International Journal of Sciences: Basic and Applied Research (IJSBAR)

ISSN 2307-4531
(Print & Online)

http://gssrr.org/index.php?journal=JournalOfBasicAndApplied

Achieving Cooperative Behavior Based on Intention Estimation by Learning Combinations of Modules

Tatsuya Sakato [a]*, Natsuki Oka [b]

[a]*Kyoto Institute of Technology, Matsugasaki, Sakyo-ku, Kyoto, Japan*

[a]*Email: sakato@ii.is.kit.ac.jp*

Abstract

A robot needs to process information appropriately depending on the environment or context. However, some of the abilities required by a robot are often common irrespective of the environment or context. In such situations, the learning agent should not learn the abilities again but use the learning results of previous tasks. In the field of the study of intellectual systems, models have been proposed that solve complex problems by combining modules, each of which serve a specific function such as recognition, planning, or action selection. The models can use the learning results of previous tasks in different environments or contexts by combining modules it has learnt. In this paper, we focus on achieving cooperative behavior based on intention estimation, and propose a model for a learning agent that can acquire combinations of modules using which the agent can achieve cooperative behavior based on intention estimation. The experimental results indicate that a desirable combination of the modules was acquired and the learning process suitably progressed.

Keywords: cooperative behavior; interaction; machine learning; module recombination.

1. Introduction

A robot needs to process information appropriately depending on the environment or context. However, some of the abilities required by a robot are often common irrespective of the environment or context.

---

* Corresponding author.

For example, the abilities of environment recognition, action selection, and action execution are required in trial and error learning tasks, and the abilities of environment recognition, action selection, action execution, and intention estimation are required in cooperation tasks. When required functions are common among tasks, a learning agent should not learn them again but rather use the learning results of previous tasks.

In the field of the study of intellectual systems, models have been proposed that solve complex problems by combining modules each of which serves a specific function such as recognition, planning, or action selection. The models can use the learning results of previous tasks in different environments or contexts by combining learnt modules. Oka proposed a module recombination model in which modules are freely combined. The model consists of a working memory and some functional modules. Inputs/outputs of the modules gather around the working memory. Each input/output has a gate, and states of the gates are controlled by a gate control module. In the model, each module serves a specific function such as recognition, planning, or action selection, and complex problems are solved by combining modules. A specification of the model is described in Section 3.

Modular structure models are inspired from the module structures of our brains. The ultimate aim is to develop agents that are capable of humanlike information processing. In this paper, we focus on cooperative behavior based on intention estimation, and propose a learning agent using the module recombination model to achieve cooperative behavior with a co-player. In this paper, the agent uses its action selection policies that are learnt for a co-player's intentions in advance, and learns a combination of modules for cooperative behavior with the co-player.

In this paper, game play between a learning agent and a co-player is used for learning. We choose games because interactions occurring while playing them such as verbal communication, joint attention, and cooperation are basic and important for children and for robots that work in environments with people. The agent and the co-player stack a block alternately as an experimental task. The agent estimates the co-player's intention based on some action selection policies for stacking blocks learnt in advance, and learns a combination of modules for performing cooperative behavior based on the estimated intention of the co-player. We assume that the agent has already learnt action selection policies for stacking blocks. Therefore, acquisition of a combination of modules for learning the policies is not discussed in this paper.

2. Related Work

In the field of the study of cooperative behavior, there are studies that use multi-agent system or robots. Ikenoue et al. indicated that two agents, among whom one learns passing and the other learns shooting, could exhibit cooperative behavior [1]. Makino et al. proposed a multi-agent reinforcement learning algorithm in which a learning agent performs cooperative behavior such that an action selection policy of the agent is decided based on a co-player's intention [2]. Nagayuki et al. proposed a learning agent that learns by a reinforcement learning method based on co-player actions [3]. The agent estimates co-player's actions from a state of an environment and selects its own actions using state space including co-player's actions. Additionally, Kobayashi et al. proposed a learning agent that solves a task efficiently by selecting performed actions from basic behavior and cooperating behavior that are learnt separately [4].

The learning agents learn cooperative behavior in those studies. However, the combinations of functions necessary for cooperative behavior are fixed in those studies. In this paper, combinations of functions necessary for cooperative behavior are not fixed. A learning agent learns combinations of functions such as environment recognition, intention estimation, plan, and execution, and exhibits cooperative behavior.

3. Module Recombination Model

In the study of intellectual systems, there are many models that solve complex problems by combining simple modules. If strong constraints are implemented, functions and combinations of modules can be learnt [5]. However, combinatorial explosion can occur and learning of combinations can become difficult if the combinations do not have strong constraints. Therefore, a method in which functions of modules are fixed and combinations of modules are learnt [6], and a method in which functions of modules are learnt and combinations of modules are fixed [7] were proposed. Against this background, Oka [8] proposed a model in which modules are freely combined, and inputs/outputs of modules are placed around working memory. Further, Sakamoto et al. indicated that learning functions of the modules as well as their combinations are tasks that can be performed simultaneously in the model [9].

In this paper, we implement the model for a cooperative task. Using the model, we indicate that the learning agent can acquire combinations of modules and using them the agent can achieve cooperative behavior based on intention estimation.

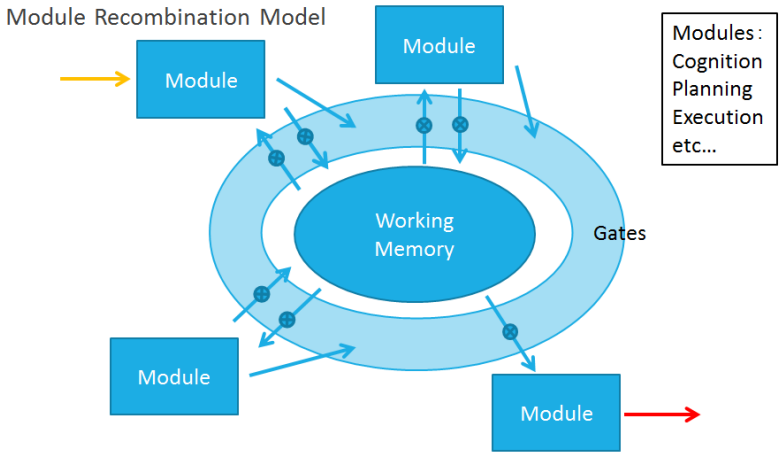

Figure 1: Module recombination model.

4. Method of Intention Estimation

We use a model of others for intention estimation. The probability that the co-player performs action a in state s based on intention I is defined as follows:

$$P(a|s,I) \qquad (1)$$

The likelihood that the intention of the co-player is I when the co-player performs action a in state s is defined

as follows:

$$\mathrm{L}(I) = \mathrm{P}(I|s,a) \qquad (2)$$

Boltzmann selection based on the likelihoods is used for intention estimation.

In this paper, the model of others is the same as an action selection policy of the learning agent. The policy is learnt using a reinforcement learning method in advance.

5. Method for Determining Cooperative Behavior

In this paper, the learning agent performs actions based on the estimated intention of the co-player for cooperative behavior.

When the co-player performs action a in state s, and the state becomes s_1, the probability that the learning agent performs a cooperative action a is determined based on (3).

$$\mathrm{P}(a|s_1,I) \qquad (3)$$

where I is the estimated intention of the co-player.

In this paper, cooperative behavior is defined as behavior wherein an agent performs the actions instead of the co-player. The learning agent performs cooperative behavior by performing actions based on the estimated intention.

6. Experiment

We examine whether a learning agent acquires a combination of modules for cooperative behavior based on the estimated intention of the co-player.

6.1. Experimental Environment

We suppose that there is a learning agent and a co-player in the experimental environment. The co-player performs actions based on its intention. The learning agent learns cooperative behavior through interactions with the co-player.

There are some distinguishable blocks and a stage for stacking the blocks in the environment (see figure 2).

6.2. Experimental Settings

We use playing blocks as the experimental task. The learning agent and the co-player stack a block alternately. Table 1 shows the procedure for round one. The learning agent receives a reward of 10.0 when a reward is given to the agent.

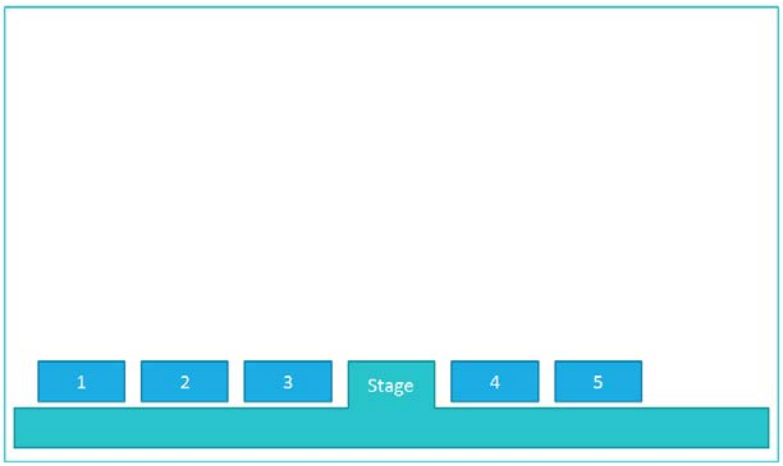

Figure 2: Experimental environment.

Table 1: Procedure for round 1.

Procedure for round one	
1:	**Initialize** the environment.
2:	The co-player decides its intention.
3:	The co-player stacks a block based on its intention.
4:	**if** the blocks are stacked in the order of the intention **then**
5:	**End** the episode.
6:	The learning agent performs an action by combining its modules.
7:	**if** the agent stacks a block following the co-player's intention **then**
8:	The learning agent receives a reward.
9:	**Update** a policy of the agent.
10:	**if** the blocks are stacked in the order of the co-player's intention **then**
11:	**End** the episode.
12:	**Go to** 3.

7. Agent Specifications for the Experiment

Figure 3 describes the configuration of the learning agent. The learning agent consists of a working memory, a state recognition module, an action selection module, an intention estimation module, an action generation module, a gate control module, and an action decision policy for each intention. As an initial state of the agent, the agent receives information about the state of the environment from the state recognition module, and the action of the co-player from the action recognition module. The agent stores the information in the working memory. All gates are kept closed in the initial state.

7.1. Working Memory

Working memory is able to store information about a state, an intention, and an action. Inputs/outputs of the

module are placed around the working memory. Modules exchange information with each other using the working memory. Each input/output has a gate, and the states of the gates are controlled by the gate control module.

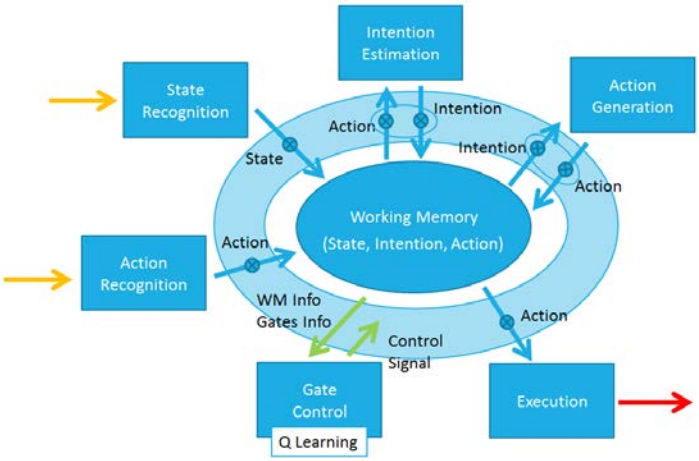

Figure 3: Configuration of a learning agent.

7.2. State Recognition Module

The state recognition module recognizes blocks on the stage as the state of the environment. The state differs depending on the blocks on the stage and the order of the blocks. The module sends information about the recognized state to the working memory when the gate of the module is opened.

7.3. Action Recognition Module

The action recognition module recognizes a block stacked by the co-player as an action of the co-player. The module sends information about a recognized action to the working memory when the gate of the module is opened. The module sends nothing if the module has no information about an action. The information in the module is cleared when the module sends it to the working memory.

7.4. Intention Estimation Module

The intention estimation module estimates the intention of a co-player based on information about the state and action in the working memory. The module sends information about the estimated intention to the working memory when the gate of the module is opened. The module estimates an intention by the method explained in Section 3. The module sends nothing if the working memory has no information about a state or an action.

7.5. Action Generation Module

The action generation module generates an action based on information about a state and an intention in the working memory. The module sends information about a generated action to the working memory when a gate of the module is opened. The module selects an action using the softmax method (4) based on a Q table of the

intention. Parameter T denotes a temperature parameter of the method. The module sends nothing if the working memory has no information about the state or intention.

$$\pi(s,a) = \frac{\exp(Q(s,a)/T)}{\sum_{a'} \exp(Q(s,a')/T)} \qquad (4)$$

7.6. Gate Control Module

The gate control module controls the gate of each module. The module uses the Q learning method to learn a combination of the modules. The module represents a state based on (i) whether the working memory has information about the state of the environment, the intention, and the action, and (ii) which gate is opened. An action of the module is defined based on which gate is opened. When the module opens gate a, and receives reward r, action value $Q(s_t, a)$ is updated as follows:

$$Q(s_t,a) \leftarrow Q(s_t,a) + \alpha(r_{t+1} + \gamma \max Q(s_{t+1},p) - Q(s_t,a)) \qquad (5)$$

where α denotes the learning rate, γ denotes the discount factor, and r denotes the reward the learning agent receives. The module decides which gate to open using the softmax method (4). Parameter T denotes the temperature parameter of the method.

7.7. Action Selection Policies

In this paper, each intention corresponds to an action selection policy that is learnt in advance in this experiment. Each action selection policy has a Q table that is learnt for stacking blocks in a particular order. Five intentions are learnt in advance as follows:

- 12345
- 54321
- 34521
- 45123
- 45312

Each number denotes the order of stacking the blocks as in Figure 4.

7.8. Parameters

In the action generation module, temperature parameter T is 0.1. In the gate control module, learning rate α is 0.1, discount factor γ is 0.9, and temperature parameter T is 0.1.

7.9. Achieving Cooperative Behavior Based on Intention Estimation using a Module Recombination Model

In this paper, the learning agent performs cooperative behavior by the method explained in Section 4 based on the intention of the co-player estimated by the method explained in Section 3. The agent can perform

cooperative behavior if the agent opens gates in the order as in the example described in figure 5. If the agent opens gates in the order from an initial state, the agent estimates an intention of the co-player, recognizes a present state, generates a cooperative action, and executes the action. The agent cannot perform cooperative behavior if the agent opens gates in the order as in the example described in figure 6. The agent imitates the co-player's action if the agent opens gates in order.

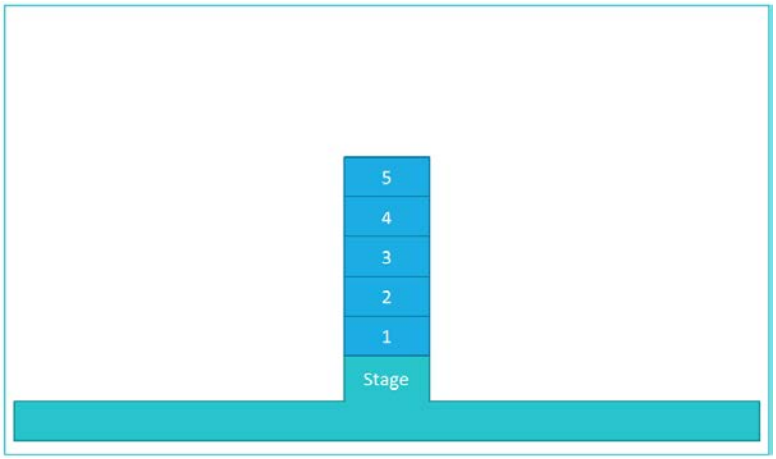

Figure 4: Example of stacking blocks in the order "12345."

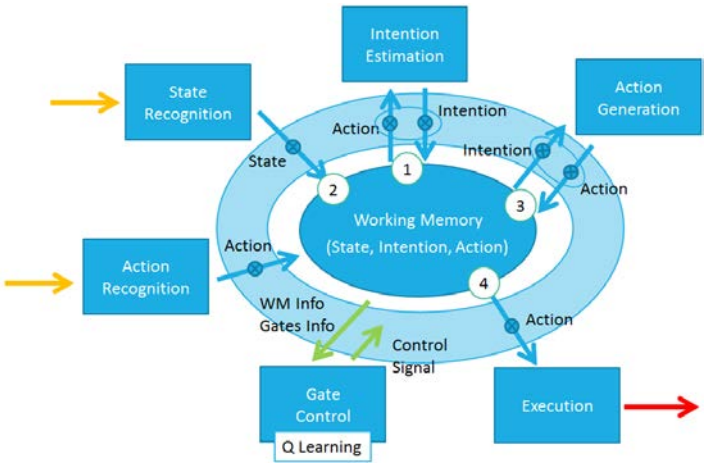

Figure 5: Example of a combination of modules for generating cooperative behavior. The learning agent performs cooperative behavior if it opens gates in the order described in this figure of the initial state.

8. Results and Discussion

An acquired combination of the modules and rewards that the agent received were evaluated.

The successfully acquired combination of the module is shown in figure 7. The gates are opened in the order described in the figure if the agent selects gates by a greedy policy from an initial state. The result indicates that a desirable combination of the modules explained in section 7.9 is acquired in the experiment.

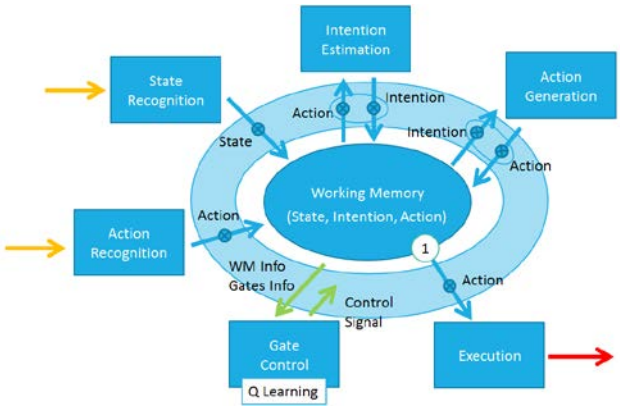

Figure 6: Example of a wrong combination of modules for generating cooperative behavior. The learning agent cannot perform cooperative behavior with this combination. The agent imitates the co-player's action with the combination.

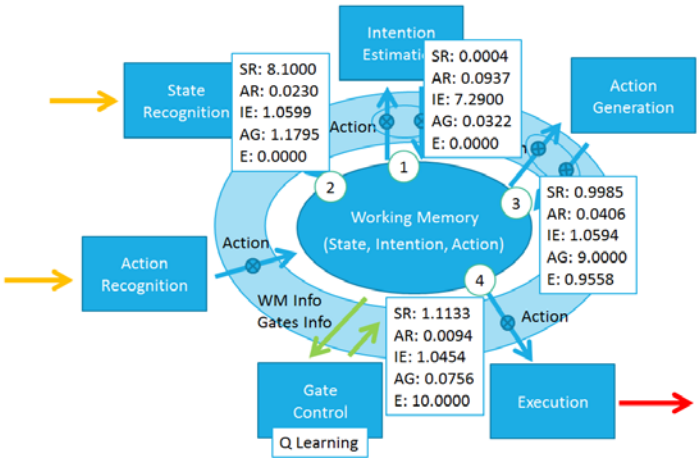

Figure 7: Successfully acquired combination of modules that is represented by numbers. Gates open in the order of numbers according to the greedy policy. Each gate opens based on Q values by the number. Each symbol denotes an initial of each module (e.g., SR denotes the gate of the state recognition module.).

An average of the sum of rewards the agent received in each of the 10 rounds of 20 attempts is shown in figure 8. The rewards the agent received increased as the experiment progressed. The result indicates that the learning process suitably progressed.

9. Conclusion

This paper proposed a model in which a learning agent could acquire a combination of modules by which the agent performs cooperative behavior based on intention estimation. Playing blocks were used for the experimental task. The experimental results indicate that a desirable combination of the modules was acquired in the experiment, and that the learning process suitably progressed.

In this experiment, each intention corresponds to an action selection policy to acquire the intention. Each action

selection policy has a Q table learned in advance. Learning each action selection policy in the experimental task is the aim of our next study.

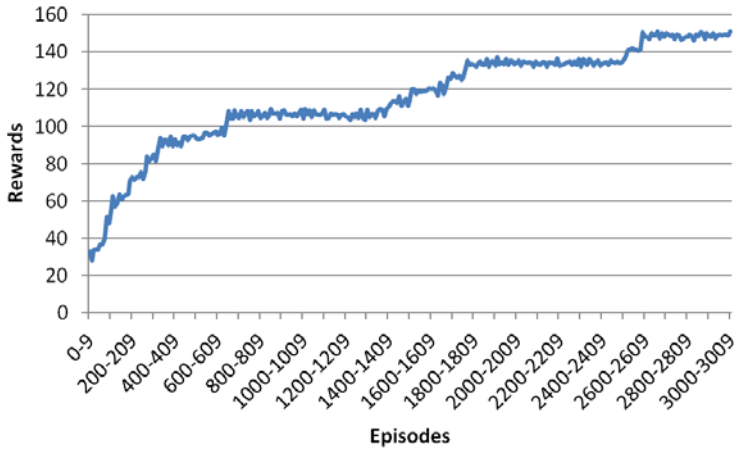

Figure 8: Rewards a learning agent received.

References

[1] S. Ikenooue, M. Asada and K. Hosoda. "Cooperative behavior acquisition by asynchronous policy renewal that enables simultaneous learning in multiagent environment," in Proc. the 2002 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems, 2002, pp. 2728-2734.

[2] T. Makino and K. Aihara. "Multi-agent reinforcement learning algorithm to handle beliefs of other agents' policies and embedded beliefs," in Proc. International Joint Conference on Autonomous Agents and Multiagent Systems, 2006, pp. 789-791.

[3] Y. Nagayuki, S. Ishii and K. Doya. "Multi-agent reinforcement learning: An approach based on the other agent's internal model," in Proc. IEEE International Conference on Multi-agent systems, 2000, pp. 215-221.

[4] K. Kobayashi, K. Nakano, T. Kuremoto and M. Obayashi. "Cooperative behavior acquisition of multiple autonomous mobile robots by an objective-based reinforcement learning system," in Proc. International Conference on Control, Automation and Systems, 2007, pp. 777-780.

[5] R. Jacobs, M. Jordan, S. Nowlan and G. Hinton. "Adaptive mixture of local experts". Neural Computation, vol. 3, pp. 79-87, 1991.

[6] A. Ogawa and T. Omori. "Acquisition of learning processing in navigation task using functional parts combination model." The Transactions of the Institute of Electronics, Information and Communication Engineers D-II, vol. J87-D-II, 2004, pp. 987-998.

[7] J.C. Sylvester, J.A. Reggia, S.A. Weems and M.F. Bunting. "Controlling working memory with learned instructions." Neural Networks, vol. 41, pp. 23-38, May 2013.

[8] N. Oka. “Apparent “free will” caused by representation of module control,” in No matter, Never mind: Proc. Toward a Science of Consciousness: Fundamental Approaches, 1999, pp. 243–249.

[9] Y. Sakamoto, T. Sakato, M. Ozeki and N. Oka. “Learning functions of modules and their free combinations in a module recombination model,” in Proc. The 26th Annual Conference of the Japanese Society for Artificial Intelligence, 2012, 3B2-R-2-6.