A Reuse of Acquired Rules in Reinforcement Learning for Tactics of *Hasami-Shogi*

Yukinobu HOSHINO[1], Hiroshi HIDAKA[2] and Katsuari KAMEI[3]
[2]Graduate School of Science and Engineering, Ritsumeikan University
[1,3]Dept. of of Human and Computer Intelligence, Ritsumeikan University
1-1-1 Noji-higashi, Kusatsu, Shiga 525-8577, JAPAN
email:{[1]hoshino, [2]hidaka, [3]kamei}@spice.ci.ritsumei.ac.jp

***Abstract*–The pattern of scene in Two-player Game is huge combination. So a game environment requires determination of the best selection by evaluating each state during a plating game. We have been conducting Reinforcement Learning system for Hasami-Shogi player of Computer. Hasami-Shogi is popular game in Japan, because regulation of this game is very easy. Japanese children have ever played Hasami-Shogi more than one time. Go is a famous Japanese game. But this game is very hard to learn those regulations. It is possible to play good moves on Hasami-Shogi that most good player of any games, like Chess-game. We would like to propose a machine learning system, which is able to learn established fuzzy rule as common knowledge. Fuzzy Environment Evaluation Reinforcement Learning allows us to make efficient use of experience with a previous state. We applied FEERL to Hasami-Shogi, and obtained good results for a 4x4 grid. We extended the rules of the 4x4 grid to a corresponding with 7x7 grid, and applied the extended rules to Hasami-Shogi of 7x7 grid. From the results, we would show validly advantage point of FEERL as a reinforcement learning method with the ability to use acquired rules effectively as experience.**

I Introduction

Sometime, people think a lot of time to determine action. Especially, Shogi and Chess game has very huge patterns of scenes, because there has many kind pieces on the board. Player evaluates many scenes to select the best move. Always, the game player search a game tree by those evaluations but player has to prune a game tree because the game tree is very large. So, it is not sure that selected move is good on a current scene. We propose to apply a Reinforcement learning to make an evaluation function of game scenes. The Reinforcement learning is a kind of machine leaning, which is able to learn without a supervisor[3][4][7][9]. However, it is very hard work that a Reinforcement learning works on all game scenes. For such case, we propose Fuzzy Environment Evaluation Reinforcement Learning (FEERL)[5][6]. This learning system is possible to get profitable rules for a reward and keep output a reasonable evaluation. Additionally, we try a Profit Sharing with Reinforced Flag for FEERL. In this learning system, we set many flags to indicate whether a rule has been used previously. At the same time, our system continuously checks the state (on/off) of the flags and determines whether the system has previously ever used the rule in a current episode. If the system finds a flag, it skips the learning process for the previously used rule. By these flags, we are able to set high-level discount rates and the profit sharing never learns invalid routes. We have applied 4x4 grid Hasami-Shogi of Two-Player game and we have taken good result. Next step, we have tried expand rule for 7x7 grid Hasami-Shogi and have applied it.

II FUZZY ENVIRONMENT EVALUATION REINFORCEMENT LEARNING

Fuzzy environment evaluation reinforcement learning (FEERL) is a kind of reinforcement learning, which is using fuzzy reasoning. This has a built-in rule base corresponding to experience, which a rule have evaluated a state. Fuzzy reasoning can evaluate the unknown state. The environment evaluate rule is based on describing the target state and its evaluation.

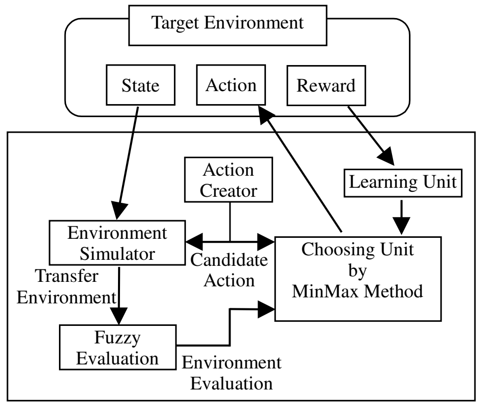

Figure 1: Diagram of FEERL

$$\boldsymbol{q} = (q_1, q_2, \cdots, q_j, \cdots, q_N) \tag{1}$$

$$R_i: \quad \text{if } \boldsymbol{q} \text{ is } \boldsymbol{p}_i \text{ then } E \text{ is } w_i$$

$$\boldsymbol{p}_i = (p_{1i}, p_{2i}, \cdots, p_{ji}, \cdots, p_{Ni}) \tag{2}$$

q is a state vector which is observed. i=1,2,...,M; M is the number of rules, was used previously. p_i is a state vector, which is already known. Specifically, the method is a machine learning algorithm, which is for determining a behavior at unknown states. For FEERL, the resemblance between the input state q and the environment evaluation rule fidelity of each for Eq.(1). The element p_{ji} of rule is given Eq.(2). As shown in Fig.2. is the total sum of fidelity against the total sum of dimension the attributes, L is the sensitivity zone of p_{ji}, and pp is a sensitivity parameter.

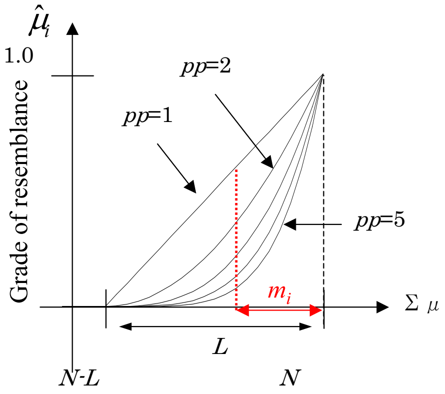

Figure 2: Fuzzy Resemblance Reasoning

Based on such fuzzy resemblance reasoning, the environment evaluation value $E(q)$ in the input state $\boldsymbol{q}$ is given Eq.(6). w is the environment evaluation value in the environment evaluation rule. Also,

$$\mu_{ji} = \begin{cases} 1 - \dfrac{|p_{ji} - q_j|}{l} & |p_{ji} - q_j| < l \\ 0 & |p_{ji} - q_j| \geq l \end{cases} \tag{3}$$

$$m_i = N - \sum_{j=1}^{N} \mu_{ji}, pp \geq 1 \tag{4}$$

$$\widehat{\mu}_i = \begin{cases} \left(1 - \dfrac{m_i}{L}\right)^{pp} & m_i < L \\ 0 & m_i \geq L \end{cases} \tag{5}$$

$$E(\boldsymbol{q}) = \begin{cases} \dfrac{\sum_{i=0}^{M} w_i \widehat{\mu}_i}{\sum_{i=0}^{M} \widehat{\mu}_i} & \sum_{i=0}^{M} \widehat{\mu}_i > 0 \\ 0 & \sum_{i=0}^{M} \widehat{\mu}_i = 0 \end{cases} \tag{6}$$

The reasoning definition of the resemblance based on an ordinary is given as

$$\widehat{\mu}_i = \textstyle\bigwedge_j \mu_{ji} \tag{7}$$

In this resemblance, the reasoning would judge a state has no resemblance with the target state, when one of μ_{ji} puts a small value. Such judgment does not some agrees with a human judgment on an observed state. Our reasoning system is able to compute Fuzzy Evaluation about next state. Selection Unit would select a next state q and an action from a Environment Evaluation $E(q)$, even if a next state have never observed previously. FEERL keep going select action by $E(q)$. FEERL is able to get reward/Penalty from Environment when FEERL have gotten a goal, which is a finish point of an episode. If FEERL took Reward/Penalty, a Learning Unit renew weight w_i with the used rule R_i, given Eq.(8).

$$w_i = (1 - \alpha)w_i + \alpha(r + \gamma\widehat{\mu}_i E^{max}) \tag{8}$$

Learning Unit would select several rules, which described $E(q)$. If all resemblance of rules are less than 0.5, FEERL add new rule R_{M+1}, that p_{M+1} will be an observed state q and w_{M+1} will be Environment Evaluation $E(q)$. E^{max} is a result have searched a game tree by a MinMax method[9].

III HASAMI-SHOGI AS TWO-PLAYER GAME

Hasami-Shogi is a kind of Classic Japanese Board Games. Player use a shogi board, size 9 x 9 grid, has 18 pieces. At Start Game, White pieces put 9 pieces on the top line of Board, and black pieces put 9 pieces on the bottom line of Board. On Fig.3, white triangles are white pieces, and black triangles are black pieces.

	a	b	c	d	e	f	g	h	i
1	▼	▼	▼	▼	▼	▼	▼	▼	▼
2									
3									
4									
5									
6									
7									
8									
9	△	△	△	△	△	△	△	△	△

Figure 3: 9x9 grid real Hasami-Shogi

All the pieces can move as the horizontally and vertically way, like a rook of Chess game. But if pieces exist in that way, a moving pieces cannot jump over other pieces. When player sandwiches opponent's pieces by two of own pieces, player can take the sandwiched pieces. It is similar to Othello game. In Hasami-Shogi, player can sandwich horizontally and vertically, but cannot sandwich diagonally. This is different point from sandwich style of Othello game. See Fig.4.

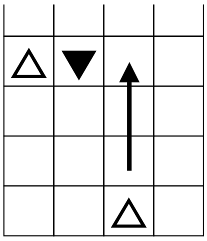

Figure 4: Sandwich a target piece

When player move in to where such that pieces are sandwiched by the opponent pieces, our sandwiched pieces are not taken from the board. When the opponent has only one piece, player wins. The piece at the edge of the board can be taken by two adjacent pieces, when white piece is on 'a1' and black promoted pieces are on 'a2' and 'd1', black can take white 'a1' piece by moving 'd1' to 'b1'. See Fig.5.

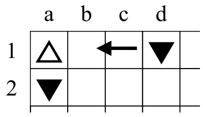

Figure 5: Take a edge piece

IV PROFIT SHARING WITH REINFORCED FLAGS

We tried to apply FEERL 4x4 grid Hasami-Shogi. To apply it, we propose new technique for Profit Sharing. We show Profit Sharing with Reinforced Flag. On General Profit Sharing, $w(s,a)$ is renewed by reward r, given Eq.(9). In this case, we cannot put high value Discount rate d because Rationality Theorem[2] would define a limit to set this value.

$$w(s,a) = w(s,a) + d^{T-t}r \tag{9}$$

In this equation, s is an observed state and a is one of actions as a possible. T is maximum long step of an episode and t is current step for the learning system to apply the rule $w(s,a)$. We have defined Reinforced Flag, which is a signal about reinforced already.

Here, we show an example of the application of Reinforcement Flags, as shown in Fig.6. x, y and z are the observable states. a and b are the actions for the 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 allots to all rules the discounted profits. If the discount rate is a high value, all rules share the high profit and increase $w(s,a)$, as in Eq.(9). So the learning system will over-learn and agents would confuse the actions selected by a high value of $w(s,a)$. 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.

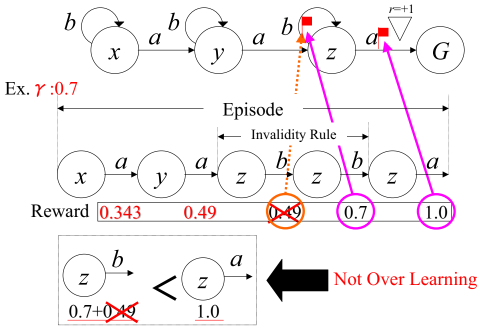

Figure 6: Frame work of the reinforced flags

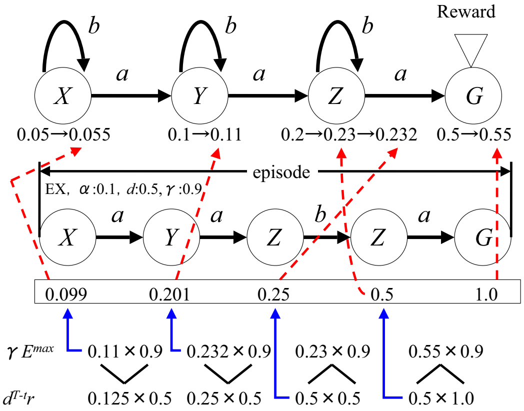

Figure 7: Frame work of profit division

V Reinforce Division type Reinforcement Learning

In this section, we address one more idea using Reinforced Flags. We show one of example on Fig.7, 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 γE_{max}, which uses Temporal Difference method. Selection conditions are very simple, as given in Eq.(11). We have fixed the learning formula as in Eq.(10). Now, we explain the framework of the profit division, given in Fig.7. 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 would work among routes with non-Markov properties. Also, Profit Sharing would 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.

$$w_i = (1-\alpha)w_i + \alpha\triangle w_i \tag{10}$$

$$\triangle w_i = \begin{cases} d^{T-t}r & d^{T-t}r \geq \gamma E_{max} \\ \gamma E_{max} & d^{T-t}r < \gamma E_{max} \end{cases} \tag{11}$$

VI FEERL to 4x4 Hasami-Shogi

We has applied FEERL to 4x4 grid Hasami-Shogi as Exp.1 and describe configuretion parameters for a FEERL, given Table.1.

Table 1: Result of all experiment

Learning rate	α=0.01
Discount rate	γ=0.9
Reward	r=1
Width of Membership	l=0.1
Sensitivity zone	L=18
Sensitivity parameter	pp=3.0
Depth game tree	6

The result of 4x4 grid games is Fig.8. This graph is describing between a number of games and the percent of wins, loses and draws. Trial is 2500 games and FEERL begin to obtain rules for win over 1400. We can consider that finished rules are complete rules for wins.

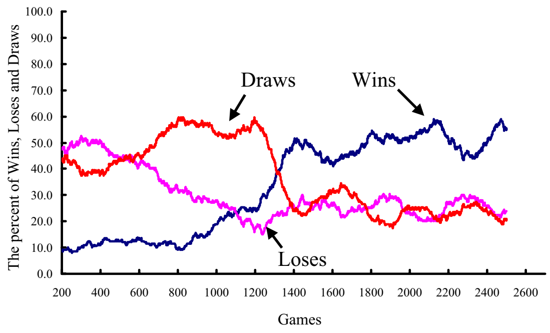

Figure 8: result4x4

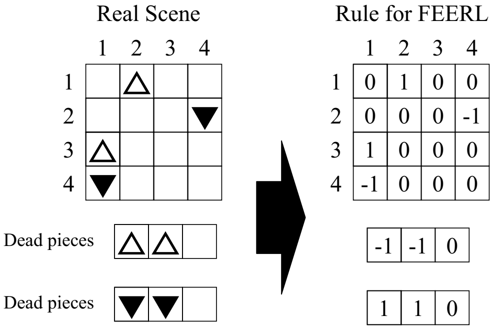

Figure 9: rule

VII Expanded rule for 7x7 grid hasami-shogi

Now, we consider the way to extended rules as a knowledge. Rule of FEERL explain an evaluation about observed states, which fit a current environment, given Fig.9. We assume that a kind of candidate actions cannot expand for similarly observed state. Because, FEERL has Action Creator Unit to create the candidate actions, given Fig.1. FEERL is able to estimate the unknown states by Fuzzy Resemblance Reasoning. Therefor, the extension rule would work on extended environment. In this paper, we have attempted one expand method to extend 4x4 grid rules to 7x7 grid rules. We have fixed Affine Transformation, which is one popular method of the data processing method, given Eq.(12). x' and y' are the position information about a source data. $SIZE'$ is a number of grid size about a source data, x, y and $SIZE$ are same kind information of a destination data. If we apply the transform 4x4 grid rules to 7x7 grid rules, $SIZE'$ would be 4 and $SIZE$ would be 7. Expand part of Affine Transformation compute to expand 2D data from the liner approximate, given Fig.10. This formula is Eq.(13).

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{SIZE-1}{SIZE'-1} & 0 & 0 \\ 0 & \frac{SIZE-1}{SIZE'-1} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} \tag{12}$$

$$\begin{aligned} B'(x,y) = & B(u,v)(1-\alpha)(1-\beta) \\ & +B(u+1,v)\alpha(1-\beta) \\ & +B(u,v+1)(1-\alpha)\beta \\ & +B(u+1,v+1)\alpha\beta \end{aligned} \tag{13}$$

The expanding would lose an information of a source data1 about Affine Transformation. We have more fixed the special rules after expanded it from 4x4 grid rules. Exactly, we have exchanged state p_i of 7x7 grid rules to most resemblance state k_s on a record of all moves. This process would support the expanding, which would lose the information. We define this technique the resemblance convert, given Fig.11.

Exp.2, we have tried normal FEERL to 7x7 Hasami-Shogi, dose not use the expand rule based on 4x4 grid

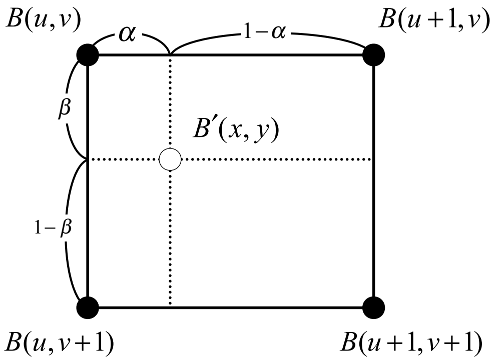

Figure 10: Affine Transformation

Figure 11: convert

rule. we have applied FEERL to 7x7 grid Hasami-Shogi. This result is just one win, so we could verify that 7x7 grid Hasami-Shogi is very hard(difficult) for normal FEERL. On Exp.3, we use Affine transition only to transport 4x4 grid rules to 7x7 grid rules. As a result, we have observed that transfer method need a more additional technique on Affine transition method. It is able to consider that ordinal Affine transition did not work. That reason is expanding of Affine transition probably lose some information by a liner approximate. We show all results on Table.2. Exp.4 took the best result. We used the resemblance convert in addition to Affine transition. Resemblance could increase by using this technique to expand rules. By that result, expanded rules, which is using the resemblance convert, could work on Hasami-Shogi.

VIII Conclusion

Reusing of Acquired rule is a kind human ability. If Reinforcement learning could take an ability to reuse rules for the similar state, the opportunity to apply a Reinforcement learning would increase to Two-player Game. Ordinary reinforcement learning could be hard to apply Two-player Game because most these games has many scene/state and actions. Because those combination is very huge. To cover those states, Reinforcement learning have to obtain an approximate function. About this problem, we have proposed any idea based on Fuzzy theorem. Also, there are many investigator to work with a Reinforcement leaning and a Approximate for the huge environment. We could propose some technique to Reinforcement learning for Two-player Game. At first, we could show a work of FEERL with those technique on 4x4 grid Hasami-Shogi. Secondly, we proposed the expand method based on Affine transition. Finally, we have compared ordinal FEERL and some expand techniques. As a result, we have verified a growing percent of wins. For the future, we would like to keep this work until coming a saturation point. Also, we would like to try 9x9 grid Hasami-Shogi and some other Two-player Games.

Table 2: Result of all experiment

Exp.	Wins	Loses	Draws
Exp.1	828	761	911
Exp.2	1	279	20
Exp.3	3	250	47
Exp.4	34	119	27

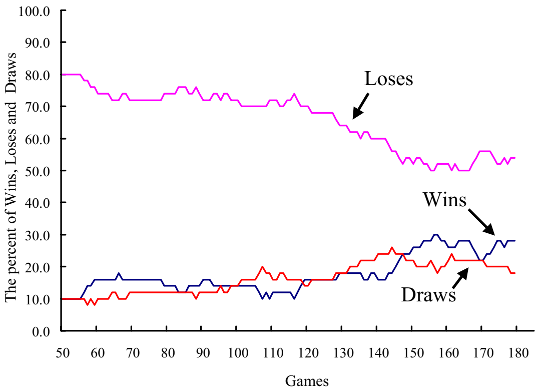

Figure 12: Result of Exp.4

References

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

[2] Miyazaki, K., and Kobayashi, S.: 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] Tatsuo UNEMI, Reinforcement leaning, Journal of the Artificial Intelligence Society, Vol.9, No.6, pp.830–836 (1994)

[4] 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)

[5] 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)

[6] 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)

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

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

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