2021 IEEE 12th International Workshop on Computational Intelligence and Applications
November 6-7, 2021, Hiroshima, Japan

# A Study on Application of Curriculum Learning in Deep Reinforcement Learning

## Action Acquisition in Shooting Game AI as Example

Ikumi KODAKA
Graduate School of Advanced Engineering
Chiba Institute of Technology
Chiba, Japan
s20s3011ea@s.chibakoudai.jp

Fumiaki SAITOH
Faculty of Advanced Engineering
Chiba Institute of Technology
Chiba, Japan
fumiaki.saitoh@p.chibakoudai.jp

2021 IEEE 12th International Workshop on Computational Intelligence and Applications (IWCIA) | 978-1-6654-4425-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/IWCIA52852.2021.9626020

***Abstract*—In recent years, deep reinforcement learning has garnered significant attention because it can be applied to higher-dimensional environments compared with traditional reinforcement learning. However, the number of trials increases in behavior acquisition, particularly in tasks with high dimensions and sparse rewards. To improve learning speed, we apply curriculum learning, which improves the learning performance by changing the difficulty of the task in a stepwise manner, to the behavior acquisition of a shooting game as well as conduct experiments. We compare the learning performance with and without the application of curriculum learning and confirm the faster behavior acquisition of the shooting game AI through experimental evaluation. Additionally, we analyze and discuss the development of other tasks and an algorithm for automatic curriculum generation.**

***Index Terms*—Deep Reinforcement Learning; Curriculum Learning; DeepQ-Network; Game AI;**

## I. Introduction

In recent years, autonomous agents using reinforcement learning have been adapted to increasingly difficult tasks [1]. Reinforcement learning is a learning method in which an agent interacts with its environment, decides on a course of action based on a strategy, and receives a reward from the environment as an evaluation. The goal of the agent is to maximize the reward obtained through a series of actions. However, in environments where rewards are sparse or states are abundant, the experience of rewarding behavior may be insufficient or the search for states may be time consuming. Additionally, a significant number of trials are required to determine the optimal behavior [2].

Deep learning has garnered attention in the field of image recognition because of its ability to extract detailed features from input data. In this context, deep reinforcement learning [3], a combination of reinforcement learning and deep learning, has enabled adaptations to higher-dimensional environments such as images. However, as the number of states in the task increases, the number of trials required for learning in deep reinforcement learning behavior acquisition increases. The significant number of trials required to acquire an action is a longstanding problem in deep reinforcement learning.

## II. Objective

Whereas deep reinforcement learning can be applied to complex tasks with many states, the computational cost is prohibitive. In addition, when the reward is sparse, many trials are required to learn the optimal behavior owing to insufficient exploration [4]. A sparse environment implies that random action selection during exploration is extremely unlikely to be rewarded. This problem is directly related to the difficulty of the target task. Furthermore, it implies that for the same task, if the difficulty level is high (low), then many (few) trials are required for the agent to learn to acquire the behavior.

Therefore, we assumed that we can avoid sparse environments caused by high task difficulty by applying curriculum learning, in which the difficulty of the task is changed in a stepwise manner during learning. In this study, we conducted verification and experiments on the application of curriculum learning in deep reinforcement learning for game AI to reduce the number of trials before an agent acquires an action or to improve the learning performance based on the same number of trials. The techniques and insights gained from these experiments are expected to be applicable not only to game AI, but also to various automated tasks, and their applications are expected to expand and further develop in the future.

## III. Deep Reinforcement Learning

In reinforcement learning, an agent in an environment receives a state $s_t$ as input at time $t$ and selects an action $a_t$ based on a strategy $\pi$, resulting in a reward $r_{t+1}$ and a change in the state to $s_{t+1}$ [5]. This sequence of events is shown in Figure 1. The agent performs this series of actions in a trial-and-error manner and aims to maximize the final reward. Q-Learning, a type of reinforcement learning, treats the states as discrete and stores the Q-values of each action for each state in a Q-table. In a low-dimensional environment, all states can be represented in a Q-table; however, in an image-based

978-1-6654-4425-5/21/$31.00 ©2021 IEEE

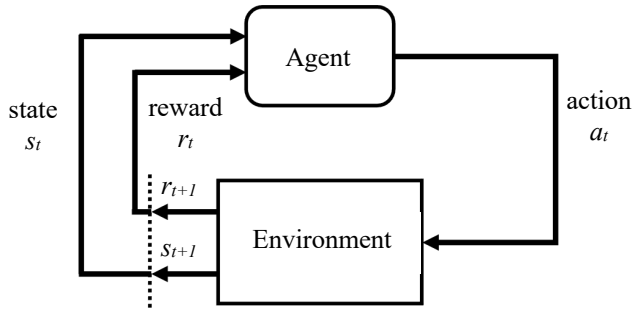


Fig. 1. Reinforcement Learning

environment, the input states become high dimensional. As the number of states increases, it becomes difficult to represent all of them in a Q-table; consequently, the measures cannot be calculated. Therefore, a deep Q-network (DQN), which uses deep learning to compute Q-learning measures, was proposed [6]. The DQN comprises a superior convolutional neural network that can extract fine features for high-dimensional states, such as images and Q-learning, and is a typical method involving conventional reinforcement learning algorithms. In the Atari 2600 game task, we obtained excellent results that were comparable or superior to human scores.

## IV. Curriculum Learning

Bengio et al. introduced curriculum learning as a machine learning concept to improve the learning efficiency in supervised learning [7]. In our experiments, we demonstrated that the prediction accuracy of models increased when the examples presented were simple at the beginning and became more difficult gradually as the learning progressed in language modeling and image classification. Narvekar et al. [8] applied this curriculum learning concept to reinforcement learning, where a new structure for curriculum learning was devised, as supervised learning and reinforcement learning are significantly different. In that study, the source of the target task, known as "Curriculum," was generated; subsequently, the agent on each task was trained, followed by the target task. Generating a task that is the source of the target task implies setting difficulty levels in stages until the target task, which is the goal, is reached. By gradually increasing this difficulty level as the agent learns, we improved the learning efficiency in reinforcement learning.

The application of curriculum learning in reinforcement learning has indicated excellent performance in various experiments, as well as in problems involving tasks that are difficult to learn using traditional reinforcement learning. Curriculum generation enables good performances to be achieved, thereby contributing to a faster learning process and the convergence of the target task [9-10].

## V. Experiment

The application of curriculum learning in deep reinforcement learning, which has been investigated in recent years, involves many experiments where the structure of the learning environment and the elements being built are changed during curriculum generation, and qualitative difficulty levels are set [11-13]. For example walls and obstacles that exist in the game stage are removed, the stage size is reduced to facilitate goal achievement, and the curriculum is applied through qualitative changes in difficulty. When applying the curriculum through qualitative change, the structure of the learning environment must be understood, and the characteristics of the obstacles during learning for the agent must be inferred. Subsequently, the structure of the learning environment as well as the elements of its construction must be modified, and a new environment must be created for each curriculum with a difficulty level that renders it viable as a curriculum.

However, studies regarding the application of curricula with quantitative changes in difficulty are scarce. Therefore, in this study, we investigated curricula with quantitative changes in the application of curriculum learning in deep reinforcement learning. Quantitative change implies that the difficulty level must be clearly defined as a numerical value. In terms of qualitative changes, when a Level 1-3 curriculum exists, Level 2 need not necessarily be twice as difficult as Level 1, and Level 3 need not be three times as difficult as Level 1. However, in the case of quantitative changes, the difficulty level must be clearly defined as a numerical value; therefore, Level 2 must be twice as difficult as Level 1, and Level 3 must be three times as difficult as Level 1. Hence, a shooting game was used as the subject of this experiment such that the curriculum can be applied through quantitative changes.

A shooting game is an environment where the player has to evade and attack the enemy while avoiding the enemy's bullets and hitting the enemy with his own attack. In this study, we generated a curriculum with varying difficulty levels by changing the number of bullets fired by the enemies. The enemy's projectiles were set to launch with an arbitrary probability at regular intervals; therefore, the difficulty can be set using this value. In other words, the difficulty of Level 1, which has a 10% probability of launching a bullet at a certain time, Level 2, which has a 20% probability of launching a bullet at a certain time, and Level 3, which has a 30% probability firing a bullet at a certain time, can be clearly expressed as Level 2 being twice as difficult as Level 1, and Level 3 being three times as difficult as Level 1. Theoretically, Level 2 is twice as difficult as Level 1, and Level 3 is three times as difficult as Level 1. By changing the probability of bullets launched by the enemies, a curriculum with quantitative difficulty levels can be created without changing the stage structure. Hence, we performed our experiment by exploiting the characteristics of this shooting game.

### A. Experimental environment

In this study, we prepared a shooting game as a learning environment for the agents. An example of this environment is shown in Figure 2. This environment is a vertical scrolling shooter with a stage (screen) size of 480 × 640. Figure 3 shows the four types of objects that comprise the game, i.e., the player, enemy, player bullets, and enemy bullets (from left to right). The environment maintains the variable step as a temporal concept, and all objects are updated based on the

rules at each step. The origin (x, y) = (0, 0) is assumed to be the upper left corner of the screen. Table 1 describes each object and the game rules.

When the player and enemy bullets, player bullets and enemy, or player and enemy are in contact with each other, each of them disappears. Eliminating an enemy results in a +1 reward, and eliminating a player earns a -1 reward. The game ends when the player disappears, and the sequence of events from start to finish is known as an episode. In other words, the game is a task for learning strategies to obtain more rewards in one episode.

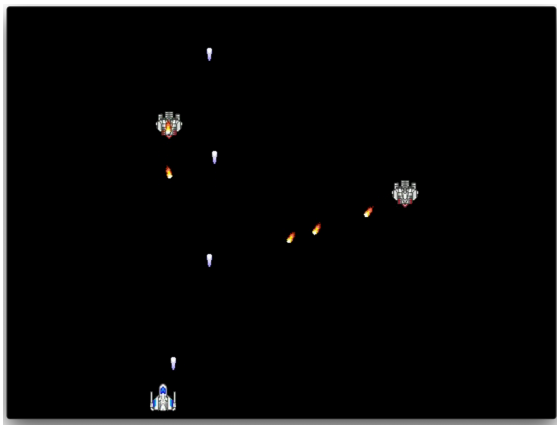

Fig. 2. Shooting Game

Fig. 3. Four types of objects

TABLE I. OBJECT AND THE GAME RULES

| objects | Rules |
|---|---|
| player (agent) | Appears at (455, 320) at the beginning. Can move +- 16px to x per step. |
| enemy | Appears at (100~440, 0) with a probability of 1/50 per step. Moves 2px to y and +1px to x per step (reverses every 100 steps) |
| player bullet | Appears at center of player every 15 steps. Moves -8px to y for each step. |
| enemy bullet | Appears at center of each enemy object at an arbitrary probability per step. Moves to the player point at the time of appearance at approximately 4px per step. |

It is noteworthy by changing the "arbitrary probability" underlined in the enemy bullet entry in Table 1, the change is quantitative, and this probability is defined as the difficulty level. For example, if the probability is set to 1/60, then each enemy object on the screen has a 1/60 probability of launching an enemy bullet at each step. If the probability is set to 1/30, then the difficulty will be twice as high as the previous probability of 1/60.

### *B. Overview of DQN*

An overview of the convolutional neural network used in the DQN for the experiments is shown in Figure 4. In the learning environment, the game screen was a 480 × 640 pixel RGB image, which is converted to an 84 × 84 pixel grayscale image when the agent observes it. As input to the DQN, four temporally sequential images were provided. Therefore, the input layer of the DQN used in this experiment was 84 × 84 × 4 dimensional data. The data were passed through three convolutional layers, and finally, two values were output at the output layer.

In the DQN used in this study, Boltzmann selection was used as the agent's action selection method [14]. Boltzmann selection is an action selection method in which each action is assigned a selection probability based on the Q-value obtained by learning. A is a set whose elements are actions; τ is a parameter that determines the amount of difference to be evaluate between each Q-value, and it is known as the temperature parameter.

$$\pi(s,a) = \frac{exp\left(\frac{Q(s,a)}{\tau}\right)}{\sum_{b \in A} \exp\left(\frac{Q(s,b)}{\tau}\right)} \quad (1)$$

The ε-greedy method [6], which is a general action selection method, cannot be regarded as an action selection method based on the learning process because the probability is determined by a predetermined value. In reinforcement learning, the balance between knowledge exploration and utilization is important. Hence, we used Boltzmann selection,

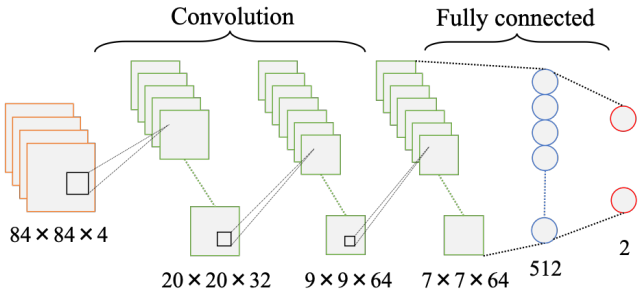


Fig. 4. Convolutional Neural Network

which uses learning-based selection probabilities, because it can be regarded as be an action selection method that allows for the flexible exploration and utilization of knowledge.

### C. Experimental setup

For our experiment, normal learning and curriculum-applied learning were represented by the DQN and CL, respectively. In this section, we compare them through experiments conducted on the DQN and CL. As mentioned earlier, the difficulty level is defined as the probability that an enemy bullet will be launched. Therefore, a probability of 1/30 was set as the highest difficulty level, and a curriculum was generated using it as the target task. In regard to the DQN, learning was performed in the target task for 10,000 episodes. In regard to the CL, the students learn based on the curriculum up to 6,000 episodes, after which they learn based on the target task for 4,000 episodes. In other words, we compared the difference in learning performance with and without the curriculum for up to 6,000 episodes using the results for the next 4,000 episodes. In addition, behavior acquisition in deep reinforcement learning has a high degree of randomness owing to its characteristics, and the learning results are likely to vary. Therefore, a five-trial experiment was conducted for each training, and the average of the experiments was used for performance comparison.

Table 2 shows the curriculum of the six patterns generated. The starting probabilities were 1/60, 1/90, and 1/120, where three and six steps were required to reach the highest difficulty level with a probability of 1/30. Each pattern is shown in Table 2. The values were obtained empirically from the results of preliminary experiments. Figures 5 and 6 show examples of the difference in enemy bullets depending on probability, using probabilities 1/30 and 1/120.

A summary of the experimental process is as follows.

Step1. Definition of the target task (DQN)

Determine the highest level of difficulty you want the agent to learn to be 1/30 of the probability.

Step2. Definition of the Curriculum (CL)

The curriculum should have three starting points: 1/60, 1/90, and 1/120, which are the target task probability multiplied by 1/2, 1/3, and 1/4. For each of these, we have a method of increasing difficulty every 1,000 episodes and a method of increasing difficulty every 2,000 episodes, for a total of six different curricula (Table 2).

Step3. Definition of the experimental method

For learning by DQN, 10,000 episodes are considered as one trial. In the case of CL learning, 6,000 episodes of curriculum-applied learning followed by 4,000 episodes of target-task learning is considered one trial. 5 trials of DQN and CL_A to F are performed respectively.

Step4. Definition of evaluation method

Average the rewards obtained in five trials for each of DQN and CL_A-F, and compare the transitions or values.

TABLE II. CURRICULUM SETUP

| Curriculum | Value of x with probability 1/x (1k=1,000episodes) | | | | | | |
|---|---|---|---|---|---|---|---|
| | 0~ | 1k~ | 2k~ | 3k~ | 4k~ | 5k~ | 6k~ |
| CL_A | 60 | 55 | 50 | 45 | 40 | 35 | 30 |
| CL_B | 60 | | 50 | | 40 | | 30 |
| CL_C | 90 | 80 | 70 | 60 | 50 | 40 | 30 |
| CL_D | 90 | | 70 | | 50 | | 30 |
| CL_E | 120 | 105 | 90 | 75 | 60 | 45 | 30 |
| CL_F | 120 | | 90 | | 60 | | 30 |

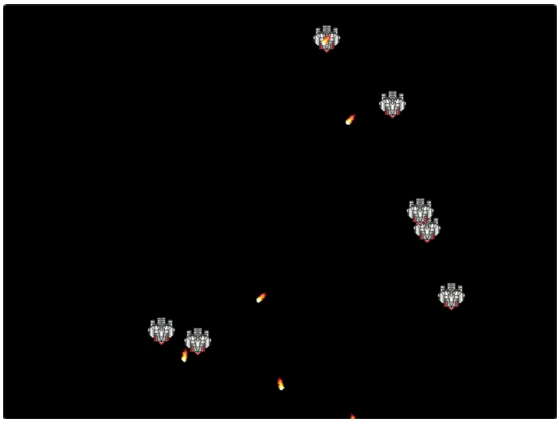

Fig. 5. Difference in enemy bullets with a probability of 1/30

Fig. 6. Difference in enemy bullets with a probability of 1/120

### D. Experimental results

Figure 7 shows the results for the DQN and CL. The graph shows the average of the rewards obtained in each episode of the five experiments, smoothed using a moving average of 301 intervals. Figure 8 shows a comparison of the learning results after 6,000 episodes. Figure 9 shows a graph of the average Q values obtained in each episode of the five experiments.

As shown in Figure 7, the results from the start of learning to 6,000 episodes were the lowest for the DQN as the curriculum was generated, and the CL increased the value of the reward obtained in the order of the curriculum difficulty. The result indicates that the curriculum rendered the task easier and less difficult.

Figure 8 shows that the overall results of the rewards obtained by the CL were generally higher than those obtained by the DQN. In general, the performance improved as the difficulty order of the curriculum increased; however, after 6,000 episodes, all the difficulty levels were the same. Therefore, in a few situations, it was overtaken, depending on the results of trial and error. When the last difficulty level in the curriculum and the highest difficulty level in the target task differed significantly, the increase in reward immediately after 6,000 episodes became lower; however, the participants gradually adapted to the target task. The reward obtained by CL_F immediately after 6,000 episodes was approximately 8.5. This value corresponds to the value of the reward obtained after approximately 3,000 episodes in the DQN. Although this difference decreased gradually, it did not disappear as the CLs adapted to the target task. Furthermore, a difference of approximately 4.0 was indicated between the reward in the DQN and the higher reward in the CL immediately after 6,000 episodes, and a difference of approximately 3.5 immediately before 10,000 episodes.

As shown in Figure 9, the increase in the Q-value was greater for the CL than for the DQN; in fact, for the CL, the increase was greater at lower levels of curriculum difficulty. This indicates that the Q-value was higher because the difficulty level was lower, and hence more possibilities to earn rewards. In this experiment, because the probability of an action to be selected depended on the Q-value of the Boltzmann selection, it was assumed that a significant increase in the Q-value based on the curriculum rendered it easier for the agents to reflect the knowledge that they obtained from their actions.

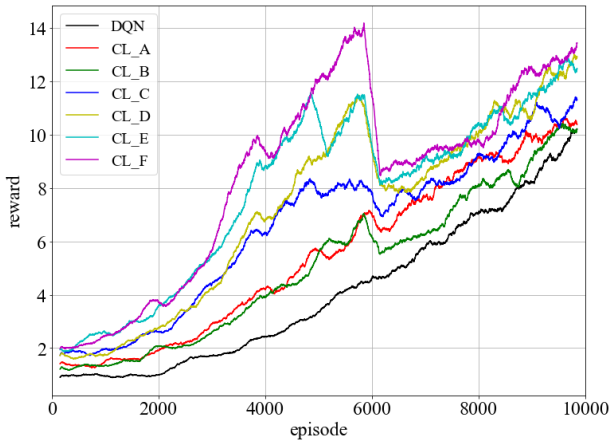


Fig. 7. The relationship between episodes and rewards

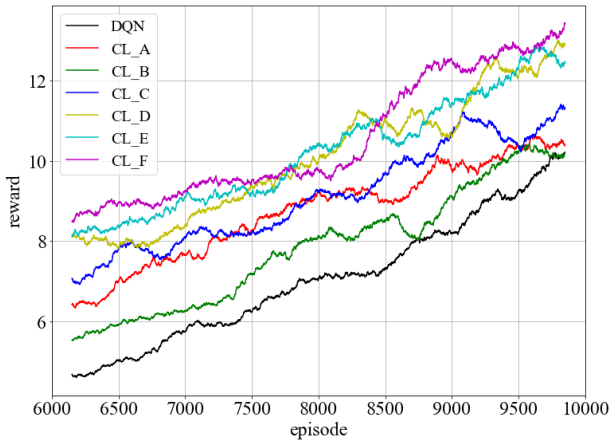


Fig. 8. The relationship between episodes and rewards (After 6,000 episodes)

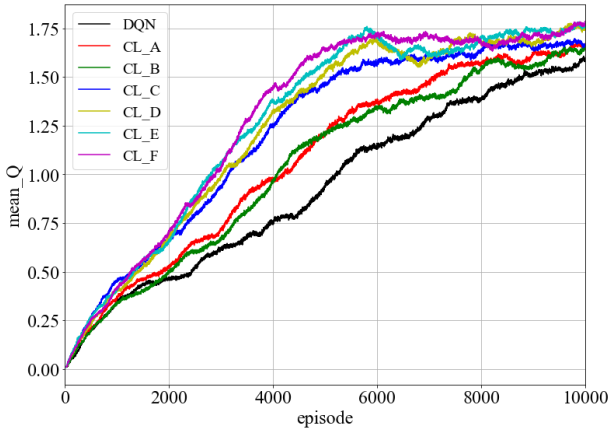


Fig. 9. The relationship between episodes and Q-value

## VI. Discussion and Summary

The experimental results showed a difference in the episodes with the same level of reward by applying curriculum learning, thereby confirming the reduction in the number of trials before the agent can acquire the action. The difference in the rewards obtained in the same episodes confirmed the improvement in learning performance for the same number of trials. We discovered that the easier the curriculum, the better was the comparative learning results in the target task; however, we could not confirm the extent to which the simplification of the curriculum was effective. Based on our experiments, we discovered that the simpler the curriculum, the lower was the increase in the reward for moving to the target task, suggesting that oversimplifying the curriculum to be generated would impose a greater impact.

In our experiment, a shooting game was used as the target. Therefore, it will be challenging to determine the tasks can be

enabled or performed effectively by the application of curriculum learning in deep reinforcement learning. In addition, we created a curriculum in which the difficulty level was varied quantitatively, beginning with a task that was several times easier than the target task. Because the learning environment by the curriculum was created by changing the numerical values, it needs not be created in advance and can be generated during learning. In this study, we generated the curriculum manually. If we can formulate the curriculum based on an index obtained in each learning process, then we can generate an optimal curriculum based on the process, including a trial-and-error learning process. The resulting curriculum generation algorithm can be applied to any environment in which the difficulty level can be quantitatively managed as a numerical value; therefore, it is expected to be further developed and applied.

## REFERENCES


[1] S. Narvekar, B. Peng, M. Leonetti, J. Sinapov, M. E. Taylor, and P. Stone, “Curriculum Learning for Reinforcement Learning Domains: A Framework and Survey, ”Journal of Machine Learning Research, 21(181):pp.1-50, 2020.

[2] J. West, F. Maire, C. Browne, and S. Denman, “Improved reinforcement learning with curriculum, ” Expert Systems with Applications, Volume 158, 113515, 2020.

[3] K. Arulkumaran, M. P. Deisenroth, M. Brundage and A. A. Bharath, “Deep Reinforcement Learning: A Brief Survey, ” in IEEE Signal Processing Magazine, vol.34, no.6, pp.26-38, Nov.2017.

[4] Naoki MIZUKAMI, Jun SUZUKI, Hirotaka KAMEKO, Yoshimasa TSURUOKA, “Deep Reinfocement Learning for Sparse Reward Environments, ” IPSJ Journal, Vol.60, No.3, pp. 956-966, 2019. (in Japanese)

[5] Junichiro YOSHIMOTO, Kenji DOYA, Shin ISHII, “Fundamental Theory and Application of Reinforcement Learning, ” Journal of the Society of Instrument and Control Engineers, Vol.44, No.5, pp.313-318, 2005. (in Japanese)

[6] V. Mnih, K. Kavukcuoglu, and D. Silver, “Playing Atari with deep reinforcement learning, ”arXiv preprint arXiv:1312.5602, 2013.

[7] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, “Curriculum learning, ”Proceedings of the 26th annual international conference on machine learning, 2009.

[8] S. Narvekar, J. Sinapov, M. Leonetti, and P. Stone, “Source Task Creation for Curriculum Learning, ” Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS), pp.566-574, 2016.

[9] S. Luo, H. Kasaei and L. Schomaker, “Accelerating Reinforcement Learning for Reaching Using Continuous Curriculum Learning, ” 2020 International Joint Conference on Neural Networks (IJCNN), pp.1-8, Glasgow, United Kingdom, 2020.

[10] A. Bassich and D. Kudenko, “Continuous Curriculum Learning for Reinforcement Learning, ” in Proceedings of the 2nd Scaling-Up Reinforcement Learning (SURL) Workshop, 2019.

[11] S. Narvekar, J. Sinapov, and P. Stone, “Autonomous Task Sequencing for Customized Curriculum Design in Reinforcement Learning, ” In Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI), Melbourne, Australia, August, 2017.

[12] F. L. Da Silva, and A. H. R. Costa, “Automatic Object- Oriented Curriculum Generation for Reinforcement Learning, ” in Proceedings of the 1nd Scaling-Up Reinforcement Learning (SURL) Workshop, 2017.

[13] M. Svetlik, M. Leonetti, J. Sinapov, R. Shah, N. Walker, and P. Stone, “Automatic Curriculum Graph Generation for Reinforcement Learning Agents, ” Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17), pp. 2590-2596, 2017.

[14] Yuto KITA, Satoshi YAMAGUCHI, “A Deep Q Network with Boltzmann Selection, ” IEEJ transactions on electronics, information and systems, Vol.137, No.12, pp.1676-1683, 2017. (in Japanese)