learning

This section describes the development of the RL model and formulates the scheduling process as a Markov decision process (MDP). Dueling DQN and prioritized replay are used in developing the agent.

4.2.1 MDP Formulation

As a general representation, the MDP can be described by five main components (S, A, P, R, γ) where S is the set of states, A is the set of actions, P is the probability transition matrix from one state to another, R is the reward function, and γ is the discount factor. Typically, in each time step t, the agent observes the states, takes an action a_t, then gets a reward or penalty r_t through the reward function R. It then moves to the next state s_{t+1} depending on P. While calculating the rewards, the effect of future rewards is taken into consideration using the discounting factor γ.

- State

The state is a crucial concept that represents the current status of the environment at a given time step, and based on the state, the agent determines its next action [37]. In our model, the state includes average processing duration for every spool, average slack, and average resources required for each spool.

- Action

Actions are crucial decisions made by an agent based on its observation of the current state. As the agent navigates the environment, it utilizes specific rules for its choices. In this context, the agent has two dispatching rules from which to select: first-come-first-serve (FCFS) or shortest processing time (SPT). The FCFS rule simply adheres to the order of spool arrival, processing spools in the sequence they were received, while the SPT rule prioritizes spools expected to be completed in the shortest duration. Given these choices, the agent dynamically identifies the optimal action for the current state during each decision-making time step.

- Reward

The reward function plays a crucial role in guiding the agent's decisions. It provides feedback from the environment in response to the agent's actions, effectively serving as an indicator of the agent's performance. The reward function deals with minimizing the number of spools that are not completed before their due date as demonstrated in Equation (1):

$$R = \begin{cases} 0, & D_i \leq t \\ -1, & D_i > t \end{cases} \quad (1)$$

where D_i represents the due date of the spool, and t represents the time step.

4.2.2 Deep Reinforcement Learning

In this model, we employ the Dueling Deep Q-Network (Dueling DQN) which builds upon the foundational principles of the Deep Q-Network (DQN). The Dueling DQN enhances the conventional DQN by decoupling the state values and the action advantages, refining the policy learning process.

The RL model is developed through the features of PyTorch, a powerful open-source machine learning framework related to RL algorithms [41]. Additionally, OpenAI's gym serves as the training ground for the RL model [42]. Coupling PyTorch's modeling prowess with OpenAI gym's environmental interface and training capabilities results in a refined and highly effective RL solution.

- Model Training

The Q-value serves as a foundational metric in the MDP, quantifying the anticipated discounted future reward when a specific action is taken. The ultimate aim of the agent is to develop an optimal policy that increases the expected rewards, a principle outlined by Mnih et al. [34].

However, employing a non-linear function estimator—such as a neural network—to approximate the Q-function presents challenges. Specifically, it can lead to instability or divergence. Two predominant strategies, as described by Wang et al. [38], offer solutions to those challenges. First, the experience replay method involves storing the agent's experiences. This data is then randomized, reducing correlations, and sampled in mini-batches to train the Q-network. The second strategy introduces an iterative update mechanism. By integrating a target Q-network into the DQN with parameter θ_i^-, correlations with targets are diminished. This target Q-network synchronises with the primary Q-network's parameters at periodic intervals, as illustrated by Mnih et al. [34].

Enhancing the DQN's architecture, we adopted the Dueling DQN. This refined structure improves efficiency by addressing Q-value overestimations. Notably, the Dueling DQN separates its estimations, separately determining the state value function and the action's advantage. These assessments then combine to forecast the action's quality, a technique explained by Liang et al. [39] and Wen et al. [40].

In each time step, the agent observes the state of the environment and chooses an action from one of the two dispatching rules: FCFS or SPT. Once an action is taken by the agent, the environment starts to send feedback in the form of reward or penalty, so that the agent can learn from this interaction and make sure to take better actions that maximize rewards in future time steps.

The model is trained on the training dataset, which is approximately 1,500 records and 10 resources for fitting the spools. Additionally, the following hyperparameters (Table 1) are used while training the model:

Table 1. Model hyperparameters

Parameter	Value
No. of training episodes	100
Learning rate	0.0025
Minibatch size	32
ε	0.9
γ	0.9

5 Model Results and Interpretation

After training, the agent was introduced to the testing dataset containing pipe spool records unseen by the agent. Consequently, the agent was evaluated on this dataset, making decisions to maximize rewards and concurrently reducing the number of spools that were not completed before their due date. Figure shows the average reward during the evaluation phase, based on the respective dispatching rule selected by the RL agent.

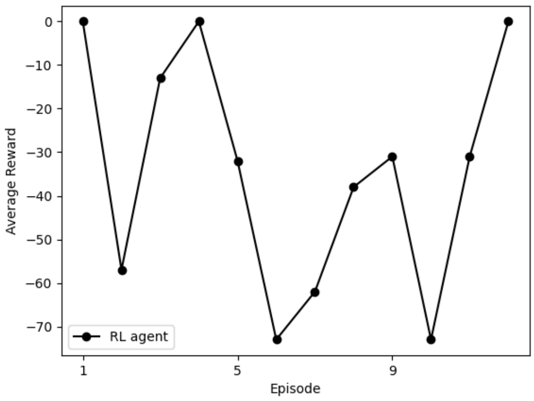

Figure 2. Average reward per episode (Exp #1)

Another experiment was conducted and the model was trained using 300 episodes—an episode is a complete cycle of scheduling all the pipe spools in the training dataset—and a learning rate of 0.001 in lieu of 100 and 0.0025, respectively. Figure 3 shows that the model's performance was relatively improved as the lowest average reward reached -50 when compared to the first experiment where the lowest average reward reached approximately -70.

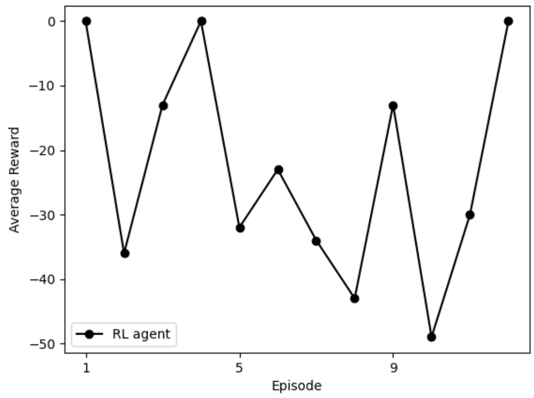

Figure 3. Average reward per episode (Exp #2)

A third experiment was conducted and the model was trained using 500 episodes and a learning rate of 0.001. Figure 4 shows that the model's performance was impacted as the lowest average reward reached approximately -140 when compared to the first and second experiments where the lowest average reward reached -70 and -50, respectively. As such, these hyperparameters had a negative impact on the model performance. Future work will investigate the hyperparameters that improve the model performance.

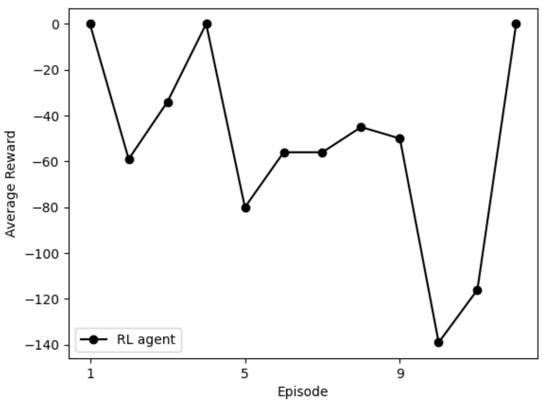

Figure 4. Average reward per episode (Exp #3)

6 Contributions and Future Work

This research introduces an innovative method for automating the scheduling of pipe spool fabrication in industrial construction projects. It uses RL, specifically employing Dueling DQN with an experience replay buffer that stores the agent's previous experiences. This method proves that an agent can be developed to schedule tasks in the simulated fabrication shop, reducing human intervention. This work serves as an initial exploration into applying RL for pipe spool scheduling in a fabrication shop.

The research was then applied to an illustrative example that employed an artificial dataset that mimicked the data collected from the fabrication shop. The analysis highlights the significant potential of incorporating RL into scheduling, offering industry professionals a decision support tool for pipe spool scheduling.

These preliminary results show that the proposed RL model has the potential to outperform traditional methods, which often require human input to deal with continuous changes. This research also refines the automation of scheduling, emphasizing the main constraint of adhering to due dates, which is imperative to preventing onsite installation delays.

The study includes certain limitations that must be addressed in future research. First, the study focused on the fabrication process only, specifically the fitting operation. Second, only two of the basic dispatching rules were included as actions to be taken by the agent. Third, a simple straightforward reward function was used to calculate the rewards/penalties based on the agent's actions. Finally, the model should be trained and evaluated on a real-world dataset from a fabrication shop. Future efforts should expand this research by encompassing diverse shop operations, by comparing the performance of the agent against the dispatching rules, and by assessing the agent's ability to shift between those dispatching rules.

7 Acknowledgements

This project was supported by a Collaborative Research and Development Grant (572086-22) from the Natural Sciences and Engineering Research Council of Canada. We would also like to express our gratitude to PCL Industrial Management Inc. for their continued support throughout the projects, and Brann Munro for her assistance with paper proofreading.

8 References

[1] Mosayebi S. P., Fayek A. R., Yakemchuk L., and Waters S. Factors Affecting Productivity of Pipe Spool Fabrication. *International Journal of Architecture, Engineering and Construction*, 1(1):30–36, 2012.

[2] Mohamed Y., Borrego D., Francisco L., Al-Hussein M., Abourizk S., and Hermann U. Simulation-based scheduling of module assembly yards: Case study. *Engineering, Construction and Architectural Management,* 14(3):293–311, 2007.

[3] Mohsen O., Petre C., Mohamed Y. Machine-Learning Approach to Predict Total Fabrication

Duration of Industrial Pipe Spools *Journal of Construction Engineering and Management, 149(2),* 2022.

[4] Safarzadeh S., Shadrokh S., and Salehian A. A heuristic scheduling method for the pipe-spool fabrication process. *Journal of Ambient Intelligence and Humanized Computing,* 9(6):1901–1918, 2018.

[5] Moghadam A. M., Wong K. Y., Piroozfard H., Asl Derakhshan A., and Shanty Hutajulu T. Solving an industrial shop scheduling problem using genetic algorithm," in *Advanced Materials Research*, 845: 564–568, 2014.

[6] Xing L. N., Chen Y. W., Wang P., Zhao Q. S., and Xiong J. A Knowledge-Based Ant Colony Optimization for Flexible Job Shop Scheduling Problems. *Applied Soft Computing Journal*, 10(3): 888–896, 2010.

[7] Bagheri A. and Zandieh M. Bi-criteria flexible job-shop scheduling with sequence-dependent setup times - Variable neighborhood search approach. *Journal of Manufacturing Systems*, 30(1):8–15, 2011.

[8] Zandieh M., Khatami A. R., and Rahmati S. H. A. Flexible job shop scheduling under condition-based maintenance: Improved version of imperialist competitive algorithm. *Applied Soft Computing*, 58: 449–464, 2017.

[9] Xu Y., Wang L., Wang S. yao, and Liu M. An effective teaching-learning-based optimization algorithm for the flexible job-shop scheduling problem with fuzzy processing time. *Neurocomputing*, 148: 260–268, 2015.

[10] Mokhtari H. and Hasani A. An energy-efficient multi-objective optimization for flexible job-shop scheduling problem. *Computers & Chemical Engineering,* 104:339–352, 2017.

[11] Gao K. Z., Suganthan P. N., Pan Q. K., Chua T. J., Chong C. S., and Cai T. X. An improved artificial bee colony algorithm for flexible job-shop scheduling problem with fuzzy processing time. *Expert Systems with Applications,* 65:52–67, 2016.

[12] Chang H. C. and Liu T. K. Optimisation of distributed manufacturing flexible job shop scheduling by using hybrid genetic algorithms. *Journal of Intelligent Manufacturing*, 28(8):1973–1986, 2017.

[13] Kardos C., Laflamme C., Gallina V., and Sihn W. Dynamic scheduling in a job-shop production system with reinforcement learning. In *Procedia CIRP*, pages 104–109, 2020.

[14] Lu M., Abourizk S. M., and Hermann U. Sensitivity analysis of neural networks in spool fabrication productivity studies. *Journal of Computing in Civil Engineering*, 15(4):299-308, 2001.

[15] Wang, P., Mohamed, Y., Abourizk, S. M., & Rawa, A. T. Flow Production of Pipe Spool Fabrication: Simulation to Support Implementation of Lean Technique. *Journal of Construction Engineering and Management*, 135(10):1027-1038, 2009.

[16] Song, L., Mohamed, Y., & AbouRizk, S. M. Early Contractor Involvement in Design and Its Impact on Construction Schedule Performance. *Journal of management in engineering*, 25(1): 12-20, 2009.

[17] Hu D. and Mohamed Y. Simulation-Model-Structuring Methodology for Industrial Construction Fabrication Shops. *Journal of construction engineering and management,* 140(5), 2014.

[18] Taghaddos H., Hermann U., AbouRizk S., and Mohamed Y. Simulation-Based Multiagent Approach for Scheduling Modular Construction, *Journal of Computing in Civil Engineering*, 28(2):263–274, 2014.

[19] Song L., Wang P., and AbouRizk S. A virtual shop modeling system for industrial fabrication shops. *Simulation Modelling Practice and Theory*, 14(5):649–662, 2006.

[20] Sadeghi N. and Fayek A. R. A framework for simulating industrial construction processes. In *Proceedings of Winter Simulation Conference*, pages 2396–2401, 2008.

[21] Wang P. and Abourizk S. M. Large-scale simulation modeling system for industrial construction," *Canadian Journal of Civil Engineering*, 36(9):1517–1529, 2009.

[22] Hu D. and Mohamed Y. Pipe spool fabrication sequencing by automated planning. In *Construction Research Congress 2012: Construction Challenges in a Flat World,* pages 495–504, 2012.

[23] Xu Y., Zhou Y., Sekula P., and Ding L. Machine learning in construction: From shallow to deep learning. *Developments in the Built Environment*, 6, 2021.

[24] Nguyen H. and La H. Review of Deep Reinforcement Learning for Robot Manipulation. In *Proceedings - 3rd IEEE International Conference on Robotic Computing,* pages 590–595, 2019.

[25] Nian R., Liu J., and Huang B. A review On reinforcement learning: Introduction and applications in industrial process control," *Computers and Chemical Engineering*, 139, 2020.

[26] Levine S., Kumar A., Tucker G., and Fu J. Offline Reinforcement Learning: Tutorial, Review, and Perspectives on Open Problems, http://arxiv.org/abs/2005.01643, 2020.

[27] Akanmu A. A., Olayiwola J., Ogunseiju O., and McFeeters D. Cyber-physical postural training system for construction workers, *Automation in construction*, 117, 2020.

[28] Mullapudi A., Lewis M. J., Gruden C. L., and Kerkez B. Deep reinforcement learning for the real time control of stormwater systems. *A Advances in water resources,* 140, 2020.

[29] Aydin M. E. and Öztemel E. Dynamic job-shop scheduling using reinforcement learning agents. *Robotics and Autonomous Systems*, 33:169-178, 2000.

[30] Wei, Y., & Zhao, M. A reinforcement learning-based approach to dynamic job-shop scheduling. *Acta Automatica Sinica*, 31(5), 2005.

[31] Shahrabi J., Adibi M. A., and Mahootchi M. A reinforcement learning approach to parameter estimation in dynamic job shop scheduling. *Computers & Industrial Engineering,* 110:75–82, 2017.

[32] Bouazza W., Sallez Y., and Beldjilali B. A distributed approach solving partially flexible job-shop scheduling problem with a Q-learning effect. *IFAC-PapersOnLine,* 50(1):15890–15895, 2017.

[33] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. Playing Atari with Deep Reinforcement Learning, *arXiv preprint arXiv:1312.5602,* 2013.

[34] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G. and Petersen, S. Human-level control through deep reinforcement learning," *Nature*, 518(7540):529–533,2015.

[35] Li Y. Deep Reinforcement Learning: An Overview. *arXiv preprint arXiv:1701.07274*, 2017.

[36] Waschneck, B., Reichstaller, A., Belzner, L., Altenmüller, T., Bauernhansl, T., Knapp, A., & Kyek, A. Optimization of global production scheduling with deep reinforcement learning, In *Procedia Cirp 72*, pages 1264-1269, 2018.

[37] Kim T., Kim Y. W., Lee D., and Kim M. Reinforcement learning approach to scheduling of precast concrete production. *Journal of Cleaner Production*, 336, 2022.

[38] Wang, Z., Schaul, T., Hessel, M., Hasselt, H., Lanctot, M., & Freitas, N. Dueling Network Architectures for Deep Reinforcement Learning. In *International conference on machine learning*, pages 1995-2003, 2016.

[39] Liang W., Xie W., Zhou X., Wang K. I. K., Ma J., and Jin Q. Bi-Dueling DQN Enhanced Two-stage Scheduling for Augmented Surveillance in Smart EMS," *IEEE Transactions on Industrial Informatics*, 2022.

[40] Wen S., Lv X., Lam H. K., Fan S., Yuan X., and Chen M. Probability Dueling DQN active visual SLAM for autonomous navigation in indoor environment. *Industrial Robot: the international journal of robotics research and application,* 48(3):359–365, 2020.

[41] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L. and Desmaison, A. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Advances in neural information processing systems*, 32, 2019.

[42] Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J. and Zaremba, W. OpenAI Gym, *arXiv preprint arXiv:1606.01540,* 2016.