The Seventeenth International Symposium on Artificial Life and Robotics 2012 (AROB 17th '12),
B-Con Plaza, Beppu, Oita, Japan, January 19-21, 2012

Reinforcement learning with phased approach for fast learning

Norifumi Hodohara, Yuichi Murakami, Shingo Nakamura and Shuji Hashimoto

Waseda University, Tokyo, 169-8555, Japan
(Tel: 81-3-5286-3233, Fax: 81-3-3202-7523)

nhodohara@shalab.phys.waseda.ac.jp

Abstract: In this paper we consider the reduction of the computational cost of reinforcement learning. When we apply the reinforcement learning to a robot with a large number of DOF, it needs a tremendous amount of time for learning because of the large state space. This problem is called "the curse of dimensionality". To solve the problem, we propose a phased approach on reinforcement learning. In the proposed method, we apply reinforcement learning to a robot with limited DOF at first, then release the restriction and resume the learning from the previous learning result. The computer simulation using arm robots having four and five joints proved the effectiveness of the proposed method. We also conducted an experiment in the case that an obstacle exists around the arm.

Keywords: Arm Robot, Reinforcement Learning, Phased Learning, Q-Learning.

1 INTRODUCTION

In recent years, robots have been expected to work not only in structured environments like factories, but also in different unstructured environments. The reinforcement learning [1], which is able to make robots acquire objective behaviors by themselves, attracts attention as a method to avoid the difficulty of programing of robot behavior in such environments. However, the reinforcement learning has a big problem called "the curse of dimensionality". This means that learning cost gets huge with increased DOF of the robot because the robot's state space exponentially expands.

Previously, some researchers have used the reinforcement learning for a robot with a number of DOF. They have manually given the environmental information to the robot to make the state space as small as possible [2][3]. However, it is workable only under the special conditions, and it is inevitable that learning time must be lengthened when large amount of information to solve a problem is not given in advance.

In this research, we attempt to reduce the learning cost by introducing a phased procedure in applying to a robot with a number of DOF. In this approach, the reinforcement learning is applied to the robot with the limited DOF beforehand, which can reduce the learning cost because the scope of state space gets narrower. Then we release the limitation of the robot's motion and restart the reinforcement learning with the state value of the previous learning result. In this way we expect that the robot learns objective task faster than the standard reinforcement learning starts with the full DOF from the beginning. In this paper, we call the first phase learning "previous learning" and final phase learning "eventual learning". To confirm the feasibility of the proposed method, we applied it to arm robots having four and five joints in the computer simulation. We also experimented in the environment with an obstacle.

2 PROPOSED METHOD

2.1 Reinforcement Learning

In this research, we use the Q-learning, the typical algorithm of reinforcement learning. Here, we briefly introduce the method.

When the agent at step t takes the action a_t based on its current state s_t, the state-action value function $Q(s_t, a_t)$ is updated as follows:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha[r_{t+1} + \gamma \max_a Q(s_{t+1},a) - Q(s_t,a_t)] \quad (1)$$

where r_t is the reward the agent receives at step t, α is a learning rate and γ is a discount rate.

We employ the Radial Basis Function (RBF) network [4] for the approximation of the state-action value function. Thereby, Q-values are treated as a continuous function and neighboring state-action values are related in the value space. We use the Gaussian distribution as the RBF and place them in the value space divided into segments uniformly. Therefore, the approximated state-action value function is given by:

$$Q_a(\boldsymbol{s}) = \sum_j w(\boldsymbol{\mu}_j) \exp\left[-\frac{|\boldsymbol{s} - \boldsymbol{\mu}_j|^2}{\sigma^2} \right] \quad (2)$$

where w, $\boldsymbol{\mu}$ and σ^2 are the weight, the average and the

variance of the RBF network. The weight w is updated as Q_a gets close to right-side value of (1).

2.2 Phased Approach

Here, we assume that an agent takes two-dimensional state $\boldsymbol{s}=(s_1, s_2)$ for explanation, and introduce an outline of the phased approach, the proposed method in this study. At first, the agent learns with a restricted dimension of state $\boldsymbol{s}$ in the previous learning phase. In this case, s_1 is fixed at constant value s_1', and the agent learns only with variable s_2 as shown in **Fig. 1.** (a). Thereby, the learning cost is reduced because it allows the agent to explore narrow area of the entire state space. However, the learning result exists in the only limited region of the space. Next, we spread out the resultant state value into the whole state space by using the Gaussian distribution function as follows:

$$w(s_1, s_2) = c \cdot w(s_1', s_2) \exp\left[-\frac{(s_1 - s_1')^2}{\sigma_s^{\,2}}\right] \tag{3}$$

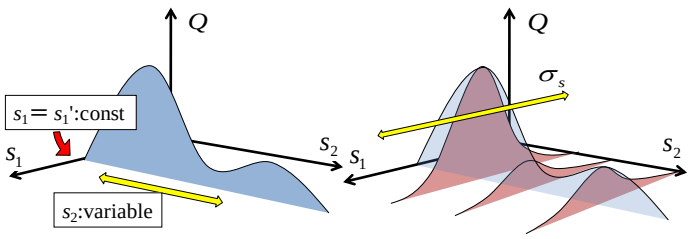
where σ_s^2 is the variance of the Gaussian and c is the constant for adjusting an effect of the previous learning. The weight of the RBF network w spreads in a direction toward s_1-axis as shown in **Fig. 1.** (b). Finally, after releasing the limitation of state $\boldsymbol{s}$, the agent restarts learning from the expanded result and acquires an optimal behavior in the eventual learning phase.

In this way, we expect that the agent learns objective task faster than simply learns the whole value space from the beginning. This approach can be expanded to the more dimensional cases.

3 ARM ROBOT

Here we introduce an arm robot with multiple joints for the experimentation. The robot is implemented with Open Dynamics Engine (ODE), the free library for simulating physical dynamics. **Fig. 2.** (a) shows the arm robot with four joints and environment created by the ODE. The joints rotate around the z-axis, therefore robot moves only in the xy-plane. Each link can overlap other links. The whole length of the robot is constant regardless of the number of joints and each link length is the same.

As shown in **Fig. 2.** (b), a movable range of the each joint is from -150 to 150 degrees, and each joint rotates 30 degrees clockwise or counter-clockwise in one step. The first joint from the root adopts absolute angle and the other joints adopt relative angle. The agent action a is defined to rotate any one of joints in 30 degrees. The state $\boldsymbol{s}$ is defined as a combination of respective joint angles, thus the number of all the states is 11^n, where the n is the number of joints.



(a) Previous learning (b) Value spreading

Fig. 1. The outline of the phased learning.

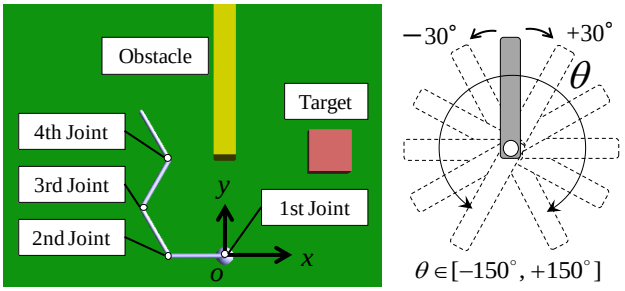
4 EXPERIMENTS

4.1 Experimental setup

We examined both the proposed method and the standard reinforcement learning with the arm robot for comparison. In the experiments of the proposed method, the agent learned with some fixed joints in the previous learning phase as shown in **Fig. 3.** (a). Then, as shown in **Fig. 3.** (b), the agent learned again by using all the joints starting with the value space expanded from the previous learning result as the eventual learning. Meanwhile, the agent began learning with all the joints in the standard reinforcement learning. Both of the methods employed the Q-learning approximated by RBF network.

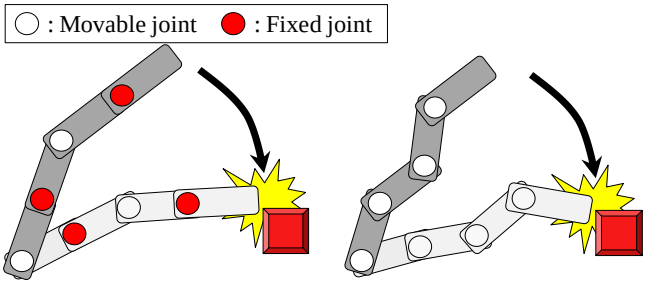
The robot's task was to reach a target with its tip. The agent received a reward only when the robot accomplished its task. On the other hand, the agent paid a penalty when the robot touched an obstacle. There were two experimental environments; with and without an obstacle. The robot having 2 joints cannot reach the target when the obstacle exists. The robot restarted from an initial position when it reached the task or touched the obstacle. If the obstacle existed, the initial position took the other side of the target across the obstacle. If it didn't exist, the initial position was took randomly.

In the Q-learning, the agent updated its state-action value $Q(s, a)$ at each step. One episode was defined as the robot reaching the target once. However, if the agent



(a) The robot having four joints (b) The movable range

Fig. 2. The arm robot simulated by ODE.



(a) Fixing a part of the joints (b) Moving the all joints

Fig. 3. Applying the proposed method to the arm robot.

couldn't accomplish the task until N_{max} steps, the next episode started from another initial position. We employed the softmax function as a policy of the agent as follows:

$$\pi(s,a) = \frac{\exp(Q(s,a)/T_B)}{\sum_b \exp(Q(s,b)/T_B)} \quad (6)$$

where $\pi(s, a)$ is the probability distribution of taking the action a based on the state s and T_B is the Boltzmann temperature.

4.2 Configuration

We conducted three experiments to compare the proposed method with the standard reinforcement learning. In the first experiment, we applied our method to the arm robot with four joints in the environment without an obstacle. The agent learned under the condition with fixing the 2nd and 4th joints as the previous learning. Then, the agent restarted the learning as the eventual learning. In the same way, we conducted another two cases: fixing 1st and 3rd joints, and 2nd and 3rd joints. In the second experiment, we utilized the arm robot with five joints without an obstacle. As the previous learning, the agent learned with the robot fixed the 2nd, 4th and 5th joints. In the third experiment, we applied the proposed method to the arm robot with four joints in the environment with an obstacle. The previous learning was conducted with fixing 2nd and 4th joints without the obstacle. The agent then learned with the obstacle in the eventual learning phase. We compared the learning cost of our method with the cost of the standard reinforcement learning.

In each experiment, the agent learned during 50 episodes as the previous learning. The shapes and locations of the target and the obstacle are shown in **Table 1**. The values of these lengths were based on the condition that the whole length of the arm robot was 1 and the origin position was placed at the root of the arm robot. In the Q-learning, a values of the reward and penalty were 1 and -1, and N_{max} was set to 10,000. The RBFs variance σ^2 equaled 0.01 in the value space divided into 11^n segments. **Table 2.** shows the detail of the other experimental parameters. These parameters were determined empirically by exploratory experiments.

Table 1. The shapes and locations of the target

	Shape	x area	y area
Target	0.20×0.20 Square	[0.40, 0.60]	[0.40, 0.60]
Obstacle	0.10×0.55 Rectangle	[-0.05, 0.05]	[0.45, 1.00]

Table 2. The experimental parameters

	α	γ	T_B	σ_s	c
Previous learning	0.4	0.6	0.005	-	-
4 joints	0.5	0.6	0.005	0.2	0.5
5 joints	0.4	0.8	0.005	0.2	0.1
4 joints with obstacle	0.4	0.8	0.005	0.3	0.2

5 REUSLTS

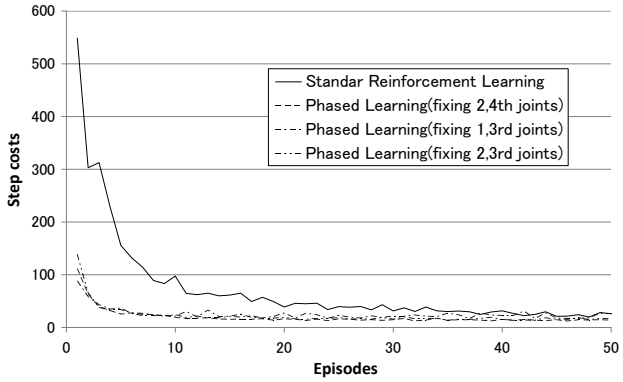
Fig. 4. shows the step cost of reaching the target at each episode number of each experiments. **Fig. 5.** shows the total cost of steps until the 50th episode finished. The every resultant value is an average of the 100 times experiments.

As shown in **Fig. 4.**, the agent with the proposed method learned the tasks with less steps in all the experiments. Thus the feasibility of the proposed method is confirmed regardless of fixed joints or the number of joints. The eventual learning resultant of our method shows the slow convergence speed in **Fig. 4. (c)**. It is considered that the value around the obstacle which built in the previous learning interrupted the robot to reach the goal target.

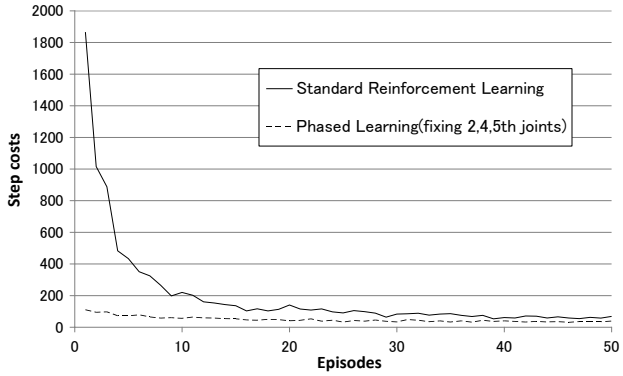
As shown in **Fig. 5.**, the number of total steps with phased learning was less than the standard reinforcement learning in all the experiments. Especially, **Fig. 5. (c)** shows higher reduction ratio of total steps than when no obstacle exists. It represented that our proposed method is more efficient in complex environment.

6 CONCLUSION

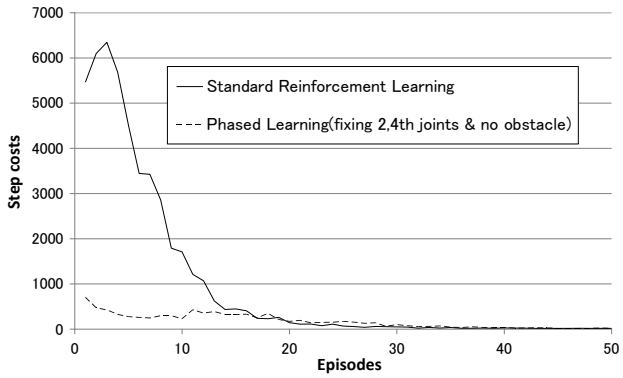
We have developed an algorithm of reinforcement learning, phased learning. In this method, an agent learned with limited robot's DOF in the previous learning phase. The proposed method allows robots having a number of DOF to learn tasks efficiently. The conducted experiments with arm robots having four and five joints proved the feasibility of the proposed method. However, in the experiment with an obstacle, the proposed method converged slower than standard method. We need to research in detail how the resultant of previous learning affects the eventual learning for further improvement.



(a)Result on four joints arm robot

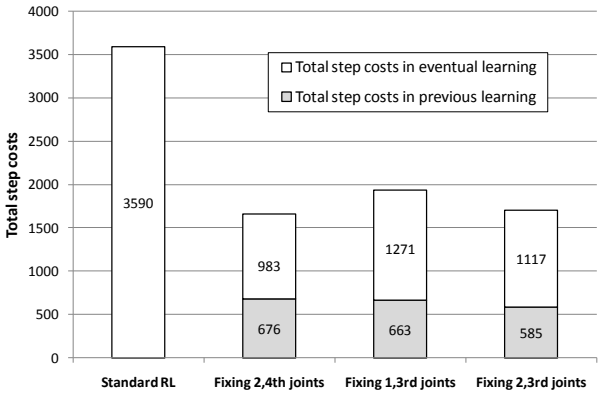


(b)Result on five joints arm robot

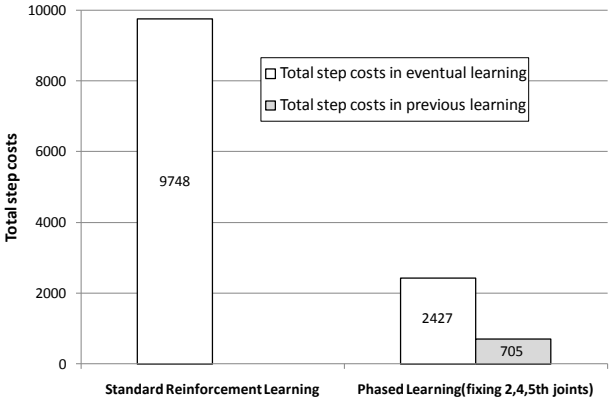


(c)Result on four joints arm robot with an obstacle

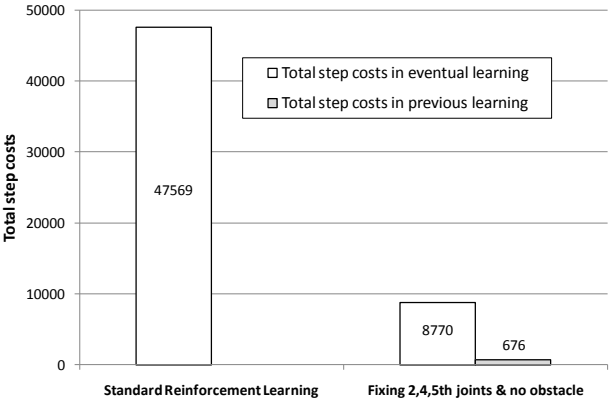
Fig. 4. Step costs as a function of episodes.



(a) Result on four joints arm robot



(b) Result on five joints arm robot



(c) Result on four joints arm robot with an obstacle

Fig. 5. Total step costs during 50 episodes.

ACKNOWLEDGMENT

This work was supported in part by the CREST project "Foundation of technology supporting the creation of digital media contents" of JST, and the "Global Robot Academia," Grant-in-Aid for Global COE Program by the Ministry of Education, Culture, Sports, Science and Technology, Japan.

REFERENCES

[1] Sutton RS, Barto AG (1988), Reinforcement Learning: An Introduction. Cambridge, MA: MIT Press

[2] Takayama A, Ito K, Minamino T(2008), Autonomous control of a snake-like robot using reinforcement learning - Discussion of the role of the mechanical body in abstraction of state-action space. Proc. of IEEE Annual Conference of the IEEE Industrial Electronics Society, pp.1584-1589.

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

[4] Fuchida T, Maehara M, Mori K, Murashima S (2000), A Learning Method of RBF Network using Reinforcement Leaning Method (in Japanese). Technical Report of IEICE, NC99-113, pp.157-163.