2005 International Symposium on Nonlinear
Theory and its Applications (NOLTA2005)
Bruges, Belgium, October 18-21, 2005

A Detection Method of Environmental Changes Using Recurrence Plots for Reinforcement Learning

Tetsuya Takahashi† and Masaharu Adachi‡

†‡Department of Electronic Engineering, Graduate School of Engineering, Tokyo Denki University.
2-2 Kanda-Nishiki-Cho, Chiyoda-ku.
Email: 04gmd13@ed.cck.dendai.ac.jp, adachi@d.dendai.ac.jp

Abstract—In this article, we propose a novel method for detecting environmental changes in the reinforcement learning. The proposed method utilizes recurrence plots of state transitions of the system, and quantifies changes of the recurrence plot by a texture analysis. It is shown that the proposed method is effective to detect environmental changes.

1. Introduction

Reinforcement learning is a learning theory that an agent learns the optimal action in a state through trials-and-errors for an unknown environment [1]. When the agent does not have knowledge about the environment, the agent can learn the optimal actions by being presented only rewords for the actions. For such an advantage, the reinforcement learning is used in wide fields such as game problems, robot control, and dynamic allocation problems [1].

Reinforcement learning is known that it can apply to a dynamic environment for that the agent learns by trials-and-errors. When the agent learns a dynamic environment, it is already reported that the method of adjusting specific parameters in the reinforcement learning is effective. The method for adjusting the parameters, such as the learning rate, the discount rate and the strength of randomness of an agent, is known as the meta-learning in the reinforcement learning [2]. By using the meta-learning, the agent can adjust quickly in the static or dynamic environment. One of the purposes of the meta-learning is an acceleration of a learning speed. Several methods of an acceleration the learning speed such as profit sharing or eligibility trace exist besides the meta-learning. However, those methods are considering past events, and it differs from the meta-learning. Some other methods of adjusting parameters are also reported. Those parameters are controlled by a certain total reward for a duration [3], by an estimated reliability using accumulated **TD** error [4] or by statistical method using probabilistic model [5]. Those proposed methods, that use total reward accumulated **TD** error or a state transition probability, are kinds of detecting methods by evaluating how much an agent's learning progressed enough or how much agent's state transition converged. It is important that the measure of an agent's learning progressed enough or not for the meta-learning. We consider that obtaining a measure for an environmental change is one of the important stage to the meta-learning. For the method of the detection of environmental changes, Tanaka et al. have reported on the method of the detection of environmental changes using a sequential Monte Carlo [6].

In the present paper, we propose a new method using recurrence plots as a method for detecting the environmental changes. The recurrence plot is effective to distinguish steady between dynamic time series data [7]. In this paper, we use the recurrence plots as a tool that distinguishes the regularity of the state transition. Furthermore, we show that a quantitative analysis of the recurrence plots by the texture analysis is also effective for the detection of the environmental changes.

In the following, section 2 describes about the Q-learning which is used for constructing controller through out the present paper. In section 3, we propose a detecting method for environmental changes by using recurrence plots. Section 4 shows a schematic of pole balancing problem that is to be learned by the Q-learning. Next, in section 5 we describe results of the detection of environmental changes. As the final, we give conclusions in section 6.

2. Q-learning

In this paper, we use a simple Q-learning [9] to learn controlling a pole balancing system. The purpose of the Q-learning is to obtain the optimal action-value-function $Q^*(s,a)$, at a state s for an action a. At discrete time t, Q value is updated by Eqs. (1) and (2).

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t)+\alpha\delta_t, \quad (1)$$

$$\delta_t = r_{t+1}+\gamma\max_a Q(s_{t+1},a)-Q(s_t,a_t), \quad (2)$$

where α, γ, and δ_t denote the learning rate, the discount rate, and the **TD** error, respectively.

In this paper, the agent uses the ε-greedy policy. In the ε-greedy policy, the agent's action is a random action at the probability of ε, otherwise, it is the best action at the current state that is determined by $Q(s,a)$. In the following numerical experiment, α, γ, and ε are fixed to 0.1, 0.99, and 0.00001, respectively.

The purpose of the reinforcement learning is a return maximization. The agent acquires the best actions that

can lead to maximum summation of the rewards through the learning. The reinforcement learning deals an immature $Q(s_t, a_t)$ and an ε-greedy policy π as an approximate value in the early stage of the learning. By iterating the learning, $Q(s_t, a_t)$ and π become the optimal value and the optimal policy. When $Q(s_t, a_t)$ is immature the difference between the current $Q(s_t, a_t)$ and the optimal $Q^*(s_t, a_t)$ is large, consequently, the agent selects a probabilistic action. Conversely, when the agent learns enough, the agent selects the best action. Therefore, it is useful to know whether the agent's action is always probabilistic or not in order to detect the progress of the learning or to detect environmental changes in the reinforcement learning.

3. Detection of environmental changes

3.1. Recurrence Plots

Recurrence plot is effective to distinguish steady between dynamic time series data [7]. Let a time series data of length N is used to make the recurrence plot.

In general, each point in a time series is presented by a vector $\boldsymbol{v}$. A distance between two vectors of the i th and the j th points $\boldsymbol{v}(i)$, $\boldsymbol{v}(j)$ in the time series is denoted by $D(i, j)$ and calculated by the following equation.

$$D(i, j) = \|\boldsymbol{v}(i) - \boldsymbol{v}(j)\|, \tag{3}$$

where $\|\cdot\|$ denotes Euclidean norm.

We make a recurrence plot using $D(i, j)$. The recurrence plot can be obtained by plotting a dot on (i, j) pixel, when the distance $D(i, j)$ is smaller than predetermined threshold θ_D .

These examples demonstrate that one can distinguish between static and dynamic time series by the textures of the recurrence plots.

3.2. Texture Analysis of a Recurrence plots

A texture analysis for quantifying recurrence plots has been proposed [8] . In the analysis, the feature of a recurrence plot is obtained by a co-occurrence matrix. The co-occurrence matrix $P_C(r, \theta, l_1, l_2)$ represents probabilities that the luminance of a pixel and another pixel that apart from each other for the interval (r, θ) are l_1 and l_2, respectively. Where r is a distance and θ is an angle between the pixels.

In the present paper, the recurrence plot is created as a binary image. A feature of the image is obtained by the following equation.

$$f_{r,\theta} = \sum_{l_1, l_2 = 0, |l_1 - l_2| = 1}^{1} P_C(r, \theta, l_1, l_2), \tag{4}$$

where $f_{r,\theta}$ represents probability that the luminances of the two pixels apart for (r, θ) are different in the recurrence plot.

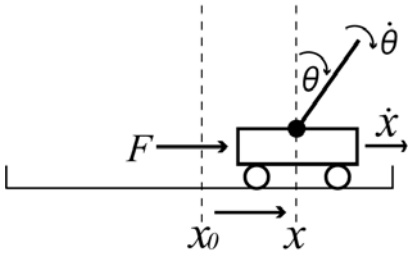


Figure 1: Shematic diagram of a pole balancing problem.

4. System to be Learned for the Control

4.1. Pole Balancing Problem

In this paper, numerical experiments are executed on the pole balancing problem [10]. The purpose of learning pole balancing problem is that the agent becomes to select appropriate forces to the cart so that the pole never fall down for predetermined duration. Fig. 1 shows a schematic representation of the pole balancing problem.

A task is composed of 900 episodes for a trial which is composed of $600,000$ time steps that correspond to 200 minutes in real time. States of reinforcement learning are decided by boxes system which is expressed by combining quantized state variables [10]. In this paper, the state variables x, $\dot{x}$, θ and $\dot{\theta}$ are quantized by the following thresholds ; $x : \pm 0.8, \pm 2.4[m]$, $\dot{x} : \pm 0.5, \pm\infty[m/s]$, $\theta : 0, \pm 1, \pm 6, \pm 12[^\circ]$, $\dot{\theta} : \pm 50, \pm\infty[^\circ/s]$, where x and $\dot{x}$ denote the position and the velocity of the cart, respectively. θ and $\dot{\theta}$ are the angle and the angular velocity of the pole, respectively.

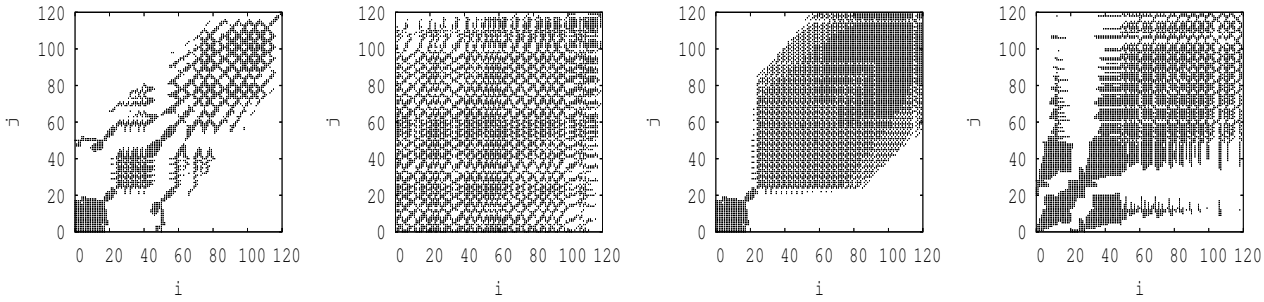
The goal of the control is to hold the pole within $x = \pm 2.4[m]$ and $\theta = \pm 12[^\circ]$ during $600,000$ time steps. When the agent succeeds to hold the pole, the agent is given $r = 0$ as a reward, otherwise the agent is given $r = -1$ as a penalty. Q-learning is executed based on above rewards. An action of the agent is selected from $F_t = 0, \pm 10, \pm 20[N]$ at time t. The angle of the pole on the cart is initialized at $6[^\circ]$.

The system's dynamics is represented by the following Eqs. (5) and (6).

$$\ddot{\theta}_t = \frac{g \sin\theta_t + \cos\theta_t \left[\frac{-F_t - ml\dot{\theta}_t^2 \sin\theta_t + \mu_c \mathrm{sgn}(\dot{x}_t)}{m_c + m} \right] - \frac{\mu_p \dot{\theta}_t}{ml}}{l\left[\frac{4}{3} - \frac{m\cos^2\theta_t}{m_c + m} \right]}, \tag{5}$$

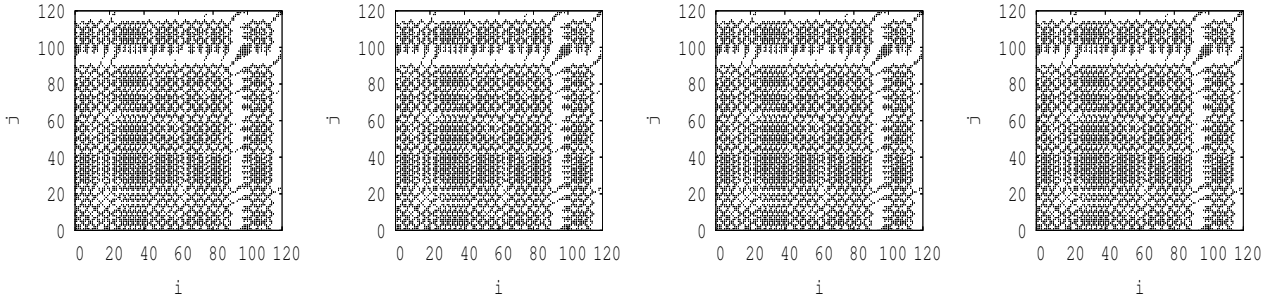
$$\ddot{x}_t = \frac{F_t + ml[\dot{\theta}_t{}^2 \sin\theta_t - \ddot{\theta}_t \cos\theta_t] - \mu_c \mathrm{sgn}(\dot{x}_t)}{m_c + m}, \tag{6}$$

where F_t denote an applied force to the cart's center of mass at time t. g, m_c, μ_c and μ_p denote the gravity, the mass of the cart and the coefficients of the friction of the cart on the track and the pole on the cart, respectively. The values



(a) 510 episode. (b) 511 episode. (c) 512 episode. (d) 513 episode.

Figure 2: Recurrence plots during the early stage of the environment 2 which means early steps of 100–220 in the environment 2.



(a) 690 episode. (b) 691 episode. (c) 692 episode. (d) 693 episode.

Figure 3: Recurrence plots during the late stage of the environment 2 which means early steps of 100–220 in the environment 2.

are fixed to $g = 9.8[m/s^2]$, $m_c = 1.0[kg]$, $\mu_c = 0.0005$ and $\mu_p = 0.000002$ in the following experiments.

4.2. Environmental Changes of the Pole Balancing Problem

In this paper, we assume that environmental changes mean changing the mass m and or length l of the pole at a given episode. One trial is composed of 900 episodes and three kinds of the environment exist in the trial. These environments are defined as followings.

- $m = 0.2[kg], l = 0.4[m]$, during episode 1–500.
- $m = 0.4[kg], l = 0.2[m]$, during episode 501–700.
- $m = 0.2[kg], l = 0.5[m]$, during episode 701–900.

These are called environment 1, environment 2 and environment 3, respectively in the following section.

5. A Detection of Environmental Changes

5.1. A Detection Method of Environmental Changes using Recurrence Plots

Figures 2 and 3 show recurrence plots in the environment 2. Each recurrence plot is created by the change of a vector of the state variables $\boldsymbol{v}= (x, \dot{x}, \theta, \dot{\theta})$, for early steps of 100–220 in each episode ($N = 120$). Each element of the state vector $\boldsymbol{v}$ using recurrence plots are normalized. The threshold θ_D used for the recurrence plots is fixed to 0.3. Figures 2(a)–(d) show the recurrence plots during the early stage of the environment 2. Figures 3(a)–(d) show the recurrence plots during the last stage of the environment 2. During the early stage of the learning for the new environment, the state transitions of the system responding to the action determined by the agent are dynamic because $Q(s_t, a_t)$ of an agent is immature. Therefore, recurrence plots are changing for every episode. On the contrarily, when the agent learns enough the state transitions are static because $Q(s_t, a_t)$ of an agent is mature. Consequently, recurrence plots are not changing in successive episodes.

5.2. Result of Texture Analysis of Recurrence Plots

We quantify recurrence plots using texture analysis, and obtain a feature vector $\bar{\boldsymbol{f}}$ for each episode. In the following, the parameters in Eq. (4), θ_a are fixed to $0°$, $45°$ and $135°$. The feature vector $\bar{\boldsymbol{f}}$ consists of [$\bar{f}_{r1,0}$, $\bar{f}_{r1,45}$, $\bar{f}_{r1,135}$, $\bar{f}_{r2,0}$, $\bar{f}_{r2,45}$, $\bar{f}_{r2,135}$, $\bar{f}_{r3,0}$, $\bar{f}_{r3,45}$, $\bar{f}_{r3,135}$], where $f_{ri,\theta}$ is calculated by the following equation.

$$\bar{f}_{ri,\theta} = \frac{1}{20} \sum_{i=ri}^{ri+20} f_{i,\theta}, \tag{7}$$

where $r1$, $r2$ and $r3$ denote 1, 21 and 41, respectively. We intend to calculate $\bar{f}$ for short, middle and long ranges. One can evaluate how much dose $\bar{\boldsymbol{f}}$ change for each episode by

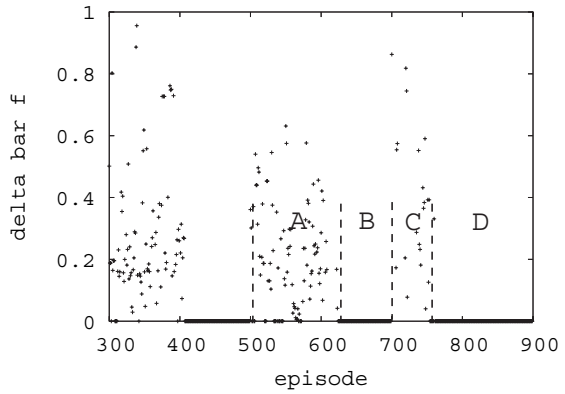


Figure 4: $\Delta\bar{\boldsymbol{f}}$ for each episode. A and B denote the early and the late stage of the environment 2. C and D denote the early and the late stage of the environment 3.

the following equation.

$$\Delta\bar{\boldsymbol{f}}(t) = \|\bar{\boldsymbol{f}}(t) - \bar{\boldsymbol{f}}(t-1)\|, \tag{8}$$

where $\|\cdot\|$ denotes Euclidean norm.

Figure. 4 show $\Delta\bar{\boldsymbol{f}}(t)$ for each episode. In the Fig. 4, A and B denote the early and the late stage of the environment 2. C and D denote the early and the late stage of the environment 3. During A or C, $\Delta\bar{\boldsymbol{f}}(t)$ become larger than zero. On the other hand, during B or D, $\Delta\bar{\boldsymbol{f}}(t)$ becomes zero. As shown above, a change of an environmental can be detected as a quantitative expression by the texture analysis of the recurrence plots.

However, the above detection of environmental change is effective not for every case. The goal of the control of the pole balancing system is to maintain the pole staying in the vicinity of an unstable fixed point. Therefore, this problem is sensitive to the input. When an agent uses a policy with the ε-greedy, the agent randomly selects actions with probability ε. Consequently, the failure of the control often happens even in after learning for long duration. The agent detects an environmental change by mistake in such a case. The agent also mistakes for the detection, when the agent explores even though the agent's learning is enough.

6. Conclusion and Discussion

We proposed a new method of detecting the environmental changes in a reinforcement learning. This method can detect environmental changes by using the recurrence plot of the state variables that can be observed by an agent. While an agent is in the early stage of the learning for new environments, the state transitions of the controlled system changes dynamically. In contrast, when an agent is in the late stage of the learning, the state transitions of the controlled system are steady. Even in the late stage of the learning, the state transitions become dynamic when the environmental changes. Therefore such environmental changes can be detected by the recurrence plots. A feature is obtained by quantifying the recurrence plots using a texture analysis. The change of the feature indicates environmental changes.

In the proposed method, there is the following problem. The agent can not distinguish between the immature learning and the environmental changes. Because the proposed method is detecting by the state transitions in the system. In the future, we need to consider the above problem. Moreover, developing a method to utilize the feature obtained by the proposed method for the meta-learning is an important future problem. Then we could consider more effective meta-learning for the return maximization than the existing ones.

References

[1] R. S. Sutton, and A. G. Barto, "Reinforcement learning: An introduction", The MIT Press, 1998.

[2] N. Schwighofer and K. Doya, "Meta-learning in Reinforcement Learning", Neural Networks vol. 16, pp. 5-9, 2003.

[3] K. Murakoshi, J. Mizuno, "A parameter control method in reinforcement learning to rapidly follow unexpected environmental changes", Biosystems, No. 77, pp. 109-117, 2004.

[4] N. Ogawa, A. Namiki, and M. Ishikawa, "Adjustment of Discount Rate Using Index for Progress of Learning", (in Japanese), TECH. REP. OF IEICE, NC2002-129, pp. 73-78, Feb. 2003.

[5] S. Ishii, W. Yoshida, and J. Yoshimoto, "Control of exploitation-exploration meta-parameter in reinforcement learning", Neural Networks, 15(4-6), 665-687, 2002.

[6] A. Tanaka, Y. Nakada, T. Matsumoto, "Reinforcement Learning under Dynamic Environment : Sequential Monte Carlo with Sample Re-initialization", TECHN, REP, OF IEICE, NC2004-186(2005-03), pp. 101-106, Mar. 2005. (in Japanese).

[7] M. Koebbe and G. Mayer-Kress, "Use of Recurrence Plots in the Analysis of Time-Series Data", NONLINEAR MODELING AND FORECASTING, pp. 361-378.

[8] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural Features for Image Classification", IEEE, Trans.Syst.Man.Cybern, vol.3, No. 6, pp. 610-621, November 1973.

[9] Watkins. C. J. C. H, "Learning from Delayed Rewards", Ph. D. thesis, Cambridge University. 1989.

[10] A.G.Barto, R.S.Sutton, C.W.Anderson, "Neuronlike Adaptive Elements That Can Solve Difficult Learning Control Problems", IEEE, Trans.Syst.Man.Cybern, vol.13, no.5, 1983.