Multi-agent Learning Mechanism Based on Diversity of Rules : from the Viewpoint of LCS *

Hiroyasu Inoue∗† Yutaka Leon Suematsu∗† Keiki Takadama†‡
Katsunori Shimohara∗† Osamu Katai∗

∗ Kyoto University Graduate School of Informatics, Yoshida Honmachi, Sakyo-ku, Kyoto 606-8501 Japan
† ATR Network Informatics Laboratories 2-2 Hikaridai, Keihanna Science City, Kyoto 619-0288 Japan
‡ Tokyo Institute of Technology 4259 Nagatsuta-cho, Midori-ku, Kanagawa 226-8502 Japan
E-mail: hir_inoue@atr.jp

Abstract

Multi-agent learning requires effective means because of its vast learning space. One means is that agents share their knowledge. However, thoughtless knowlege sharing can disturb its learning process. In this paper, we investigate the relatioship between the effect of knowledge sharing and information on other agents' positions.

We proposed a novel rule-sharing mechnism employing LCS. Through experiments using simplified soccer, we demonstrated that if agents can find other agents, sharing rules is effective; if not, then sharing rules is ineffective.

Keywords: multi-agent, learning, rule sharing, learning classifier system

1 Introduction

Arai [1] and others have shown that multi-agent learning includes many difficulties. Those problems include the simultaneous learning problem, the reward-assignment problem, and so on. Another problem is that even if a single agent has a small learning space, the whole learning space is enormous because of the number of possible combinations. Thus, a more effective method is required for multi-agent learning. One means of this is that agents share their knowledge.

Generally, it seems that two conditions should be fulfilled to achieve knowledge sharing. The first is that representation of knowledge is sharable, while the second condition is that imported knowledge from other agents can be interpreted. One approach that fulfills these conditions is that agents have the same mechanical functions so as to interprete knowledge from each other, and to employ a rule-based system for importing and exporting their knowledge. Though such agents can share knowledge, thoughtless knowlege sharing can disturb the learning process, since knowledge sharing homogenizes agents' knowledge. Inoue [2] showed that the effect of knowledge diversity has a relationship with position information on other agents. This paper focuses on this problem.

In this paper, we investigate the relationship between the effect of knowledge sharing and information on other agents' positions.

This paper is organized as follows. Section 2 starts with a description of the rule-sharing mechanism. Section 3 explains the experimental settings, and Section 4 provides the results. Section 5 we discuss the results, and conclude in Section 6.

2 Rule-Sharing Mechanism

There are various possible mechanisms for sharing rules. For the purpose of this paper, we should consider a general mechanism. An inevitable aspect of the general mechanism is sharing priorities of rules. In accordance with this, in this section we propose a novel mechanism based on the Learning Classifier System (LCS: proposed by Holland [4]).

LCS is a general learning system in which a set of condition-action rules called classifiers compete to control a target system, gaining reward from the environment. We employ ZCS (Zeroth level Classifier System) proposed by Wilson [5], which has no message list, and propose the Rule-Sharing Learning Classifier System (RS-LCS), extending ZCS to share rules between agents.

RS-LCS has the following differences to ZCS: (1) RS-LCS does not use GA because the effectiveness of GA is ambiguous and still under discussion. Also, RS-LCS does not use #. # is used to condition parts of rules and matches any condition. # does not work

*This research was conducted as part of 'Research on Human Communication' with funding from the National Institute of Information and Communications Technology.

effectively without GA. (2) The covering process in RS-LCS creates rules so that all kinds of actions can be chosen. (3) RS-LCS does not use the roullette selection process. Twenty percent of all actions are random, while the rest are chosen from the rule that matches conditions and has the highest priority. (4) RS-LCS does not use a bucket brigade process; instead it uses a profit-sharing process proposed by Grefenstette [6] because the profit-sharing process fits multi-agent learning. We employ Miyazaki's profit-sharing process [7], and its decreasing ratio is 0.9.

LCS, including ZCS, is basically a mechanism for single agents. Hence, RS-LCS should include an extension for rule sharing. This paper has the assumption that agents simultaneously acquire a reward from the environment. Based on this, agents share their rules every ten reward events. In the rule-sharing process, all of the agents' rules are merged, an if any of those rules have the same condition parts and action parts, their strengths are averaged. The merged rules are then returned to each agent.

3 Experimental Setting

3.1 Simplified Soccer

We employ simplified soccer as an example. The following is a discription.

Figure 1 shows the landscape of simplified soccer. (The meaning of dotted lines is described in the following section.) The field has 8×30 squares, and there are two teams, a left team and a right team, and a ball. Each team has three agents. The vertical line on the right side is the left team's goal, and vice versa.

At every step, all agents move simultaneously, to the left, right, above, or below squares. Agents can also stay in the same squares, and the agents and the ball can freely move into the same square. If at least one agent moves into the same square as the ball, the ball is moved (kicked). If more than two agents move into the same square, one agent is selected randomly, and the ball is moved. If the left team's agent moves the ball, the ball moves 12 squares to the right, and also moves vertically and horizontally by less than two squares. This is also applied to the right team, but the direction of movement is left. If the ball goes over the left goal, the left team gets a goal, and vice versa. If the ball goes over the horizontal lines, it goes back to within the range of the horizontal lines.

At the beginning of experiments, or after a goal, positions of the agents and the ball are reset. The ball is set randomly from a choice four squares in the center of the field. The left agents are set to the left half of the field randomly, and the right agents are also set in the same way.

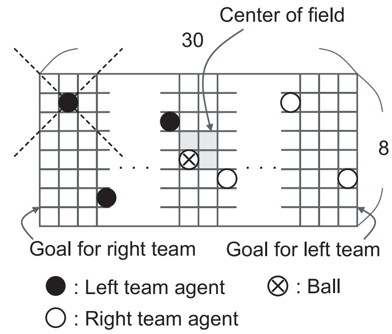

Figure 1: Simplified Soccer

Each right agent has a greedy mechanism, which always makes an agent chase the ball. Through all experiments, the right agents' mechanism is the greedy mechanism. On the other hand, left agents have RS-LCS implemented. By fixing the right agents, performance comparisons with the left agents are possible.

The greedy mechanism does not need any additional description. To use RS-LCS, however, sensor description is necessary, which is provided in the following section.

3.2 Left Agent Sensor

In Section 2, the substance of the sensor was not mentioned. In fact, the sensor includes six bits in total.

The first four bits are assigned to information on positions of other agents in the left team [1] . The dotted lines in Fig. 1 are two imaginary lines for an agent. Based on those two lines, the field is divided into four areas. The squares on the border are included in the right area, or left area, and the square in which the agent sits is included in the right area. If at least one other of the left agents is found in an area, the bit for that area is set to 1, if not, it is set to 0.

Next, three bits are assigned to directions of the ball. If the ball is in the right area, 000 is set. (The distance is not considered.) In the same way, left, above, below are set to 001, 010, 011, respectively. The remaining areas, upper-right, upper-left, lower-right, and lower-left, are set to 100, 101, 110 ,111.

The last one bit is assigned to the distance of the ball. The horizontal difference of position between the agent and the ball is calculated and its absolute value is taken. If the sum of the horizontal value and the

[1] Position information of right team's agents can be included. But because of large learning space experiments are difficult to conduct. Hence this information is omitted here.

vertical one is larger than 10, the bit is set to 0. If not, it is set to 1.

3.3 Reward Assignment

As we mentioned in Section 1, the reward-assignment problem for multi-agent is still being discussed. Hence we describe a specific method of reward assignment for the simplified soccer problem, because the validity of reward assignment cannot be discussed.

To assign a reward, an episode of moving the ball is recorded. This episode records a left agent moving the ball every step. If no left agent moves the ball, that is also recorded. When the left team scores a goal, left agents obtain a reward. The last agent that moved the ball gets a 1.0 reward. After that the episode is traced back, and at every one step the reward is multiplied by 0.9. If a record of an agent is found, the reward is given at that step. However if the agent has already received a reward, the reward is not given. After tracing back the episode, the agents which did not receive a reward get a 0.0 reward, and the episode is cleared.

If the right team scores a goal, all left agents get 0.0 reward. The episode is then cleared.

3.4 Experimental Combinations

The purpose of this paper is to verify whether or not sharing rules is always effective. Therefore we conduct the following experiments. (1) Left agents cannot find other teammates, and cannot share their rules. (2) Left agents cannot find other teammates, and can share their rules. (3) Left agents can find other teammates, and cannot share their rules. (4) Left agents can find other teammates, and can share their rules.

4 Results

Figure 2 shows the two experimental results. The vertical axis represents the number of goals every 2,000 steps, and the horizontal axis representss the steps. Both results are the average of ten trials. In both experiments, agents cannot find other teammates. In one experiment, agents can not share their rules, whereas in the other, they can. At the start, the number of no share agents' goals is smaller than that of share agents' goals. At the end, however, no share agents overtake share agents.

Figure 3 shows the two other experimental results. The axes of Figure 3 are identical to those in Fig. 2. In both experiments, agents could find other teammates. At the start, the number of share agents' goals is larger

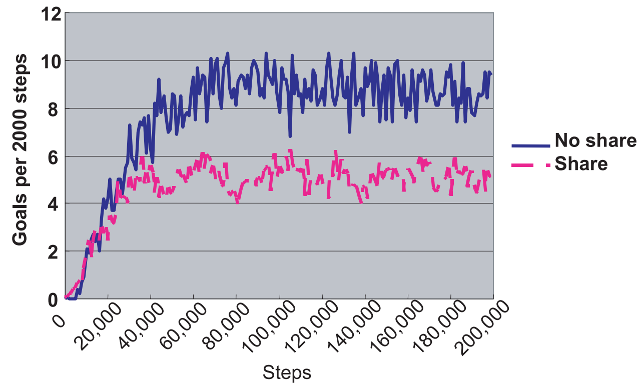

Figure 2: Result (Agents cannot find other teammates)

than that of no share agents' goals, but the number of finite goals is similar between two agents.

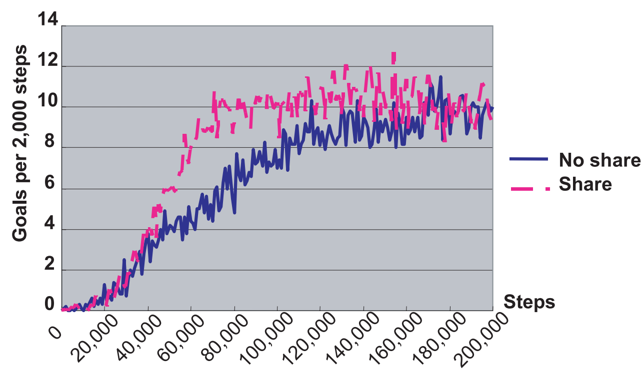

Figure 3: Result (Agents can find other teammates)

5 Discussion

5.1 Construction on Results

If left agents take the same greedy strategies as right agents, the performance should be identical between them. The performance, however, is not identical: the left agents are superior to the right agents. (Because of space restrictions, the right agents' results can not be included.) This is because each left agent performs different roles, which enables them to pass the ball to each other so as to outwit right agents. The important point is the method of performing those roles. The difference between Fig. 2 and 3 can be analyzed in light of this point.

In these experiments, role assignments are similar to positioning. That is, left agents try to take their appropriate posiions. If left agents cannot find other teammates (Fig. 2), each agent has to take a specific position; otherwise they may conflict in their positions. This specialty can only be realized by the difference of rules. Hence, if agents share their rules, that specialization collapses, and the performance deteriorates.

If left agents can find other teammates (Fig. 3), none of the agents have to take a specific position because they can arrange their positions dynamically. Thus even if agents share their rules, the performance does not deteriorate.

Sharing rules makes agents homogeneous, and vice versa. As we mentioned in Section 1, there are concerns with the results. This is because heterogeneity between agents is necessary when agents cannot find other agents, as Inoue [2] demonstrated.

From the above discussions, we could show the relationship between the effect of sharing rules and information of other agents' positions. Especially, if agents can find other agents, rule sharing is effective. If not, rule sharing is not effective.

5.2 Open Problems

This research has only just started. Therefore, many open problems still remain, as the described bellow.

5.2.1 Constraints of Problems

In this paper, agents do not communicate with each other. It should be investigated as to whether the finding of this paper is still valid, if agents do communicate with each other.

Rule-sharing has a cost, which is not discussed in this paper. Not only the ability to find other agents, but also the cost should be taken into account.

5.2.2 Rule-Sharing Mechanism

A rule-sharing mechanism has many parameters. For example, the number of rules for sharing, the timing of rule sharing, and so on. The sensitivities of these parameters need to be investigated.

This paper's rule-sharing mechanism has all agents merge their rules. However, there are various kinds of merger. For example, specific two agents always merge their rules.

Another important open problem is the contradiction of rules between agents. There are some solutions to this problem proposed by Inoue [8], but these are not insufficient.

Although this paper's rule-sharing mechanism has agents merge their rules, agents do not necessarily merge their knowledge. One such mechanism has been proposed by Inoue [9]. In that mechanism, agents retain other agents' rules as different rule sets and exploit them according to the situation the agent faces.

6 Conclusion

Multi-agent learning requires effective means because of its vast learning space. One means is that agents share their knowledge. We proposed a novel rule-sharing mechnism employing LCS. Through experiments using simplified soccer, we demonstrated the relationship between the effect of sharing rules and information of other agents' positions. In particular, if agents can find other agents, sharing rules is effective. If not, it is not effective.

References

[1] Arai S (2001), Multiagent reinforcement learning frameworks: Steps toward practical use (in Japanese). *Journal of Artificial Intelligence* 16(4):476-481

[2] Inoue H, Takadama K, Shimohara K, et al. (2003), Exploring Conditions of Heterogeneous Knowledge for Effective Multi-agent Systems (in Japanese). *The Second Joint Agent Workshops and Symposium*, 208-215

[3] Takadama K, Hajiri K, Nomura T, et al. (1998), Organizational Learning Model for Adaptive Collective Behaviors in Multiple Robots. *Advanced Robotics (The International Journal of the Robotics Society of Japan)*, 12(3):243-269

[4] Holland H (1985), Properties of the Bucket Brigade Algorithm. *The 1st International Conference on Genetic Algorithms* 1-7

[5] Wilson S (1994), ZCS: A Zeroth Level Classifier System. *Evolutionaly Computation* 2(1):1-18

[6] Grefenstette J.J. (1988), Credit assignment in rule discovery systems based on genetic algorithms. *Machine Learning*, 3:225-245

[7] Miyazaki K, Arai S, Kobayashi S (1999), A theory of profit sharing in multi-agent reinforcement learning (in Japanese). *Journal of Artificial Intelliegence*, 14(6):1156-1164

[8] Inoue H, Takadama K, Shimohara K, et al. (2002), Learning Classifier System Enabling Adaptation to Mixed Known Environments (in Japanese). *The Journal of Three Dimensional Images*, 208-213

[9] Inoue H, Takadama K, Shimohara K, et al. (2000), Agent Architecture Based on Self-Reflection Learning Classifier System. *The 5th International Symposium on Artificial Life and Robotics*, 454-457