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

Adaptive reinforcement learning based on degree of learning progress

Akihiro Mimura[1] and Shohei Kato[1]

[1] Dept. of Computer Science and Engineering, Graduate School of Engineering, Nagoya Institute of Technology,
Gokiso-cho, Showa-ku, Nagoya, Aichi 466-8555, Japan
(Tel: +81-52-732-5625, Fax: +81-52-732-5625)
({mimura, shohey}@juno.ics.nitech.ac.jp)

Abstract: In this paper, we propose adjustment method named *adaptive learning rate considering learning progress* (ALR-P). The learning rate is a meta parameter that balances trade-off between speed and stability of the learning. Conventionally, designer had to manually set up a fixed learning rate. However, it is difficult for learning agent to adapt dynamic environment with a fixed learning rate. ALR-P enables adaptive adjustment of learning rate based on degree of learning progress for every steps. The degree of learning progress is calculated based on TD-error which is a difference of predicted and observed rewards. Only TD-error which can be calculated easily and simply is used in the ALR-P, so it can be applied into any types of reinforcement learning. We confirm effectiveness of ALR-P under a number of dynamic environments through the maze problem in which the environmental changes occurred.

Keywords: reinforcement learning, learning rate, dynamic environment, adaption, degree of learning progress

1 INTRODUCTION

Recently, there are a lot of researches about robotics using the reinforcement learning. The reinforcement learning is a kind of the machine learning which the learning agent learns and acquires appropriate control rules through the trial and error [1]. Because the learning agent acquires the control rules autonomously, for instance, reinforcement learning is often used for autonomous behavior acquisition of robots [2][3].

However, if meta parameters of the learning such as learning rate and the discount rate are inappropriate, the learning agent cannot behave sufficient performance. In general, the optimal value does not exist in these meta parameters, and the designer should adjust them properly according to the target task and progress of learning. In addition, it is difficult to learn appropriately under dynamic environment. When the reinforcement learning is applied to real environmental problem such as robot control, robustness for environmental change and noise should be taken into consideration. However, a lot of conventional methods are based on the premise that environment is static, and do not take environmental change into consideration.

In this paper, we focus on learning rate, which is one of the meta parameters of the reinforcement learning, and propose the dynamic adjusting method for learning rate.

2 REINFORCEMENT LEARNING

The agent learns control rule by repeating a sequence of taking an action based on action-value (Q-value) and receiving a reward. Q-value is predicted value of the total amount of reward that the agent will receive over the future. The reward indicates what is good in an immediate sense.

Softmax method is used for action selection in this paper, which is often used in many reinforcement learning. Softmax method determines selection probabilities of actions based on Q-values for the corresponding actions. Selection probability of action a in state s is shown as $\Pr(a|s)$, and is determined by the following equation.

$$\Pr(a|s) = \frac{\exp(Q(s,a)/\tau)}{\sum_{a' \in A} \exp(Q(s,a')/\tau)}, \tag{1}$$

where A, $Q(s,a)$, and τ show set of actions that the agent can select in the state s, action-value of the action a in the state s, and the temperature, respectively. The agent receives the reward that is the evaluation value of the selected action from the environment.

The agent dynamically updates the action-value function based on the received rewards. In this paper, we use Q( )-learning [4] as learning algorithm. In Q( )-learning, the action-value function is updated according to the following equation.

$$Q(s,a) \leftarrow Q(s,a) + \quad \delta(s,a)e(s,a), \tag{2}$$

where , $\delta(s,a)$, and $e(s,a)$ show the learning rate, the TD-error of action a in the state s, and the eligibility trace, respectively. TD-error is calculated by the following equation.

$$\delta = r + \gamma \max_{a' \in A'} Q(s',a') - Q(s,a), \tag{3}$$

where r, γ, s', and A' show the reward, the discount rate, the following state after the action a, and set of actions that

the agent can select in state s', respectively. Roughly, TD-error indicates the difference of the predicted value and the observed value in the state-action pair. Eligibility trace is index which shows whether the updating of the corresponding Q-value is proper or not.

3 LEARNING RATE

As shown in the Equation (2), a learning rate is a meta parameter which determines updating width of action-value function. Generally, in the case of that the learning rate is high, the progress of learning is fast although the learning is not steady. On the other hand, the learning progresses slowly though the learning is steady in the case of that the learning rate is low. This means that the learning rate is a parameter that balances trade-off between speed and stability of the learning.

In this paper, we propose adjustment method named *adaptive learning rate considering learning progress* (ALR-P). ALR-P adjusts learning rate according to each state (or state-action pair) for every step. The learning in the early stage or environment changes should be progressed speedy even if it can be roughly. because the learning agent have to acquire more proper action-value function as soon as possible in that cases. Meanwhile, the learning in the end stage should be progressed more precisely in order to converge action-value on optimal value in high accuracy. From the above things, the learning rate should be adjusted considering the learning progress; the learning rate should be high to value the learning speed in a state that the action-value cannot be correctly estimated, while the one in a state that the action-value can be correctly estimated should be low to value the stability.

4 ALR-P

ALR-P is a method of adjusting learning rate for every step considering learning progress, and we newly propose *degree of learning progress* in order to realize the adjustment of learning rate in this method. In this method, we focus on TD-error which can be thought a difference of predicted and observed rewards. In a state such as early stages of learning and environmental change, TD-error is high because prediction of reward is not proper. On the other hand, in a state such as converging stages of learning, TD-error is low and approaches 0 because prediction of reward is proper. TD-error changes for every steps, since the suitable learning can be not realized if the learning rate is adjusted based on the each change. Thus, we define the expected value of absolute TD-error as the degree of learning progress, and use the degree of learning progress to adjust the learning rate. The degree of learning progress in state s and action a, $d(s,a)$ is defined by the following equation.

$$d(s,a) := E[|\delta(s,a)|]. \tag{4}$$

Table 1. Q( )-learning applying ALR-P

Initialize $Q(s,a)$ arbitrarily for all s,a
Initialize $e(s,a)=0$ and $d(s,a)=0$ for all s,a
Repeat (for each episode):
  Initialize s,a
  Repeat (for each step of episode):
    Choose a' from s' using policy derived
    from Q (e.g., softmax-method)
    $\delta(s,a) \leftarrow r(s,a) + \gamma \max_{a'} Q(s',a') - Q(s,a)$
    $d(s,a) \leftarrow d(s,a) + \omega[|\delta(s,a)| - d(s,a)]$
    If $d(s,a) > d_m(s,a)$:
      $d_m(s,a) \leftarrow d(s,a)$
    $\alpha = d(s,a)/d_m(s,a)$
    $e(s,a) = 1$
    For all s,a:
      $Q(s,a) \leftarrow Q(s,a) + \alpha\delta e(s,a)$
      $e(s,a) \leftarrow \gamma\lambda e(s,a)$
    $s \leftarrow s'$
    until s is terminal

However, to calculate the expected value, it is necessary to save all history of TD-errors in each state-action pair. Since huge calculation memories and costs are needed for calculating the expected value, it is approximated by the following exponential moving average (EMA).

$$d(s,a) \leftarrow d(s,a) + \omega[|\delta(s,a)| - d(s,a)], \tag{5}$$

where ω shows updating width. By using the EMA, only the present TD-error is required to calculate the degree of learning progress, and the maximum $d(s,a)$ in each state-action pair is saved as $d_m(s,a)$. Using the calculated values from Equation (5), the learning rate is adjusted by the following equation for every steps.

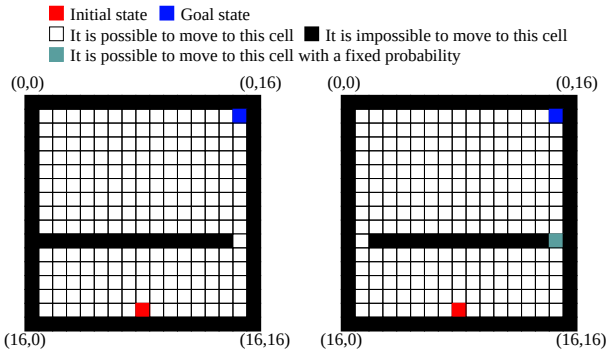
$$= \frac{d(s,a)}{d_m(s,a)}. \tag{6}$$

Because the learning rate is adjusted based on the degree of learning progress that each state-action pair has, so the learning rate can be adjusted according to the each state-action pair.

In ALR-P, only TD-error is used to calculate the degree of learning progress. Because the TD-error is a simple value generally calculated in existing reinforcement learning methods, it is easy to build ALR-P into almost all reinforcement learning methods such as Q-learning, Sarsa, Actor-Critic and so on. Q( )-learning applying ALR-P is shown in Table 1.

5 ADAPTIVE LEARNING EXPERIMENT

We conduct comparative experiments with ALR-P and conventional methods, and verify effectiveness of ALR-P.



(a) Until 100th episode (b) From 101st episode

Fig. 1. Maze problem

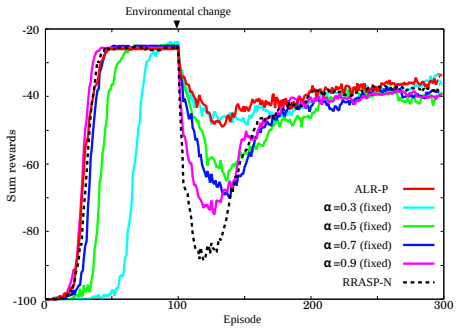


Fig. 2. Transition of sum rewards in the Setting A

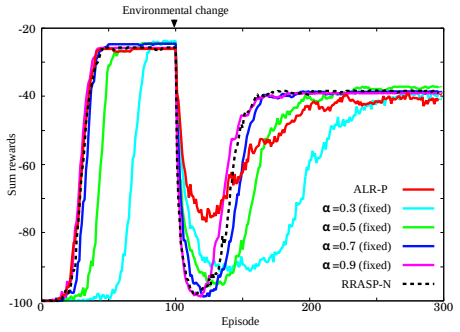


Fig. 3. Transition of sum rewards in the Setting B

5.1 Maze problem with environmental changes

We apply ALR-P to the maze problems (see Fig. 1) in which the environmental changes are occurred. Each cell shows state, and the coordinates are defined as follows: left-uppermost is $(0, 0)$, right-uppermost is $(0, 16)$, left-lowermost is $(16, 0)$, and right-lowermost is $(16, 16)$. An initial state and a goal state are set to $(15, 8)$ located in lower part of the maze and $(1, 15)$ located in upper-right part, respectively. The environmental changes in this experiment is defined as follows; until 100th episode (Fig. 1(b)), agent can reach goal state by passing through $(10, 15)$, from 101th episode (Fig. 1(b)), $(10, 1)$ is opened up along the left side, and $(10, 15)$ changes whether transition is possible or not with a fixed probability for every steps. And we prepared two transition settings, *Setting A*: agent can through the $(10, 15)$ with a possibility 1/7, and *Setting B*: agent can through the cell with a possibility 1/14. In Setting A, the expected value of sum rewards of right side path which contains $(10, 15)$ is higher than one of the left side path which contains $(10, 1)$. On the other hand, in Setting B, the expected values of sum rewards of both side paths are equivalent.

Comparative methods are RRASP-N [5] and the conventional general methods that learning rate is each fixed to 0.3, 0.5, 0.7, and 0.9. RRASP-N is a method to adjust learning rate to minimize square TD-errors. Q( )-learning is used as learning algorithm, and softmax method is used as action selection rule. This experiment consists of 300 episodes, and one episode is that learning agent reaches in goal state or passes over 100 steps. The agent moves to top, bottom, right, or left cell, and receives reward that is -1 in a step.

5.2 Experimental results

5.2.1 Transition of sum rewards

The above-mentioned experiments were conducted 100 times for the each method. The results shown in Fig. 2 and Fig. 3 indicate that the transition of sum rewards in the Setting A and Setting B, respectively. In Setting A, as comparing the results of the conventional methods that learning rate is each fixed, it was confirmed that advantage by magnitude relation of learning rate is reversed before and after environmental changes. Also in Setting B, there is no conventional method that always keeps advantage without differences of learning rate. Therefore, high learning performance could not be behaved with a stationary learning rate in this problem.

From Fig. 2 and Fig. 3, we confirmed that ALR-P behaved high performance, irrespective of environmental changes. Before the environmental changes, ALR-P converged learning fast. After the environmental changes, ALR-P behaved relatively high performance, and progresses learning without greatly reducing sum rewards even though immediately after the environmental changes. On the other hand, RRASP-N was reduced sum rewards greatly in the both settings.

5.2.2 Transition of learning rate

Fig. 4 shows an example of transition of learning rate in state-action pair that state is $(11, 15)$ and action is moving to up. From this figure, we confirm that after environmental changes learning rate was adjusted high again. Because the location of state $(11, 15)$ was near the state in which environmental changes occurred, it is thought that the state was profoundly affected by the environmental changes. ALR-P

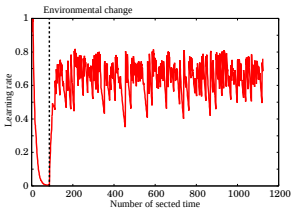


Fig. 4. Transition of learning rate in state-action pair that state is $(11, 15)$ and action is moving to up

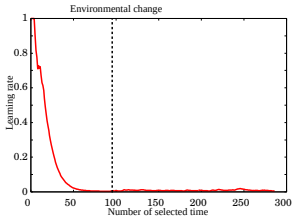


Fig. 5. Transition of learning rate in state-action pair that state is $(15, 8)$ and action is moving to right

Table 2. Success rate of the task and selection path at success episode after environmental changes in the Setting A

Learning rate	Success rate (%)	Selection rate of $(10, 1)$ (%)	Selection rate of $(10, 15)$ (%)
ALR-P	94.25	16.29	83.51
$\alpha = 0.3$ (fixed)	86.08	8.49	91.51
$\alpha = 0.5$ (fixed)	82.59	31.62	68.38
$\alpha = 0.7$ (fixed)	84.43	56.12	43.88
$\alpha = 0.9$ (fixed)	85.16	67.58	32.42
RRASP-N	83.08	84.38	15.62

Table 3. Success rate of the task and selection path at success episode after environmental changes in the Setting B

Learning rate	Success rate (%)	Selection rate of $(10, 1)$ (%)	Selection rate of $(10, 15)$ (%)
ALR-P	83.20	64.67	35.33
$\alpha = 0.3$ (fixed)	62.51	66.06	33.94
$\alpha = 0.5$ (fixed)	75.84	86.26	13.74
$\alpha = 0.7$ (fixed)	81.13	94.99	5.01
$\alpha = 0.9$ (fixed)	84.45	97.21	2.79
RRASP-N	83.08	96.83	3.17

could promote relearning by adjustment of the learning rate according to the variation of TD-error.

Fig. 5 shows an example of transition of learning rate in state-action pair that state is initial state $(15, 8)$ and action is moving to right. From this figure, we confirm that the learning rate was not high even though it was after the environmental changes. Because the location of state $(15, 8)$ was far from the state in which environmental changes occurred, it is thought that the state is not seriously affected by the environmental changes. From these results, it was confirmed that ALR-P could adjust learning rate to appropriate value on each state.

5.2.3 Success rate of task

The results shown in Table 2 and Table 3 indicate that success rate of the task and selection path at success episode after environmental changes in the Setting A and Setting B, respectively. From these tables, we confirm that success rate of ALR-P is relatively higher than other methods in the both settings. The results suggest that ALR-P enables adaptive learning under dynamic environment and has general versatility.

From the tables, we confirmed that ALR-P showed the relatively higher selection rate of $(10, 15)$ than other methods. In Setting B, although expected values of sum rewards of right and left side paths are equivalent, the right side path can be the shortest. Therefore, it can be said that ALR-P not only maintains high success rate, but also searches shorter path.

When RRASP-N is applied, selection rate of $(10, 1)$ is relatively higher than other methods. Thus, it is thought that RRASP-N maintains high success rate by selecting left side path in which environmental change does not occur.

6 CONCLUSION

In this paper, we propose adjustment method named adaptive learning rate considering learning progress (ALR-P). We confirmed that learning agent can adapt to dynamic environment by using ALR-P through maze problem with environmental changes.

For example, in robotic control of humanoids under real world, it is necessary to consider risk such as tumble. ALR-P which can adapt appropriately to dynamic environment is expected to decrease influence of breakdown caused by abrasion of joint, disturbance and so on.

ACKNOWLEDGMENT

This work was supported in part by the Ministry of Education, Science, Sports and Culture, Grant.in.Aid for Scientific Research under grant #20700199.

REFERENCES

[1] Sutton R. S and Barto A. G. (1998). *Reinforcement Learning: An Introduction*. The MIT Press.

[2] Mimura A, Nishibe S, and Kato S. (2011), Kinetic chained throwing humanoid robots by using reinforcement learning. *12th International Symposium on Advanced Intelligent Systems*, pp. 188–191.

[3] Riedmiller M, Gabel T, Hafner R, and Lange S. (2004), Reinforcement learning for robot soccer. *Autonomous Robots*, 27(1), pp.55–73.

[4] Watkins C. J. C. H and Dayan P. (1992), Q-leaning. *Machine Learning*, 8(3-4), pp.279–292.

[5] Noda I. (2010), Adaptation of stepsize parameter to minimize exponential moving average of square error by newton's method. *AAMAS*, pp. M–2–1.