An Action Selection Method Based on Estimation of Other's Intention in Time-Varying Multi-agent Environments

Kunikazu Kobayashi, Ryu Kanehira, Takashi Kuremoto, and Masanao Obayashi

Yamaguchi University, Tokiwadai 2-16-1, Ube, Yamaguchi 755-8611, Japan
{koba,wu,m.obayas}@yamaguchi-u.ac.jp
http://www.nn.csse.yamaguchi-u.ac.jp/k/

Abstract. An action selection method based on the estimation of other's intention is proposed to treat with time-varying multi-agent environments. Firstly, the estimation level of other's intention is stratified as active, passive and thoughtful levels. Secondly, three estimation levels are formulated by a policy estimation method. Thirdly, a new action selection method by switching three estimation levels is proposed to cope with time-varying environments. Fourthly, the estimation methods of other's intention are applied to the Q-learning method. Finally, through computer simulations using pursuit problems, the performance of the estimation methods are investigated. As a result, it is shown that the proposed method can select the appropriate estimation level in time-varying environments.

Keywords: Multi-agent system, Reinforcement learning, Intention estimation, Action selection, Pursuit problem .

1 Introduction

Multi-agent systems can emerge intellectual behavior such as cooperative behavior toward a goal of agent group through mutual interaction among individual agents. In general, multi-agent systems can cope with intractable problems that single-agent systems cannot solve and dynamical environments [1]. As giving agents a reinforcement learning function, multi-agent systems can maximize its potential abilities such as cooperativeness and robustness [2,3].

To realize cooperative behavior in multi-agent systems, if agents are able to communicate with others using some kind of communication tool, agents can pick up on other's intention. Agents however have to estimate the other's intention if agents are unable to communicate with others by restrictions of robot hardware and external environments. In the present paper, we assume intention as agent's behavior with a goal and a plan after Bratman's definition [4]. In this situation, agents are required to accurately estimate the other's intention and to cooperatively act toward a goal of agent group.

Nagayuki et al. presented a policy estimation method which can estimate the other's action to be taken based on the observed information about the other's action sequence [5,6]. They successfully applied it to the Q-learning method [7] which is one of reinforcement learning methods and showed to get effective the other's policy. Meanwhile,

B.-L. Lu, L. Zhang, and J. Kwok (Eds.): ICONIP 2011, Part III, LNCS 7064, pp. 76–85, 2011.

Yokoyama et al. proposed an approach to model action decision based on the other's intention according to an atypical situation such as human-machine interaction [8,9]. They defined three estimation levels of the other's intention and presented a computational model of action decision process to solve cooperative tasks through a psychological approach.

Although the approach of Nagayuki et al. assumes the policy estimation as the other's action prediction, they don't consider a deep intention estimation at all, i.e. a self-action prediction by others. The self-action therefore consists of a self-experience and the other's action prediction. On the other hand, The approach of Yokoyama et al. estimates the other's intention but has to learn in advance by classifying action probabilities according to goals and cannot cope with time-varying environments.

In the present paper, we propose an action selection method based on the estimation of the other's intention to treat with time-varying multi-agent environments. In Section 2, we briefly outline the Q-learning method. In Section 3, we give three estimation levels of the other's intention based on the work of Yokoyama et al. and formulate these three estimation levels using the policy estimation method of Nagayuki et al. We furthermore propose a new action selection method by switching the three estimation levels to cope with time-varying environments. At the same time, all the estimation methods are applied to the Q-learning method. In Section 4, we investigate the performance of the estimation methods through computer simulations using pursuit problems. As a result, we confirm that the proposed method can select the appropriate estimation level in time-varying environments.

2 Reinforcement Learning

Reinforcement learning is a machine learning technique that a decision-making agent takes actions and then receives rewards in an environment, and finally acquires the optimum policy by trial and error [2,3].

The Q-learning method by Watkins et al. is a representative reinforcement learning technique and guarantees that a value function will converge to the optimal solution by appropriately adjusting a learning rate in Markov decision process environments [7]. A state-action value function $Q(s, a)$ is updated by (1) so as to take the optimal action by exploring it in a learning space.

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha\left(r + \gamma \max_{a' \in A} Q(s',a')\right), \tag{1}$$

where s' is the next state after an agent takes action a, r is a reward at the state s', A is a set of all possible actions, α is a learning rate ($0 < \alpha < 1$), γ is a discount rate ($0 \leq \gamma \leq 1$).

Probabilistically, an agent selects action a at state s according to policy $\pi(s, a)$. Throughout the present paper, we employ the Boltzmann method defined by (2) as the policy.

$$\pi(s,a) = \frac{\exp\left(\beta Q(s,a)\right)}{\sum_{b \in A} \exp\left(\beta Q(s,b)\right)}, \tag{2}$$

where β is a parameter to control randomness of action selection called as inverse temperature parameter. The policy $\pi(s, a)$ referes to a probability to select action a at state s.

3 Intention Estimation Levels and Their Application to Reinforcement Learning

3.1 Intention Estimation Levels

In the present paper, intention estimation refers to the estimation of an action sequence toward a goal. We formulate three estimation levels according to the depth of the intention estimation which referes to the work of Yokoyama et al. [8,9]. Then we propose a new estimation level which can switch three levels depending on the situation. Note that we abbreviate level as Lv. and intention estimation as IE.

Lv.0 IE. We sometimes behave without awareness of others. We call this as active behavior and label Lv.0 IE. Lv.0 IE assumes an action selection mechanism which an agent approaches a self-goal without intention estimation of others.

Lv.1 IE. We often select actions by predicting the other's actions. We call this as passive behavior and label Lv.1 IE. Lv.1 IE is an action selection mechanism by predicting the other's actions based on an other's action history.

Lv.2 IE. We often decide actions by estimating the other's intention. We call this as thoughtful behavior and label Lv.2 IE. Lv.2 IE is an action selection mechanism not only by predicting the other's actions but also by estimating the other's intention based on the other's situation.

Lv.3 IE. We often choose actions by changing estimation levels depending on the situation. We label this Lv.3 IE. Lv.3 IE is an approach to switch the above three estimation levels, i.e. Lv.0, Lv.1, and Lv.2 IEs depending on the situation.

In the next section, we implement these estimation levels with reinforcement learning.

3.2 Application to Reinforcement Learning

At first, we formulate the estimation levels described in 3.1 in order to apply them to reinforcement learning. In the present paper, we employ the Q-learning method described in 2 as a reinforcement learning method. After that, we propose a new reinforcement learning system which can switch three estimation levels depending on the situation.

Action Selection Method at Lv.0 IE. Lv.0 IE realizes active action selection without considering the other's intention. The learning at Lv.0 IE therefore employs the standard Q-learning method.

To begin with, let us denote a self-state as s_s, a self-action as a_s ($\in A_s$), and an other's action as a_o ($\in A_o$). Note that A_s and A_o refer to the sets of all possible actions by self and the other, respectively. In the present paper, both action elements of self and those of the other assume completely identical, i.e. $A_s = A_o$. Let us denote a Q-function as $Q(s_s, a_s, a_o)$. An update rule of $Q(s_s, a_s, a_o)$ is represented by

$$Q(s_s, a_s, a_o) \leftarrow (1-\alpha)Q(s_s, a_s, a_o) + \alpha\left(r + \gamma \max_{a'_s \in A_s} \bar{Q}(s'_s, a'_s)\right), \tag{3}$$

where s'_s is a next self-state and $\bar{Q}(s_s, a_s)$ is the average of $Q(s_s, a_s, a_o)$ with respect to a_o.

$$\bar{Q}(s_s, a_s) = \sum_{a_o \in A_o} \frac{1}{|A_o|} Q(s_s, a_s, a_o), \tag{4}$$

where $|A|$ denotes the number of elements in set A.

We employ the Boltzmann method in (2) as action selection. Note that the Q-function in (2) should be replaced by (4). At Lv.0 IE, self-action a_s with the higher value of $\bar{Q}(s_s, a_s)$ tends to be selected.

Action Selection Method at Lv.1 IE. Lv.1 IE realizes passive action selection with predicting the other's actions. The learning at Lv.1 IE is assumed as the Q-learning method based on other's action estimation.

We employ the policy estimation method by Nagayuki et al. [5,6] for Lv.1 IE. The method predicts an other's action using a policy estimation function $P_s(s_s, a_o)$. The P-function $P_s(s_s, a_o)$ is updated by (5) for all the other's actions to be taken, i.e. $a_o (\in A_o)$

$$P_s(s_s, a_o) \leftarrow (1-\rho) P_s(s_s, a_o) + \begin{cases} \rho & (a_o = a_o^*), \\ 0 & (\text{otherwise}), \end{cases} \tag{5}$$

where a_o^* is the actual other's action and ρ is a positive parameter ($0 \leq \rho \leq 1$). As updating P-value by (5), P-value with a_o^* increases and the other P-values decrease. Repeatedly updating P-values, an agent can predict other's actions. Note that $\sum_{a_o \in A_o} P_s(s_s, a_o) = 1$ holds at any time.

An update rule of $Q(s_s, a_s, a_o)$ at Lv.1 is denoted by

$$Q(s_s, a_s, a_o) \leftarrow (1-\alpha) Q(s_s, a_s, a_o) + \alpha \left(r + \gamma \max_{a'_s \in A_s} \bar{Q}(s'_s, a'_s) \right), \tag{6}$$

where $\bar{Q}(s_s, a_s)$ is a weighted average of $Q(s_s, a_s, a_o)$ with respect to a_o.

$$\bar{Q}(s_s, a_s) = \sum_{a_o \in A_o} P_s(s_s, a_o) Q(s_s, a_s, a_o). \tag{7}$$

We also employ the Boltzmann method in (2) at Lv.1 IE as action selection. Note that the Q-function in (2) should be replaced by (7). As introducing the policy estimation method into Q-learning, Q-values are able to update by predicting other's actions. At Lv.1 IE, action a_s with the higher value of $\bar{Q}(s_s, a_s)$, i.e. the average of $Q(s_s, a_s, a_o)$ with respect to policy estimation function P_s tends to be selected. In this way, the prediction of other's actions reflects self-action selection. As a result, an agent can gradually predict other's actions.

Action Selection Method at Lv.2 IE. Lv.2 IE realizes thoughtful action selection which an agent decides a self-action by estimating the other's intention based on other's situation. The agent should therefore consider the self-intention which is estimated by the other.

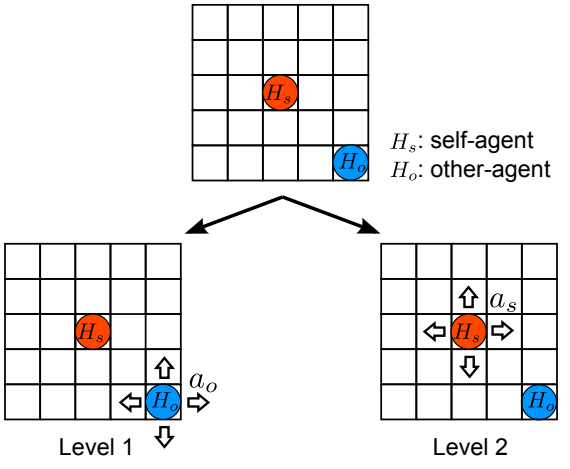

Fig. 1. Difference of policy estimation between Lv.1 and Lv.2

In the present paper, the estimation of self-intention by the other realizes by replacing the other's position with the self-position as shown in Fig.1. The P-function is updated in the similar with Lv.1 IE. Since the P-function at Lv.2 IE is assumed as the self-policy estimation by the other, it is denoted by P_o. The P-function at state s_o is updated by (8) for all the self-actions to be taken, i.e. $a_s (\in A_s)$

$$P_o(s_s, a_s) \leftarrow (1-\rho) P_o(s_s, a_s) + \begin{cases} \rho & (a_s = a_s^*), \\ 0 & (\text{otherwise}), \end{cases} \tag{8}$$

where a_s^* is the actual self-action. The P-value that an agent actually took increases according to (8). The agent predicts an action which the other desires for the self.

An update rule of $Q(s_s, a_s, a_o)$ using P_o at Lv.2 is denoted by

$$Q(s_s, a_s, a_o) \leftarrow (1-\alpha) Q(s_s, a_s, a_o) + \alpha \left(r + \gamma \max_{a'_s \in A_s} \bar{Q}(s'_s, a'_s) \right), \tag{9}$$

where $\bar{Q}(s_s, a_s)$ is a weighted average of $Q(s_s, a_s, a_o)$ with respect to a_s.

$$\bar{Q}(s_s, a_s) = \sum_{a_o \in A_o} P_o(s_s, a_s) Q(s_s, a_s, a_o). \tag{10}$$

We also employ the Boltzmann method in (2) at Lv.2 IE as action selection. Note that the Q-function in (2) should be replaced by (10). At Lv.2 IE, self-action a_s with the higher value of $\bar{Q}(s_s, a_s)$, i.e. the average of $Q(s_s, a_s, a_o)$ with respect to policy estimation function P_o tends to be selected.

Action Selection Method at Lv.3 IE. Lv.3 IE is an approach to switch three estimation levels, i.e. Lv.0, Lv.1, and Lv.2 IEs depending on the situation.

Since an observed state will change with time in real environments, an agent has to appropriately select an action in time-varying environments. If the estimation level is fixed, however, the agent has difficulty adjusting to the environment. The agent therefore needs to appropriately change the estimation levels.

We propose a selective method of the estimation levels in (11) to cope with time-varying environments.

$$c = \arg \max_{i \in \{0,1,2\}} PQ_i, \tag{11}$$

where PQ_i $(i = 0, 1, 2)$ is defined as follows.

$$\begin{cases} PQ_0 = \displaystyle\sum_{a_s \in A_s} \sum_{a_o \in A_o} \frac{1}{|A_o|} Q(s_s, a_s, a_o), \\ PQ_1 = \displaystyle\sum_{a_s \in A_s} \sum_{a_o \in A_o} P_s(s_s, a_o) Q(s_s, a_s, a_o), \\ PQ_2 = \displaystyle\sum_{a_s \in A_s} \sum_{a_o \in A_o} P_o(s_s, a_s) Q(s_s, a_s, a_o). \end{cases} \tag{12}$$

The proposed selective method is described as follows. Firstly, we calculate PQ_i, i.e. the product sum of P-values and Q-values. Note that P-values at Lv.0 IE mean the equal probability because they don't predict actions and estimate intention. Secondly, we compare the values of PQ_i and choose the estimation level c that has the maximum value of PQ_i. Note that we use the update rules of P-values and Q-values as described before. We can therefore select an estimation level according to the learning situation of P-values and Q-values.

4 Computer Simulation

4.1 Problem Setting

A pursuit problem is a well-known multi-agent problem which plural hunters pursuit preys (or a prey) and catch them in a grid field. The followings are assumed in the present paper.

- 9×9 grid field with a torus structure in Fig.2.
- Two hunters (H_1 and H_2) and two preys (P_1 and P_2) in the field. Initially, H_1 and H_2 are located in the center of the field, P_1 is located near from the hunters, and P_2 is located far from the hunters as shown in Fig.2(a). It allows that hunters and preys are occupied in the same cell.
- Two hunters can observe all the cells (complete observation) and act according to their own estimation levels. The hunters can synchronously move up, down, left, or right by one cell or stay on the same cell.
- A goal state is assumed that each hunter is occupied in one of four adjacent cells. An example of the goal state is depicted in Fig.2(b).

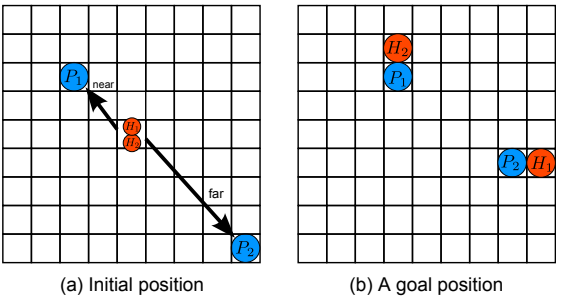

Fig. 2. (a) Initial position of two hunters (H_1 and H_2) and two preys (P_1 and P_2), (b) an example of a goal position

4.2 Simulation Setting

Two hunters get a positive reward $r = 50$ if a goal state is reached and get a negative reward $r = -0.01$ if otherwise. The number of steps is limited to 30,000 and we start a next trial if it reaches the limit.

We prepare the following two kinds of simulation setting according to behavioral patterns of two preys, P_1 and P_2.

- Simulation 1
  - P_1 can only move up.
  - P_2 can only move right.
- Simulation 2
  - Two preys can only move right before 1,500 episodes.
  - Two preys can only move left after 1,500 episodes.

Under this simulation setting, each hunter has to choose a different prey with the other hunter as a target. Since the initial positions of P_1 and P_2 are different, one hunter needs to choose P_2 as a target with considering the other hunter. In simulation 2, hunters are required to adjust to the change of the environment.

The parameters were selected as $\alpha = 0.1$, $\gamma = 0.99$, $\beta = 10$, and $\rho = 0.75$. Initial Q-values and P-values were set to 0.1 and 0.2, respectively. These parameters were selected so as to get the best performance through preliminary simulations.

4.3 Simulation Results

The learning curves for four combinations of estimation levels in Simulation 1 and 2 are shown in Figs.3 and 4, respectively. In all the simulations, the number of steps is averaged for 10 trials. In this figure, Lv.i-j ($i, j \in \{0, 1, 2, 3\}$) refers to pairs of estimation levels which assigned for two hunters. For example, we denote Lv.0-2 if H_1 is Lv.0 IE and H_2 is Lv.2 IE. We pick up the representative pairs of estimation levels out of 10 pairs, i.e. Lv.0-0, Lv.1-1, Lv.2-2, and Lv.3-3. We also enlarge the learning curves around the last episodes, i.e. from 2,800 to 3,000 episodes for comparison. The

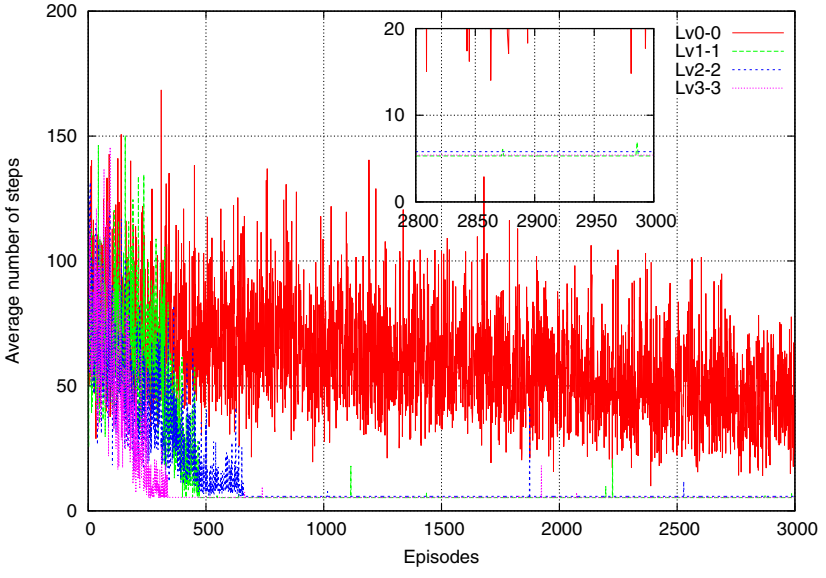

Fig. 3. Learning curves of in Simulation 1

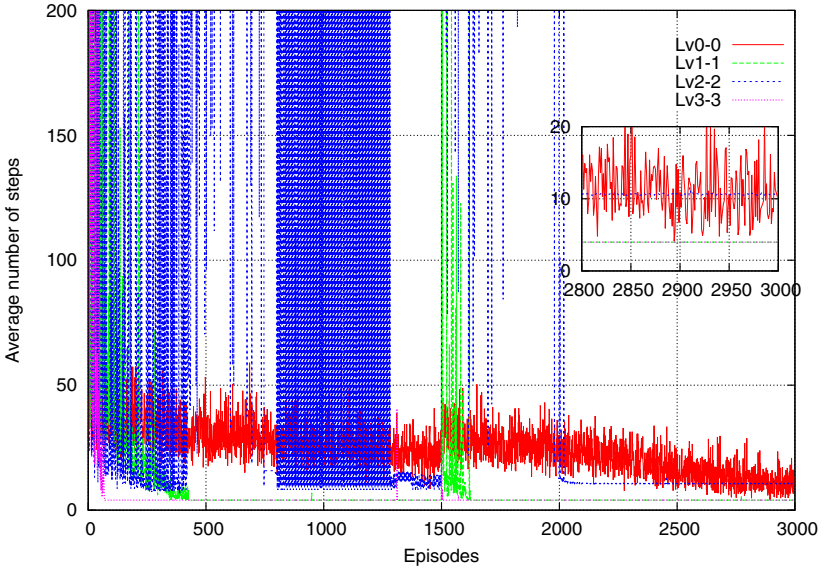

Fig. 4. Learning curves in Simulation 2

transition diagrams of estimation levels of two Lv.3's hunters in simulation 1 and 2 are shown in Figs.5 and 6, respectively. Although we need to update both P-values and Q-values, we initialized Q-values with the learned Q-values without loss of generality. We got the learned Q-values at Lv.0-0 after 3,000 episodes. In this situation, the agents don't have any advantage or disadvantage.

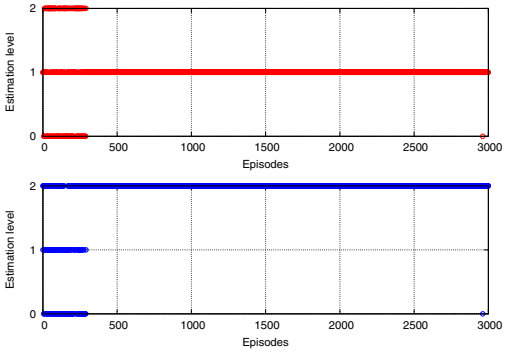

Fig. 5. Transition diagrams of estimation levels of two Lv.3's hunters in Simulation 1

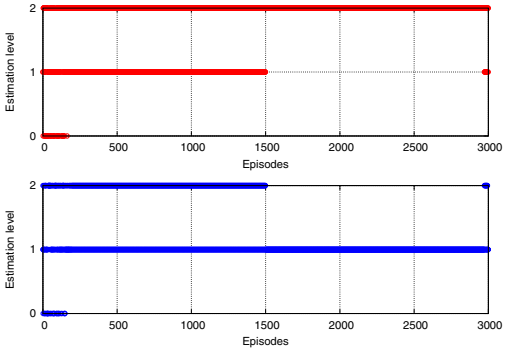

Fig. 6. Transition diagrams of estimation levels of two Lv.3's hunters in Simulation 2

As seen in Figs.3 and 4, the convergence of Lv.3-3 (combination of the proposed method) is faster than other combinations of estimation levels, i.e. Lv.0-0, Lv.1-1, and Lv.2-2. In simulation 2, as the environment is changed at 1,501 episode, the other combinations other than Lv.3-3 get increase their average number of steps. As seen in Figs.5 and 6, the estimation levels of two hunters begin at Lv.0 IE and then transit to Lv.1 IE and Lv.2 IE. Finally, H_1 and H_2 choose Lv.1 IE and Lv.2 IE in simulation 1, respectively and they select Lv.2 IE and Lv.1 IE, respectively. All the combinations of estimation levels without Lv.3 IE are six. After we conduct simulations with all the combinations, we found that Lv.1-2 showed the best performance in both simulation

1 and 2. Consequently, Lv.3-3 automatically searched the best combination, i.e. Lv.1-2. This result agrees with the work of Yokoyama et al.[8,9] That is, Yokoyama et al. pointed out that the best performance out of six combinations was Lv.1-2.

5 Summary

In the present paper, we have proposed an action selection method based on the estimation of the other's intention to treat with time-varying multi-agent environments. Firstly, we have stratified the estimation levels of the other's intention as active, passive and thoughtful levels incorporating the work of Yokoyama et al. Secondly, we have formulated three estimation levels using the work of Nagayuki et al. Thirdly, we have proposed a new action selection method by switching the three estimation levels to cope with time-varying environments. Fourthly, the estimation methods of the other's intention has been applied to the Q-learning method. Finally, through computer simulations using pursuit problems, we have investigated the performance of the estimation methods. As a result, we have confirmed that the proposed method could select the best combination of estimation levels even in time-varying environments.

Acknowledgments. This work was partly supported by Grant-in-Aid for Scientific Research (No.20500207, 20500277, and 23500181) from MEXT, Japan.

References

1. Stone, P., Veloso, M.: Multiagent Systems: A Survey from a Machine Learning Perspective. Autonomous Robots 8(3), 345–383 (2000)
2. Kaelbling, L.P., Littman, M.L., Moore, A.P.: Reinforcement Learning: A Survey. Journal of Artificial Intelligence Research 4, 237–285 (1996)
3. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press (1998)
4. Bratman, M.E.: Intention, Plans and Practical Reason. Harvard University Press (1987)
5. Nagayuki, Y., Ishii, S., Ito, M., Shimohara, K., Doya, K.: A Multi-Agent Reinforcement Learning Method with the Estimation of the Other Agent's Actions. In: Proceedings of the Fifth International Symposium on Artificial Life and Robotics, vol. 1, pp. 255–259 (2000)
6. Nagayuki, Y., Ito, M.: Reinforcement Learning Method with the Inference of the Other Agent's Policy for 2-Player Stochastic Games. Transactions on the Institute of Electronics, Information and Communication Engineers J86-D-I(11), 821–829 (2003) (in Japanese)
7. Watkins, C.J.C.H., Dayan, P.: Q-learning. Machine Learning 8(3-4), 279–292 (1992)
8. Yokoyama, A., Omori, T., Ishikawa, S., Okada, H.: Modeling of Action Decision Process Based on Intention Estimation. In: Proceedings of Joint 4th International Conference on Soft Computing and Intelligent Systems and 9th International Symposium on advanced Intelligent Systems, vol. TH-F3-1 (2008)
9. Yokoyama, A., Omori, T.: Model Based Analysis of Action Decision Process in Collaborative Task Based on Intention Estimation. Transactions on the Institute of Electronics, Information and Communication Engineers J92-A(11), 734–742 (2009)