Proceedings of the 2022 Winter Simulation Conference
B. Feng, G. Pedrielli, Y. Peng, S. Shashaani, E. Song, C.G. Corlu, L.H. Lee, E.P. Chew, T. Roeder, and P. Lendermann, eds.

REINFORCEMENT LEARNING-BASED TRANSPORTATION AND SWAY SUPPRESSION METHODS FOR GANTRY CRANES IN SIMULATED ENVIRONMENT

Namkyoun Kim
Minhyuk Jung
Inseok Yoon
Moonseo Park
Changbum R. Ahn

Department of Architecture and Architectural Engineering
Seoul National University
Gwanak-ro 1, Gwanak-gu
Seoul, 08826, SOUTH KOREA

ABSTRACT

To improve the productivity and safety of cranes, deep reinforcement learning (DRL) has received widespread attention as a framework for developing automated control methods. However, the major challenge of DRL is sample efficiency, which is further exacerbated by the operational and kinematic characteristics of the crane. Our study proposes an approach to improve the sample efficiency in training control policies for two subtasks: horizontal transportation and sway suppression. To do this, we built a simulation environment and defined the state of the environment and the reward. Then, we performed experiments to find out whether three DRL techniques (reward shaping, curriculum learning, and generative adversarial imitation learning) can mitigate the sample efficiency degradation caused by operational and kinematic characteristics. The results show that the techniques used in our experiment are effective in the improvement of the sample efficiency and learning performance of the DRL model for crane operation.

1 INTRODUCTION

Cranes are widely used in construction sites to transport heavy materials and are one of the major causes of dangerous accidents (Ramli et al. 2017; Fang and Cho 2017). Many previous studies have attempted to develop automated crane control methods to prevent accidents caused by operator error, and deep reinforcement learning (DRL) framework has received widespread attention for use in developing automated control methods (Zhao et al. 2020; Sallab et al. 2017). This framework has an advantage in that the control methods learn from collected data samples through interacting with an environment without explicit instructions (Yu 2018).

One of the key challenges of the reinforcement learning process is sample efficiency. Sample efficiency is the degree to which samples valid for training are included among the data (Botvinick et al. 2019). Low sample efficiency makes it necessary to interact with a huge number of environments to obtain the number of samples required for training a desired policy (Yu 2018; Kiran et al. 2021). When the reinforcement learning framework is applied to cranes, the sample efficiency problem is further exacerbated by two issues: operational and kinematic characteristics of the crane.

First, in terms of the operational characteristic, the conditions of a crane task vary according to the location of the destination, type of material, and so forth (Fang and Cho 2017). In order to achieve a reasonable control policy, it is necessary to provide a sufficient number of task execution environments that a crane agent might encounter in the real world (Whiteson et al. 2011; Matsumoto et al. 2020). In addition,

978-1-6654-7661-4/22/$31.00 ©2022 IEEE

since it takes a long time for a crane to complete a task, the number of samples constituting one task is large, and the rewards becomes sparse and delayed. As a result, the total number of samples required for training increases significantly, reducing sample efficiency.

Next, in terms of kinematics, cranes are underactuated systems in which the degree of freedom is greater than the number of manipulable actions (Sawodny et al. 2002). Due to this underactuated feature, the action of the crane causes a sway of the payload, making it difficult to predict the position of the payload (Yang et al. 2019). As a result, it complicates the distribution between the action and environment state in the samples and causes a decrease in sample efficiency.

Therefore, it is necessary to improve the sample efficiency that is degraded due to the characteristics of the crane. Our study aims to propose an approach to improve sample efficiency in developing crane automated control policies through reinforcement learning frameworks. To do this, we built a simulation environment in which an agent interacted with cranes. In addition, to define data samples for reinforcement learning, we selected the reward and state of the environment to be observed by the agent. We then proposed an approach using DRL assistive techniques such as reward shaping, curriculum learning (CL), and generative adversarial imitation learning (GAIL) to improve the sample efficiency of crane operation. We performed experiments to find out whether the proposed techniques (reward shaping, CL, and GAIL) can mitigate the learning performance degradation caused by operational and kinematic characteristics. The experiment results show that the DRL methods used in our experiment are effective in the improvement of sample efficiency and learning performance of the DRL model for crane operation.

2 REALTED WORK

Previous studies on crane automation with the DRL framework have focused on the sample efficiency problem. Ding (2018) used reward shaping and CL to improve the sample efficiency of reinforcement learning in set down tasks of an offshore crane. Reward shaping is a commonly used method to improve sample efficiency in reinforcement learning. Reward shaping guides agents through the learning process in a way that provides more frequent reward signals for appropriate actions (Grzes 2017). This type of reward is called dense reward, and th e study showed that reward shaping is useful in a domain in which reward is sparse due to the working environment of cranes. Next, CL is an assistive technique of DRL that gradually increases the difficulty of a problem. The agent transfers the experience gained from an easy problem to learning a more difficult task, thereby speeding up performance convergence (Narvekar et al. 2020). In this study, the problem difficulty was adjusted by increasing the dimensions of the environment, which is related to the state of the environment to be observed and the number of manipulable actions from one-dimension (1D) to two-dimension (2D). The results of the study showed that CL is particularly useful when the dimension of the environment is large and reward is sparse and delayed.

Matsumoto et al. (2020) applied reinforcement learning to the operation of two machines to reduce the sway of a load suspended on a mobile boom crane and to load soil into an excavator bucket. In the study , reward shaping, frame skipping, and imitation learning (IL) were used and evaluated to improve the sample efficiency of reinforcement learning. Frame skipping is a method in which the agent maintains the same action for several time steps, thereby reducing the combination of state and action space. It is useful for reducing the total number of samples but prevents fine grained action. IL uses a supervised learning scheme that mimics expert demonstrations using algorithms such as behavior cloning (Kober and Peters 2010). However, in the unseen episodes in the demonstration data, a serious distribution shift problem occurs, and the agent does not know which action to choose (Kiran et al. 2021). Therefore, in this study, the policy trained with IL was used only as the initial policy of the RL model to increase the training efficiency.

It is noted that studies on crane automation deploy DRL assistive techniques such as reward shaping, IL, or CL to resolve sample efficiency problems. However, discussion on how to apply the techniques for different tasks of cranes is still lacking. Operational and kinematic characteristics of cranes lower sample efficiency in different ways. Therefore, it requires approaches appropriate to the main causes of low sample efficiency on each subtask of cranes. In addition, there were insufficient training episodes for tasks to train a robust agent. Work conditions of tasks should be generated so as not to be limited to specific work through environment randomization.

3 DRL MODEL DEVELOPMENT

DRL is a machine learning field that develops sequential decision-making processes so that an agent observes the current state of an environment and takes an action that maximizes the reward. The policy chooses the action from a list of all options. At each step, the agent is in a certain state and the reward is assigned after completing an action and moving to the next state (Sutton and Barto 2018). Until the episode reaches the termination condition, the agent continues to interact with the environment.

This section describes the process of building an environment for the agent to explore, including the crane model that the agent will control, the action space, and the observation space. Then, we configured the training episode and designed the reward function to train the control policy networks for crane tasks. We used Unity 3D to create the simulated environments with which the crane agent interacted (Juliani et al. 2018).

3.1 Scope of the Method

Several types of cranes, such as overhead, gantry, and rotary, are widely used in many industrial sectors (Ramli et al. 2017). In our work, we chose a gantry crane operating in a precast concrete field yard in which there were a few factors that can affect learning performance. At the field yard, a gantry crane transports produced material into the yard and releases stocked material onto a truck.

The control of transportation is the most difficult task in a crane operation cycle and comprises three phases: payload lifting, horizontal transportation, and sway suppression (Wu and Xia 2014). Our work focused on two successive subtasks: horizontal transportation (Task 1) and payload suppression (Task 2). Task 1, which moved the load near the destination without excessive sway, is greatly affected by the characteristics of the operational environment (e.g., long time to complete the task and various conditions of a crane task). On the other hand, Task 2, which positioned the load accurately on the destination and removed payload sway for rigging works, is affected by the kinematics characteristics of the crane. Therefore, an approach to improve the sample efficiency of DRL should be presented considering the characteristics that mainly affect each task.

3.2 Agent Crane Modeling and Action Space

Figure 1(a) shows the layout of the workplace and task-related parameters. The crane agent had six discrete actions with three behaviors (i.e., backward, forward, and idle) in each action branch of gantry and trolley. The length of the cable was 5 m. The maximum speed of gantry and trolley motion were 20 and 30 m/min, respectively. Also, our agent controlled a gantry crane with two actuators in horizontal and single pendulum motion, as illustrated in Figure 1(b). We implemented the pendulum motion based on the joint distance constraint. A rope consisting of small segments and the joints that connected the small segments was used to imitate the physical behavior of the payload.

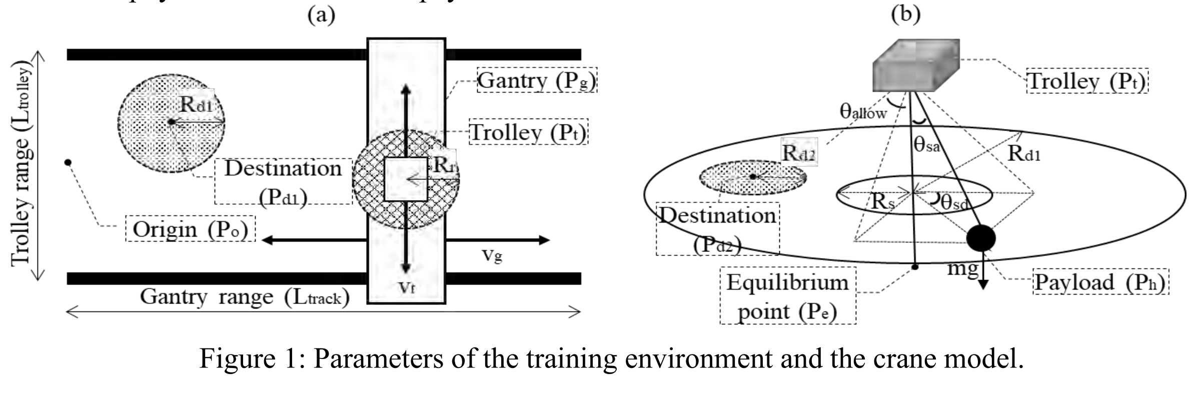

Figure 1: Parameters of the training environment and the crane model.

3.3 Observation Space

We investigated an observation space that facilitated training a policy network through a preliminary test before the experiments. In the preliminary test, each observation vector and reward criterion were applied to train the policy for at least 1 million time steps, and they were selected based on the learning speed of the policy. The agent directly observed the Cartesian position under the assumption that an external perception system (e.g., odometry and computer vision) would measure the state of the environment (e.g., the location of a destination and the status of a crane).

Table 1 describes the selected observation space, which consisted of the current state of each component of the crane in terms of the information of the location (e.g., location of the destination and distance) and the payload (e.g., angle, direction, and angular velocity). We separated sway information into P_e as location information, and θ_{sd} and θ_{sa} as sway information for both tasks. In addition, the relative coordinates were suitable for Task 2.

Table 1: Observation space and reward function of Tasks 1 and 2.

Observation space		**Reward function (value)**		
Task 1	Task 2	Type	Task 1	Task 2
Trolley position (P_t), equilibrium position (P_e), destination position (P_d), sway direction (θ_{sd}), and sway angle (θ_{sa})	Relative position between (P_e) and (P_d), relative position between (P_e) and (P_h), sway direction (θ_{sd}), sway angle (θ_{sa}), and angular velocity ($d\theta_{sa}/dt$)	Goal-related	Reach to destination (+1.0), sway angle over (-3.0/maxstep)	Under both thresholds of distance and energy (+1.0), distance over R_{d1} (-1.0)
		Dense	The closest distance to the current step in 1D (0.5/maxstep, + closer, - far)	Distance under R_{d2} (+0.15), distance over R_{d2} (-0.2)
		Encour-aging	Small deduction (-1.0/maxstep)	Small deduction (-1.0/maxstep)

3.4 Training Environment

Since the policy network for the two tasks required separate learning environments, we describe the process of building the environment and defining the reward function for each task.

3.4.1 Training Environment for Task 1

Training episode: In the traveling task environment, the gantry crane agent was trained to maneuver the hook horizontally to the desired position (P_{d1}) without exceeding the allowable sway angle (θ_{allow}) in the workplace (L_{track} = 85m, $L_{trolley}$ = 15 m). Figure 1 shows the layout and parameters of the worksite. R_{d1} was the threshold distance at which a reward was given when the hook reached the destination (P_{d2}). We randomized the positioning parameters to train a robust policy. After the position of the trolley (P_t) was assigned in the workspace, the destination (P_{d1}) was created at a distance more than R_r from the position of the hook (P_h) to prevent the crane from colliding with an obstacle or completing the task as soon as an episode started. The maximum time step of an episode for the task (maxstep0 was set to 8,000 to prevent infinite exploration.

Reward function: For the purpose of the task, when the distance between P_h and P_{d1} was less than the threshold (R_{d1}), the agent received a high-sparse reward for achieving a long-term goal. However, the reward could be dedicated to unlimited exploration with sparse feedback for the agent to learn from, especially in an environment in which countless steps are expected to complete a task. Therefore, we defined dense rewards using the following criteria: (1) event criteria: a closure far from the destination; (2) distance criteria: the distance at the current step, the minimum distance so far; (3) dimension criteria: 1D and 2D; and (4) distance measurement object: hook and equilibrium point. Through the preliminary test, the dense reward was given to the agent when the 1D distance from the current step was closer or farther than the minimum distance so far. The distance was measured by the position of the equilibrium point.

The payload sway reward, a larger value than the distance-related dense reward, was deducted whenever the payload angle exceeded θ_{allow}. Although the rewards depend on the type of task, a small negative reward was given to the agents at every time step to encourage early achievement of the task objective. Table 1 gives the list of reward functions.

3.4.2 Training Environment for Task 2

Training Episode: The purpose of the crane agent in Task 2 was to lower the residual sway and learn how to minimize the worktime to maneuver the hook accurately to the destination (P_{d2}) after Task 1. To provide the agent with various scenarios, sway angle (θ_{sa}) was set to within 15° and sway direction (θ_{sd}) was set in all directions at the start of an episode. P_{d2} was created in the shape of a torus. The major radius of the torus was R_{d1} and the minor radius was R_s to prevent immediate task completion. We set the maximum time step for the task episode to 2,000.

Reward function: To achieve the goal of the task, the payload must be located at P_{d2} within a distance threshold (R_{d2}), and the mechanical energy (E_{tot}) of a single pendulum should be lower than a certain energy threshold. E_{tot} is the sum of kinetic energy (E_k) and potential energy (E_p).

$$E_p = mgl(1 - \cos\theta_{sa}),\ E_k = ml^2\left(\frac{d\theta_{sa}}{dt}\right)^2,\ E_{tot} = E_p + E_k.$$

When the agent satisfied the two thresholds, a high-sparse and goal-related reward were obtained, and the episode ended. However, successful policy training is difficult when the sparse reward is used alone. Thus, we granted dense rewards to guide the agent to reach and stay inside P_{d2}. In this method, a small reward was given when the distance between the hook and P_{d2} was less than the distance threshold (R_{d2}), and a reward was lost when it departed from P_{d2} after obtaining the reward. On the other hand, as a failure condition, if the hook was farther from P_{d2} than R_{d1}, it was considered that the task had failed and the episode was terminated with a negative reward. Those rewards are summarized in Table 1.

4 EXPERIMENTS

4.1 Experimental Settings

In the interactive environment developed for this study, we trained the feed-forward neural networks based on data observed in the environment during crane operation. To find out how to efficiently train policies in terms of the required number of samples and performance of the policy, we conducted experiments with DRL assistive techniques, which are reward functions, CL, and GAIL.

Each training session was executed for at least 40 million time steps. The simulation time step per second was set to 50. Agents' policies are represented by a feed-forward neural network and optimized using Proximal Policy Optimization (PPO) algorithms (Schulman et al. 2017).

4.2 Experiments for the Transportation Task (Task 1)

Task 1 took a long time to complete because it moved the load over a long distance. Thus, in reinforcement learning frameworks, the sample efficiency decreases as rewards become delayed and sparse due to the scale of the workspace. In the experiments for Task 1, to figure out the effects of dense reward and CL on the reward sparseness problem, the training process and results were compared in four baselines according to whether dense rewards are added and whether CL is used: Default; Dense; CL; and Dense and CL. The CL adjusted the difficulty of lessons by gradually increasing L_{track} and decreasing R_d, as given in Table 2. L_{track} and R_d contributed to the sample efficiency with time-step and completion probability, respectively. When the agent gets a reward above the threshold, it moves on to the next lesson.

Table 2: Curriculum adaptation.

Lesson	Length of L_{track} (m)	Radius of R_{d1} (m)	Reward Threshold
1	From 49 to 25	From 1.9 to 2.5	0.60
2	From 69 to 45	From 1.4 to 2.0	0.65
3	From 85 to 65	From 1.0 to 1.5	0.70
4	85	1.0	-

4.3 Experiments for the Transportation Task (Task 2)

In Task 2, the sample efficiency was lowered by the state of the payload, which is complicated by the pendulum motion. In the experiment, GAIL was used to alleviate the problem. GAIL is a technique that rewards agents for imitating a series of demonstrations using an adversarial approach (Ho and Ermon 2016). In GAIL, the discriminator allocates rewards in proportion to the degree to which the agent's observations and behaviors are similar to those of the demonstration.

The training process was compared in three baselines according to the use of GAIL and the value of strength: Default; GAIL with the strength of 0.1 (GAIL 0.1); and GAIL with the strength of 0.9 (GAIL 0.9). Strength is one parameter of GAIL that determines how much of the demonstration data to imitate. We used it as one of the baselines because the expert demo can be suboptimal. We collected expert demonstrations for 100,000 steps in 180 episodes (550 steps per task). Four display screens were provided to the experts to collect the most optimal demonstration as possible, as shown in Figure 2.

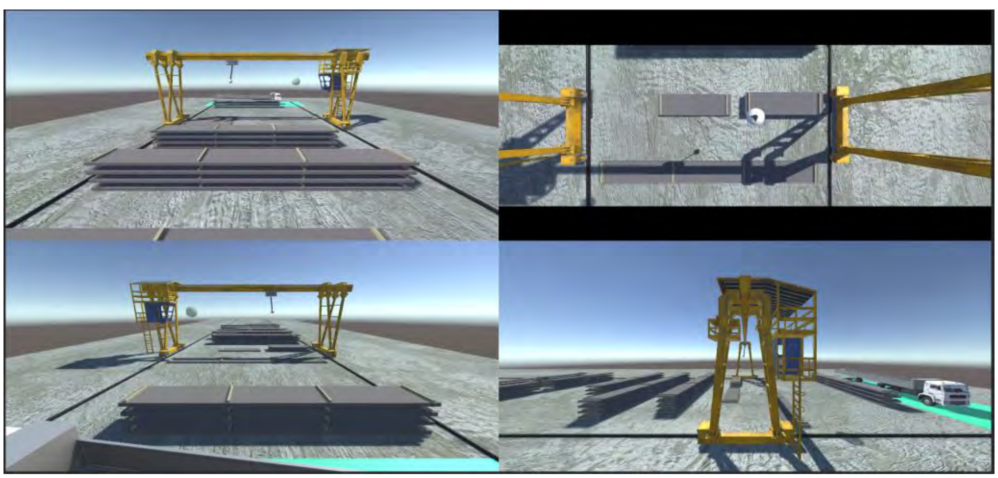

Figure 2: Training environment to collect experts' demonstrations.

4.4 Results and Discussion

This section demonstrates the training progress of the agents and compares the performance of the presented training methods in terms of cumulative reward and training speed.

Figure 3 describes the progress of learning policies in terms of the cumulative rewards of the baselines on Task 1: Default, Dense, CL and Dense and CL. In the case of the default policy, the agent failed to find an appropriate behavior in a given number of training steps (Figure 3(a)). This is due to the sparseness of the goal-achievement reward. The experimental results also show that there was a steady increase in dense. However, even if the maximum step of the session was increased, it did not converge to the task goal in the end (Figure 3(b)). The agent was focused on obtaining rewards from the dense reward.

On the other hand, in two cases: CL; and Dense and CL, the agents successfully learned the optimal policy with the reward increasing sharply after a certain threshold. This implies that once the optimal behavior is found, learning proceeds dramatically since the task is relatively easy. In our cases, dense and CL showed superior performance in finding the optimal policy with significantly fewer steps required for training. Although there is a policy loss in the section in which the lesson is passed, the agent adapted well to the changing work environment and found the optimal immediately.

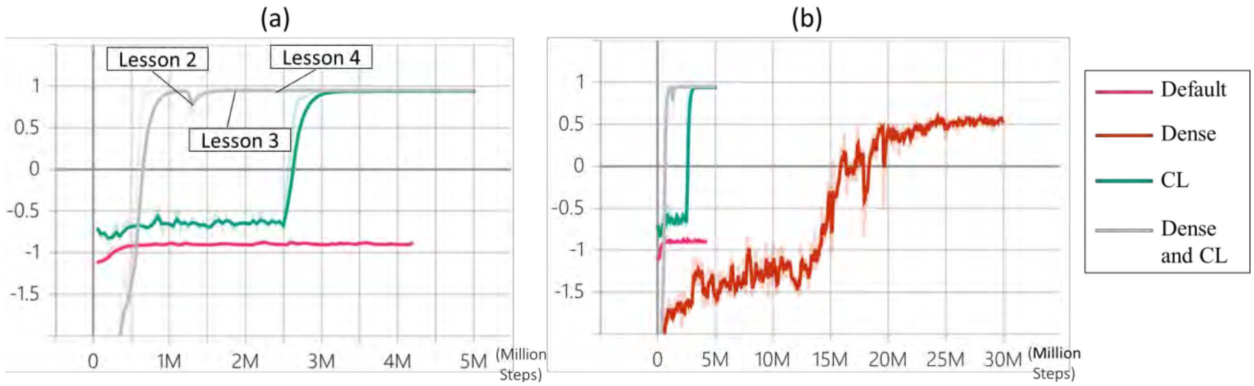

Figure 3: Learning curves of each baseline at Task 1: (a) 5M steps and (b) 30M steps.

Figure 4 shows the progress of different learning policies in Task 2: Default, GAIL 0.1, and GAIL 0.9. The training process of Task 2 was stable at the beginning, but the agents failed to reach the performance of the expert, in which the average reward was 0.61 and the average of the steps was 550. There are vast fluctuations in the whole process. This means that the agent is still looking for an optimal policy network. Meanwhile, GAIL 0.9 had a higher success rate while copying the demonstration at the beginning. Although the performance of the agents is not enough, we found that GAIL is effective for training when compared with default.

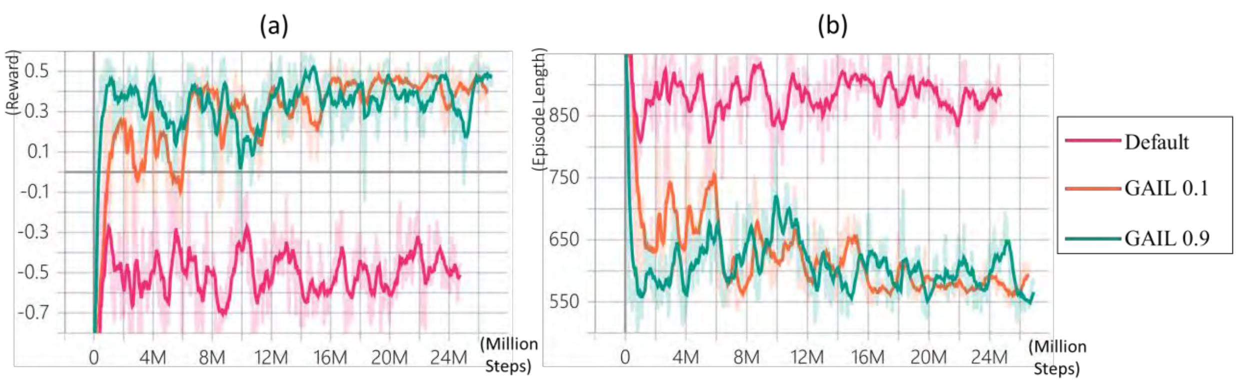

Figure 4: Task 2 (a) learning curves and (b) episode length of each baseline.

5 CONCLUSION

This research examined adopting a DRL framework to train control policies for two subtasks of a gantry crane and proposed an approach to improve sample efficiency while taking the characteristics of crane operation into account. We built interactive simulated environments for two subtasks of the crane and defined the data sample that facilitates learning through preliminary experiment. We then performed experiments to find out whether three DRL technology techniques (reward shaping, curriculum learning, and generative adversarial imitation learning) can improve the learning performance degradation caused by operational and kinematic characteristics. Considering the characteristics that the task is affected by, we proposed an approach to apply CL to Task 1 and GAIL to Task 2. The results of the experiments show that the proposed approach applying the DRL assistive methods (reward function, CL, and GAIL) are effective in training for crane control policy. Namely, the agent is trained quickly and the performance of the policy is higher. Specifically, the methods are adopted considering the cause of low sample efficiency: the operational and kinematic properties of cranes. The number of samples required for learning in Task 1 increased due to the characteristics of the crane's work environment, and CL that adjusts the work space and threshold distance was found to be effective. In Task 2, it was found that GAIL using expert demonstration data was effective for the agent to learn the complexity of crane operation.

The limitations of this study and recommendations for future research are as follows. The proposed approach is highly likely to be effective for overhead cranes and bridge cranes with similar operating systems to gantry cranes, but its effectiveness will be limited for rotary cranes with slewing motion. In the DRL framework for the rotary crane, it is required to newly define most of the observation space, reward function, and lessons in CL. Besides, we simplified the states of the crane and the environment by limiting obstacles and the crane's dynamic motion (e.g., length of the wire rope and motion speed). In future research, it would be necessary to develop a crane control policy in a higher dimension of the states.

ACKNOWLEDGMENTS

This research was supported by a grant (21CTAP-C163785-01) from Technology Advancement Research Program funded by Ministry of Land, Infrastructure, and Transport of Korean government, and was also supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No.2021R1C1C2011913).

REFERENCES

Botvinick, M., S. Ritter, J.X. Wang, Z. Kurth-Nelson, C. Blundell, and D. Hassabis. 2019. "Reinforcement Learning, Fast and Slow". *Trends in Cognitive Sciences* 23(5):408-422.

Ding, M. 2018. *Reinforcement Learning For Offshore Crane Set-down Operations*. Ph.D. thesis, Faculty of Science and Engineering, University of Groningen, Groningen, Netherlands. http://fse.studenttheses.ub.rug.nl/id/eprint/18927, accessed 13th May 2022.

Fang, Y. and Cho, Y.K. 2017. "Effectiveness Analysis from a Cognitive Perspective for a Real-time Safety Assistance System for Mobile Crane Lifting Operations". *Journal of Construction Engineering and Management* 143(4):p.05016025.

Grzes, M. 2017. "Reward Shaping in Episodic Reinforcement Learning". In *Sixteenth International Conference on Autonomous Agents and Multiagent Systems,* May 8th-12th. Sao Paulo, Brazil.

Ho, J. and S. Ermon. 2016. "Generative Adversarial Imitation Learning". In *Advances in Neural Information Processing Systems 29*, December 5th-10th, Barcelona, Spain, 4565-4573.

Juliani, A., V.P. Berges, E. Teng, A. Cohen, J. Harper, C. Elion, C. Goy, Y. Gao, H. Henry, M. Mattar, and D. Lange. 2018. "Unity: a General Platform for Intelligent Agents". *arXiv preprint*. https://arxiv.org/abs/1809.02627, accessed 9th February 2022.

Kiran, B.R., I. Sobh, V. Talpaert, P. Mannion, A.A. Al Sallab, S. Yogamani. and P. Pérez. 2021. "Deep Reinforcement Learning for Autonomous Driving: a Survey". *IEEE Transactions on Intelligent Transportation Systems* 23(6):4909-4926.

Kober, J. and J. Peters. 2010. "Imitation and Reinforcement Learning". *IEEE Robotics & Automation Magazine* 17(2):55-62.

Matsumoto, K., A. Yamaguchi, T. Oka, M. Yasumoto, S. Hara, M. Iida, and M. Teichmann. 2020. "Simulation-Based Reinforcement Learning Approach towards Construction Machine Automation". In *Proceedings of the International Symposium on Automation and Robotics in Construction*, October 27th-28th, Kitakyushu, Japan, 457-464.

Narvekar, S., B. Peng, M. Leonetti, J. Sinapov, M.E. Taylor, and P. Stone. 2020. "Curriculum Learning for Reinforcement Learning Domains: a Framework and Survey". *arXiv preprint*. https://arxiv.org/abs/2003.04960, accessed 11th April 2022.

Ramli, L., Z. Mohamed, A.M. Abdullahi, H.I. Jaafar, and I.M. Lazim. 2017. "Control Strategies for Crane Systems: a Comprehensive Review". *Mechanical Systems and Signal Processing* 95:1-23.

Sallab, A.E., M. Abdou, E. Perot, and S. Yogamani. 2017. "Deep Reinforcement Learning Framework for Autonomous Driving". *Electronic Imaging* 2017(19):70-76.

Sawodny, O., Aschemann, H. and Lahres, S. 2002. "An Automated Gantry Crane as a Large Workspace Robot". *Control Engineering Practice* 10(12):1323-1338.

Schulman, J., F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. 2017. "Proximal Policy Optimization Algorithms". *arXiv preprint*. https://arxiv.org/abs/1707.06347, accessed 11th April 2022.

Sutton, R.S. and A.G. Barto. 2018. *Reinforcement Learning: An Introduction*. Cambridge: MIT press.

Whiteson, S., B. Tanner, M.E. Taylor, and P. Stone. 2011, April. "Protecting against Evaluation Overfitting in Empirical Reinforcement Learning". In *2011 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*, April 11th-15th, 120-127.

Wu, Z. and X. Xia. 2014. "Optimal Motion Planning for Overhead Cranes". *IET Control Theory & Applications* 8(17):1833-1842.

Yang, R., C. Jiang, Y. Miao, J. Ma, X. Zhang, T. Yang, and N. Sun. 2019. "A Flexible Rope Crane Experiment System". *Applications of Modelling and Simulation* 3(1):11-17.

Yu, Y., 2018, July. "Towards Sample Efficient Reinforcement Learning". In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, July 13th-19th, Stockholm, Sweden, 5739-5743.

Zhao, W., J.P. Queralta, and T. Westerlund. 2020, December. "Sim-to-Real Transfer in Deep Reinforcement Learning for Robotics: a Survey". In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, December 1st-4th, Canberra, ACT, Australia, 737-744.

AUTHOR BIOGRAPHIES

NAMKYOUN KIM is an PhD candidate in the Department of Architecture and Architectural Engineering at Seoul National University. He received a bachelor's degree from SNU in 2016. In 2018, he graduated master's course for construction management at SNU. His research interests focus on sensing technology and reinforcement learning framework for automation of heavy equipment. His e-mail address is dewichon@snu.ac.kr.

MINHYUK JUNG is a Research Professor in the Department of Architecture and Architectural Engineering at Seoul National University. He completed his PhD course at Seoul National University in 2016. His research interests focuses on the improvement of construction and supply chain systems using (1) real-time data collection and analysis in a construction environment with machine learning, (2) construction system modeling using multi-agent simulation techniques, and (3) reinforcement learning-based autonomous operation system for construction equipment. His e-mail address is archidea914@snu.ac.kr.

INSEOK YOON is in an Ph.D. course in the Department of Architecture and Architectural Engineering at Seoul National University. He received a bachelor's degree from SNU in 2016. His research interests focus on simulation technology to improve productivity of construction projects. His e-mail address is yoon92411@naver.com.

CHANGBUM RYAN AHN is an associate professor in the Department of Architecture and Architectural Engineering at Seoul National University. He earned a Ph.D. degree in Civil Engineering from the University of Illinois at Urbana-Champaign. His research interests focus on data-driven techniques and smart-sensing technologies for analyzing and identifying humans' collective behavior patterns derived from wearable sensors and/or crowdsourced dataset, in order to design and build (1) safe construction workplaces, (2) smart and connected urban communities, and (3) intelligent and energy-efficient building systems. His email address is cbahn@snu.ac.kr.

MOONSEO PARK got into Department of Architecture of Seoul National University in 1985, completed the courses for a bachelor's degree in 1989, and graduated master's course for City Planning at SNU in 1992. In 1998-2001, he received master's degree and doctor's degree for Project Management in MIT. After graduation, he worked for the Dept. of Building in National University of Singapore as an Assistant Professor. Since 2005 he has been working as a Professor at the Department of Architectural Engineering of Seoul National University. Currently, his major research area is systematic approach for construction, knowledge-based construction etc. His email address is mspark@snu.ac.kr.