2003 International Symposium on Advanced Intelligent Systems
September 25-28, 2003, Jeju, Korea

Research of Foresight Knowledge by CMAC based Q-learning in Inhomogeneous Multi-Agent System

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

Abstract – **A purpose of our research is an acquisition of cooperative behaviors in inhomogeneous multi-agent system. In this research, we used the fire panic problem as an experiment environment. In Fire panic problem, a fire exists in the environment, and follows in each steps of agent's behavior, and this fire spreads within the constant law. The purpose of the agent is to reach the goal established without touching the fire, which exists in the environment. The fire heat up by a few steps, which exists in the environment. The fire has unsureness to the agent. The agent has to avoid a fire, which is spreading in environment. The acquisition of the behavior to reach it to the goal is required. In this paper, we observe how agents escape from the fire cooperating with other agents. For this problem, we propose a unique CMAC based Q-learning system for inhomogeneous multi-agent system.**

I INTRODBEHAVIOR

The generally CMAC[2][3] based Q-learning[1][6] have some layers, which tallied Q-values, and has a possible to support huge Q-values on large environment. Our proposal technique has just two kinds of layers on system. One is Foresight Knowledge Layer. On the layer, Q-value is not renewed by the reinforcement function. Other layer is Learning Layer. Learning system is able to renew Q-value by the reinforcement function. About this fire panic experiment, we have two experiments to learn an escape way about agents, and have two type agents, which are given a different view area and different rules.

In first experiment, one kind of Agent, which has small view, learns Q-learning about some ways for an exit. So, the first agents will learn a best escape way by Q-learning. Most agents are able to learn a escape way from the fire. And we make a layer of Q-value from all agents. So, this layer is Foresight Knowledge Layer.

At next experiment, we have injected other kind of agents, has a wide view. Normally, all agents has confuse by new states, which there is new type agents. In this case, all agents have to learn behaviors for the escape again, like a bland new rules (knowledge) and we show cooperative behaviors, witch the agents got through the Q-learning. But our proposal technique is able to support Q-value by using Foresight Knowledge Layer as a basic behavior, and has possible to suppress a confusion of agents.

II Q-LEARNING

Q-learning is a kind of unsupervised learning system. This system reinforces any evaluates to select a behavior and a policy by reward/penalty. Argent chose a behavior with the observed state. This behavior is hopefully the best behavior, using some reinforced rules as $Q(s, a)$. Also, a is a behavior and s is the observed state. If there are no fitting rules, a new rule is created on the rule database. Q-learning is able to learn under the uncertain information. The learning systems are composed three units, such as sensitive unit, the learning unit, and the selective unit as shown in fig.1. The learning is going by renewing a evaluate with a behavior. If agents have taken a positive reward, all evaluates are renewed to up. So, the learning unit is renewed all evaluates with the used rules by a reward or a penalty. The renewal function f is given from Eq.1. $Q(s, a)$ is a choosing evaluation to take a renewal in Eq.2. In these cases, γ is the discount rate, α is the learning rate, is the learning rate, and the three parameters are in the range [0,1]. Also, Q_{max} is max evaluation of a next state under the best policy, which is a time until getting a reward from the start. r is +1 as a reward or -1 as a penalty. Fit rules are selected by these renewed evaluates. Eq.2 explain that

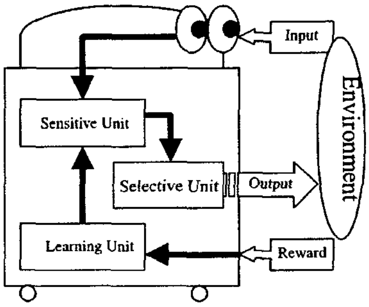

Fig. 1 Framework of Q-learning

e reward is an important matter, and all evaluates e renewed equally with the reward.

$$f = \gamma Q_{max} - Q(s,a) + r \qquad (1)$$

$$Q(s,a) \longleftarrow Q(s,a) + \alpha f \qquad (2)$$

n agent select one action with observed view. As the lective unit on Q-learning, we always use Selection robability based on Bolltmann Distribution, is given qu.3

$$p(a|s) = \frac{\exp\left(\frac{Q(s,a)}{T}\right)}{\sum\limits_{b \in actions} \exp\left(\frac{Q(s,b)}{T}\right)} \qquad (3)$$

III The fire panic problem

We show the fire panic problem on fig.2. In this roblem, a fire exists in the environment, and follows each steps of agent's behavior, and this fire spreads. purpose of the agent is to reach the goal established ithout touching the fire, which exists in the environment. The fire heat up by a few steps, which exists this environment. The fire has un-sureness to the gent Agents have to avoid a fire, which is spreadng in environment. An acquisition of the behavior to each it to the goal is required.

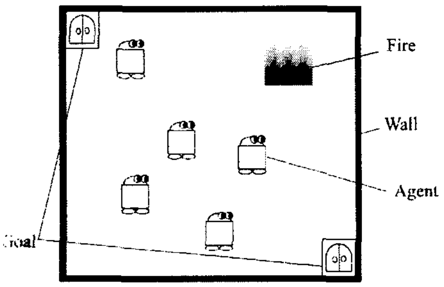

Fig. 2 The fire panic problem

We use two type agents, A1 and A2. There are inhonogeneous multi-agents, which has different view area. View area of A1 is 3 × 3. View area of A2 is 5 × 5. Also, ll agents make sure all objects, Fire, Wall, Agent, nd Goal. All agents are able to select behaviors, Up, Down, Left, and Right. There are not able to go on Wall and Agents.

In the fire panic problem, one action is one step. If teps are over 100, then one trial is done. All Agents earn behaviors by Q-learning.

- If the agent got a goal, penalty r=+1 is given the agent.
- If a trial finished, agents on field is given noreword r=0.
- If the agent touched Fire, the agent has dead and is given penalty r=-1.

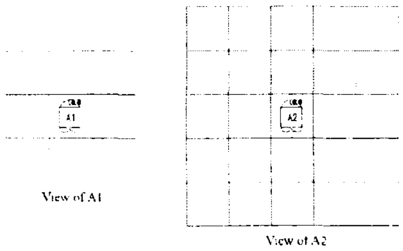

Fig. 3 View area of Agents

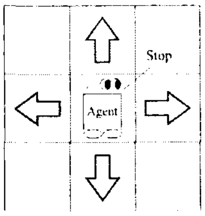

Fig. 4 Behavior of Agents

Also, we show parameters of Q-learning on Table.1.

Table 1 Parameters about Q-learning

α(learning rate)	0.01
γ(discount rate)	0.9
T(Temperatures param)	0.075

IV CMAC based on Q-learning

CMAC (Cerebellar Model Articulation Computer) is the effective method of learning to the problem, which has the large state space. CMAC has excellent generalization ability, and consists of some layers of rules, called tiling. The agent who uses CMAC uses the average of the evaluation value, which it could get for the behavior referring to more than one tiling. We give the framework of CMAC on fig.5.

Evaluation value $Q(x,a)$ is shown equ.4. Hear, i is a number of the current layer. And $q_i(x,a)$ is a output Q-value on the layer of i. m is the number of all layers to use on CMAC.

$$Q(x,a) = \frac{1}{m}\sum_{i=1}^{m} q_i(x,a) \qquad (4)$$

V First Experiment(Non-CMAC)

At first, we have tried learning of A2 between 100,000 trials. Then, after it passes, we inject A1 to the field,

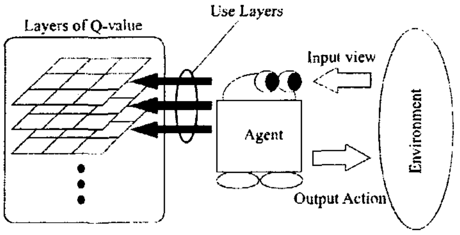

Fig. 5 Framework of CMAC

and we have kept on 100,000 trials which live together A1 and A2.

We give figure6 which show the environment of 1st and 2nd trial. We show a number of goal arrivals by every 1,000 trials on fig.7 and fig.8. As for A1, we could observe the behavior to reach it between the short trial to the goal from the fig.7. As for A2, we could take the number of goal arrival decreases right after injected A1. Though it followed in a trial and rose again, the number of goal arrival didn't reach it to the number of goal arrival before A1 is injected.

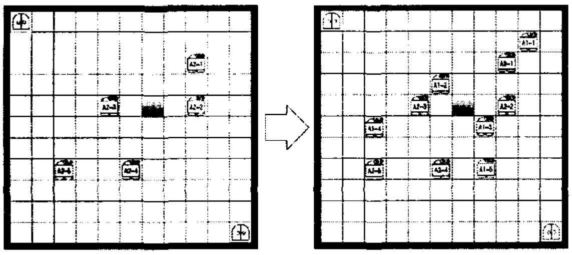

Fig. 6 1st and 2nd trial on Exp.1

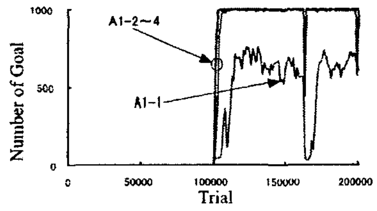

Fig. 7 Number of Goal, taken A1 on Exp.1

VI THE CONSIDERATION OF THE ACQUIRED BEHAVIOR

A1 could learn the behavior to reach it to the goal in the short trial after it was thrown into the environment first. A1 repeats behavior about the direction for walls until A1 did not found it. After A1 touched a wall, A1 have selected the behavior to reach that wall to the goal. However, A2 make confuse the view information by an injected A1, because A2 have never known A1. Therefore, most A2 has taken random behavior right after A1 is injected. It can think that this phenomenon is because a change occurred by A1 come to the view information of A2. The outline of a change in view information is shown in the fig.9.

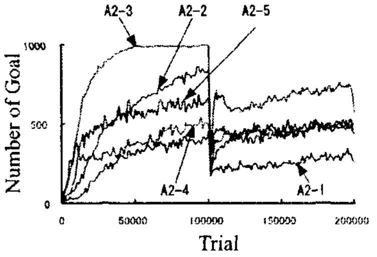

Fig. 8 Number of Goal, taken A2 on Exp.1

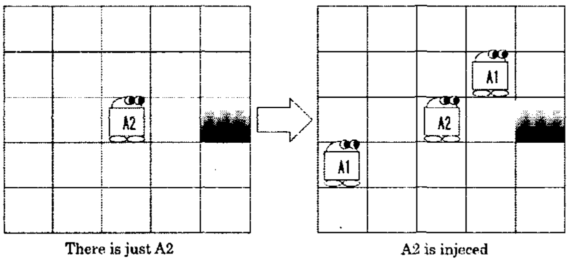

Fig. 9 View information of A2 after A1 is injected

VII SECOND EXPERIMENT(WITH IN CMAC)

The evaluation value found in the above process is applied to Q-learning, and behavior is selected on a probability with the Boltzmann distribution from the viewpoint. CMAC is introduced in A2 that view is 5x5, and it decides to learn in accordance with Q-learning. The use a proposal idea of CMAC is shown in the fig.10. We have injected A2 first as well as the first experiment, and A2 have learned the behavior to a first layer between 100,000 trials. On last 100,000 trials, A2

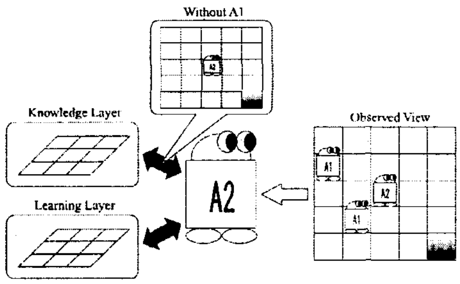

Fig. 10 Freamwork of CMAC

have learned the behavior to a second layer, however Q-value to select behavior is make from an average of Q-values of two layers. So, when reward, punishment is given to it by an environment, only the learning unit updates a Q-value of a second layer. CMAC was introduced in A2, and keep on using it until done. The number of goal arrival of every 1,000 trials of A1 is shown in the fig.11. A1 gets the behavior to reach it between the short trial to the goal after it is thrown into the environment as well as the preceding learning introduction experiment. As for A1-1 as well, the environment where arrival is difficult shows the number of goal arrival of about 200-300 times to the goal, too. The number of goal arrival of every 1,000 trials of A2 is shown in the fig.12. It is compared before the CMAC introduction, and we can observed that the declines of the number of goal arrival right after A1 is injected, given at fig.12. And, it can confirm that the number of goal arrival at 200,000 trials rises so that it is about the same as the number of goal arrival just before A1 is injected. A1 could get the behavior to reach it to the goal in the short trial after it was thrown into the environment as well as the preceding learning introduction experiment. Behavior was repeated in the direction a wall was thought to exist, and A1 have taken the behavior to reach that wall to the goal when it touched a wall. Though, view information changed after A1 is injected, there were a few declines of the number of goal arrival in A2. This can be thought that it could make use of the rule, which got it in preceding learning effectively to be a cause, by the introduction of CMAC. Then, the number of goal arrival rose even in the value after 200,000 trials.

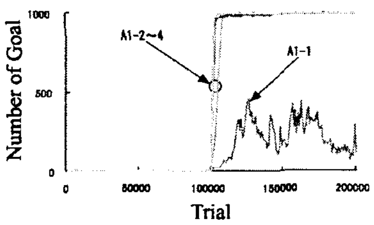

Fig. 11 Number of Goal, taken A1 on Exp.2

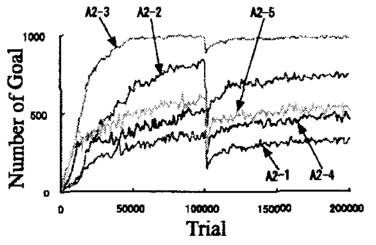

Fig. 12 Number of Goal, taken A2 on Exp.2

VIII CONCLUSION

In this paper, the verification to get the cooperation behavior of the inhomogeneous multi-agent system was done by using the fire panic problem. Concretely, we set up a difference in the view of agents, and verified which occurred from the inhomogeneous of each agent about the cooperation behavior. As a result of the experiment, it could be confirmed that A2 have followed A1.From the difference in the view, it confirmed that the difference was seen in the contents of learning and the learning speed. For making use the rule, which agents have taken by preceding, learning effectively, we have proposed the learning method that Q-learning was combined with CMAC was introduced and verified. The proposal of the technique, which it is made that it gets efficient cooperation behavior in, is shown as the future subject. Many phenomena that the flame, which existed in the environment, obstructed the following behavior of A2 were seen in the experiment. Because it is unintentional and spreads, it can be said that it is the existence, which has un-sureness about the flame. For our future work, verification is more necessary about the efficient way of learning and we have to research about the technique of the cooperation behavior acquisition.

REFERENCES

[1] C.J.C.H.Watkins, P.Dayan:"Technical Note:Q-Learning", Machine Learning, Vol.8, No.3, pp.279–292(1992)

[2] R.S.Sutton : Integrated architectures for learning, planing, and reacting based on approximate dynamic programming ; *Proceedings of the 7th International Conference on Machine Learning,* pp. 216–224 (1990)

[3] R.S.Sutton and A.G.Barto : Reinforcement Learning ; *The MIT Press* (1998)

[4] Y.KAKAZU:"RecentStudies of Multi-Agent Systems", Journal of System, Control and Information, Vol.41, No.8, pp.291–296(1997)

[5] T.ISHIDA: "Discussion on Agents", Journal of Japanese Society for Artificial Intelligence, Vol.10, No.5, pp.663–667(1995)

[6] T.UNEMI:"Reinforcement Learning", Journal of Japanese Society for Artificial Intelligence, Vol.9, No.6, pp.830–835(1994)