Cooperative Behavior Acquisition for Multiple Autonomous Mobile Robots

Koji Nakano*

Masanao Obayashi**

Kunikazu Kobayashi**

Takashi Kuremoto**

* Graduate School of Science and Engineering
Yamaguchi University
2-16-1 Tokiwadai Ube, Yamaguchi 755-8611 Japan

** Faculty of Engineering
Yamaguchi University
2-16-1 Tokiwadai Ube, Yamaguchi 755-8611 Japan

Abstract

This paper proposes a method for multiple autonomous mobile robots to acquire cooperative behaviors through a garbage-collection problem. In the proposed method, robots select the most available target garbage for cooperative behaviors by visual information in unknown environments, and move to the target avoiding obstacles. The learning system in the robot uses Profit sharing (PS), which is one of the reinforcement learning, and the feature of this method is using two kinds of PS-tables. The one is to learn cooperative behaviors using information of other robot's positions, the other is to learn how to control movements. This paper demonstrates effectiveness of the proposed method through simulation and real experiments.

1. Introduction

Recently, many researches on solving problems cooperatively with plural agents have been studied enthusiastically. Specially, a research field that agents get cooperative behavior through reinforcement learning in a dynamic environment has gotten a lot of attention.

Reinforcement learning is a method that agents will acquire the optimum behavior by trial and error by being given rewards in an environment as a compensation for its behaviors. Most of studies on reinforcement learning have been done for a single agent learning in a static environment. Q-learning which is a typical learning method is proved that it converges to an optimum solution for Markov Decision Process (MDP). However, in a multiagent environment, as plural agents' behavior may effect state transition, the environment is considered as non Markov Decision Process (non-MDP), and we must face critical problems whether it is possible to solve.

In such situation, Arai et al.[1] evaluated both Q-learning and PS for the pursuit problem which is one of the multiagent tasks, and suggested that PS is suited to a multiagent environment more than Q-learning.

In this paper, we propose a learning method for a multiagent environment based on PS. And the feature of this method is using two kinds of PS-tables. The one is to learn cooperative behaviors using information of other agent's position and present state, the other is to learn how to control own basic behavior like movements.

We apply the proposed method to garbage collection problem which is one of the multiagent tasks, and demonstrate effectiveness of the proposed method through computer simulation and real experiment.

2. Reinforcement Learning

Reinforcement learning is to learn what to do (how to map situations to actions) so as to maximize a numerical reward signal [2].

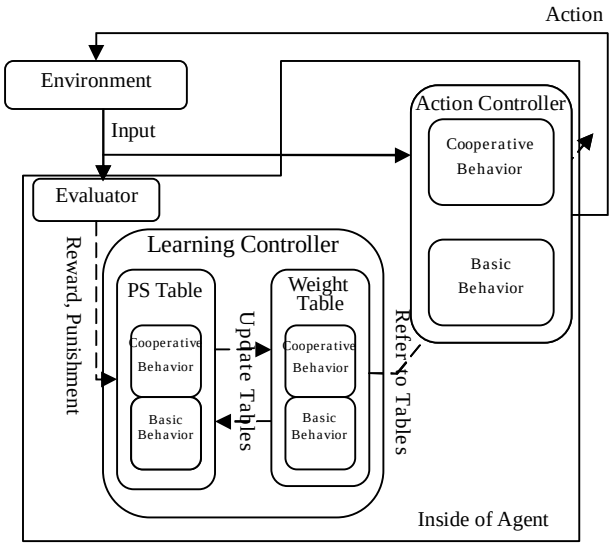


Figure 1. System architecture.

Profit sharing (PS) is one of the learning methods. PS defines a pair of state s and action a as a rule. Rules are stored in PS table each step, and when a learner achieves a goal PS reinforces weight w(s,a) based on PS table. W(s,a) is updated with equation (1).

$$w(s_t, a_t) \leftarrow w(s_t, a_t) + f(t, r) \qquad (1)$$

w(s,a) : weight of rule (s,a)
f : reinforcement function
t : time , r : reward

3. Proposed System

3.1 System architecture

Figure 1 shows proposed system architecture. The system is composed of three parts; action controller, learning controller and evaluator. The feature of the system is to divide behavior of agent into cooperative and basic behavior to learn separately. The learning of cooperative behavior is using information of the other agent's position and present state. The learning of basic behavior is to learn how to control own basic behavior like movements. In a general learning method, when an agent acquires a reward it can hardly estimates own action whether it can cooperate or not. To solve this problem, the proposed system divides the learning into two kinds of behavior, and each behavior is evaluated using different criteria.

3.2 Action controller

The agent controls its action based on w(s,a). It selects an action using following Boltzmann distribution which is one of the probability action selections (equation (2)).

$$p(a \mid s) = \frac{e^{w(s,a)/T}}{\sum_{a_i \in A} e^{w(s,a_i)/T}} \qquad (2)$$

p(a|s) : probability of a on s
T : temperature parameter
A : set of all actions

3.3 Learning controller

The agent learns using PS (as shown in Section 2.1). The parameter w(s,a) is updated by equation (3) and the reinforcement function uses geometric decreasing function.

$$w(s_t, a_t) \leftarrow w(s_t, a_t) + r \cdot \gamma^t \qquad (3)$$

γ : rate of attenuation

3.4 Evaluator

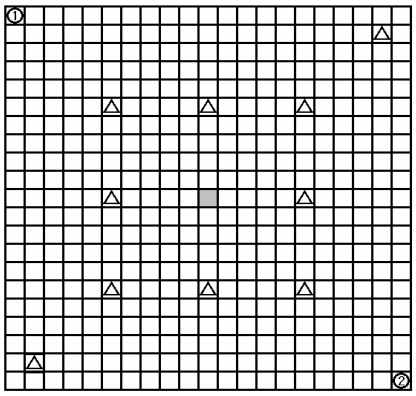
The behavior of the agent is evaluated using next state s'. The evaluation of behavior is similarly established by two criteria.

4. Experiment

We apply the proposed method to garbage collection problem which is one of the multiagent tasks. There are plural agents, garbage and one trash can in the environment, and agents collect garbage and take it to the trash can. Agents learn cooperative behavior and basic behavior by themselves.

4.1 Computer Simulation

Figure 2 illustrates the simulation environment which field size is 21x21 and there are 10 garbage, 2 agents and a trash can on the field. One trial is defined as until all garbage are collected, and 100 trials are considered as 1 episode. We calculate the



①,② :agent
△ :garbage
■ :trash can

Figure 2. Initial positions of agents, garbage, and trash can.

Table 1. The average number of steps to the goal one agent can observe the other or not.

	agent can observe	agent cannot observe
conventional method	118.7	111.1
proposed method	113.3	123.8

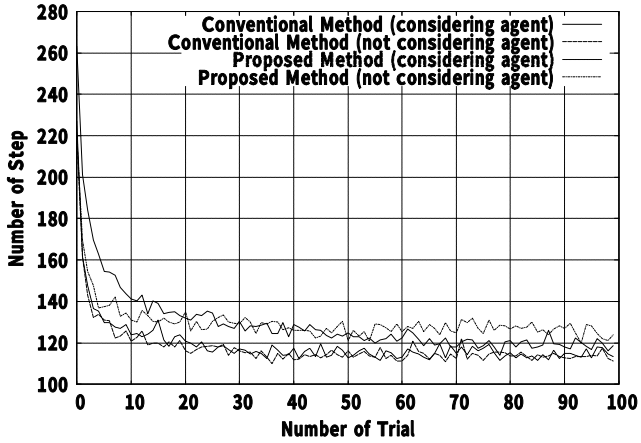
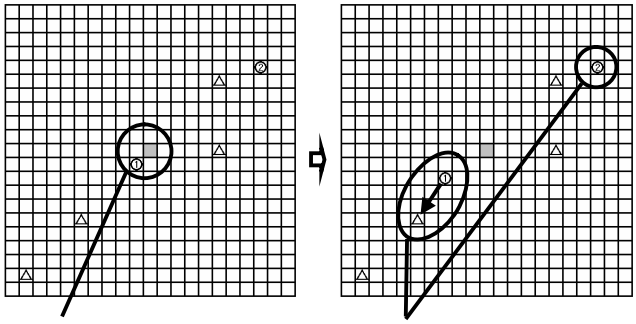


Figure 3. Change of the number of steps.

number of average steps after repeating 100 episodes. At this time, w(s,a) are initialized for each episodes. To verify the effectiveness of the proposed method, we compare the proposed method with the conventional method which also learns using a PStable.

Figure 3 and Table 1 show the result of the



(a)Agent 1 arrives at the trash can.

(b)Agent 1 goes to garbage opposite to agent 2

△ :garbage
①、② :agent
■ :trash can

Figure 4. Cooperative behavior acquired by the experiment with observing the other agent.

experiment. In the case that one agent can observe the other, the agent using the proposed method learns faster than the agent using conventional method. However, when the agent is compared with the agent which is using the conventional method and do not observe the other agent, the performance of the proposed agent is similar to that of the conventional agent.

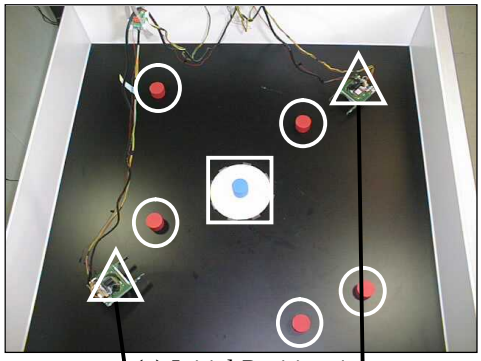
Figure 4 illustrates a cooperative behavior observed in the experiment in which agent observes the other one. After agent 1 took garbage to the trash can (Figure 4 (a)), it do not select the garbage near agent 2 as the object, but another one opposite to agent 2 (Figure 4 (b)). Such behavior often occurred after learning with observing the other agents.

4.2 Real Experiment

Figure 5 illustrates the field of the experiment which field size is 1x1(m) large and has 5 garbage, 2 robots and a trash can. We try following three kinds of the experiments.

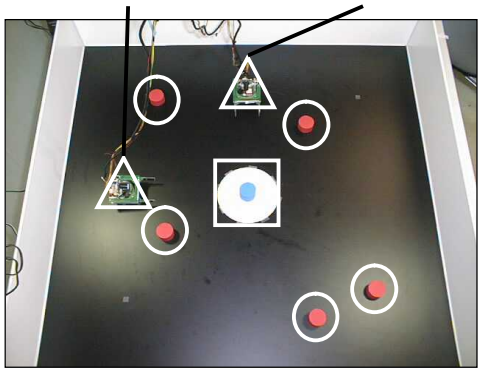
Exp 1: Using weights which are learned in the computer simulations repeating for 10 trials in Figure 5(a)

Exp 2: Using weights which are learned in the real experiments repeating for 10 trials in Figure 5(a)



(a) Initial Position 1

Agent 1 Agent 2



(b) Initial Position 2

△：robot
○：garbage
□：trash can

Figure 5. Initial positions of robots, garbage, and trash can.

Table 2. The number of average steps to garbage collection in the experiment 1-3.

	Number of average steps
Exp 1	201.9
Exp 2	178.2
Exp 3	161.1

Exp 3: Using weights which are the same as Exp 2 repeating for 10 trials in Figure 5(b)

Table 2 illustrates the result of the experiment. Table 2 illustrates that the number of average steps of Exp 2 is decreased compared with Exp 1. Thus the weights learned in computer simulation are available for the real environment, and furthermore, the proposed method can learn flexibly in real environment. On the other hand, the number of average steps in Exp 3 is not increased compared with Exp 2. Thus the weights learned in real environment are applicable to different environments, and this shows that the proposed system is robust.

5. Conclusion

In this paper, we proposed the method which separates the learning of the cooperative behavior and the learning of the basic behavior for a multiagent environment. We demonstrated availability of the method through computer simulation and real experiment, and confirmed that the agents learned quickly and behaved cooperatively.

However, there are two problems in the proposed method. The first is that most of cooperative behaviors are emerged by agents acting basic behaviors each other, so it is difficult to separate the learning of the cooperative behavior and the learning of the basic behavior in other multiagent tasks. Therefore, to apply the proposed method to many multiagent tasks, it should be generalized more. The second is that separating two ways of learning may restrict emergences of cooperative behavior. The cooperative behavior should be emerged fundamentally. Therefore we should establish the framework of the learning of the cooperative behavior not restricting emergences of the cooperative behavior.

References

[1] S. Arai, K. Miyazaki, S. Kobayashi: Methodology in Multi-Agent Reinforcement Learning -Approaches by Q-Learning and Profit Sharing-, Artificial intelligence, Vo.13, No.4, pp.609-618, 1998 (in Japanese)

[2] Richard S. Sutton, Andrew G. Barto: Reinforcement Learning An Introduction, MIT Press, 1998