Reinforcement Learning with dual tables for a partial and a whole space

Nobuo Shibata and Hirokazu Matsui

Graduate School of Engineering, Mie University, Tsu, Mie, Japan
E-Mail: shibata@eds.elec.mie-u.ac.jp

Abstract— **The reduction on the trial frequency is important for reinforcement learning under an actual environment.**

We propose the Q-learning method that selects proper actions of robot in unknown environment by using the Self-Instruction based on the experience in known environment. Concretely, it has two Q-tables, one is smaller, based on a partial space of the environment, the other is larger, based on the whole space of the environment. At each learning step, Q-values of these Q-tables are updated at the same time, but an action is selected by using Q-table that has smaller entropy of Q-values at the situation. We think that the smaller Q-table is used for the knowledge storing as self-instructing. The larger is used for the experiment storing.

We experimented the proposed method with using an actual mobile robot. In the experimental environment, exist a mobile robot, two goals and one of a red, a green, a yellow and a blue object. The robot has a task to carry a colored object into the corresponding goal. In this experiment, the Q-table for the whole has a state for the view of the object and the goals with the colors, the Q-table for the partial has the state without color information. We verified that the proposed method is more effective than the ordinaries in an actual environment.

Keywords—**Reinforcement learning**

Introduction

When human being encounters a known environment, he acts based on his experience. When he encounters an unknown environment, he acts based on his knowledge, that is compressed his experience. Even if not effective, he stacks the experience for the next chance and he will not take the action from next time.

We propose a learning method that uses experience and knowledge for an autonomous robot, based on the above mentions. We have already shown the proposed method is more efficient than the ordinary method on the simulations[1]. In this paper, we experiment the proposed method with using an actual mobile robot in an actual environment. We show that the proposed method is effective in an actual environment as well as a simulation.

Principle

Q-learning

Outline of the ordinary Q-learning: Q-learning is one of the reinforcement learning methods[2].

The ordinary Q-learning has a Q-table, that is collected Q-values, decided by each pair of one of states S and one of actions A. S is the discrete states that an agent can recognize in the environment, and A is the discrete actions that an agent can take in the environment. Q-learning consists of two procedures, one is Q-table updating, the other is Action selecting by the Q-table.

Q-table is updated by each Q-value updating as the follow Eq.1.

$$Q(s_t,a_t) \leftarrow \left(Q(s_t,a_t)+\alpha(r+\gamma \max_a Q(s_{t+1},a)-Q(s_t,a_t)\right) \quad (1)$$

where, t is time, s_t is the state at the time t. $Q(s,a)$ is the Q-value at state s and action a. $r(s,a)$ is a reward decided by each pair of s and a. α is learning rate and γ is discount rate.

Action a is selected by the values of $Q(s,a)$ at state s.The larger $Q(s,a)$ is at the state s, the more selectable the action a is. The agent has learned the suitable state-action pair after enough repeats of updating Q-table and selecting action.

Algorithm

Concept: Here, we describe the proposed reinforcement learning method with Self-Instruction. The proposed methods make two Q-tables (Fig.1) update at the same time, one is larger for experience storing, the other is smaller for knowledge storing. We think the following, an experience is one example in the environment, so we set the Q-table for the experience to the whole space of the environment, and knowledge is compressed the experience in the environment,

so we set the Q-table for the knowledge to a partial space of the environment. The smaller Q-table (for the knowledge storing) makes the learning finish earlier. The larger Q-table (for the experience storing) makes the learning be even in more detail.

This research aims at the method that makes be learned the environment earlier and in more detail by using these two Q-tables at the same time while switching by information entropy by each Q-table.

Materialization: Here, we materialize the concept in the above paragraph to the algorithm of the proposed method, as shown in Fig.2.

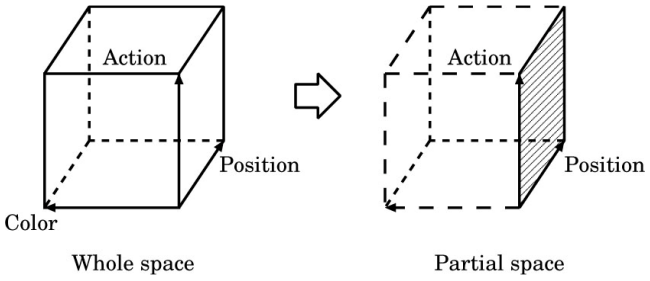


Fig.1. Q-tables for the partial space and the whole space

(1) Set Environment: The learning environment is prepared for a learning agent. In the environment, the proper action is defined. The agent can know indirectly the environment by getting rewards.

(2) Init Q-tables: The agent initializes the two Q-tables to set each Q-value to an init value.

(3) Select Q-table: The agent selects the Q-table with lower information entropy $H(s)$, out of the partial and the whole Q-table, that is calculated by Eq.2.

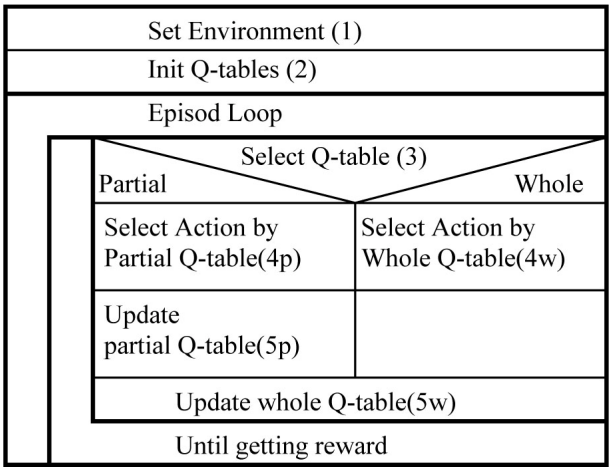


Fig.2. NS chart of the proposed method

$$H(s)=\sum_{a\in A} p(a\mid s)\log_2 \frac{1}{p(a\mid s)} \tag{2}$$

where $p(a\mid s)$ is a probability of selecting action a at the state s, that is defined by the following Eq.3. We think that it means Q-table is effective, that the entropy of Q-table is low.

(4) Select Action (for both Q-tables): The agent decides the action by the Boltzmann selection used generally on Q-learning. The selection probability of action a is shown by Eq.3.

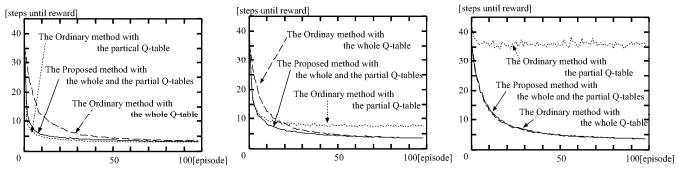
$$p(a\mid s_t)=\frac{\exp\left(\frac{Q(s_t,a)}{T}\right)}{\sum\exp\left(\frac{Q(s_t,a')}{T}\right)} \tag{3}$$

where $p(a\mid s_t)$ is probability of selecting action a on state s_t, t is times, T is temperature.

Update Q-table (for both Q-tables): Q-value is updated by Eq.1. The set of equations is used standard with updating Q-value.

Simulation of different effectiveness of partial Q-table

We show results of average of the 1000 trials simulation in Fig.3. Fig.3(a) is a result in the case that partial Q-table is 100% effective, In other words, the states component of partial Q-table can distinguish the needed situations by 100%. The case of Fig.3(b) and the (c) are 50% and 0% effective, respectively. The ordinary method uses only the partial Q-table or only the whole Q-table. As the results of (a)(b)(c) three cases, the proposed method is not less effective than the ordinaries in any case.



(a)100% effective (b)50% effective (c)0%effective

Fig.3. Results of computer simulation

EXPERIMENTS

In this chapter, we verify that the proposed method is effective by using an actual mobile robot in an actual experiment. We take the following steps to verify it, since the actual experiment needs to a lot of time. At first step, we simulate the actual environment that is very close to the actual experiment to find the comparing points of the proposed method and the ordinary method. Second, at the comparing points in the actual experiment, we compare the proposed with the ordinaries.

Experiment environment

Experiment environment consists of a mobile robot, a goal and a colored object in a field. Object color is randomly selected out of 4 colors, red, blue, green and yellow by each episode. The field is a tray whose size is 0.84[m]x0.54[m]. We use MieC as mobile robot, MieC has a camera for recognizing an object and a goal, two crawlers as drive unit, a blade with magnet for capturing a magnetic object. We use 4 colored poles as object in Fig.4. The pole has iron inside for being captured by MieC. The goals are separated into A area and B area in Fig.5.

Action set and State sets

The action set: The drive unit of MieC is constructed by two crawlers. In this experiment, the crawlers is driven dependently, and make 4 actions "go forward","go backward", "turn left" and "turn right".

The state sets: The state sets consist of the position state set and the color state set.

The position state set has 160 states that are each view position and size of the object and goal. The object image is divided into 6 states, combinations of positions(left, center, or right) and sizes(large or small). The goal image is divided into 18 states, combination of positions(left, center, or right), sizes(large or small) and directions(left, front, right). In addition to these 78 states, we add states that only the object or only the goal view or lose in the image.

The color state set has 5 states. They are four states of object color (Red, Green, Blue, Yellow) and one state of unknown object color.

To apply the proposed method to this experiment, the whole Q-table is constructed by the position state set and the color state set, the partial Q-table is constructed by only the position state sets(without the color state sets).

Experiment

In the experiment, MieC learns the task to carry the object to the corresponding goal area, given reward at the success. We define an episode to be time until getting reward, and define one trial to be 500 episodes. The initial position of the agent, object and goal are shown in fig.5. These positions are not changed in each episode. The agent gets reward in the case that the object carried into the corresponding goal(Red object:goal A, Blue:B, Green and Yellow:A and B). The Q-values are initialized to 0.0, the reward r is set to 1.0, the learning rate α is set to 0.3, and the discount rate γ is set to 0.85. The temperature T of Boltzmann selection is set to 0.07.

At first steps, we verify that the state set and action set are suitable for experiment by using the simulator before experiment of actual world. The position and direction of agent are changed 0.5 and 0.1[deg] per move respectively. It is assumed one step until the current state changes to next state. The agent keeps taking the same action until the current state changes. The results of the simulation are shown

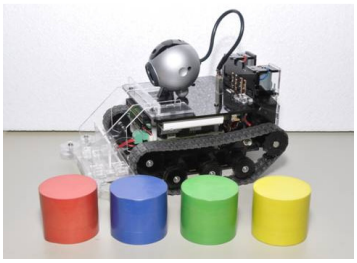


Fig.4. Autonomous mobile robot "MieC"

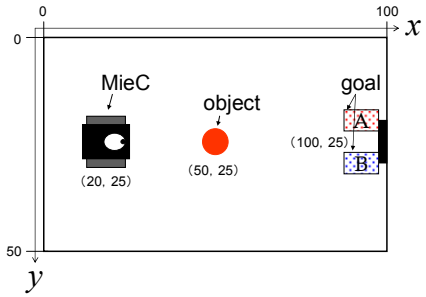


Fig.5. Simulator

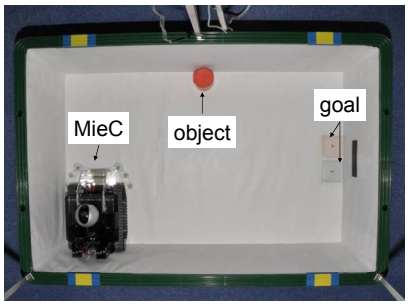


Fig.6. Environment of experiment

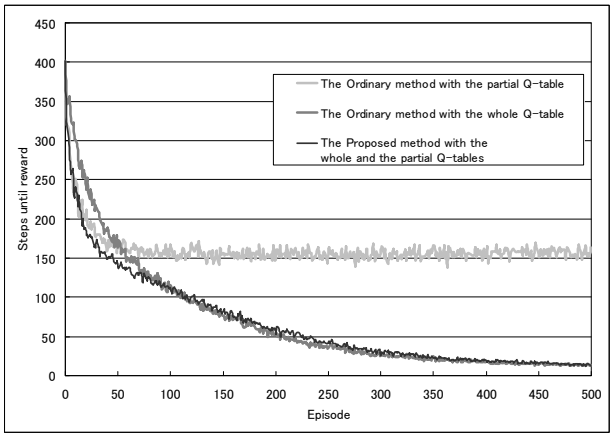


Fig.7. Results of computer simulation

in Fig.7. Each graph is the average of 1000 trials.

(1)The agent with the proposed method learned earlier than the agent with the ordinary method with the whole Q-table.

(2)The agent with the proposed method learned less steps to goal than the agent with the ordinary method with the partial Q-table.

From the point (1) and (2), we verified that the proposed method was more effective than the ordinary methods. And we verify that in the actual experiment at the point (1) and (2), the proposed method is more efficient.

Actual experiment

We observe the period from episode 1 to 16 for point (1), the period from episode 501 to 516 for point (2) to verify the point (1) and (2). We consider the sum of the steps in a period instead of the average of some trials, since in the simulation, the results were the average of 1000 trials, but in the actual experiment, it needs so much time that it cannot.

The results of the actual experiment are shown in Table1 and Table2. The values of result at the point (1) are the sum of steps from episode 1 to 16.

The values of result at the point (2) are the average of steps by episode from episode 501 to 516. In actual experiment, the episodes from 501 to 516 are learned by using the learned Q-table until episode 500 by the simulator instead of the actual experiment.

We verified that the proposed method was more effective than the ordinary methods at the point (1) and (2) in an actual experiment as well as simulation. We consider the difference between the actual and the simulation values to be an error of distribution of the sum of steps instead of trial average.

Table 1 Result of actual experiment at the point (1)

	The ordinary method with the whole Q-table		The proposed method with the whole Q-table and the partial Q-table	
	Actual	Simulation	Actual	Simulation
Steps	2435	3352.8	2199	2760.7
Time[sec]	2843		2317	

Table 2 Result of actual experiment at the point (2)

	The ordinary method with the partial Q-table		The proposed method with the whole Q-table and the partial Q-table	
	Actual	Simulation	Actual	Simulation
Steps	81	151.6	8	12.5

CONCLUSIONS

We verified that the proposed method is more effective than the ordinaries in an actual environment. But the actual experiment needs too much time. So we intend to make the environment of actual experiment that is able to learn full-automatically.

In addition, we intend to expand the proposed method into a method that applies the learned Q-table to a learning of different form robot.

REFERENCES

[1] Osamu NISHIMURA, Hirokazu MATSUI, Chieko HIOKI, Yoshihiko NOMURA:Reinforcement Learning with Self-Instruction by using dual Q-tables, AROB 11th 2006

[2] R. S. Sutton and A. G. Barto: Reinforcement Learning: An Introduction, The MIT Press, 1998