Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

# Implementation of Fuzzy Q-Learning Based on Modular Fuzzy Model and Parallel Structured Learning

Toshihiko WATANABE
Faculty of Engineering
Osaka Electro-Communication University
Neyagawa, Osaka, JAPAN
t-wata@isc.osakac.ac.jp

*Abstract*— **In order to realize intelligent agent such as autonomous mobile robots, Reinforcement Learning is one of the necessary techniques in control system. Fuzzy Q-learning is one of the promising approaches for implementation of reinforcement learning function owing to its high ability of model representation. However, in applying fuzzy Q-learning to actual application, the number of iterations for learning also becomes huge as well as almost all Q-learning application. Furthermore convergence performance is often deteriorated owing to its complicated model structure. In this study, implementation method of fuzzy Q-learning is discussed in order to improve the learning performance of fuzzy Q-learning. The modular fuzzy model construction method based on fuzzy Q-learning is proposed in this paper. Multi-grain configuration of modular fuzzy model is compared with parallel structured learning scheme. Through numerical experiments of mountain car task and Acrobot task, I found that the proposed construction of modular fuzzy model improved the performance of fuzzy Q-learning.**

*Keywords*—**reinforcement learning, Q-learning, fuzzy Q-learning, modular fuzzy model, Acrobot, mountain car task**

## I. INTRODUCTION

Reinforcement learning[1-4] is one of the necessary techniques to realize intelligent agent such as autonomous mobile robots. The reinforcement learning is attractive technique because it is simply based on rewards concept, and need not teaching data. Many bench mark simulations show the effectiveness of the learning technique. Fuzzy Q-learning is one of the promising approaches for implementation of reinforcement learning function owing to high ability of model representation. Fuzzy system has been studied as modeling methodology of human related systems. Fuzzy model for reinforcement learning can also be expected to deal with nonlinearity and complexity as in the human related systems. However there exist some problems in order to apply reinforcement learning method to actual applications. One of the main problems of fuzzy Q-learning is huge iteration needed in order to formulate the model iteratively by Q-learning algorithm. Though the problem is also revealed in almost all standard Q-learning applications, the problem tends to be serious in fuzzy Q-learning in the cause of the high redundancy of model representation. As the other problem, convergence performance of learning is often deteriorated through learning iteration. It is obviously desirable to perform reinforcement learning of good precision for application.

In order to deal with these problems, implementation methods of fuzzy Q-learning are investigated in this study. The modular fuzzy model is applied based on the fuzzy Q-learning in this paper. In addition to the conventional construction of the modular fuzzy model suitable for high dimensional task, I propose multi-grain configuration of modular fuzzy model. As another implementation of fuzzy Q-learning, parallel learning structure for modular model based on fuzzy Q-learning algorithm is also discussed. In this study, reinforcement learning problem having continuous state and discrete action is considered to evaluate the essential performance of reinforcement learning.

The paper is organized as follows. In section 2, I introduce and propose modular fuzzy model for reinforcement learning. In section 3, Q-learning algorithm based on the proposed model architecture is described. In section 4, parallel structured learning of modular model is described. The results of numerical experiments using “mountain car task” and “Acrobot task” are shown in section 5. Finally, conclusion is drawn in section 6.

## II. MODULAR FUZZY MODEL FOR REINFORCEMENT LEARNING

### A. Structure of Modular Fuzzy Model

As a fuzzy model having high applicability, Single Input Rule Modules(SIRMs)[7][8] was proposed. The idea is to unify reasoning outputs from fuzzy rule modules comprised with single input and single output formed fuzzy if-then rules. The number of rules can be drastically reduced as well as bringing us high maintainability in actual application. However, its disadvantage of low precision is inevitable in order to apply the method to high dimensional or complicated problems. The modular fuzzy model[9] is extension of the SIRMs method by relaxing the restriction of the input space, i.e. single, to arbitrary subspace of the rule for constructing the model of huge multi-dimensional space. In many application cases of huge multi-dimensional space, human expertise or know-how can almost be expressed approximately using some inputs among whole input variables, e.g. fuzzy control and expert system. The concept of the model is intuitively representing such observations. Description of the model is as follows:



978-1-4244-2794-9/09/$25.00 ©2009 IEEE

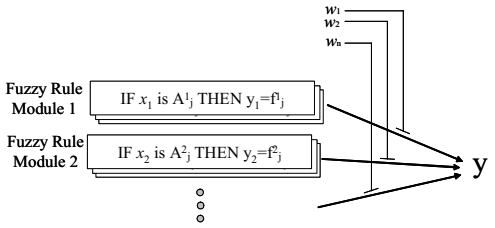


Fig. 1. Modular Fuzzy Model

$$\begin{aligned} &Rules-1:\{if\ P_1(x)\ is\ A_j^1\ then\ y_1=f_j^1(P_1(x))\}_{j=1}^{m_1}\\ &\qquad\vdots\\ &Rules-i:\{if\ P_i(x)\ is\ A_j^i\ then\ y_i=f_j^i(P_i(x))\}_{j=1}^{m_i}\\ &\qquad\vdots\\ &Rules-n:\{if\ P_n(x)\ is\ A_j^n\ then\ y_n=f_j^n(P_n(x))\}_{j=1}^{m_n} \end{aligned} \tag{1}$$

where "*Rules-i*" stands for the *i*-th fuzzy rule module, $P_i(x)$ denotes predetermined projection of the input vector $x$ in *i*-th module, $y_i$ is the output variable, and $n$ is the number of rule modules. The number of constituent rules in the *i*-th fuzzy rule module is $m_i$. $f$ is the function of consequent part of the rule like *TSK*-fuzzy model[13]. $A_j^i$ denotes the fuzzy sets defined in the projected subspace. Fig.1 shows the conceptual diagram of modular fuzzy model.

The membership degree of the antecedent part of *j*-th rule in "*Rules-i*" module is calculated as:

$$h_j^i = A_j^i(P_i(x^0)) \tag{2}$$

where $h$ denotes the membership degree and $x^0$ is an input vector. The output of fuzzy reasoning of each module is decided as the following equation.

$$y_i^0 = \frac{\sum_{s=1}^{m_i} h_s^i \cdot f_s^i(P_i(x^0))}{\sum_{s=1}^{m_i} h_s^i} \tag{3}$$

The final output of the Modular Fuzzy Model is formulated as:

$$y^0 = \sum_{i=1}^{n} w_i \cdot y_i^0 \tag{4}$$

where $w_i$ denotes the parameter of importance of the *i*-th rule module. The parameter can be predetermined or modified by learning algorithms[9]. By not using whole Cartesian products of input states, the "curse of dimensionality" in high dimensional problem can be avoided in this structure.

### B. Definition of Projection

In order to construct the modular fuzzy model, the necessary projection of each "Rule Module" should be defined. I explain the basic idea of the definition using examples. Assume that $x$ is 4 dimensional input vector as:

$$x = [x_1, x_2, x_3, x_4]^T \tag{5}$$

The conventional SIRMs architecture adopts the projection as follows:

$$\{P_1(x), P_2(x), P_3(x), P_4(x)\} = \{\{x_1\}, \{x_2\}, \{x_3\}, \{x_4\}\} \tag{6}$$

Though the projection in the modular fuzzy model can be arbitrarily defined, the primary idea of the modular fuzzy model is to define the projection from combination of each variable as:

$$\begin{aligned} &\{P_1(x), P_2(x), P_3(x), P_4(x), P_5(x), P_6(x)\}\\ &= \{\{x_1, x_2\}, \{x_1, x_3\}, \{x_1, x_4\}, \{x_2, x_3\}, \{x_2, x_4\}, \{x_3, x_4\}\} \end{aligned} \tag{7}$$

where pair inputs are used for rule modules. In the same way, triplet inputs can also be formulated as:

$$\begin{aligned} &\{P_1(x), P_2(x), P_3(x), P_4(x)\}\\ &= \{\{x_1, x_2, x_3\}, \{x_1, x_2, x_4\}, \{x_1, x_3, x_4\}, \{x_2, x_3, x_4\}\} \end{aligned} \tag{8}$$

In this manner, the construction of projection leads to reduction of the number of total rules. The number of total rules in modular fuzzy model can be less than the conventional Cartesian product typed fuzzy model when the number of modules is limited. Furthermore, projections for the rule modules can be selected from the whole combination. We can also utilize the same projection for the plural modules. Though the problem of deciding the number of appropriate rule modules still remains, it is decided by increasing step by step through modeling from one dimension, i.e. equivalent to SIRMs, in this study.

### C. Multi-grain Configuration of Modular Fuzzy Model

As described above, the modular fuzzy model can be constructed by combining rule modules of different projections in the input space. In another point of view, the rule modules in the modular fuzzy model can be structured using coarse graining of the whole input. The coarse grain corresponds to abstract knowledge and the finer grain corresponds to concrete knowledge, of agent task. The coarse grained module is expected to accelerate exploration behavior through reinforcement learning. On the contrary, the finer grained module is expected to improve precision of the learning. My idea is to unify these multi-grain modules as modular fuzzy model to improve the learning performance. Concretely, the coarse grained module is constructed by coarse fuzzy partition, the finer grained module is constructed by finer fuzzy partition, of the whole input space.

## III. Q-Learning Based on Modular Fuzzy Model

In Q-learning reinforcement learning algorithm, an optimal policy is found by maximizing rewards received over time. Q-function is defined as values for each pair of state and action:

$$Q(x_t, a_t) = Q(x_t, a_t) + \alpha\left(r_t + \gamma \max_{\eta} Q(x_t, \eta) - Q(x_t, a_t)\right) \tag{9}$$

where $Q$ is Q-value, $x_t$ is the state vector(input) at *t*-th step, $a_t$ is action at *t*-th step, $r_t$ denotes the reward, $\gamma$ is a discount factor, and $\alpha$ is a learning rate.

In this paper, Q-function is approximated by the modular fuzzy model. Eq. (1) is modified to FCS(Fuzzy Classifier System) form[10][11] as:

$$Rules-i:\{if\ P_i(x_t)\ is\ A_j^i\ then\ a_t\ is\ c_j^k\ with\ q_{ij}^k\}_{j=1}^{m_i} \tag{10}$$

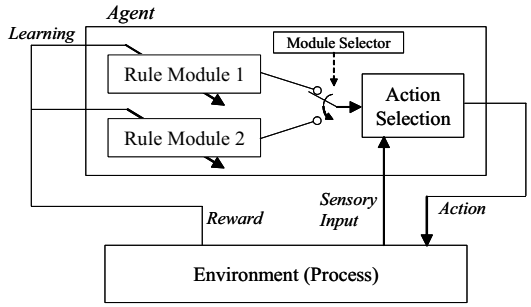


Fig. 2. Parallel Structured Learning Model

where $c^k_j$ is a $k$-th concrete action and $q^k_{ij}$ is the Q-value of the $j$-th rule in *Rules-i* module. The output Q-value is calculated as the same way in (3):

$$Q_i(x_t, c^k) = \frac{\sum_{s=1}^{m_i} h_s^i \cdot q_{is}^k}{\sum_{s=1}^{m_i} h_s^i} \tag{11}$$

$$Q(x_t, c^k) = \sum_{i=1}^{n} w_i \cdot Q_i(x_t, c^k) \tag{12}$$

From (9), we have error to be modified as:

$$\Delta Q(x_t, a_t) = \alpha\left(r_t + \gamma \max_{\eta} Q(x_t, \eta) - Q(x_t, a_t)\right) \tag{13}$$

The changes of $q$ values in the rules can be formulated by calculating the gradient of (11) and (12) as:

$$\Delta q_{ij}^k = \Delta Q(x_t, a_t) \cdot \frac{\partial Q(x_t, a_t)}{\partial q_{ij}^k} = \Delta Q(x_t, a_t) \cdot w_i \cdot \frac{h_j^i}{\sum_{s=1}^{m_i} h_s^i} \tag{14}$$

where $a_t$ is $c^k$ in this equation.

## IV. Parallel Structured Learning

In another point of view for reinforcement learning, modular structured model can be implemented as parallel module of Q-table model unlike unification of the modules. The basic idea for tile coding application is presented in [5,6]. In this paper, the idea is extended to fuzzy model. As described in previous section, multi-grain configuration of modular fuzzy model is expected to improve learning performance by aggregating different level of knowledge acquired by reinforcement learning. Parallel structure of modular model attempts to utilize the different module more clearly by switching. Fig.2 shows the conceptual diagram of the parallel structured learning. Each module is learned parallel by Q-learning method. The module selector switches the module to use for action selection of the agent corresponding to convergence status of learning. In typical reinforcement learning process, the coarse grained model is appropriate to be used for action in early trials. To the contrary, finer grained model is appropriate to be used in latter trials. It should be noted that the learning is performed parallel through the whole trials utilizing independence property of action and learning in Q-learning method.

In the concept of parallel learning, the main issue is to decide appropriate criterion for switching in the module selector. The criterion should be based on judgment of learning convergence of the module. In this study, the criterion is based on the number of iterations as well as quantification of action policy as the following formulation:

$$z = \sum_{s \in S} \max_{\eta} Q(s, \eta) \tag{15}$$

where $S$ denotes the universal set of states. In other words, $S$ is the set of center of fuzzy rules in the module. The center of fuzzy rule is defined as the state of maximum membership value of the antecedent part. As it is not practical to numerically integrate the maximum value of Q in continuous state problem, representative states, i.e. the centers of fuzzy rule, are considered in this paper.

There exists an another issue in switching the module. When the switching is performed, the knowledge acquired in the coarse grained module becomes useless as it is. It is desirable to utilize the information by exploration behavior before the switching as much as possible. In this study, the acquired model in coarse grained module is reflected to finer grained module simultaneously at the switching timing as follows:

$$q_{is}^k = (1-\beta) q_{is}^k + \beta Q_j(x_t^s, c^k), \quad s = 1,2,...,m_i \tag{16}$$

where $i$ denotes the module number of the finer grained module, $j$ denotes the module number of the coarse grained module, and $x_t^s$ is the center of $s$-th rule in the coarse grained module. $\beta$ is a parameter such that $0<\beta<1$.

## V. Numerical Experiments

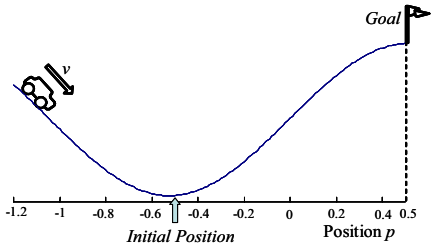
In the mountain car task[12] and the Acrobat task[14], I evaluate the performance of fuzzy Q-Learning based on the modular fuzzy model and parallel structured learning. These two tasks are dynamical system problems of continuous state and discrete action. In the experiments, several learning models described in this paper are applied to evaluate in terms of the learning speed and convergence property. The action of the agent is decided based on $\varepsilon$-greedy strategy. In the strategy, a random action is chosen with probability $\varepsilon$ and a greedy action exploiting the known Q-values is chosen with probability 1-$\varepsilon$. The value of $\varepsilon$ is fixed as 0.2 through the whole simulations.

### A. Simulation Settings

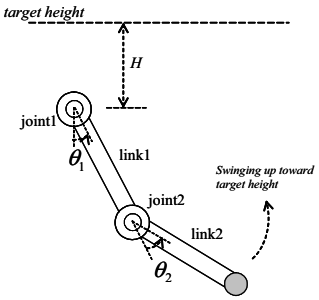
#### 1) Mountain car task

Fig.3(a) shows the mountain car task. The objective of the task is to reach the car(agent) to the goal at the top of the right hand mountain. A unique characteristic of the task is that the force of gravity is stronger than the motor power. Accordingly, the agent should acquire rules of climbing after speeding up by round behavior utilizing the slopes, in order to reach the goal.

The mountain car task has two continuous state variables, the position of the car, $p_t$, and the velocity of the car, $v_t$. At the start of each trial, the initial state is set as $p_0$=-0.5 and $v_0$=0 in this study. The action, $a_t$, takes on values in {+1,0,-1} corresponding to forward thrust, no thrust, and reverse thrust.




(a) Mountain Car Task




(b) Acrobot Task

Fig. 3. Tasks for Numerical Experiments

The state transition is according to the following discrete system of simplified physics:

$$v_{t+1} = v_t + 0.001a_t - G\cos(3p_t)$$
$$p_{t+1} = p_t + v_{t+1}$$

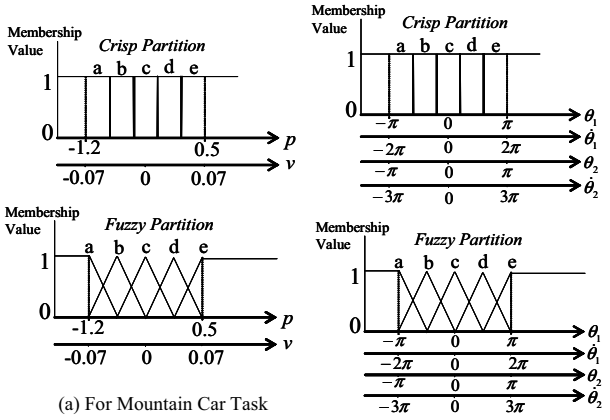
where $t$ is the number of iteration and $G$=0.0025 is the force of gravity. The state variables are bounded by -1.2<$p$<0.5 and -0.07<$v$<0.07. If $p$ is limited at this boundary, $v$ is also reset to zero. The agent obtains the reward and trial terminates when the position $p_t$ is over 0.5.

*2) Acrobot task*

Acrobot is a mimic system of the "horizontal bar" of human gymnastics. The objective of the task is to swing the tip of the robot up to the predefined height as shown in Fig.3(b). A unique characteristic of the task is that the robot adjusts only the second joint torque being affected by its dynamics to achieve the goal.

The dynamical equations of the Acrobot are as follows:

$$d_{11}\ddot{\theta}_1 + d_{12}\ddot{\theta}_2 + h_1 + \phi_1 = 0$$
$$d_{21}\ddot{\theta}_1 + d_{22}\ddot{\theta}_2 + h_2 + \phi_2 = T$$
$$d_{11} = m_1 r_1^2 + m_2 l_1^2 + m_2 r_2^2 + 2m_2 l_1 r_2 \cos\theta_2 + I_1 + I_2$$
$$d_{12} = m_2 r_2^2 + m_2 l_1 r_2 \cos\theta_2 + I_2$$
$$d_{21} = m_2 r_2^2 + m_2 l_1 r_2 \cos\theta_2 + I_2$$
$$d_{22} = m_2 r_2^2 + I_2$$
$$h_1 = -m_2 l_1 r_2 \left(2\dot{\theta}_1 + \dot{\theta}_2\right)\dot{\theta}_2 \sin\theta_2$$
$$h_2 = m_2 l_1 r_2 \dot{\theta}_1^2 \sin\theta_2$$
$$\phi_1 = \left(m_1 r_1 + m_2 l_1\right) g \sin\theta_1 + m_2 r_2 g \sin\left(\theta_1 + \theta_2\right)$$
$$\phi_2 = m_2 r_2 g \sin\left(\theta_1 + \theta_2\right)$$




(a) For Mountain Car Task

(b) For Acrobot Task

Fig. 4. Fuzzy and Crisp Partition for Simulation

where $\theta_1$, $\theta_2$ are the angles of joint1 and joint2, $\dot{\theta}_1$, $\dot{\theta}_2$ are the angular velocities of joint1 and joint2, and $\ddot{\theta}_1$, $\ddot{\theta}_2$ are the angular accelerations of joint1 and join2, respectively. $T$ is the torque applied to the joint2. $m_i$ is mass, $l_i$ is length, $r_i$ is length to center of mass, and $I_i$ is moment of inertia of link$i$, respectively. Actual setting of constant parameters is as $m_i$ =1, $l_i$ =1, $r_i$ =1, $I_i$=1 ($i$=1,2), and g=9.8 in this study. The angular velocities are bounded by $\dot{\theta}_1 \in [-4\pi, 4\pi]$ and $\dot{\theta}_2 \in [-9\pi, 9\pi]$. The target height $H$ is set as 0.5.

The equations of motion are numerically solved by means of Runge-Kutta method in 0.05sec interval. The sampling period is 0.2 sec. The agent chooses the torque $T$ from {-1,0,1} as action at every sampling time(step).

### *B. Evaluation of Crisp Model and Fuzzy Model*

First of all, Q-learning agent based on "crisp" tile coding and fuzzy Q-learning agent based on fuzzy model coding are applied to the two tasks to evaluate essential learning performance. The state variables of the agent are ($p$,$v$) in mountain car task and ( $\theta_1$, $\theta_2$, $\dot{\theta}_1$, $\dot{\theta}_2$ ) in acrobot task, respectively. The crisp partition and fuzzy partition for the models are shown in Fig.4. In this study, it is assumed that the partition is decided at even intervals. The number of trials in each learning simulation is set as 2000. The results are averaged over 30 times simulations.

*1) Mountain car task*

Fig.5 shows the converged episode length by Q-learning agent based on "crisp" tile coding. Fig.6 shows the converged episode length and Fig.7 shows the averaged episode length in early 50 trials, by fuzzy Q-learning agent based on fuzzy model coding. From these results, the better number of each partition is selected. Fig.8 shows the learning process of the agents. From the results, it is clear that there exists serious problem of huge episode length in early trials by fuzzy Q-learning. Furthermore, the average value of converged episode lengths by crisp Q-learning agent is much smaller than one by fuzzy Q-learning agent. However, better results are attained

occasionally by fuzzy Q-learning agent as shown in Fig.9. This figure shows the distribution of converged episode lengths in each simulation.

*2) Acrobot task*

Fig.10 shows the converged episode length by Q-learning agent based on "crisp" tile coding. Fig.11 shows the converged episode length and Fig.12 shows the averaged episode length in early 50 trials, by fuzzy Q-learning agent based on fuzzy model coding. From these results, the better number of each partition is selected. Fig.13 shows the learning process of the agents. From the results, it can be seen that the episode length in early trials by fuzzy Q-learning is smaller than one by crisp Q-learning agent. As for the average value of converged episode length, the crisp Q-learning agent is better than the fuzzy Q-learning agent. However, better results are attained occasionally by fuzzy Q-learning agent as shown in Fig.14. This characteristic of fuzzy Q-learning is the same as in mountain car task. The main issues of learning by fuzzy Q-learning are to improve the convergence performance and to sustain huge episode length in early trials.

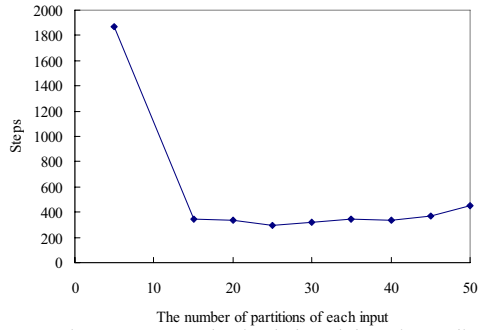



Fig. 5. Converged episode length by crisp coding

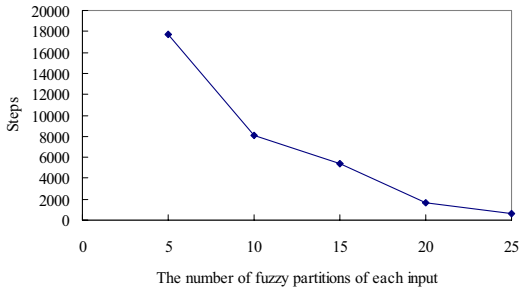



Fig. 6. Converged episode length by fuzzy coding

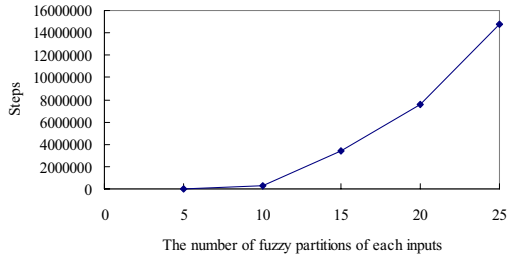



Fig. 7. Averaged episode length in early 50 trials by fuzzy coding

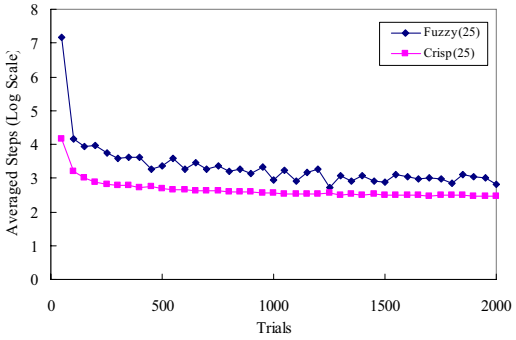



Fig. 8. Learning results of mountain car task

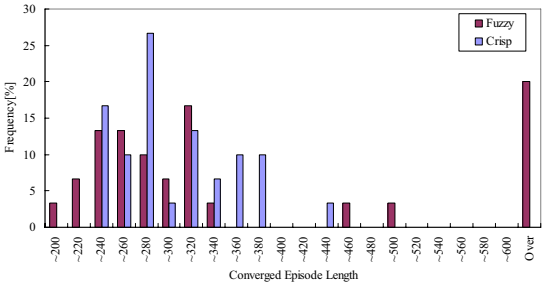



Fig. 9. Distribution of converged episode length

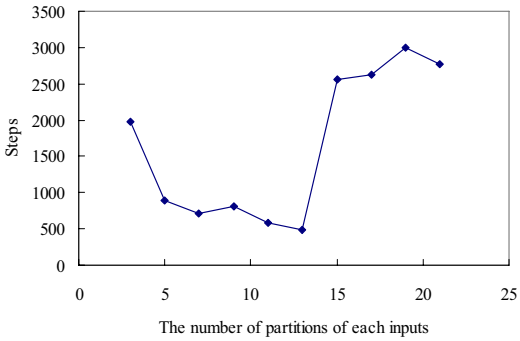



Fig. 10. Converged episode length by crisp coding

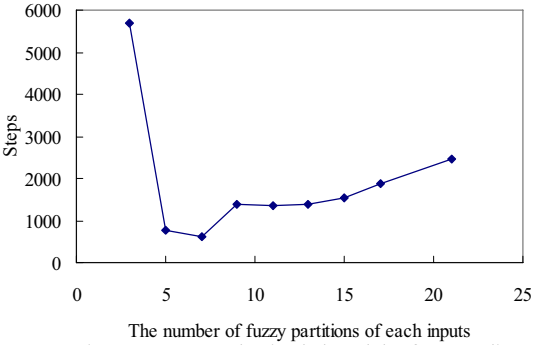



Fig. 11. Converged episode length by fuzzy coding

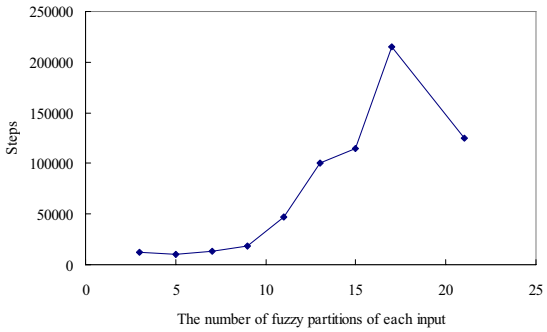



Fig .12. Averaged episode length of early 50 trials by fuzzy coding

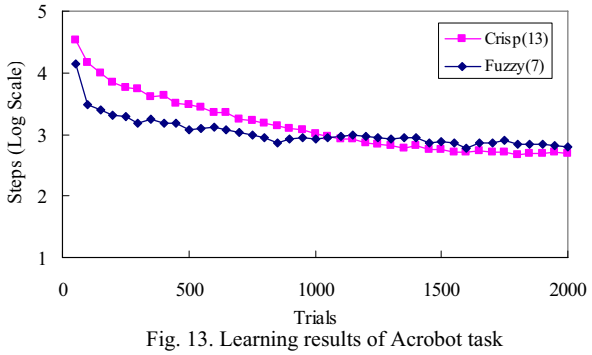



Fig. 13. Learning results of Acrobot task

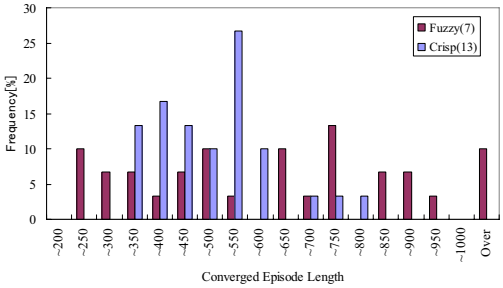



Fig. 14. Distribution of converged episode length

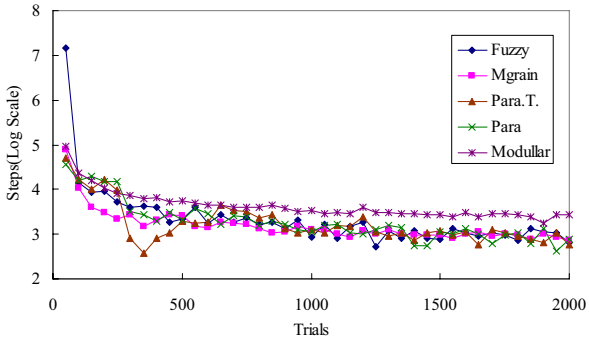



Fig. 15. Learning Results of Mountain Car Task

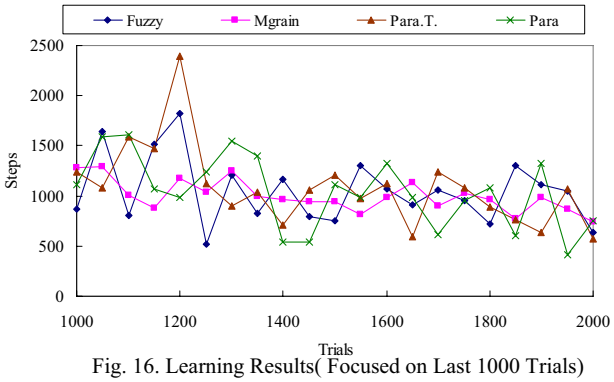



Fig. 16. Learning Results( Focused on Last 1000 Trials)

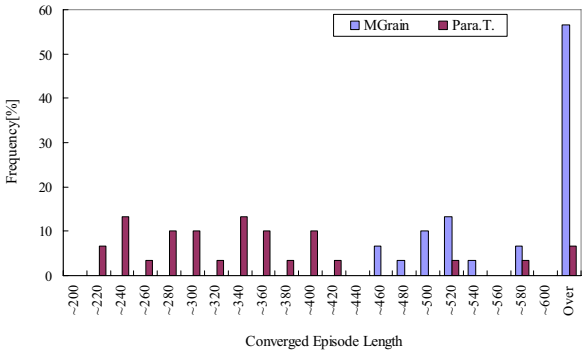



Fig. 17. Distribution of Converged Episode Length

### C. *Results of Mountain Car Task*

Fig.15 shows the result by multi-grain modular fuzzy model("Mgrain"), parallel structured learning("Para"), parallel structured learning with transformation("Para.T."), and modular fuzzy model ("Modular"). Huge iteration in early trials is drastically suppressed by these proposed methods compared with conventional fuzzy Q-learning. The convergence performance except the modular fuzzy model is fair as shown in Fig. 16. However, convergence quality of learning is not improved by the multi-grained modular fuzzy model as shown in Fig.17. From these results, the multi-grained modular fuzzy model can achieve the stable learning process making sacrificing precision of learning slightly. As for effect of the transformation in parallel structured learning, the significance is also investigated by *t*-test. The result is that null hypothesis, i.e. the means do not differ, is rejected with statistical significance level of 0.05. The transformation is significant in the mountain car task.

### D. *Results of Acrobot Task*

Fig.18 shows the result of Acrobot task. Huge iteration in early trials is also drastically suppressed by the proposed methods compared with conventional fuzzy Q-learning. The convergence performance is fair as shown in Fig.19. Especially, the convergence performance of learning is improved by the modular fuzzy model as shown in Fig.20. As for the effect of the transformation in parallel structured learning, it is not significant, as the performance is relatively fair by conventional fuzzy Q-learning agent. The modular fuzzy model of projection type configuration attains good performance unlike in the mountain car task. The performance of the modular fuzzy

model is investigated in detail. The results of converged value by the modular fuzzy model with different construction are summarized in Table.1. In the Table, "2+4" denotes the modular fuzzy model with projection of the module as $\{\{\ \{\theta_1,\dot{\theta}_1\},\{\theta_1,\theta_2\},\{\theta_1,\dot{\theta}_2\},\{\dot{\theta}_1,\theta_2\},\{\dot{\theta}_1,\dot{\theta}_2\},\{\theta_2,\dot{\theta}_2\}\ \},\ \{\theta_1,\dot{\theta}_1,\theta_2,\dot{\theta}_2\}\ \}$. From the results, the performance of the modular fuzzy model containing the single input rule modules such as "1+4", "1+2+4", "1+3+4", and "1+2+3+4" is deteriorated in comparison with the other constructions. The single input rule module obviously falls into well known "incomplete perception problem" by itself. It can be considered that the bad influence caused the performance deterioration. The results by the modular fuzzy model in the mountain car task are also caused by the influence. It can be considered that the situation should be avoided in construction of modular fuzzy model.

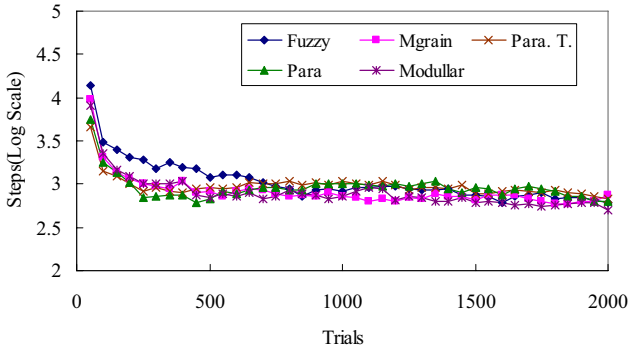


Fig. 18. Learning Results of Acrobot Task

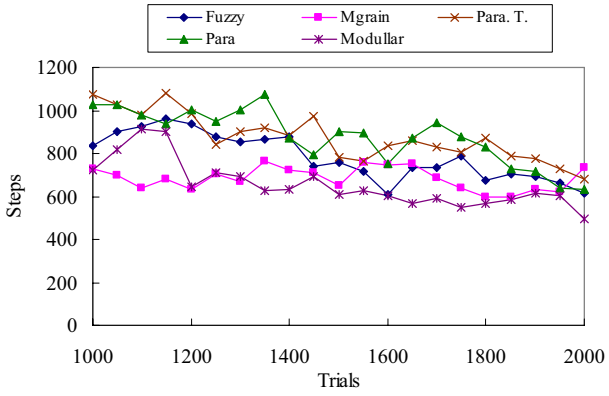


Fig. 19. Learning Results( Focused on Last 1000 Trials)

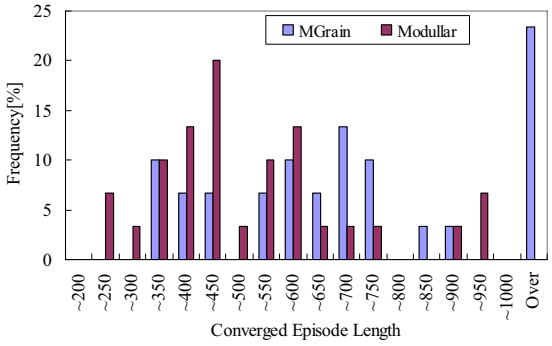


Fig. 20. Distribution of Converged Episode Length

Table. 1. Comparison of Modular Fuzzy Model

| Model | **1+4** | **2+4** | **3+4** | **2+3** |
|---|---|---|---|---|
| Steps | 1123 | 578 | 539 | 970 |
| Model | **1+2+4** | **1+3+4** | **2+3+4** | **1+2+3+4** |
| Steps | 951 | 1019 | 498 | 907 |

## VI. CONCLUSION

In this paper, the implementation of fuzzy Q-learning based on the modular fuzzy model and parallel structured learning was described. Through numerical experiments, I showed that performance of reinforcement learning could be improved by the proposed methods. Especially huge iteration at the early trials in reinforcement learning was avoided. Although the technique in this study formulate except the eligibility trace in order to investigate the essential performance of fuzzy Q-learning, evaluation using the eligibility trace is also needed for further improvement of the learning performance.

## REFERENCES


[1] R. S. Sutton and A. G. Barto, "Reinforcement Learning", *MIT Press*, 1998.

[2] C. J. H. Watkins and P.Dayan, "Technical Note: Q-Leaning", *Machine Learning*, Vol.8, pp.58-68, 1992

[3] K. Miyazaki, H. Kimura, and S. Kobayashi, "Theory and Application of Reinforcement Learning Based on Profit Sharing, " *J. of JSAI*, Vol.14, No.5, pp. 800-807, 1999.

[4] K. Miyazaki, S. Arai, and S. Kobayashi, "A Theory of Profit Sharing in Multi-agent Reinforcement Learning," *J. of JSAI*, Vol. 14, No.6, pp.1156-1164, 1999.

[5] A. Ito and M. Kanabuchi, "Speeding up Multi-Agent Reinforcement Learning by Coarse-Graining of Perception –Hunter Game as an Example-," *Trans. of IEICE*, Vol.J84-D-1, No.3, pp.285-293, 2001.

[6] K. Fujita and H. Matsuo, "Multi-agent Reinforcement Learning with the Partly High-Dimensional State Space," *Trans. of IEICE*, Vol.J88-D-1, No.4, pp.864-872, 2005.

[7] H. Seki, H. Ishii, and M. Mizumoto: "On the Generalization of Single Input Rule Modules Connected Type Fuzzy Reasoning Method," *IEEE Trans. on Fuzzy Systems*, Vol.16, No. 5, pp.1180-1187, 2008.

[8] N. Yubazaki, J. Yi, M. Otani and K. Hirota, "SIRMs Dynamically Connected Fuzzy Inference Model and Its Applications," *Proc. IFSA'97*, vol.3, pp.410-415, 1997.

[9] T. Watanabe and Y. Takahashi, "Hierarchical Reinforcement Learning Using A Modular Fuzzy Model for Multi-Agent Problem," *Proc. of the 2007 IEEE International Conference on Systems, Man, and Cybernetics*, 2007.

[10] P. Y. Glorennec, "Reinforcement Learning: Overview," *Proc. of ESIT*, pp. 17- 35, 2000.

[11] D. Gu and H. Hu, "Fuzzy Multi-Agent Cooperative Q-Learning," *Proc. of the 2005 IEEE International Conferece on Information Acquisition*, pp.193-197, 2005.

[12] R. S. Sutton, "Generalization in Reinforcement Learning: Successful Examples Using Sparce Coarse Coding," *Advances in Neural Information Processing Systems*, Vol.8, pp.1038-1044, MIT Press, 1996.

[13] T. Takagi and M. Sugeno: "Fuzzy Identification of Systems and Its Applications to Modeling and Control," *IEEE Transaction on Systems, Man, and Cybernetics*, Vol. 15, pp. 116-132, 1985.

[14] S. P. Singh and R. S. Sutton: "Reinforcement Learning with Replacing Eligibility Traces," *Machine Learning*, Vol.22, pp.123-158, 1996.