A Proposal of Reward Division Type Q-Learning with Reinforcement Flags and Its Application to a Maze Problem

Yukinobu HOSHINO and Katsuari KAMEI
Dept. of of Human and Computer Intelligence, Ritsumeikan University
1-1-1 Noji-higashi, Kusatsu, Shiga 525-8577, JAPAN
email:{hoshino, kamei}@spice.ci.ritsumei.ac.jp

***Abstract*–A recently, Reinforcement Learning has been recognized as a technique of the knowledge acquisition in the intelligent systems. The reinforcement learning creates behavior information by rewards, and is a kind of machine learning for adaptation to unknown environments through trial and error. It is expected to apply to the behavior acquisition in agents such as robots. A profit Sharing has been combined with Q-learning, than Q-PSP is a technique that it aims for performance improvement. Problems of these techniques were investigated by this research. In this paper, we propose a reward division type Q-learning with reinforced flags. Our proposed technique consists of two reinforcement types. One is the reward distributive law with an intensified flag, and the other is the reward division type Q-learning. We apply this technique to a maze problem and show improvements in learning speed.**

I Introduction

Reinforcement learning[6] is applicable to some problems of behavior acquisition in agents such as robots. Profit Sharing is a general technique for Reinforcement learning. This technique has a high learning speed, but it tends to learn local behavior, rather than optimal behavior. On the other hand, Q-learning [10], another technique for reinforcement learning, achieves optimal behavior but has a slow learning speed. In order to improve Reinforcement Learning, some systems have been developed to combine Profit Sharing and Q-learning [2][4][5]. Of these methods, the most common technique is Q-PSP learning [3]. In this paper, we propose a unique technique to combine Profit Sharing and Q-learning. This paper describes experiments comparing Profit Sharing, Q-learning, and Q-PSP, and shows the effectiveness of our proposed technique.

II Reinforcement Learning

Reinforcement learning is a kind of unsupervised learning system[1][11][12]. This system reinforces weights for action selection to create a policy by reward/penalty. Agent select actions, the best actions, using reinforced rules. If there are no fitting rules, new rules are created on the rule base. Reinforcement learning is able to learn under specific information that is uncertain or delayed. Learning systems have three units, the sensory unit, the learning unit, and the selection unit. as shown in Fig.1. Learning proceeds by renewal of weights through action selection.

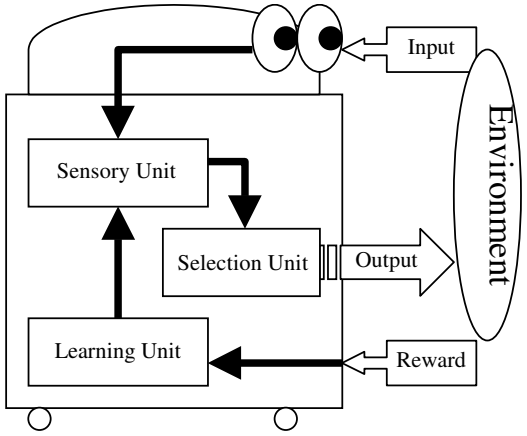

Figure 1: Frame work of Reinforcement learning

III Profit Sharing with Reinforced Flags

We will apply Reinforcement Learning to a Maze problem. For this application, we propose a new technique for Profit Sharing. Here, we describe Profit Sharing with Reinforced Flags. On General Profit Sharing, rule's weight $w(s_t, a_t)$ is renewed by reward r, as given Eq.(1).

$$w(s_t, a_t) = w(s_t, a_t) + d^{T-t}r \tag{1}$$

In this case, we cannot assign a high Discount rate d because Rationality Theorem [2] determines a limit for setting this value.

In this equation, s_t is an observed state and a_t is one of a set of actions available for selection at the current period t. T is maximum long time of an episode and t is current period for the learning system to apply the rule $w(s_t, a_t)$. We have defined the Reinforced Flag, which shows whether a rule has been reinforced previously.

Here, we show an example of an application of Reinforcement Flags, as shown in Fig.2. x, y and z are the observable states. a and b are the actions for the

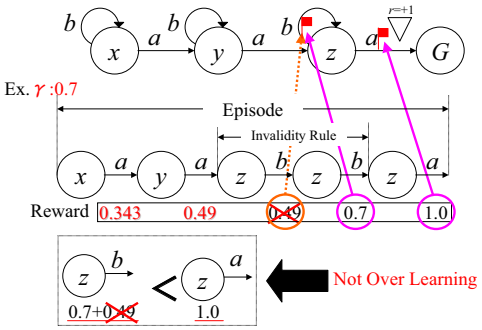

Figure 2: Frame work of the reinforced flags

learning system to select. G is the goal point of the episode. The learning system receives a reward when selecting the action a under the observed state z and reaching the goal point G. On a normal learning process, the learning system distributes to all rules the discounted profits. If the discount rate is a high value, all rules share the high profit and increase $w(s_t, a_t)$, as in Eq.(1). So the learning system will over-learn and agents will confuse the actions selected by a high value of $w(s_t, a_t)$. However, Reinforced Flags make the system able to allow a high discount rate because the learning process checks Reinforced Flags (on/off) to determine whether the system has previously ever used the rule in the episode. If the system finds a flag, it skips the learning process for the previously used rule, which is then called an Invalid Rule. Using these flags, the discount rate keeps a high level and the profit sharing never learns invalid routes.

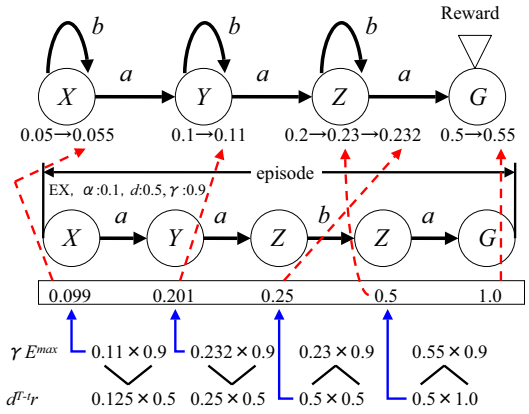

Figure 3: Frame work of profit division

IV Reward Division Type Reinforcement Learning

In this section, we describe one more idea using Reinforced Flags. We show one of examples in Fig.3 in order to explain the computing process of Reward Division type Reinforcement Learning. Here, we apply Reinforced Flags to TD learning, called Temporal Difference method [1]. Our proposal idea has a unique unit which selects one of two candidate profits. One candidate profit is $d^{T-t}r$, which uses a discount reward similar to Profit Sharing. The other profit is $\gamma w(s_{t+1}, a_{t+1})$, which uses Temporal Difference method. Selection conditions are very simple, as given in Eq.(3). We have revised fixed the learning formula as in Eq.(2). Now, we explain the framework of the profit division, given in Fig.3. By using this method, an agent is able to learn by Temporal Difference method during the beginning steps. During the final steps, an agent will shift to Profit Sharing method. In this way, Temporal Difference method will work among routes with non-Markov properties. Also, Profit Sharing will work among route with Markov properties. This is because, in most maze problems, the states near the goal point have an optimal direction to reach the goal point.

$$w(s_t, a_t) = (1-\alpha)w(s_t, a_t) + \alpha \triangle w(s_t, a_t) \qquad (2)$$

$$\triangle w(s_t, a_t) = \begin{cases} d^{T-t}r \\ \quad (d^{T-t}r \geq \gamma w(s_{t+1}, a_{t+1})) \\ \\ \gamma w(s_{t+1}, a_{t+1}) \\ \quad (d^{T-t}r < \gamma w(s_{t+1}, a_{t+1})) \end{cases} \qquad (3)$$

V Experiment 1

We solved a maze problem to compare ordinary Reinforcement learning method to the proposed techniques. This maze has two routes to a goal. However application to a small maze could not determine the different points between these learning methods. We designed a unique maze, in which an agent has to break a wall by picking up a tool, as shown in Fig.4. In this maze, 'T' is the tool. The task is to reach 'G', which is the Goal point, from 'S', which is the Start point, via 'T'. The gray zone is a wall that the agent can break using the tool. So, the agent has to learn the route to pick up the tool and break the wall. We designed the agent, which

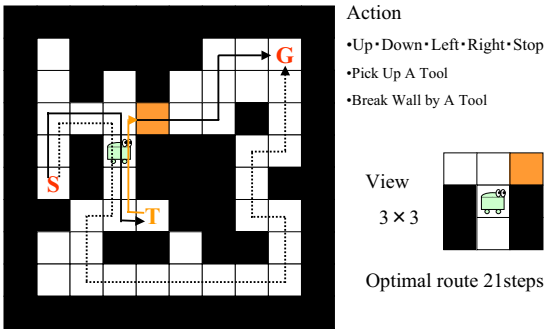

Figure 4: Maze of Experiment 1.

has to spend one step to pick up the tool and break the wall. Each agent has a 3x3 grid view, one step in each direction, and is able to select from seven actions. Movement actions are up, left, right, down and stop (stay). Two additional actions are picking up the tool, and breaking a wall. We compared the ordinary Reinforcement Learning with the proposed methods. The learning parameters are shown in table.1. The

Table 1: Learning parameters

Method	α	γ	d
Q-Learning	0.1	0.9	–
Profit Sharing Plan	–	–	0.5
Q-PSP Learning	0.1	0.9	0.5
Reward Division type Q-Learning	0.1	0.9	0.5
Profit Sharing with Reinforced Flags	0.1	–	0.9
Reward Division type Q-Learning with Reinforced Flags	0.1	0.9	0.9

number of steps, which is 'G', converged on almost the same value and become stable on Fig.5. We conducted 500 trials for all methods. Q-learning and the proposed methods converged on about 21 to 30 steps. In particular, Q-learning and Reinforced Division type Q-learning with Reinforced Flags obtained the optimal steps. However, Profit Sharing did not converge in increments. The reason may be that agents are confused by the "Alias Problem". Also, this maze has many steps to finish each episode. In such cases, the reinforced function of Profit Sharing has tiny values from the beginning point 'S' and the agent is confused by the short reinforcement.

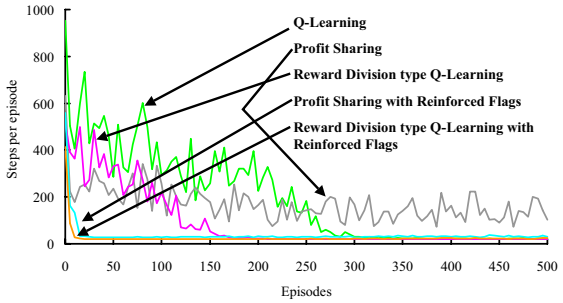

Figure 5: Learning Curve between Episodes and Steps per episode in Experiment.1.

We show a detailed learning process in Fig.6 explaining Profit Sharing with Reinforced Flags and Reward Division type Q-learning with Reinforced Flags. These learning speeds are very fast. Especially, Reward Division type Q-learning with Reinforced Flags take converges on, at most, 25 steps, which is a optimal step on this maze. We are considering that this phenomena was given by a efficient work of a part of the Temporal Difference method.

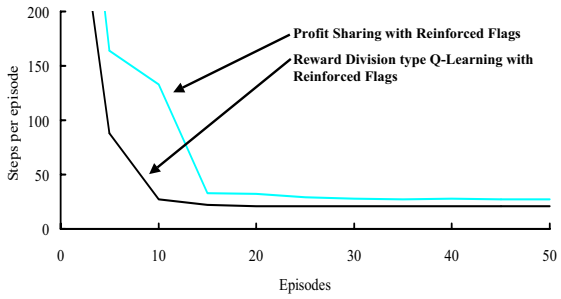

Figure 6: Detail Learning Curve during 0–50 episodes on Experiment 1.

VI Alias Problem

Ordinarily, Q-learning uses a reward and a value of $Q_{max}(s_{t+1}, a_{t+1})$ to learn. A reward is a wake signal and $Q_{max}(s_{t+1}, a_{t+1})$ is the main element in learning. In the Alias Problem, as shown in Fig.7, $Q_{max}(s_{t+1}, a_{t+1})$ would be same value at state 'A' and 'B', when an agent learns. Around 'A', the Q-value would be a high value from $Q_{max}(s_{t+1}, a_{t+1})$. States between 'A' and 'B' would take similar Q-values by those Q_{max}. However, their real values would be less than at 'A' or 'B' because the discount rate γ would discount Q-values. As a result, Q-values around 'A' and 'B' would be similar values and the agent would be confused when actions are selected by Q-values because they have the same evaluation even when agents select the best action by Q-value.

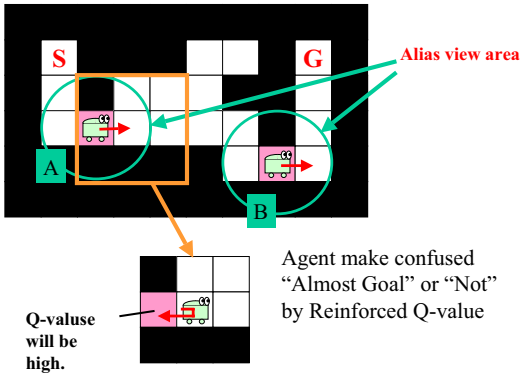

Figure 7: Alias Problem

VII Experiment 2

This maze has two tools and two walls, as well as a more complicated route than Exp.1 in which an agent has to pick up tools and break walls. Those procedures have a definite order: tool 'T1', wall 'W1', tool 'T2'

and wall 'W2'. The agent has to learn the route to 'G' from 'S' via those procedures. The optimal step number is very long at 45 steps, as shown in Fig.8. Fig.9 shows the results of each learning method for Exp.2. This phenomenon is almost the same as that shown in Exp.1. Profit Sharing did not converge. Q-learning was much slower than trial Q-learning in Exp.1. Fig.10 shows a comparison of Q-learning and Reward Division type Q-learning.

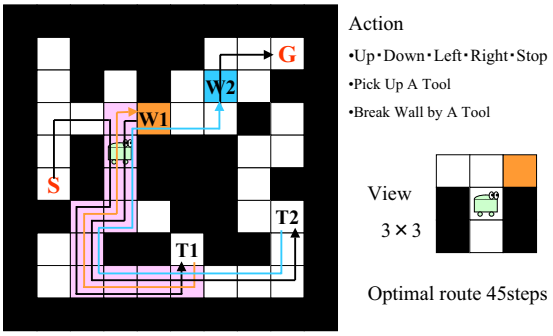

Figure 8: Maze of Exp.2

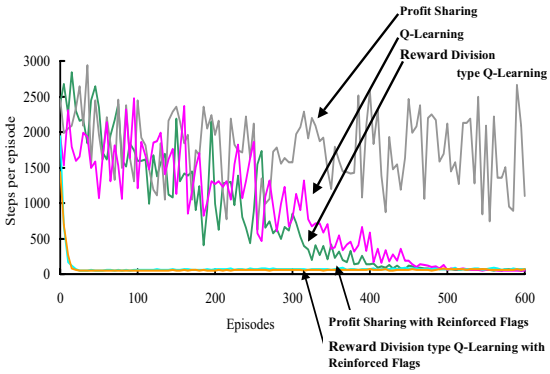

Figure 9: Learning Curve between Episode and Steps per Episode in Experiment 2.

In Exp.1, Q-learning obtained the optimal step number. We consider that Q-Learning learned the optimal route for the maze in Exp.1. However under Exp.2, Q-learning continuously confused route by our idea. Therefore, Alias zone provides a reason for Q-learning to confuse learning in an optimal solution. See Fig.10.

Fig.11 is a detail graph of Profit Sharing with Reinforced Flags and Reward Division type Q-learning with Reinforced Flags. However, Reward Division type Q-learning with Reinforced Flags and Reinforcement never make stable increments compared with the results in Exp.1. Instead, the steps keep on wavering, see Fig.11. As a result, we are considering that the main process is the effect of the reinforced flags to increase learning speed. Also, Temporal Difference method is able to work for learning an optimal route but its learning speed is not fast. The Reward Division method and the refinanced flags are very particular in learning the optimal route.

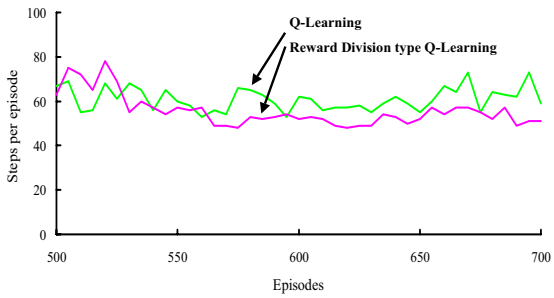

Figure 10: Stable steps about Q-Learning and Reward Division type Reinforcement Learning between 500–700 episodes on Experiment 2.

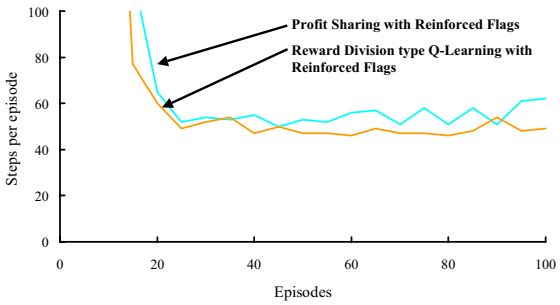

Figure 11: Detail Learning Curve between 0–100 episodes on Experiment 2.

VIII Conclusion

We have applied several learning methods to two types of maze problem, and compared ordinal learning methods with proposed methods, including Profit Sharing with Reinforced Flags. In Exp.1, reinforcement learning method acquired the ability to take the best route. The proposed methods had the best performance for the learning speed. In Exp.2, the results described a very similar phenomenon. However, those methods never proceed in stable increments. In this paper, we proposed two methods, the Reinforced Flags method and the Reward Division method. We verified that the Reinforced Flags method increases the learning speed. However, if an episode is made from a long series of steps, it is not able to learn the optimal route quickly. Put another way, Reward Division method shifts the Profit Sharing Learning from Temporal Difference method in each episode. We show results comparing the proposed system with other reinforcement learning systems. Finally, Reward Division type Q-learning with Reinforced Flags achieves a better result than Q-learning for a specific maze with the Alias Problem and episodes with long steps. For future work, we would like to try many types of maze, each with its own specific problems. The selection unit on Reward division is very simple, so we consider that this unit needs more dynamic condition in order to solve problems and to design conditions for all cases.

References

[1] Tom Mitchell: Machine Learning, *McGraw Hill*, 1997.

[2] Kazuteru MIYAZAKI, Shigenobu KOBAYASHI: On the Rationality of Profit Sharing in Multi-agent Reinforcement Learning, International Conference on Computational Intelligence and Multimedia Applications 2001, pp.123–127 (2001).

[3] Tadashi Horiuchi, Akinori Fujino, Osamu Katai, Tetsuo Sawaragi: Q-PSP Learning: An Exploitation-Oriented Q-Learning Algorithm and Its Applications, International Conference on Evolutionary Computation pp.76-81 (1996)

[4] Kazuteru MIYAZAKI, Shigenobu KOBAYASHI: Rationality of Reward Sharing in Multi-agent Reinforcement Learning, Second Pacific Rim International Workshop on Multi-Agents (PRIMA'99), pp.111-125 (1999).

[5] Kazuteru MIYAZAKI, Masayuki YAMAMURA, Shigenobu KOBAYASHI: MarcoPolo : A Reinforcement Learning System considering tradeoff exploration and exploitation under Marcovian Environments, Proceedings of IIZUKA'96, pp.561-564 (1996).

[6] Tatsuo UNEMI, Reinforcement leaning, Journal of the Artificial Intelligence Society, Vol.9, No.6, pp.830–836 (1994)

[7] Tadashi HORIUCHI, Akinori FUJINO, Osamu KATAI, Testuo SAWARAGI, Fuzzy Interpolation-Based Q-Learning with Continuous Inputs and Outputs, Journal of The Society of Instrument and Control Engineers, Vol.35, No.2, pp.271–279 (1999)

[8] Yukinobu HOSHINO, Katsuari KAMEI, A Proposal of Reinforcement Learning with Fuzzy Environment Evaluation Rules and Its Application to Chess, Journal of Japan Society for Fuzzy Theory and Systems, Vol. 13, No.6, pp.626–632 (2001)

[9] Yukinobu HOSHINO, Katsuari KAMEI, An Application of FEERL (Fuzzy Environment Evaluation Reinforcement Learning) to LightsOut Game and Avoidance of Detour Actions in Search, Transactions of the Institute of Systems, Control and Information Engineers, Vol. 14, No. 8, pp.395–401 (2001)

[10] Christopher J. C. H. Watkins , Peter Dayan: Technical Note:Q-Learning, Machine Learning, Vol.8 No.3, pp.279–292, 1992

[11] R.S.Sutton and A.G.Barto: Reinforcement Learning, *The MIT Press*

[12] Gerhard Weiss: Multiagent systems: a modern approach to distributed artificial intelligence, *The MIT Press*, Cambridge, MA, USA