Self-Generation of Reward by Moderate-Based Index for Senor Inputs

Self-Generation of Reward by Moderate-Based Index for Senor Inputs

Paper:

Self-Generation of Reward by Moderate-Based Index for Senor Inputs

Kentarou Kurashige and Kaoru Nikaido

Department of Information and Electronic Engineering, Muroran Institute of Technology
27-1 Mizumoto-cho, Muroran, Hokkaido 050-8585, Japan
E-mail: {kentarou@epsilon2.csse, s2124127@mmm}.muroran-it.ac.jp
[Received August 17, 2014; accepted December 19, 2014]

In conventional reinforcement learning, a reward function influences the learning results, and therefore, the reward function is very important. To design this function considering a task, knowledge of reinforcement learning is required. In addition to this, a reward function must be designed for each task. These requirements make the design of a reward function unfeasible. We focus on this problem and aim at realizing a method to generate a reward without the design of a special reward function. In this paper, we propose a universal evaluation for sensor inputs, which is independent of a task and is modeled on the basis of the indicator of pleasure and pain in biological organisms. This evaluation estimates the trend of sensor inputs based on the ease of input prediction. Instead of the design of a reward function, our approach assists a human being in learning how to interact with an agent and teaching it his/her demand. We recruited a research participant and attempted to solve the path planning problem. The results show that a participant can teach an agent his/her demand by interacting with the agent and the agent can generate an adaptive route by interacting with the participant and the environment.

Keywords: reward generation, reinforcement learning, pleasure and pain, robot-human interaction, inborn index and immunity evaluation

1. Introduction

Reinforcement learning (RL) [1] is the popular learning method for real robots [2–4]. Of course it is not still sufficient to use RL for real robots and there are many researches to improve this method by extracting useful knowledge from an agent's experience [5] or by combining with other methods [6, 7]. These researches focused on knowledge of RL, a value function, and aimed at realizing a speedy learning. As the other important component of RL, there is a reward which affects the learning performance, and on which we focus in this study. In general, each reward is produced by a reward function, which is predefined to achieve a task. The most common problem is the difficulty of designing the reward function. It is very difficult to design one for a complex task in a complex environment.

To overcome this difficulty, there are some researches to generate an appropriate reward for a given task. In [8], the researchers focus on the learning of rewards by a tutor, so there is no need to design a reward function. Especially, this method can allow the existence of an unreliable tutor. However, the tutor must have expert knowledge of the rewards for the given task and check the series of rewards through trials. In [9], a human being who knows the rewards is not needed. The method is one that is combined with evolutionary computation (EC) and generates rewards or policies for selecting actions. Therefore, instead of designing a reward function, we need to design an evaluation function on EC. Unfortunately, this is the same difficulty as the design of a reward function. [10–12] focus on the generation of rewards without the use of any other evaluation function. In these studies, a pain signal is defined as a disagreeable state that has no relation with the given task. The aim of these studies is the generation of appropriate actions by interacting with a human being or an environment. However, it is very difficult to control a pain signal. In these cases, we need to design the "environment" or the "task" to give an agent an appropriate pain signal. Other words, the design of pain signals considering an interaction between an agent and environment must be needed. This seems to be a different type of difficulty as compared to the above, but the basic problem is the same. There is a need to design something for a given concrete task.

We tackle this difficulty and aim at the development of a method to generate a reward without designing anything for a task. As an approach, we consider the universal evaluation for sensor inputs, which is independent of the task. To design this evaluation, we assumed that an agent takes an action in environment and designed the following cases: If an agent faces similar circumstances frequently and can predict the trend of sensor inputs, then the inputs have a great potential for selecting the appropriate action. The influence of whether these inputs are useful or useless is dependent on each task, but the agent can estimate the influence of the task. In contrast, if the agent faces a new situation or cannot predict the trend of sensor inputs because of a violent change in data, these inputs have a less potential for the agent. Even if the agent could use the sensor inputs sometimes, there is no guarantee that

the agent can use them the next time. In addition to this usefulness of sensor inputs, we considered pleasure and pain in biological organisms [13–15], which are universal indexes. Further, we developed a concrete algorithm of the universal evaluation for sensor inputs and proposed a method to generate a reward with the universal evaluation. In our approach, the method has no special gateway for a reward related to a task. To teach a given task, a human being must interact with an agent through some devices and must learn the way to interact with it. In this paper, we propose a method so as to be a natural interaction for an outside of the agent and easy to learn this interaction. There are many researches on the interaction between a human being and an agent [16–18], but these are the one-sided interaction from a human being to an agent. We try to generate a reward by true interaction and mutual learning between a human being and an agent.

In Section 2, we provide details about the proposed method. In Section 3, we explain the experiment and the results. Finally, in Section 4, we present the conclusions of this study.

2. Moderate-Based Reward Generation with Sensor Inputs

2.1. Guideline for Handling Sensor Inputs

We modeled the indicator of pleasure and pain in biological organisms in response to sensory inputs and used the model in an agent to indicate the evaluation of sensory inputs.

We can consider the ease of predicting the input as an indicator of pleasure or pain in a biological organism, e.g., if the organism is being stroked by a human. The input is a constant force in constant motion. It can predict the input easily, because the input is stable. Therefore, it obtains pleasure. On the other hand, if it is beaten by a human, the input is a sudden and strong force. Therefore, predicting the input becomes difficult, because the input is unstable. Therefore, it receives pain. Thus, we can assume that it obtains pleasure from a stable input and receives pain from an unstable input.

The strength of the input is also another indicator. When strong forces are applied on the organism's body, it feels pain. Even the mental state of a human being becomes destabilized when isolated for a long time in a sensory deprivation environment. As indicated in previous studies [13, 14], we can consider that pain is also received when the sensory input is extremely weak. Therefore, we can consider that pleasure is obtained from the input of average strength between the upper limit and no strength.

We decided to generate the reward by calculating the evaluation value of the sensor input by using these indicators.

2.2. Algorithm for Reward Generation

We consider the strength of the input by using the average of the data values and consider the ease of prediction by using the data variance and the variance of the data variance. First, we define three evaluation values corresponding to the three statistical values for $input_i$ at elapsed time t starting from the interaction.

To calculate the statistical values in each interaction, the average of the input data, etc., we define the working-set window T_{window} and use the input data in this window size.

We define the evaluation value $A_{i,t}$, which is related to the strength as Eq. (1).

$$A_{i,t} = \exp\left\{ \frac{-\left(avg_{i,t} - \left(\frac{\max_{input_i}}{2} + \delta_{i,t}\right)\right)^2}{2\left(\max_{input_i} \cdot N_i \left(1 - \left(\frac{|\delta_{i,t}|}{\frac{\max_{input_i}}{2}}\right)\right)\right)^2} \right\} \quad \ldots\ldots (1)$$

Here, the subscript i denotes the number used for distinguishing among the sensors. $avg_{i,t}$ represents the average of the data values, $\max_{input_i}$ indicates the upper limitation value of the sensor $input_i$, and N_i is a constant value. Further, $\delta_{i,t}$ denotes the value used for adjusting the range of the input data. Basically the trend of the input data is different between situations, especially between people. By the use of this parameter, this equation can be adapted to each situation, or each interaction with each human being. The $\delta_{i,t}$ is updated by using Eq. (2).

$$\delta_{i,t} \leftarrow \delta_{i,t} + \beta_i \left\{ avg_i - \left(\frac{\max_{input_i}}{2} + \delta_{i,t}\right)\right\} \quad \ldots (2)$$

Here, β_i denotes the constant value and the range of $\delta_{i,t}$ is $-\max_{input_i}/2 + \beta_i \leq \delta_{i,t} \leq \max_{input_i}/2 - \beta_i$.

Next, we define two evaluation values $B_{i,t}$ and $C_{i,t}$, which are related to the data variance and the variance of the data variance for the ease of prediction, respectively, as Eqs. (3) and (4).

$$B_{i,t} = \exp\left\{ \frac{-v_{1,i,t}^2}{2\left(\frac{\max_{input_i}^2}{4} \cdot M_i\right)^2} \right\} \quad \ldots (3)$$

$$C_{i,t} = \exp\left\{ \frac{-v_{2,i,t}}{2\left(\frac{\max_{input_i}^4}{64} \cdot L_i\right)^2} \right\} \quad \ldots (4)$$

Here, $v_{1,i,t}$ denotes the data variance for $input_i$, and $v_{2,i,t}$ represents the variance of $v_{1,i,t}$. Further, M_i and L_i are constant values. M_i, L_i and N_i to calculate $A_{i,t}$ are determined by considering a characteristic of $sensor_i$ and do not change for given tasks.

With $A_{i,t}$, $B_{i,t}$, and $C_{i,t}$, we define the evaluation $eval_{i,t}$ as shown in Eq. (5).

$$eval_{i,t} = A_{i,t} \cdot B_{i,t} \cdot C_{i,t} \quad \ldots\ldots (5)$$

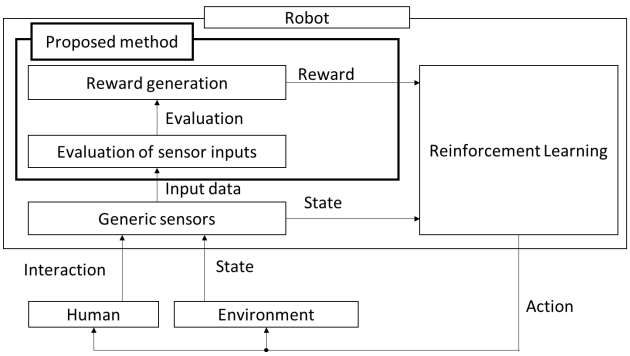


Fig. 1. Overall learning process.

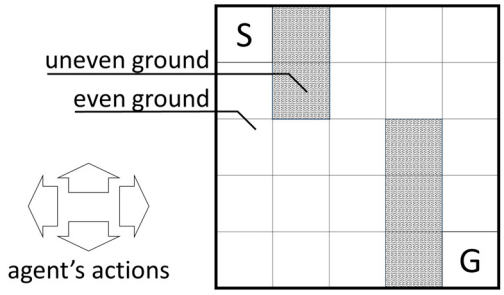


Fig. 2. Path planning problem on an open grid field.

The value of $eval_{i,t}$ is accumulated through the interactions, and therefore, we define the reward as the average of $eval_{i,t}$. In this study, the end of the interactions is the end to the input data entered by a human being. Further, the elapsed time is discrete time and increases by one for each input from a human being. Therefore, we define the reward shown in Eq. (6) by using the total number of input values, n, at each interaction.

$$reward = \frac{\sum_{t=0}^{n} \sum_{i} w_i \cdot eval_{i,t}}{n} \quad \ldots\ldots\ldots \quad (6)$$

Here, w_i denotes the weight for $input_i$. This parameter shows the priority of $sensor_i$. In this study, sensors for a human being and environment are separated clearly. So the ratio of these parameters for each sensor shows the relative priority between inputs from a human being and environment.

2.3. Learning Process by Proposed Method

The process of reinforcement learning used in the proposed method is as follows: First, the agent selects an action. Second, the agent detects a change in the sensor input from its interactions with a human being and from changes in the environment. Third, the agent evaluates the sensor input by using the proposed method. Fourth, the agent generates its own reward on the basis of the evaluation. Fifth, the agent updates the Q-value on the basis of the reward. Then, the agent returns to the first step. An overview of the learning process is shown in **Fig. 1**.

3. Experiment and Results

3.1. Outline of the Experiment

We performed a computer simulation to confirm that the proposed method can generate appropriate reward values by an interaction with the environment and a human being. In this study, we assume that the agent has some sensors to interact with the environment and the human being. We experiment on two types of interactions with the environment and two types of interactions with a research participant. Further, we discuss the difference in the actions that the agent takes for each interaction. In this section, we first explain the task and the environment. Next we explain the settings of the interactions, and the details of this experiment. Finally we present the results of the experiment.

3.2. Path Planning for an Open Grid Space

We targeted the path planning problem and gave the agent a task to reach a goal point from a start point on an open grid field. We set two types of grounds in the environment. One was an even ground, i.e., a level ground in good condition. The other was an uneven ground with a rough terrain; the agent jolted over the uneven ground. We show the field used for the experiment in **Fig. 2**.

The agent starts from point "S," and the task is to reach the point "G." The agent can move in four directions, namely up, down, left and right, and must take 5 s for one action.

3.3. Interaction with the Environment

To achieve the task, the agent goes through the even/uneven ground and recognizes the conditions of the ground. In this study, we prepared two types of grounds and performed a computer simulation. Here, we assumed that the agent had a virtual one-degree accelerometer and could sense the vertical vibration of its body. To simulate this, we defined the sensor input as Eq. (7).

$$input_{acc}(t) = const_{acc} + \sigma_* \cdot n(t) \quad \ldots\ldots \quad (7)$$

Here, $const_{acc}$ has a constant value and $n(t)$ denotes a random number between 0 and 1. We set σ_* for each type of ground: σ_{even} for the even ground and σ_{uneven} for the uneven ground. In the experiment, we set $const_{acc} = 50$, $\sigma_{even} = 1$ and $\sigma_{uneven} = 60$. The agent took 5 s per action, and the sampling time of the sensor input was 10 ms.

3.4. Interaction with a Human Being

Here, we explain the interaction between the agent and a participant. We assumed that the agent had some touch sensors and human beings could interact with it by touching its body (sensor). Therefore, we tried to have the agent interact with a participant by using a touch device.

In this experiment, a participant could watch the map and the position of the agent and the movement of the

Table 1. Specifications of the input device.

Model name	Nexus7 (2012)
Manufacturer	ASUS
OS	Android 4.e
Display size	7 inch
Display aspect ratio	16:10
Display resolution	1280×800 px

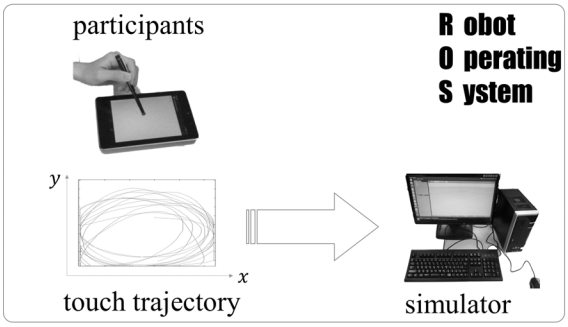


Fig. 3. Interaction by using a touch device.

agent. Further, the participant could evaluate the agent's action by writing a trajectory with a stylus pen on a touch device. In this study, we used Nexus7 (2012) as a touch device and used ROS [19] as the framework of distributed processing. We present the device specifications in **Table 1** and illustrate the outline of the interaction between the agent and a participant in **Fig. 3**.

In this study, we focus on the movement of a stylus pen as the result of an interaction; therefore, we use the series of velocities of a stylus pen position as the input data for interaction (Eq. (8)) and use the average value to calculate the reward as shown in Eq. (9).

$$input_1(t) = \frac{||\boldsymbol{p(t)} - \boldsymbol{p(t-1)}||}{T_{sampling}} \quad \ldots\ldots \quad (8)$$

$$avg_1 = \frac{T_{sampling}}{T_{window}} \sum_{t=1}^{T_{window}} input_1(t) \quad \ldots\ldots \quad (9)$$

Here, $\boldsymbol{p(t)}$ denotes the position of a stylus pen at t. $T_{sampling}$ represents the sampling time required to get to this position and is a constant value. In each interaction, the agent acquires the series of velocities of the stylus pen and uses them as the input data. Here, we define the maximum number of input values. If the number of input values exceeds this value, the proposed method neglects the old data and use the latest input data. We define $\max_{num_{input}}$ as the number of input values. T_{window} is the period to gather input values and is calculated as $\max_{num_{input}} / T_{sampling}$. We present the parameters related to the interaction between the agent and a participant in **Table 2**.

Table 2. Parameters for the interaction with participants.

$T_{sampling}$	10 ms
T_{window}	1 sec
$\max_{num_{input}}$	500
$\max_{input_1}$	100

3.5. Instructions for a Participant

For this experiment, we recruited a 24-years-old man as the participant. This participant knows the way to teach the agent his evaluation by drawing a trajectory on the touch device.

In this study, we instructed the participant to interact with the agent in two ways. One was an elaborate interaction in which the participant interacted with the agent for every action, and the other was a crude interaction in which the participant interacted with the agent only when the agent reached goal point. The instructions for both interactions were as follows:

Elaborate instruction:

- The aim is to lead the agent to the goal point.
- You can praise or scold the agent to achieve the aim.
- You can praise or scold the agent by drawing a trajectory on the touch device.
- You can praise the agent by drawing the trajectory gently.
- You can scold the agent by drawing the trajectory violently.
- You can praise or scold the agent for each action that it performs.

Crude instruction:

- The aim is to lead the agent to the goal point.
- You can praise or scold the agent to achieve the aim.
- You can praise or scold the agent by drawing a trajectory on the touch device.
- You can praise the agent by drawing the trajectory gently.
- You can scold the agent by drawing the trajectory violently.
- You can praise or scold the agent for each action that the agent performs between the first trial and the fifth trial.
- After the sixth trial, you can praise or scold the agent only once it reaches the goal point. This means that after the sixth trial, you cannot interact with the agent before it reaches the goal point.

Here, we did not instruct the participant to avoid the uneven ground and to reach the goal point by using the shortest possible route.

Table 3. Parameters for the agent.

Initial Q-value	0.0
Number of trials	30
α	0.3
γ	0.3
ε	0.1

Table 4. Parameters for the proposed method.

$N_i, i = acc, touch$	0.00001
$M_i, i = acc, touch$	0.333
$L_i, i = acc, touch$	0.15
Initial value of δ_1	0.0
β_1	0.001
w_{acc}	1.0
w_{touch}	3.0

3.6. Settings for the Learning of the Agent

Here, we discuss the settings for the learning of the agent. In this study, the agent learned by the Q-learning method discussed in Eq. (10) and selected actions by using the ε-greedy method [1].

$$Q(s,a) \leftarrow Q(s,a) + \alpha\{r + \gamma max_{a'} Q(s',a') - Q(s,a)\} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (10)$$

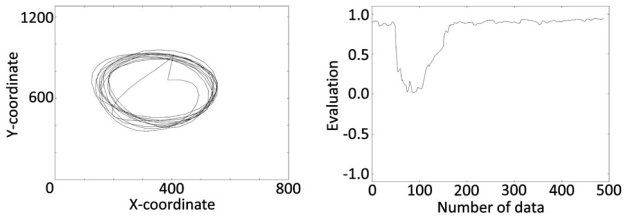
Further, we list the parameters for the agent in **Table 3**.

3.7. Experimental Procedure

We defined one step as the cycle of the selection of an action by the agent, the interaction by the participant, and learning of an appropriate action by the agent. Further, we defined one trial as a series of steps from the state that the agent was on at the start point to the one at which the agent reached the goal. The details of one trial are as follows:

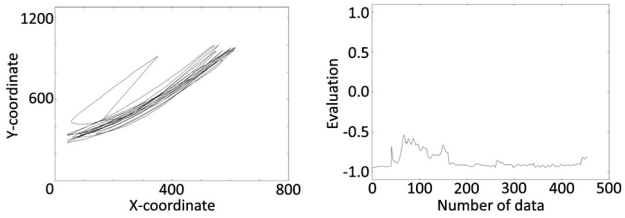
1. The agent is set on the start point.
2. The agent selects an action by using the ε-greedy method.
3. The participant interacts with the agent by drawing a trajectory on the touch device.
4. The agent generates a reward value on the basis of the interaction of the participant.
5. The agent learns with the reward value by using Q-learning.
6. If the agent does not reach the goal point, return to 2.

We list the parameters of the proposed method used in this experiment in **Table 4**.



(a) Input from the participant (b) Evaluation of praise input

Fig. 4. Result for the praise input.



(a) Input from the participant (b) Evaluation of the scolding input

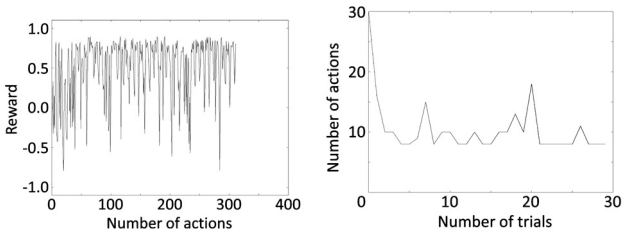
Fig. 5. Result for the scolding input.

3.8. Experimental Results

We show the typical results of the interaction with the participant in **Figs. 4** and **5**. **Fig. 4(a)** shows the trajectory that the participant drew as praise for an appropriate action. The participant drew this trajectory by stylus and the system got the points at each sampling timing. The proposed method generated the evaluation values using these points as input data. **Fig. 4(b)** shows the transition of the evaluation for the praise input. In this case, the evaluation values keep high for the most part. The evaluation values around 100th data become low because the trajectory went wild of the circle which is shown in **Fig. 4(a)**. In contrast, **Fig. 5(a)** shows the trajectory as the participant's scolding for an inappropriate action. In this case, the proposed method generated the evaluation as shown in **Fig. 5(b)** by using the points as input data.

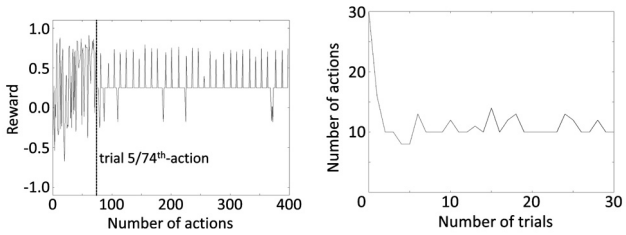
Next we show the results of the learning with the elaborate interaction. **Fig. 6(a)** shows the transition of the reward that the proposed system generated by interacting with the environment and the participant. In this case, the participant interacted with the agent for every action; therefore, the reward changed violently. **Fig. 6(b)** shows the transition of the number of actions that the agent performed and their convergence into eight actions. The route that the agent took at trial 30 is shown in **Fig. 7**. This result shows that the agent took the shortest route from the start point to the goal point.

Finally, we show the results of the learning with the crude interaction. **Fig. 8(a)** shows the transition of the reward. The participant interacted with the agent for every action until trial 5 at which the number of actions was 74. Therefore, the trend of the graph until trial 5 is similar to the one generated by the elaborate interaction. After trial 6, the participant interacted with the agent only when



(a) Transition of the rewards (b) Transition of the number of actions

Fig. 6. Results for the elaborate interaction.



(a) Transition of the rewards (b) Transition of the number of actions

Fig. 8. Results of the crude interaction.

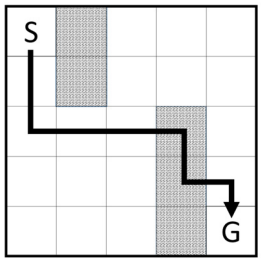


Fig. 7. Route selected as the result of the elaborate interaction.

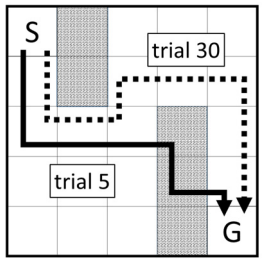


Fig. 9. Route selected as the result of the crude interaction.

the agent reached the goal point. Hence, the interaction with the environment mainly affected the transition of the reward and the change of the reward became mild. We show the transition of the actions and the route that the agent took in **Figs. 8(b)** and **9**, respectively. These show that the agent changed the route after a change in the interaction. The solid line in **Fig. 9** shows the route that the agent took at trial 5. This route reflects the participant's consideration and is the shortest path from the start point to the goal point. On the other hand, the dotted line in **Fig. 9** shows the route that the agent took at trial 30. After trial 6, the effect on the interaction with the participant decreased and the effect on the interaction with the environment increased. As a result, the agent changed the route to avoid the uneven ground. This shows that the participant can make the agent learn the desired movement by the elaborate interaction, and the agent learns a movement adapted to the environment for the crude interaction.

4. Conclusion

In this study, we focused on the generation of a reward with a universal evaluation of sensor inputs. To realize this, we considered two indexes, namely the input strength and the ease of input prediction, and developed a concrete algorithm. The proposed method has no special gateway to obtain the information of rewards for a given task that a human being wants to achieve; therefore, the interaction between the agent embedded in this method and the human being becomes important. The agent must learn the appropriate actions demanded by the human being through sensor inputs, and the human being must learn the way to teach his/her demand through sensor inputs. We attempted to experimentally solve the path-planning problem and show that the agent could learn the appropriate actions by interacting with the environment and the participant. The results of the interaction with the participant shows that this participant could teach the agent his/her demand and that the proposed method generated an appropriate reward. Further, the results of learning by an elaborate interaction shows that the agent learned the task and found the route to the goal. In addition to this, the results of learning by a crude interaction shows that the agent found the route by avoiding uneven ground. In ordinary way to adopt RL for this problem, a reward function is set at first and the agent find a route based on this function. So a human being must re-design the function if he/she want to change the route the agent took. In this study, we show that the proposed method generated the reward and the agent learned the appropriate actions for the given task and environment without the re-design of some mechanisms for particular task.

In the future, we try to equip an agent plural types of sensor. The ability to recognize an outside depends on a type of sensor, number of sensor and how to mount it on an agent. We confirm that an agent embedded the proposed method can adapt their behavior with plural sensors. And we intend to have the participants instruct an agent of a definite route as their task. In this study, we confirmed that the participant could teach an agent the demand for each action; therefore, in the future, we will confirm that the participants can teach the demand through trials. Here, we showed the participants could teach their demand interactively. In the early stage, it is needed to monitor each behavior carefully. But to keep watching all the time makes the burden too heavy for a human being. We think that a method to predict a reward calculated by

inputs from a human being can leave a human being to monitor an agent's behavior. By trying to integrate a prediction method into our system, the proposed method will be user-friendly system. Further, we will conduct the experiment for multiple participants and send out questionnaires. By using the data from the participants, we plan to keep improving the universal evaluation and enable the method to feel and interact with the outside in the same way as though it were a living thing.

References:

[1] R. S. Sutton and A. G. Barto, "Reinforcement Learning," The MIT Press, 1998.

[2] M. Riedmiller, T. Gabel, R. Hafner, and S. Lange, "Reinforcement learning for robot soccer," Autonomous Robots, Vol.27, No.1 pp. 57-73, 2009.

[3] R. Yamashina, M. Kuroda, and T. Yabuta, "Caterpillar Robot Locomotion Based on Q-Learning using Objective/Subjective Reward," Proc. of IEEE/SICE Int. Symposium on System Integration (SII 2011), pp. 1311-1316, 2011.

[4] M. Hara, N. Kawabe, J. Huang, and T. Yabuta, "Acquisition of a Gymnast-Like Robotic Giant-Swing Motion by Q-Learning and Improvement of the Repeatability," J. of Robotics and Mechatronics, Vol.23, No.1, pp.126-136, 2011.

[5] K. Inoue, T. Arai, and J. Ota, "Acceleration of Reinforcement Learning by a Mobile Robot Using Generalized Inhibition Rules," J. of Robotics and Mechatronics, Vol.22, No.1, pp. 122-133, 2010. Vol.22, No.1, 2010.

[6] S. Aoyagi and K. Hiraoka, "Path Searching of Robot Manipulator Using Reinforcement Learning – Reduction of Searched Configuration Space Using SOM and Multistage Learning –," J. of Robotics and Mechatronics, Vol.22, No.4, pp. 532-541, 2010.

[7] K. Yamada, "Expression of Continuous State and Action Spaces for Q-Learning Using Neural Networks and CMAC," J. of Robotics and Mechatronics, Vol.24, No.2, pp. 330-339, 2012.

[8] P. Weng, R. Busa-Fekete, and E. Hüllermeier, "Interactive Q-Learning with Ordinal Rewards and Unreliable Tutor," ECML/PKDD Workshop Reinforcement Learning with Generalized Feedback, 2013.

[9] S. Whiteson, "Evolutionary Computation for Reinforcement Learning" in M. Wiering and M. van Otterlo (Eds.), Reinforcement Learning: State of the Art, pp. 325-358, Springer, 2012.

[10] K. Kurashige and Y. Onoue, "The robot learning by using "sense of pain"," Proc. of Int. Symposium on Humanized Systems 2007, pp. 1-4, 2007.

[11] J. A. Starzyk, "Motivation in Embodied Intelligence," in Frontiers in Robotics, Automation and Control, I-Tech Education and Publishing, pp. 83-110, Oct. 2008.

[12] J. A. Starzyk, "Motivated Learning for Computational Intelligence," in B. Igelnik (Ed.), Computational Modeling and Simulation of Intellect: Current State and Future Perspectives, IGI Publishing, ch.11, pp. 265-292, 2011.

[13] S. Sugimoto, "The Effect of Prolonged Lack of Sensory Stimulation upon Human Behavior," Philosophy, Vol.50, pp. 361-374, 1967.

[14] S. Sugimoto, "Human Mental Processes under Sensory Restriction Environment," The Japanese Society of Social Psychology, Vol.1, No.2, pp. 27-34, 1986.

[15] N. Matsunaga, A. T. Zengin, H. Okajima, and S. Kawaji, "Emulation of Fast and Slow Pain Using Multi-Layered Sensor Modeled the Layered Structure of Human Skin," J. of Robotics and Mechatronics, Vol.23, No.1, pp. 173-179, 2011.

[16] J. Zhen, H. Aoki, E. Sato-Shimokawara, and T. Yamaguchi, "Obtaining Objects Information from a Human Robot Interaction using Gesture and Voice Recognition," IWACIII 2011 Proc., 101_GS1_1, 2011.

[17] S. Hashimoto, A. Ishida, M. Inami, and T. Igarashi, "TouchMe: An Augmented Reality Interface for Remote Robot Control," J. of Robotics and Mechatronics, Vol.25, No.3, pp. 529-537, 2013.

[18] N. Kubota and Y. Urushizaki, "Communication Interface for Human-Robot Partnership," J. of Robotics and Mechatronics, Vol.16, No.5, pp. 526-534, 2004.

[19] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "ROS: An open-source Robot Operating System," ICRA Workshop on Open Source Software, 2009.

Name:
Kentarou Kurashige

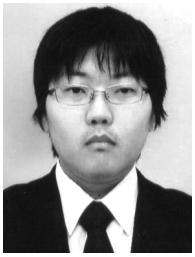
Affiliation:
Muroran Institute of Technology

Address:
27-1 Mizumoto-cho, Muroran-shi, Hokkaido 050-8585, Japan

Brief Biographical History:
2002 Received Ph.D. degree from Nagoya University
2002-2005 Research Associate, Fukuoka University
2005- Research Associate, Muroran Institute of Technology

Main Works:
• Y. Kishima, K. Kurashige, and T. Kimura, "Decision Making in Reinforcement Learning Using a Modified Learning Space Based on the Importance of Sensors," J. of Sensors, Vol.2013, Article ID 141353, 2013. doi:10.1155/2013/141353
• K. Kurashige and Y. Miyazaki, "Use of the Knowledge of Perceptual State Transition in Reinforcement Learning," JSCSE, Vol.3, No.2, pp. 1-12, 2013.
• K. Kurashige, N. Kitayama, and M. Kiyohashi, "Proposal of Method "Motion Space" to Express Movement of Robot," J. of Advanced Computational Intelligence and Intelligent Informatics, Vol.16, No.6, pp. 704-712, 2012.
• K. Kurashige, Y. Onoue, and T. Fukuda, "From Automation To Autonomy, Machine Learning edited by Abdelhamid Mellouk," Abdennacer Chebira, In-Tech, pp. 39-52, Feb. 2009.

Membership in Academic Societies:
• The Institute of Electrical and Electronics Engineers (IEEE) Robotics and Automation Society
• The Institute of Electrical and Electronics Engineers (IEEE) Systems, Man, and Cybernetics Society
• The Robotics Society of Japan (RSJ)
• Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT)
• The Japanese Society for Artificial Intelligence (JSAI)



Name:
Kaoru Nikaido

Affiliation:
Muroran Institute of Technology

Address:
27-1 Mizumoto-cho, Muroran-shi, Hokkaido 050-8585, Japan

Brief Biographical History:
2013 Received Bachelor degree from Muroran Institute of Technology
2013- Master Student, Muroran Institute of Technology

Main Works:
• K. Nikaido and K. Kurashige, "Self-Generation of Reward by Sensor Input in Reinforcement Learning," 2013 Second Int. Conf. on Robot, Vision and Signal Processing (RVSP), pp. 270-273, 2013.

Membership in Academic Societies:
• The Robotics Society of Japan (RSJ)