REAL WORLD ROBOT NAVIGATION BY TWO DIMENSIONAL EVALUATION REINFORCEMENT LEARNING

Hiroyuki Okada
TOKAI UNIVERSITY, School of Science
Kitakaname 1117, Hiratsuka City,Kanagawa 259-1292,Japan

Keywords: Reinforcement learning, Reward, Punishment, Mobile robots.

Abstract: The trade-off of exploration and exploitation is present for a learnig method based on the trial and error such as reinforcement learning. We have proposed a reinforcement learning algorism using reward and punishment as repulsive evaluation(2D-RL). In the algorithm, an appropriate balance between exploration and exploitation can be attained by using interest and utility. In this paper, we applied the 2D-RL to a navigation learning task of mobile robot, and the robot found a better path in real world by 2D-RL than by traditional actor-critic model.

1 INTRODUCTION

Reinforcement learning is attracting attention in the field of machine learning and robotics (Kaelbling, 1996). Reinforcement learning refers to general learning to obtain appropriate action strategies by trial and error without an explicit target system model; instead, learning is accomplished using rewards obtained in the execution environment as the result of self-generated actions (Unemi 1994). This learning method is now being actively studied as a framework for autonomous learning because actions can be learned using only scalar evaluation values and without explicit training.

The purpose of reinforcement learning is to maximize the total rewards depending on the present and future the environment. This kind of learning has two properties. One is optimality (exploration), that is, to ultimately obtain as many rewards as possible. The other is efficiency (exploitation), which is to obtain rewards even in the middle of the learning process. These properties are in a tradeoff relationship (Yamamura 1995). If exploration is overemphasized, convergence into an optimum policy is much longer as the environment becomes more complicated. Furthermore, only small rewards can be obtained in the learning process. Conversely, if exploitation is always emphasized, the learning results decrease to the local minimum and no optimum policies may be available at the end of the learning process.

Most of the reinforcement learning algorithms proposed at comparatively early stages emphasize either exploration or exploitation. For example, Q-Learning (Watkins 1992) guarantees the maximum expected acquisition reward per unit of action at the end of a learning process. This method can be regarded as emphasizing exploration but not considering the efficiency of exploitation in the learning process. Q-Learning determines the efficiency of exploitation during ad hoc learning at each step in action selection (Sutton 1998). It has not yet solved the tradeoff problem. Another method is profit sharing (Grefenstette 1988), which reinforces past actions only when a reward is given. This method is good at exploitation during the learning process but not efficient for whole exploration.

Miyazaki et al. conducted a series of studies (Miyazaki 1997a)(Miyazaki 1995)(Miyazaki 1997b) on the tradeoff between exploration and exploitation in reinforcement learning. Miyazaki proposed a reinforcement learning system (MarcoPolo) with an action determiner, which consisted of an environmental identifier for exploration and a reward getter for exploitation. Miyazaki proved the effectiveness of reinforcement learning based on an arbitrary tradeoff ratio specified by the user. However, at the design or implementation stage, MarcoPolo requires an arbitrary setting of a tradeoff ratio for exploration and exploitation. The user must determine the appropriate ratio on a trial-and-error basis.

Okada H. (2004).
REALWORLD ROBOT NAVIGATION BY TWO DIMENSIONAL EVALUATION REINFORCEMENT LEARNING.
In *Proceedings of the First International Conference on Informatics in Control, Automation and Robotics*, pages 249-255
DOI: 10.5220/0001136602490255

In the conventional framework of reinforcement training, a one-dimensional scalar value is used to represent an evaluation reward. It is the only reinforcement signal for learning and developing an optimum policy. When there are positive and negative rewards, however, using only a scalar-value reward may result in a tradeoff between exploration and exploitation. Uchibe et al. (Uchibe 1999) proposed a method to make a reward function multidimensional to enable simultaneous learning of several functions, and they verified that coordinated actions can be realized in a multi-agent environment. Uchibe's method seems effective when there are positive and negative rewards because of its capability to make a reward function multidimensional and to handle a reward as vector data. However, multidimensional conversion increases the number of parameters in the reward function and attenuation matrix, thereby making it difficult to determine optimum values.

Because no clear principle has been defined for reflecting a multidimensional evaluation on a one-dimensional action, it is difficult to convert the results and transfer them into another system.

Knowledge obtained from rats and monkeys about operand-conditioned subjects (Miller 1959)(Ison 1967) and from humans having damaged brains (Milner 1963) indicates that distinguishing between the evaluations of successes and failures has a tremendous effect on action learning (Yamakawa 1992)(Okada 1997)(Okada 1998). With this in mind, the authors propose reinforcement learning by a two-dimensional evaluation. This evaluation involves an evaluation function based on the dimensions of reward and punishment. An evaluation immediately after an action is called a reward evaluation if its purpose is to obtain a favorable result after repeated attempts to learn an action, or punishment evaluation if its purpose is to suppress an action.

Reinforcement learning using the two dimensions of reward and punishment separates the conventional one-dimensional reinforcement signal into reward and punishment. The proposed method uses the difference between reward evaluation and punishment evaluation (*utility*) as a factor in determining the action and their sum (*interest*) as a parameter in determining the ratio of exploration to exploitation. *Utility* and *interest* are rough ways to define the principle of reflecting multidimensional evaluation on a one-dimensional action.

Chapter 2 describes the formulation of the proposed reinforcement learning method based on the two dimensions of reward and punishment. Chapters 3 prove the usefulness of the proposed system by describing the learning process of an autonomous mobile robot. Finally, Chapter 4 summarizes the study.

2 REINFORCEMENT LEARNING BASED ON TWO DIMENSIONS OF REWARD AND PUNISHMENT

2.1 Basic Idea

Two-dimensional reinforcement learning basically consists of two aspects. One is to distinguish between reward evaluation and punishment evaluation forecasts. The other is to determine an action according to the combined index of positive and negative reward forecasts.

2.1.1 Search by *interest* and resource allocation

The conventional reinforcement learning method uses only the difference (*utility*) between reward and punishment reinforcement signals in an evaluation to determine an action. In comparison, the proposed method determines the sum (*interest*) of reward and punishment evaluation signals and considers it as a kind of criticality. Criticality can be considered to be curiosity or motivation in living things, and it used to determine which processing should be noted. In other words, not only in reinforcement learning but in any other kind of trial-and-error learning it can be used to determine the ratio of exploration search to exploitation action.

2.1.2 Distinction of the time discount ratio of forecast reward

In reinforcement learning, a forecast reward is discounted more if it's more likely to be received in the future. This discount ratio is called the time discount ratio (γ) of the forecast reward. The value of γ ranges from 0 to 1.0. If the value is 0, only the current reinforcement signal is noted and its future reinforcement is disregarded. If the value is 1.0, the evaluation of action is considered until the distant future.

In many practical problems, a reward reinforcement signal is related to the method used to move toward a goal and a forecast reward signal is used for learning a series of actions to reach the goal. To consider the effect of a goal that is far away, the γ setting must therefore be large.

Meanwhile, if a punishment reinforcement signal for avoiding a risk has an effect too far away from the risk, an avoidance action may be generated in many input states. In turn, the search range of the operating subject is reduced, thereby lowering the performance of the subject. Therefore, to generate a punishment reinforcement signal for initiating an action

to avoid an obstacle only when the obstacle is immediately ahead, the value of γ must be small in the signal.

For example, how can a robot moving toward a goal avoid at an appropriate distance an object in its path ? If the environment has an avoidance circuit designed to function immediately before a punishment state (collision against an obstacle) and the circuit is surely operable, the robot can only learn the action of avoiding the punishment target only if the γ setting is 0. However, in an uncertain environment or if a dead-end punishment is in the passageway, the value of γ must be raised to maintain a greater distance from the punishment target.

Considering these factors, a different γ value is set for each evaluation of reward and punishment under the following policies:

- Reward evaluation
  The γ setting is large and is effective at a long range. This evaluation is mainly used to reach a goal.
- Punishment evaluation
  The γ setting is small and is effective at a short range. This evaluation is mainly used to avoid risk.

2.2 Actor-Critic Architecture Based on Two-dimensional Evaluation

2.2.1 Actor-Critic architecture

Figure 1 shows the actor-critic architecture (Barto 1983) based on the proposed two-dimensional evaluation. Critic consists of a Reward section for reward evaluation and a Punishment section for punishment evaluation. Each section receives a state (s), a reward evaluation (r_R), and a punishment evaluation (r_P) according to the environment, and each section learns the forecast values. Both r_R and r_P are positive values and *interest* (δ^+) and *utility* (δ^-) are defined according to the TD differences (δ_R, δ_P) related to the forecasts of reward and punishment evaluations, which are shown below:

$$\delta^- = \delta_R - \delta_P \quad : \textit{Utility} \tag{1}$$
$$\delta^+ = |\delta_R| + |\delta_P| \quad : \textit{Interest} \tag{2}$$

Actor learns an action strategy using δ^- (*utility*) as a de facto reinforcement signal and δ^+ (*interest*) to determine the ratio of exploitation action to environmental search action.

2.2.2 State evaluation (Critic)

The reward evaluation, $V_R(s(t))$, and punishment evaluation, $V_R(s(t))$, to state $s(t)$ at time t are defined as follows:

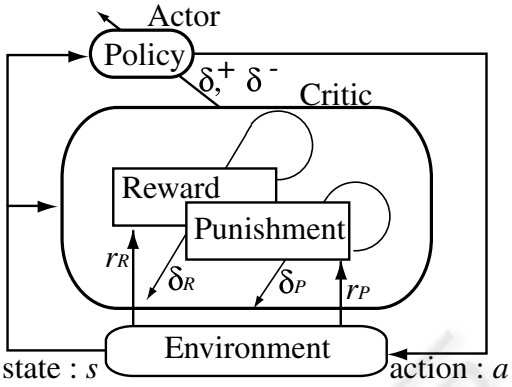

Figure 1: Outline of actor-critic architecture based on two-dimensional evaluation.

$$V_R(s(t)) = r_R(t) + \gamma_R r_R(t+1) + \gamma_R^2 r_R(t+2) + \ldots = \sum_{i=t}^{\infty} \gamma_R^{i-t} r_R(i) \tag{3}$$

$$V_P(s(t)) = r_P(t) + \gamma_P r_P(t+1) + \gamma_P^2 r_P(t+2) + \ldots = \sum_{i=t}^{\infty} \gamma_P^{i-t} r_P(i) \tag{4}$$

where $r_R(t)$ and $r_P(t)$ are the reward and punishment evaluation values (positive), respectively, at time t. $r_R(t)$ represents the time discount ratio of the reward evaluation forecast and $r_P(t)$ represents the time discount ratio of the punishment evaluation forecast.

Based on equations (3) and (4), the following relationship can be established between the evaluation forecast value $\hat{V}(s(t))$ at the current time and $\hat{V}(s(t+1))$ at the subsequent time:

$$\hat{V_R}(s(t)) = r_R(t) + \gamma_R \hat{V_R}(s(t+1)) \tag{5}$$
$$\hat{V_P}(s(t)) = r_P(t) + \gamma_P \hat{V_P}(s(t+1)) \tag{6}$$

By learning to approximate forecast errors ($\delta_R(t)$, $\delta_P(t)$) to 0, status evaluations can be accurately forecast.

$$\delta_R(t) = r_R(t) + \gamma_R \hat{V_R}(s(t+1)) - \hat{V_R}(s(t)) \tag{7}$$
$$\delta_P(t) = r_P(t) + \gamma_P \hat{V_P}(s(t+1)) - \hat{V_P}(s(t)) \tag{8}$$

where $\delta_R(t)$ represents forecast errors related to reward evaluations and $\delta_P(t)$ represents those related to punishment evaluations.

2.2.3 Determination of action (Actor)

Actor is used to develop an action strategy that maximizes the reward forecast by Critic. The proposed

method determines action strategy $\pi(s, a)$ for taking action $a(t)$ in state $s(t)$ at time t as follows:

$$\begin{aligned} \pi(s(t), a(t)) &= \Pr\{a(t) = a | s(t) = s\} \\ &= \frac{\exp(\frac{p(s(t),a(t))}{\delta^+(t)})}{\sum_b \exp(\frac{p(s(t),b(t))}{\delta^+(t)})} \end{aligned} \quad (9)$$

where $p(s(t), a(t))$ indicates whether it is preferable to take action $a(t)$ in state $s(t)$ at time t. *Interest*$(\delta^+(t))$ realizes a search function for large errors in TD learning. As *Interest* becomes greater, actions become more random with a greater priority placed on the search. If *Interest* is small, a slight difference in $p(s(t), a(t))$ has a great effect on action selection. This difference is corrected using *Utility*$(\delta^-(t))$ as expressed below, where positive constant β represents the learning rate:

$$p(s(t), a(t)) \leftarrow p(s(t), a(t)) + \beta\delta^-(t) \quad (10)$$

At the initial stage of learning, both δ_R and δ_P become 0 and consequently, δ^+ may also become 0. If this occurs, actions are taken at random.

3 SEARCHING FOR A GOAL IN PHYSICAL ENVIRONMENT INCLUDING MANY POSSIBLE PATHS

To confirm the effectiveness of two-dimensional evaluation reinforcement learning in a physical environment, an actual mobile robot searched for a path in an office. Using a computer with a mobile robot sensor database, the authors first confirmed that an optimum path can be found if resources are appropriately allocated between search and execution processing. Then, the learning results were downloaded into the actual mobile robot for the experiment.

3.1 Mobile Robot Experiment in Physical Environment

During a search for a path by a mobile robot in a physical environment, controlling the robot using learning results from a mobile robot simulator is difficult because the simulator cannot satisfactorily express the complexities of a real-world environment. The moving speed and maintenance costs also make it difficult for a mobile robot to learn by repetitive trial and error. To eliminate the discrepancy between an actual mobile robot and a simulator, the authors developed a network-distributed mobile robot experimentation system (MEMORABLE: Multilevel Environment for Mobile Robotics Capability Experiments) including a database of sensor data collected by mobile robots in a physical environment (Okada 1999). By using data actually measured by sensors to train a mobile robot, MEMORABLE realized a robust method of learning capable of handling the complexities of the real world, and MEMORABLE reduced the learning time to less than that required in the conventional method on a mobile robot simulator. Furthermore, the system enables objective evaluations for determining whether the proposed learning method is also effective in a physical environment.

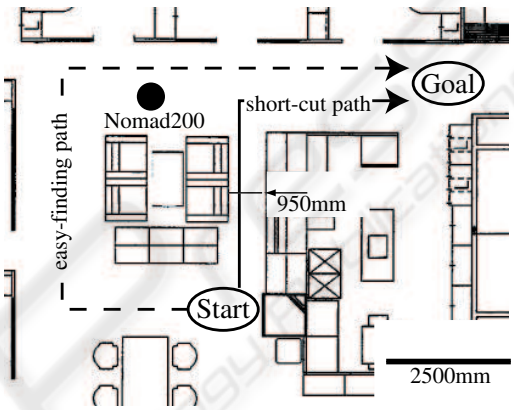

Figure 2: Office map explored by Nomad200. There are two ways to the goal, easy-finding path(dotted line) and short-cut path(solid line).

3.2 Experimental Settings

The mobile robot (Nomad 200) used for the experiment has a cylindrical shape of 60 cm in diameter and 80 cm in height. Sixteen sonar and infrared sensors are arranged on the circumference of the body to measure obstacle distances, but the distance sensors have an effective range for measuring distances. The sonar sensors can measure distances from 40 to 600 cm, and the infrared sensors can measure distances within 40 cm. In the current task, Nomad 200 moves 10 cm to the front, back, right, or left 10 per step, using a gyro to keep its front side facing north (upward in Figure 2).

The purpose of the robot is to search for a path from the starting position to the goal, without colliding with any obstacles, in the office depicted in Figure 6. The evaluations (r_R, r_P) expressed below are given to the robot. The conventional method of reinforcement learning used for comparison was based on the Actor-Critic architecture that does not distinguish between reward and punishment. For this learning process, the evaluations for one-step movement and collision were set to negative values (-0.01 and -1.0,

respectively).

$$r_R = 5.0 \quad \text{for reaching the goal} \quad (11)$$

$$r_P = \begin{cases} 1.0 & \text{for collision} \\ 0.01 & \text{for moving action} \end{cases} \quad (12)$$

The robot begins at the starting position and continues moving until it collides against an obstacle. If a collision occurs, the robot has to begin again from the starting position. The time discount rates for reward and punishment evaluation forecasts in the experiment were set to $r_R = 0.9$ and $r_P = 0.1$, respectively. For the conventional method, nnn the time discount rate for reward was set to 0.90.

3.3 Search for Shortest Path by Two-dimensional Evaluation Reinforcement Learning

There are two kinds of paths from the starting point to the goal (Figure 2). One is a path that is easy to find and secure way through wide passageway, but it is a roundabout route. The other kind of path is a shortcut, which is a short route with a high risk of collision through a narrow passageway. In the experiment, searching for a shorter path without collisions results in greater rewards because negative rewards are given for single-step movements and collisions against obstacles.

As Figure 2 shows, a narrow passageway for the 60-cm cylindrical body of Nomad 200 is about 95 cm wide. Moving through this passageway raises the probability of a collision against an obstacle. Therefore, although a shortcut to the goal is shorter, taking this path at the initial stage of learning often results in negative rewards because of collisions against obstacles. In the easy-to-find path, the probability of reaching the goal is high, even at the initial stage of learning when the robot action is unstable. This is because the path to the goal is through a wide passageway. However, the distance to the goal is long, and the total rewards are small in the end.

In this experiment, a search results in a selection conflict between a shortcut found only after repeated collisions and an easy-to-find path found with a priority on avoiding any obstacles immediately ahead. Using the conventional method may find a path through the wide passageway, where it is comparatively easy to obtain rewards, and it may fail to find the shortest path to the goal. In contrast, using the two-dimensional evaluation reinforcement learning method can balance the acquisition of new information with some risk and the action of exploitation based on past experience. Use of this proposed method is expected to solve the above conflict.

3.4 MEMORABLE Experimental Results

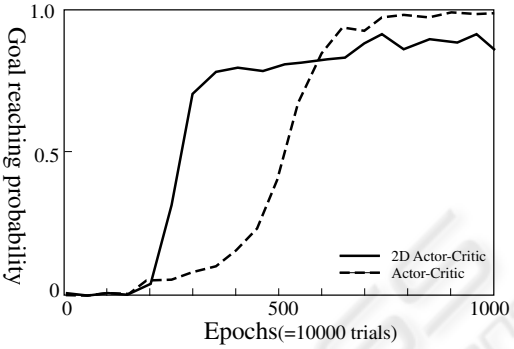

Figure 3: Goal reaching probability of proposed method(solid line) and traditional actor-critic method(dotted line).

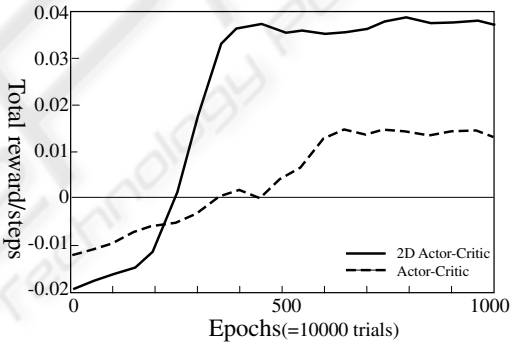

Figure 4: Reward per step of proposed method(solid line) and traditional actor-critic method(dotted line).

3.4.1 Probability of reaching goal and exploitation

One phase of the experiment consists of learning and test trials. The robot begins at the starting position and continues moving until it collides against an obstacle. If a collision occurs, the robot begins again at the starting position. This series of actions is called a trial. A learning trial is repeated 1000 times as an epoch. Then, a test trial is repeated 100 times using different random numbers to calculate the probability of reaching the goal and the reward obtained per step.

Figure 3 shows probability of reaching the goal, and Figure 4 shows the average reward obtained per step. The solid line indicates the results of using the proposed method, and the dotted line indicates the results of using the conventional method.

In using the proposed method, the probability of reaching the goal quickly increased after about 200

epochs and remained at between 80 and 90% after 400 epochs. In using the conventional method, the probability of reaching the goal gradually increased after about 300 epochs and reached 90% after 700 epochs. The probability of reaching the goal was higher for the conventional method than for the proposed method because easy-to-find paths were generated in many cases.

When exploitation per step was compared for the proposed and conventional methods, the difference was remarkable. Using the conventional method generates a long path to the goal, an easy-to-find path. This reduces the average reward received because of the effect of the negative reward given for every step. Conversely, using the proposed method generates a short path, a shortcut. This increases the average reward per step after 400 epochs when the number of collisions against a wall decreases.

3.4.2 Comparison of search strategies

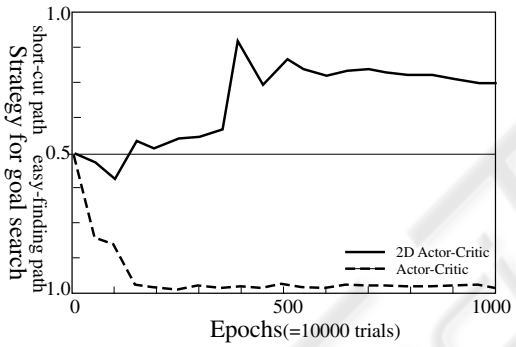

Figure 5: Goal searching strategy of proposed method(solid line) and traditional actor-critic method(dotted line).

Figure 5 compares the ratio of easy-to-find paths to shortcuts as selected by the proposed method with the equivalent ratio for the conventional method. As the figure shows, the conventional method searched for the easy-to-find paths at a rate of 95% or more after about 150 epochs. The instruction to avoid any obstacle immediately ahead is fixed. This means that the results by the conventional method are reflecting risk avoidance actions.

At the initial stage of learning before 400 epochs, the proposed method searched for easy-to-find and shortcuts in a 50:50 ratio. This confirms a good balance between attempts to obtain new information with some risk and actions for exploitation.

When a robot moves in a physical environment, the learning of risk avoidance actions is important. However, the maximum operating time depending on battery power, the maximum path length, and other cost factors must be considered. The optimum action is not simply risk avoidance. This experiment confirmed the effectiveness of the proposed method that automatically determines the tradeoff between risk avoidance and exploitation.

3.5 Trials by Actual Mobile Robot

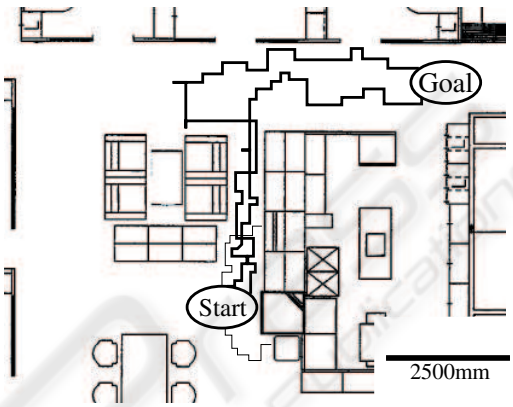

Figure 6: Execution by real mobile robot. Successful cases(thick line) and unsuccessfull cases(thin line).

The learning results were imported into an actual mobile robot to prove that searching for a goal is possible in a physical environment. Subsequently, three trials out of ten ended halfway because of collisions against obstacles, but the remaining seven trials ended in successful shortcut being taken to reach the goal. These successes prove the proposed method can be effective in searching for a goal in a physical environment.

For comparison, the learning results from the mobile robot simulator attached to Nomad were imported into an actual mobile robot, and the same experiment was conducted. Although a path to reach the goal was found on the simulator, the actual mobile robot failed on all of the 10 trials. Figure 10 shows the paths of the mobile robot in two successful trials after training by MEMORABLE (thick line) and two unsuccessful trials after training by the mobile robot simulator (thin line).

The performance of robots in MEMORABLE seemed to indicate learning after repeated attempts to perform tasks such those in the current experiment according to the data collected by an actual robot.

4 CONCLUSION

To solve the problem of tradeoff between exploration and exploitation actions in reinforcement learning,

the authors have proposed two-dimensional evaluation reinforcement learning, which distinguishes betweenbbbb reward and punishment evaluation forecasts.

In the proposed method of reinforcement learning using the two dimensions of reward and punishment, a reinforcement signal dependent on the environment is distinguished into reward evaluation after successful action and punishment evaluation after an unsuccessful action. The proposed method uses the difference between reward evaluation and punishment evaluation (*utility*) as a factor for determining the action and the sum (*interest*) as a parameter for determining the ratio of exploration to exploitation.

This paper has described an experiment with a mobile robot searching for a path and the subsequent conflict between exploration and exploitation actions. The results of the experiment prove that using the proposed method of reinforcement learning using the two dimensions of reward and punishment can generate a better path than using the conventional reinforcement learning method. MEMORABLE enables the proposed method to be effective for actual robots searching a path in a physical environment.

ACKNOWLEDGEMENT

This study was conducted as part of the Real World Computing Program.

REFERENCES

L.P.Kaelbling, K.L.Littman and A.W.Moore(1996). Reinforcement learning :A survey. In *Journal of Artificial Intelligence Research*. vol.4, pp237-285.

T.Unemi(1994). Reinforcement Learning. In *Journal of Japanese Society for Artificial Intelligence*. vol.9, no.6, pp830-836.

M.Yamamura, K.MIyazaki and S.Kobayashi(1995). A Survey on Learning for Agents. In *Journal of Japanese Society for Artificial Intelligence*. vol.10, no.5, pp23-29.

C.J.Watkins and P.Dayan(1992). Learning. In *Machine Learning*. vol.8, pp.55-68.

R.S.Sutton and A.G.Barto(1998). *Reinforcement Learning*. MIT Press.

J.J.Grefenstette(1998) Credit Assignment in Rule Discovery Systems Based on Genetic Algorithms. In *Machine Learning*. vol.3, pp225-245.

K.Miyazaki, M.Yamamura and S.Kobayashi(1997) A Theory of Profit Sharing in Reinforcement Learning. In *Journal of Japanese Society for Artificial Intelligence*. vol.9, no.4, pp104-111.

K.Miyazaki, M.Yamamura and S.Kobayashi(1998) k-Certainty Exploration Method: An Action Selector on Reinforcement Learning to Identify the Environment. In *Journal of Japanese Society for Artificial Intelligence*. vol.10, no.3, pp124-133.

K.Miyazaki, M.Yamamura and S.Kobayashi(1997) MarcoPolo: A Reinforcement Learning System Considering Tradeoff Expolitation and Exploration under Markovian Environment In *Journal of Japanese Society for Artificial Intelligence*. vol.12, no.1, pp78-89.

E.Uchibe and M.Asada(1999) Reinforcement Learning based on Multiple Reward Function for Cooperative Behavior Acquisition in a Multiagent Environment In *RSJ'99*. vol3, pp.983-984.

N.E.Miller(1959). Liberalization of basic S-R concepts:extensions to conflict behavior, motivation and social learning. In *Koch.S(Ed), Psychology:A Study of a Science*. New York:McFraw-Hill.

J.R.Ison and A.J.Rosen(1967). The effect of amobarbital sodium on differential instrumental conditioning and subsequent extinction. In *Psyhopharmacologia*. vol.10, pp417-425.

B.Milner(1963). Effects of different brain lesions on card sorting. In *Archives of Neurology*. vol.9, pp10-100.

H.Yamakawa(1992). Intelligent System Based on Reinfoecement Learning. In *Ph.D thesis Tokyo University*.

H.Okada and H.Yamakawa(1997). Neuralnetowrk model for attention and reinforcement learning. In *SIG-CII-9710*. pp4-14.

H.Okada, H.Yamakawa and T.Omori(1998). Neural Network model for the preservation behavior of frontal lobe injured patients. In *Proc. of ICONIP'98*. pp1465-1469.

A.G.Barto, R.S.Suttond and C.W.Anderson(1983). Neuronlike Adaptive Elements That Can Solve Difficut Learning Control Problems. In *IEEE Transaction on Systems, Man and Cybernetics*. vol.13, no.5, pp834-846.

H.Okada, O.Ito, Y.Hagihara, K.Niki and T.Omori(1999) Multilevel Environment for Mobile Robotics Capability Experiments (MEMORABLE). In *Journal of the Robotics Society of Japan*. vol.17, no.6, pp142-150.