Adaptive decision making using a chaotic semiconductor laser for multi-armed bandit problem with time-varying hit probabilities

Akihiro Oda¹, Takatomo Mihana¹, Kazutaka Kanno¹, Makoto Naruse², and Atsushi Uchida¹a)

¹ Department of Information and Computer Sciences, Saitama University, 255 Shimo-Okubo, Sakura-ku, Saitama City, Saitama 338-8570, Japan
² Department of Information Physics and Computing, Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

¹a) auchida@mail.saitama-u.ac.jp

Received July 10, 2021; Revised August 30, 2021; Published January 1, 2022

Abstract: We numerically demonstrate the principle of adaptive decision making for solving multi-armed bandit problems in dynamically changing reward environments. We use the tug-of-war method by comparing a threshold and a chaotic temporal waveform generated from a semiconductor laser observed in an experiment. We propose a method for detecting dynamic changes in hit probabilities by evaluating short-term standard deviations of the estimated hit probabilities. Furthermore, the threshold is forced to be initialized when changes in the hit probabilities are detected. We perform adaptive decision making in time-varying hit probabilities, including cases in which the differences in the hit probabilities are small. The proposed method paves the way for ultrafast photonic decision making in dynamically changing environments for various applications, such as cognitive wireless communications and robot control using reinforcement learning.

Key Words: decision making, reinforcement learning, chaos, semiconductor laser, multi-armed bandit problem, AI photonics

1. Introduction

Reinforcement learning has been used for artificial intelligence without training data, such as in robot control, computer gaming, and Internet advertising [1–3]. The multi-armed bandit problem is a fundamental concept of reinforcement learning, where a player repeatedly selects multiple slot machines with initially unknown hit probabilities to maximize the total reward obtained from the slot machines under a finite number of trials. Multi-armed bandit problems can be applied to cognitive...
wireless communications [4, 5] and robot-arm grasping tasks [6]. Exploration and exploitation are required for solving multi-armed bandit problems. Exploration is a process that searches for a slot machine with the highest hit probability. Exploitation is the selection of the slot machine that has the highest hit probability estimated from the exploration. More exploration is required to determine the slot machine with the highest hit probability, whereas fewer resultant exploitation opportunities lead to a reduced total reward. In contrast, more exploitation is preferable to gain the total reward; however, less exploration results in missing the slot machine with the highest hit probability. Therefore, there is a tradeoff between exploration and exploitation. This is known as the exploration–exploitation dilemma [1, 2].

The tug-of-war method has been proposed to solve the multi-armed bandit problem [7–9]. This method is based on the biological mechanism of amoeboid feeding activity [10]. The tug-of-war method provides faster decision making than software algorithms [7]. Recently, photonic implementations of the tug-of-war method have been demonstrated using quantum dots [11, 12], single photons [13], entangled photons [14], and chaotic semiconductor lasers [15–23]. Furthermore, photonic decision making of the multi-armed bandit problem has been reported using chaotic temporal waveforms with a threshold [15–17], lag synchronization of chaos in coupled semiconductor lasers [18], laser network dynamics [19, 20], and mode-switching dynamics in a ring-cavity laser [21].

In previous studies, decisions were made under a condition in which the hit probabilities of the slot machines were fixed. A few studies have reported cases in which the hit probabilities dynamically change over time [15, 16]. In [16], the effect of the memory parameter \(\alpha\) of the threshold adjuster was investigated to realize adaptive decision making with time-varying hit probabilities. However, this study was limited to the parameter setting of the hit probabilities for simplification, that is, \(P_1 + P_2 = 1\), where \(P_i\) denotes the hit probability of slot machine \(i\). Such simplification is not generally applicable. Hence, it is important to study a novel technique for adaptive decision making under time-varying hit probabilities or dynamically changing reward environments.

In this study, we investigate adaptive decision making for solving multi-armed bandit problems using slot machines with time-varying hit probabilities. We use the tug-of-war method and chaotic temporal waveforms generated from a semiconductor laser. We compare the chaotic temporal waveforms and a threshold value for decision making. Moreover, we introduce a new algorithm to detect changes in the hit probabilities. We also initialize the threshold of the chaotic temporal waveform when a change in hit probabilities is detected. Finally, we investigate the dependence of decision-making performance on parameter settings.

2. Tug-of-war method

We describe the tug-of-war method for solving the multi-armed bandit problem with two slot machines, referred to as a two-armed bandit problem, using a chaotic temporal waveform and a threshold [15, 16]. This method is schematically illustrated in Fig. 1(a). A chaotic temporal waveform is observed in a semiconductor laser with optical feedback in the experiment and stored in a digital oscilloscope with an 8-bit vertical resolution. The chaotic temporal waveform is used to determine the slot machine selection. A threshold is introduced in the chaotic temporal waveform, which is sampled at the sampling interval \(t_s\), and the sampled value is compared with the threshold. Slot machine 1 (\(S_1\)) is selected if the sampled value is larger than the threshold, and slot machine 2 (\(S_2\)) is selected otherwise. If \(S_1\) is selected and the result shows “hit,” then the threshold is decreased so that the probability of selecting \(S_1\) becomes higher at the next selection. However, if \(S_1\) is selected and the result shows “miss,” then the threshold is increased so that the probability of selecting \(S_1\) becomes lower at the next selection. This procedure is repeated until the threshold reaches the maximum or minimum value of the chaotic temporal waveform, and the final decision is made.

We explain a method for controlling the threshold of the chaotic temporal waveform [15, 16]. The threshold \(T(t)\) is controlled by the threshold adjuster \(TA(t - 1)\) as follows:
Fig. 1. (a) Schematic of tug-of-war method using chaotic temporal waveform. (b) Setting of time-varying hit probabilities for two slot machines.

\[
T(t) = \begin{cases} 
T_{max} & \text{if } (\text{int})TA(t-1) > T_{max} \\
(\text{int})TA(t-1) & \text{if } T_{min} \leq (\text{int})TA(t-1) \leq T_{max} \\
T_{min} & \text{if } (\text{int})TA(t-1) < T_{min},
\end{cases}
\]

(1)

where \(k\) denotes the threshold coefficient and \((\text{int})\) indicates the conversion of \(TA(t-1)\) to an integer.

The threshold \(T(t)\) is saturated at \(T_{min}\) or \(T_{max}\).

The threshold adjuster \(TA(t)\) is determined by the past result of \(TA(t-1)\) as follows:

\[
TA(t) = X + \alpha TA(t-1)
\]

(2)

where \(\alpha\) is the memory parameter of the threshold adjuster that determines the amount of information from the past results to determine the present \(TA(t)\) [16]. \(X\) is the step size of the threshold adjuster, and \(X\) is determined by the result of slot machine selection [9], as shown in Table 1 and expressed in Eqs. (3) and (4).

\[
\Delta = 2 - (\hat{P}_1 + \hat{P}_2)
\]

(3)

\[
\Omega = \hat{P}_1 + \hat{P}_2
\]

(4)

where \(\Delta\) and \(\Omega\) represent the step sizes of the threshold adjuster when the results of the selected slot machine are “hit” and “miss,” respectively. We use a new definition of \(\Delta\) and \(\Omega\) [18] to avoid excessively large or small values (\(\Delta = 1\) and \(\Omega = (\hat{P}_1 + \hat{P}_2)/(2 - (\hat{P}_1 + \hat{P}_2))\) were used in [15,16]). \(\hat{P}_i\) is the estimated hit probability for slot machine \(i\) \((S_i)\) and is obtained as follows:

\[
\hat{P}_i = \frac{H_i}{C_i}
\]

(5)

where \(H_i\) and \(C_i\) are the number of hits and the number of plays for \(S_i\), respectively. From Eqs. (1)–(4) and Table I, \(T(t)\) and \(TA(t)\) are affected by the estimated hit probabilities \(\hat{P}_i\) and the step size

| \(S_1\) | Hit  | Miss |
|--------|------|------|
| \(-\Delta\) | \(\Omega\) |
| \(S_2\) | \(\Delta\) | \(-\Omega\) |
of the threshold adjuster $\Delta$ and $\Omega$. Therefore, it is important to obtain accurate values of $\hat{P}_i$ for successful decision making.

We consider a situation in which the hit probabilities of the two slot machines dynamically change over time, as summarized in Fig. 1(b). In the initial 1000 iterations, for example, the hit probabilities of slot machines 1 and 2 are configured as $P_1 = 0.1$ and $P_2 = 0.3$, respectively. The correct decision is the selection of slot machine 2 because $P_1 < P_2$ (red in Fig. 1(b)). In the subsequent 1000 plays, the hit probabilities of slot machines 1 and 2 are changed to $P_1 = 0.9$ and $P_2 = 0.7$, respectively. Here, the decision to choose slot machine 1 is correct because $P_1 > P_2$. Then, the hit probabilities are configured as $P_1 = 0.1$ and $P_2 = 0.3$ again for the subsequent 1000 plays, followed by $P_1 = 0.9$ and $P_2 = 0.7$ for the next 1000 plays. Thus, the reward environmental changes are incorporated in every 1000 plays. We denote this change as $(P_1, P_2) = (0.1, 0.3)$ or $(0.9, 0.7)$. The slot machine selection with 4000 consecutive plays is repeated 1000 times (denoted as 1000 cycles) to evaluate the correct decision rate.

Figure 2(a) shows the experimentally observed chaotic temporal waveform of the laser used for decision making [15, 16]. We use a semiconductor laser with time-delayed optical feedback to generate chaotic temporal waveforms [24]. These chaotic temporal waveforms are digitized at 8-bit vertical resolution in the range of -128 to 127 and are observed with an average period of 150 ps (i.e., a peak frequency of 6.6 GHz in the frequency spectrum). The sampling interval of the chaotic temporal waveform is set to $t_s = 10$ ps for decision making. Figure 2(b) shows a histogram of the chaotic temporal waveform of the laser intensity. The histogram has a peak at approximately zero and is slightly asymmetric. We use this chaotic temporal waveform for decision making to solve the two-armed bandit problem with time-varying hit probabilities.

Figure 3 shows the results of decision making using the conventional tug-of-war method [15, 16]. We use a semiconductor laser with time-delayed optical feedback to generate chaotic temporal waveforms [24]. These chaotic temporal waveforms are digitized at 8-bit vertical resolution in the range of -128 to 127 and are observed with an average period of 150 ps (i.e., a peak frequency of 6.6 GHz in the frequency spectrum). The sampling interval of the chaotic temporal waveform is set to $t_s = 10$ ps for decision making. Figure 2(b) shows a histogram of the chaotic temporal waveform of the laser intensity. The histogram has a peak at approximately zero and is slightly asymmetric. We use this chaotic temporal waveform for decision making to solve the two-armed bandit problem with time-varying hit probabilities.

Figure 3 shows the results of decision making using the conventional tug-of-war method [15, 16]. Figure 3(a) shows the estimated hit probabilities $\hat{P}_1$ and $\hat{P}_2$ (solid lines) of the two slot machines when the true hit probabilities $P_1$ and $P_2$ (dotted lines) are changed. The estimated hit probabilities do not exactly follow the true hit probabilities, and the estimation of the hit probabilities fails, except for the first 1000 plays.

We introduce the correct decision rate (CDR) as a quantitative measure of the decision-making performance. We perform the slot machine selection for $m$ plays as one cycle, and the selection of $m$ plays is repeated for $n$ cycles. The CDR is defined as follows [15, 16]:

$$CDR(t) = \frac{1}{n} \sum_{i=1}^{n} C(i, t),$$

where $C(i, t)$ is 1 when the slot machine with the highest hit probability is selected at the $t$-th play and $i$-th cycle and 0 otherwise. CDR is the ratio of selecting the slot machine with the highest hit
Fig. 3. Result of the decision making using the conventional tug-of-war method [15, 16]. (a) True (dotted lines) and estimated (solid lines) hit probabilities of two slot machines and (b) correct decision rate (CDR). Time-varying hit probabilities of \((P_1, P_2) = (0.1, 0.3)\) or \((0.9, 0.7)\) are used. (a) Example of one cycle and (b) average over 1000 cycles.

The CDR over \(n\) cycles. A \(CDR(t)\) of 1 indicates that decision making is successful at the \(t\)-th play on the cycle average.

Figure 3(b) shows the correct decision rate (CDR) using the conventional tug-of-war method. The CDR approaches 1 for the first 1000 plays. However, the CDR becomes almost zero when the hit probabilities change after 1000 plays. Therefore, correct decision making is not accomplished after 1000 plays.

The reason for this failed decision making is as follows. In the parameter setting of \((P_1, P_2) = (0.1, 0.3)\) or \((0.9, 0.7)\), the slot machine with the highest hit probability is switched between the two slot machines every 1000 plays. Therefore, adaptive decision making is required to increase the maximum reward. In addition, the situation of \(P_1 + P_2 \neq 1\) is assumed in this case, which becomes more difficult to solve than the case in [16], where \(P_1 + P_2 = 1\) was assumed (i.e., \((P_1, P_2) = (0.4, 0.6)\) or \((0.6, 0.4)\)).

The change in the sum \(P_1 + P_2\) strongly affects the step size of the threshold adjusters \(\Delta\) and \(\Omega\), as shown in Eqs. (3) and (4), respectively. Moreover, \(P_1 + P_2\) changes every 1000 plays in the case of Fig. 3 (i.e., \(P_1 + P_2 = 0.4\) or 1.6). This results in an uncertainty of \(\Delta\) and \(\Omega\), and the threshold adjuster \(TA(t)\) does not move correctly.

3. Proposed method

We propose a novel scheme for adaptive decision making when hit probabilities dynamically change over time. We introduce three processes to realize adaptive decision making: (i) Introduction of the memory effect in estimated hit probabilities (ii) Detection of the changes in hit probabilities by monitoring the short-term standard deviation of the estimated hit probabilities (iii) Initialization of the threshold value for a chaotic temporal waveform when changes in the hit probabilities are detected. We explain these three processes below.

First, we introduce a memory parameter for the estimated hit probabilities for a short duration. In the previous method, the estimated hit probabilities were calculated for all past plays [16]. Therefore, it is difficult to detect the changes in hit probabilities using this method. Here, we introduce the memory effect with respect to the estimated hit probabilities \(\hat{P}_i(t)\) as follows:

\[
\hat{P}_i(t) = \frac{H_i(t)}{C_i(t)}
\]

\[
H_i(t) = \begin{cases} 
1 + \beta H_i(t-1) & \text{(Hit)} \\
\beta H_i(t-1) & \text{(Miss)} 
\end{cases}
\]
First, we calculate the estimated hit probabilities based on the proposed method. We use the same parameter values used in our numerical simulations as summarized in Table II.

$$C_i(t) = 1 + \beta C_i(t-1)$$

where $\beta$ is the memory parameter of the estimated hit probability $(0 \leq \beta \leq 1)$. $\beta$ is an important parameter for detecting changes in hit probabilities and indicates the extent to which the system memorizes past events. A larger $\beta$ implies a larger memory effect, and $\beta = 1$ is equivalent to the former scheme [16].

To detect the changes in hit probabilities, we introduce the short-term standard deviation of the estimated hit probabilities at the $t$-th play for slot machine $i$ is calculated as follows:

$$\sigma_i(t) = \begin{cases} \sqrt{\frac{1}{t} \sum_{j=1}^{t} (\bar{P}_i(j) - \bar{P}_i)^2} & (t < W) \\ \sqrt{\frac{1}{W} \sum_{j=t-W+1}^{t} (\bar{P}_i(j) - \bar{P}_i)^2} & (t \geq W) \end{cases}$$

where $\bar{P}_i$ is the mean of $\bar{P}_i(j)$ over $t$ ($t < W$) or $W$ ($t \geq W$). $W$ is the window size used to calculate short-term standard deviations. Further, $W$ is an important parameter for detecting the changes in $\bar{P}_i(t)$. The changes in $\bar{P}_i(t)$ can be detected by monitoring $\sigma_i(t)$.

Finally, we introduce the initialization of the threshold adjuster $TA(t)$ when changes in $\bar{P}_i(t)$ are detected based on $\sigma_i(t)$. The threshold adjuster $TA(t)$ is initialized to zero as follows:

$$TA(t) = \begin{cases} 0 & (\sigma_i(t) > \sigma_d \text{ and } T(t) = T_{\text{min}} \text{ or } T_{\text{max}}) \\ X + \alpha TA(t-1) & \text{(otherwise)} \end{cases}$$

where $\sigma_d$ is the detection criterion for the standard deviation for the initialization. $TA(t)$ is initialized when either $\sigma_1(t)$ or $\sigma_2(t)$ exceeds $\sigma_d$ and $T(t)$ is saturated at $T_{\text{min}}$ or $T_{\text{max}}$. The threshold $T(t)$ is also initialized to zero when $TA(t-1) = 0$, according to Eq. (1). The effect of initialization of $T(t)$ and $TA(t)$ is important for deleting the past history of $TA(t)$. The parameter values used in our numerical simulations are summarized in Table II.

### 4. Numerical results

First, we calculate the estimated hit probabilities based on the proposed method. We use the same condition for the changes in the true hit probabilities as in Fig. 1(b), that is, $(P_1, P_2) = (0.1, 0.3)$ or $(0.9, 0.7)$ for every 1000 plays. Figure 4(a) shows the estimated hit probabilities $\hat{P}_1$ and $\hat{P}_2$ (solid lines) of the two slot machines when the true hit probabilities $P_1$ and $P_2$ (dotted lines) are changed every 1000 plays. Evidently, $\hat{P}_1$ and $\hat{P}_2$ follow the changes in $P_1$ and $P_2$, unlike the former results shown in Fig. 3(a). Therefore, $\hat{P}_1$ and $\hat{P}_2$ are appropriately and accurately obtained.

### Table II. Parameter values used in numerical simulations.

| Parameter                                      | Variable | Value     |
|------------------------------------------------|----------|-----------|
| Hit probability for slot machine 1             | $P_1$    | 0.1 ↔ 0.9 |
| Hit probability for slot machine 2             | $P_2$    | 0.3 ↔ 0.7 |
| Threshold coefficient                          | $k$      | 8         |
| Range of chaotic temporal waveform             | $R$      | $-128 \sim 127$ |
| Maximum threshold                              | $T_{\text{max}}$ | 128       |
| Minimum threshold                              | $T_{\text{min}}$ | $-128$    |
| Total number of plays                          | $m$      | 4000      |
| Total number of cycles                         | $n$      | 1000      |
| Sampling interval of chaotic temporal waveform | $t_s$    | 10 ps     |
| Window size of standard deviation               | $W$      | 200       |
| Detection criterion of standard deviation       | $\sigma_d$ | 0.10    |
Figure 4(b) shows the changes in the standard deviations $\sigma_1(t)$ and $\sigma_2(t)$ of the estimated hit probabilities $\hat{P}_1$ and $\hat{P}_2$, respectively. Several peaks of $\sigma_1(t)$ and $\sigma_2(t)$ are observed when $P_1$ and $P_2$ are changed at 1000, 2000, and 3000 plays, respectively. The changes in $P_1$ and $P_2$ can be detected by comparing $\sigma_1(t)$ and $\sigma_2(t)$ with the detection criterion $\sigma_d$, indicated by the dotted line in Fig. 4(b). Therefore, the changes in $P_1$ and $P_2$ can be detected when either $\sigma_1(t)$ or $\sigma_2(t)$ exceeds $\sigma_d$.

In the proposed method, the threshold $T(t)$ of the chaotic temporal waveform is initialized when changes in $\hat{P}_1$ and $\hat{P}_2$ are detected. Figure 5(a) shows the threshold $T(t)$ of the chaotic temporal waveform for one cycle. In Fig. 5(a), the threshold is initialized to zero at approximately 1000, 2000, and 3000 plays when either $\sigma_1$ or $\sigma_2$ exceeds $\sigma_d$, as shown in Fig. 4(b). Resetting the threshold to zero is effective in initializing the decision-making process after $P_1$ and $P_2$ are changed every 1000 plays. Figure 5(b) summarizes the selected slot machine among different plays. Immediately after the change in the hit probabilities, both $S_1$ and $S_2$ are selected in different trials. Nevertheless, it should be emphasized that $S_1$ is mostly selected from 1000 to 2000 plays and from 3000 to 4000 plays, while $S_2$ is mostly selected from 0 to 1000 plays and from 2000 to 3000 plays. This implies that the slot machine selections are correct in time-varying reward environments.

Figure 6 shows CDR for different plays over 1000 cycles. The value of CDR increases every 1000
Fig. 6. Correct decision rate (CDR) for decision making using the proposed method. Time-varying hit probabilities of \((P_1, P_2) = (0.1, 0.3)\) and \((0.9, 0.7)\) are used.

Fig. 7. CDR for different detection criteria \(\sigma_d\). Red: \(\sigma_d = 0.02\), blue: \(\sigma_d = 0.07\), and black: \(\sigma_d = 0.10\). Time-varying hit probabilities of \((P_1, P_2) = (0.4, 0.6)\) or \((0.7, 0.5)\) are used.

plays after \(P_1\) and \(P_2\) are changed at 1000, 2000, and 3000 plays and reaches \(~1\) for all cases, unlike those in Fig. 3(b). The convergence of CDR to 1 indicates that correct decision making is achieved using the proposed method when \(P_1\) and \(P_2\) are changed. The correct decision can be made even though \(P_1 + P_2\) is significantly changed every 1000 plays because the step sizes \(\Delta\) and \(\Omega\) are estimated adaptively in our proposed method.

5. Parameter dependence

In this section, we consider a more challenging condition for the changes in the hit probabilities. We configure the condition of \((P_1, P_2) = (0.4, 0.6)\) or \((0.7, 0.5)\), where the difference in the hit probabilities is smaller than for the condition in the previous section.

Figure 7 shows CDRs for different parameter settings of \(\sigma_d\) under the condition of \((P_1, P_2) = (0.4, 0.6)\) or \((0.7, 0.5)\). In the case of \(\sigma_d = 0.10\) as in the previous section, CDR does not reach \(~1\) between 1000 and 2000 plays and between 3000 and 4000 plays, as shown by the black curve in Fig. 7. The actual changes in the hit probabilities are not well detected in this case. However, when \(\sigma_d\) decreases to 0.07 (the blue curve), CDR is improved and nearly reaches 1. When \(\sigma_d\) is decreased further to 0.02 (the red curve), the CDR decreases again. Therefore, the CDR curve depends on the value of \(\sigma_d\) because \(\sigma_1\) and \(\sigma_2\) change differently for different settings of \(P_1\) and \(P_2\).

In Fig. 7, the convergence value of CDR decreases as \(\sigma_d\) is decreased (e.g., CDR before 1000 plays). However, fast convergence is observed for a smaller \(\sigma_d\) after the hit probabilities are changed (e.g.,
CDR after 1000 plays). Therefore, there is a tradeoff between fast adaptation to the change in the hit probabilities and a large convergence value of CDR as $\sigma_d$ is changed.

We investigate the dependence of the decision-making performance on the parameter value of $\sigma_d$. Here, we introduce a figure of merit called the average hit rate (AHR) for $m$ plays and $n$ cycles as defined by [15, 16]

$$AHR = \frac{1}{mn} \sum_{j=1}^{n} \sum_{t=1}^{m} Y(j, t)$$  \hspace{1cm} (12)

where $Y(j, t)$ is 1 when the result of slot machine selection is hit at the $t$-th play and $j$-th cycle and 0 otherwise. AHR is the ratio of hits to the total number of plays on the cycle average. AHR of $\text{Max}(P_1, P_2)$ indicates that the maximum total reward is obtained.

Figure 8(a) shows AHR as $\sigma_d$ is varied under the condition of $(P_1, P_2) = (0.4, 0.6)$ or $(0.7, 0.5)$. AHR decreases for a larger $\sigma_d$, whereas it saturates for a smaller $\sigma_d$. The maximum AHR of 0.6211 is obtained at $\sigma_d = 0.07$. AHR decreases for a larger $\sigma_d$ because the detection time of the change in $\sigma_i$ is delayed. Conversely, AHR is slightly decreased and almost constant for a smaller $\sigma_d$. As shown in Fig. 7, the convergence value of CDR is decreased for a small $\sigma_d$, however, convergence is achieved faster after the hit probabilities are changed. Therefore, AHR is almost constant for a small $\sigma_d$.

We also change the window size $W$ used to calculate $\sigma_1(t)$ and $\sigma_2(t)$. Figure 8(b) shows AHR as $W$ is varied. The maximum AHR of 0.6053 is obtained at $W = 100$. AHR decreases significantly for a smaller $W$ because $\sigma_i$ is changed with large fluctuations and the detection of the change in $\sigma_i$ is difficult. As $W$ increases, the change in $\sigma_i$ becomes smoother and can be detected correctly. In contrast, AHR decreases slightly for a larger $W$ because the detection time of the change in $\sigma_i$ is delayed.

The parameter values of $\sigma_d$ and $W$ need to be optimized to maximize AHR for more difficult problems with smaller differences in the hit probabilities. For example, the window size $W$ needs to be optimized so that the change in $\sigma_i$ can be detected as the hit probabilities are changed. In a real-world situation, the hit probability may change randomly. To optimize $\sigma_d$ for random changes in the hit probabilities, we first need to adjust the window size $W$ to detect the change in $\sigma_i$, and then we set an appropriate $\sigma_d$.

We select the optimal value of the memory parameter of the estimated hit probabilities $\beta = 0.960$ for $(P_1, P_2) = (0.1, 0.3)$ or $(0.7, 0.9)$. The optimal value of $\beta$ may be changed for different settings of the hit probabilities. However, the value of $\beta$ is not sensitive to the hit probabilities in our parameter conditions. The value of $\beta$ is more critical if the changes in the hit probabilities occur more frequently.
6. Conclusions
In this paper, we proposed a novel method of adaptive decision making for solving the multi-armed bandit problem using slot machines with time-varying hit probabilities. Three processes were introduced to accomplish adaptive decision making in dynamically changing reward environments: memory effect in the estimated hit probabilities, short-term standard deviation of the estimated hit probabilities for change detection, and threshold initialization of the chaotic temporal waveform for decision making. The estimated hit probabilities followed the true hit probabilities appropriately, and changes in the hit probabilities were successfully detected by measuring the short-term standard deviations of the estimated hit probabilities. The threshold of the chaotic temporal waveform was initialized when changes in hit probabilities were detected. Thus, adaptive decision making was achieved using the proposed principle. We also examined the dependence of decision-making performance on the parameter settings. Our approach paves the way for photonic ultrafast decision making in dynamic environments, such as cognitive wireless communications and robot control using reinforcement learning.

Acknowledgments
We acknowledge the support of the Japan Society for the Promotion of Science (JP19H00868, JP20K15185, and JP20H00233), JST CREST (JPMJCR17N2), and the Telecommunications Advancement Foundation.

References
[1] H. Robbins, “Some aspects of the sequential design of experiments,” Bulletin of the American Mathematical Society, vol. 58, no. 5, pp. 527–536, 1952.
[2] R.S. Sutton, and A.G. Barto, Reinforcement Learning: An Introduction, The MIT press, Cambridge, 1998.
[3] D. Silver, A. Huang, C.J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lancotot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, “Mastering the game of Go with deep neural networks and tree search,” Nature, vol. 529, no. 7587, pp. 484–489, 2016.
[4] K. Kuroda, H. Kato, S.J. Kim, M. Naruse, and M. Hasegawa, “Improving throughput using multi-armed bandit algorithm for wireless LANs,” NOLTA, vol. 9, no. 1, pp. 74–81, 2018.
[5] S. Takeuchi, M. Hasegawa, K. Kanno, A. Uchida, N. Chauvet, and M. Naruse, “Dynamic channel selection in wireless communications via a multi-armed bandit algorithm using laser chaos time series,” Scientific Reports, vol. 10, p. 1574, 2020.
[6] O.B. Krömer, R. Detry, J. Piater, and J. Peters, “Combining active learning and reactive control for robot grasping,” Robotics and Autonomous Systems, vol. 58, no. 9, pp. 1105–1116, 2010.
[7] S.J. Kim, M. Aono, and M. Hara, “Tug-of-war model for the two-bandit problem: Nonlocally-correlated parallel exploration via resource conservation,” BioSystems, vol. 101, no. 1, pp. 29–36, 2010.
[8] S.J. Kim and M. Aono, “Amoeba-inspired algorithm for cognitive medium access,” NOLTA, vol. 5, no. 2, pp. 198–209, 2014.
[9] S.J. Kim, M. Aono, and E. Nameda, “Efficient decision making by volume-conserving physical object,” New Journal of Physics, vol. 17, no. 8, p. 083023, 2015.
[10] T. Nakagaki, H. Yamada, and Á. Tóth, “Maze-solving by an amoeboid organism,” Nature, vol. 407, no. 68031, p. 470, 2000.
[11] S.-J. Kim, M. Naruse, M. Aono, M. Ohtsu, and M. Hara, “Decision maker based on nanoscale photo-excitation transfer,” Scientific Reports, vol. 3, p. 3297, 2013.
[12] M. Naruse, W. Nomura, M. Aono, M. Ohtsu, Y. Sonnefraud, A. Drezet, S. Huant, and S.J. Kim, “Decision making based on optical excitation transfer via near-field interactions between quantum dots,” Journal of Applied Physics, vol. 116, no. 15, p. 154303, 2014.
[13] M. Naruse, M. Berthel, A. Drezet, S. Huant, M. Aono, H. Hori, and S.-J. Kim, “Single-photon decision maker,” Scientific Reports, vol. 5, p. 13253, 2015.
[14] N. Chauvet, D. Jegouso, B. Boulanger, H. Saigo, K. Okamura, H. Hori, A. Drezet, S. Huant, G. Bachelier, M. Naruse, “Entangled-photon decision maker,” Scientific Reports, vol. 9, p. 12229, 2019.
[15] M. Naruse, Y. Terashima, A. Uchida, and S.J. Kim, “Ultrafast photonic reinforcement learning based on laser chaos,” Scientific Reports, vol. 7, p. 8772, 2017.
[16] T. Mihana, Y. Terashima, M. Naruse, S.J. Kim, and A. Uchida, “Memory effect on adaptive decision making with a chaotic semiconductor laser,” Complexity, vol. 2018, p. 4318127, 2018.
[17] M. Naruse, T. Mihana, H. Hori, H. Saigo, K. Okamura, M. Hasegawa, and A. Uchida, “Scalable photonic reinforcement learning by time-division multiplexing of laser chaos,” Scientific Reports, vol. 8, p. 10890, 2018.
[18] T. Mihana, Y. Mitsui, M. Takabayashi, K. Kanno, S. Sunada, M. Naruse, and A. Uchida, “Decision making for the multi-armed bandit problem using lag synchronization of chaos in mutually coupled semiconductor lasers,” Optical Express, vol. 27, no. 19, pp. 26989–27008, 2019.
[19] T. Mihana, K. Fujii, K. Kanno, M. Naruse, and A. Uchida, “Laser network decision making by lag synchronization of chaos in a ring configuration,” Optical Express, vol. 28, no. 26, pp. 40112–40130, 2020.
[20] Y. Han, S. Xiang, Y. Wang, Y. Ma, B. Wang, A. Wen, and Y. Hao, “Generation of multi-channel chaotic signals with time delay signature concealment and ultrafast photonic decision making based on a globally coupled semiconductor laser network,” Photonics Research, vol. 8, no. 11, pp. 1792–1799, 2020.
[21] R. Homma, S. Kochi, T. Niiyama, T. Mihana, Y. Mitsui, K. Kanno, A. Uchida, M. Naruse, and S. Sunada, “On-chip photonic decision-maker using spontaneous mode switching in a ring laser,” Scientific Reports, vol. 9, p. 9429, 2019.
[22] A. Oda, T. Mihana, K. Kanno, M. Naruse, and A. Uchida, “Decision making for multi-armed bandit problem with variable hit probabilities using chaotic laser signals,” Proc. NOLTA’19, pp. 473–476, December 2019.
[23] A. Uchida, K. Kanno, S. Sunada, and M. Naruse, “Reservoir computing and decision making using laser dynamics for photonic accelerator,” Japanese Journal of Applied Physics, vol. 59, no. 4, p. 040601, 2020.
[24] A. Uchida, Optical Communication with Chaotic Lasers, Applications of Nonlinear Dynamics and Synchronization, Wiley-VCH, Weinheim, 2012.