

Basic Learning Characteristics of Digital Spike Maps

Takashi Ogawa[†] and Toshimichi Saito[†]

[†]EE Dept., Hosei Univ., Tokyo 8584, Japan

Abstract—This paper studies learning algorithm of the digital spike maps. The map is equivalent to a simple one-dimensional cellular automaton and can generate various digital spike-trains. In order to approximate a class of spike-trains, we present a learning algorithm with self-organizing function. Performing a basic numerical experiment, we have clarified that the map can learn a typical class of teacher signals. The results contribute to bridge between spiking neural systems and digital dynamical systems.

1. Introduction

Spiking signals play important roles in a variety of neural systems [1]-[9]. In biological/artificial neural systems, roughly speaking, spiking signals are generated by integrate-and-fire dynamics. Analysis of them is basic to understand information processing function in the brain [2] [3]. Such neural systems can exhibit chaos, synchronization and related rich bifurcation [4]-[8]. Analysis of these phenomena is recognized as meaningful. The spiking signals are simple, low power and suitable for various engineering applications including image segmentation [9], A/D converters [10], UWB communication [11] [12] and neural prosthesis [13].

Inspired by such neural systems, a variety of spiking neurons have been presented [2] [7]. The digital spiking neuron (DN) is one of them [14] [15]. The DN is constructed by coupling plural shift registers and can exhibit rich digital spike-trains. Adjusting the wiring pattern between plural shift registers, the DN can learn (approximate) a class of teacher spiking signals.

This paper studies the digital spike map (DSM) and its learning algorithm. The DSM can be regarded as a simple class of cellular automata (CAs, [16] [17]) and can output various spike-trains. The learning algorithm is simple and includes a self-organizing function. Although there exists various candidates of teacher spike signals, we use the teacher signal constructed by the Izhikevich neuron [2]. The Izhikevich neuron is a dynamical systems inspired by biological spiking neuron. Performing a basic numerical experiment, we have clarified that the DSM can learn typical teacher signals of the Izhikevich neuron. A basic method is used to measure distance among spike-trains [14]. Please note the following for novelty and significance.

1. The DSM is a simple model to describe digital spike-trains and can be regarded as a variant of cellular automaton

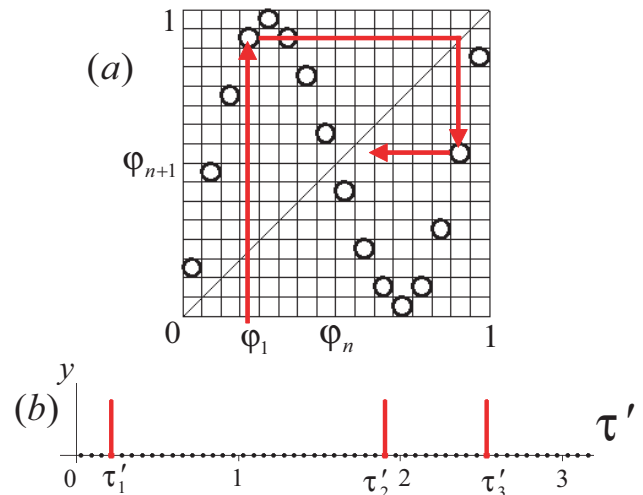


Figure 1: An example of the digital spike map (DSM) (a) and corresponding the digital spike-train (DST) (b) where $\varphi_n = \tau'_n \bmod 1$.

(CA) with rich dynamics/applications [16] [17]. The DSM can bridge between spiking neural systems and CA.

2. The DSM learning is simple, successive and self-organizing. It is different from the DN learning that is based on change of the wiring pattern and the genetic algorithm [14].

2. Digital Spike Map

In this section, we define the DSM. The domain I_D of the DSM is a set of lattice points in the unit circle $I = [0, 1)$,

$$I_D \equiv \{\alpha_1, \alpha_2, \dots, \alpha_M\}, \alpha_i = \frac{2i-1}{2M} \quad (1)$$

where $i = 1 \sim M$ and M is the number of the lattice points. The i -th lattice point α_i is the center of the i -th subinterval:

$$\Delta_i \equiv \left[\frac{i-1}{M}, \frac{i}{M} \right) \quad (2)$$

The DSM is a discrete map from I_D to itself:

$$\varphi_{n+1} = Q(\varphi_n), \varphi_n \in I_D \quad (3)$$

where φ_n denotes the n -th digital spike phase where n is a positive integer. For an initial value φ_1 , the DSM outputs a

sequence of lattice points

$$\{\varphi_1, \varphi_2, \dots, \varphi_N\}.$$

This phase sequence corresponds to the spike-train of N spike positions

$$\{\tau'_1, \tau'_2, \dots, \tau'_N\} = \{\varphi_1, \varphi_2 + 1, \dots, \varphi_N + (N - 1)\} \quad (4)$$

where $\tau'_k \in [k - 1, k)$. φ_k satisfies the condition $\varphi_k = \tau'_k \bmod 1$. The spike-train is described by

$$y(\tau) = \begin{cases} 1 & \text{for } \tau = \tau'_n \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where $n = 1 \sim N$. We refer to $y(\tau)$ as a digital spike-train (DST) hereafter. Fig.1 shows an example of the DSM and corresponding DST.

3. Learning algorithm

We introduce the learning algorithm. First, the teacher signal is a sequence of digital spike phases $\{\theta'_1, \dots, \theta'_N\}$, $\theta'_n \in I_D$, corresponding to a spike-train in Eq. (4).

In the learning a pair of the phases is necessary. We define the pairs are presented successively: let the first pair of the digital spike phases (θ'_1, θ'_2) be presented at step $s = 1$ and let the s -th pairs $(\theta'_s, \theta'_{s+1})$ be presented at step s .

Step 1: Let $s = 0$. $Q(a)$ is initialized by

$$Q(a_i) = \frac{2i - 1}{2M}, \quad i = 1 \sim M. \quad (6)$$

Step 2 (update of the teacher signal.): The s -th pair of the teacher signal $(\theta'_s, \theta'_{s+1})$ is presented. Then the DSM of a_s is updated as shown in Fig.2, black circles:

$$Q(a_s) = a_{s+1}. \quad (7)$$

We refer to the black circle as "winner". This output is permanent and can not change afterward. It causes a restriction for the teacher signals: for any phase $a_i \in I_D$, the next phase a_{i+1} is given uniquely.

Step 3 (Update of set of lattice points.): We refer to the winner before step s as "past winner". Let N_r (respectively, N_l) be the sets of lattice points between the winner a_s and the right-closest past winner a_r (respectively, the left-closest past winner a_l). For the lattice points,

$$Q(a) = \begin{cases} F_r(a) & \text{for } a \in N_r \\ F_l(a) & \text{for } a \in N_l \end{cases} \quad (8)$$

where F_r (respectively, F_l) implies the linear interpolation between a_s and a_r (respectively, a_l). This update is temporal and $Q(a)$ can change if some teacher signal is applied to the position of a .

Note that this interpolation relates to self-organizing and the DSM can learn by insufficient teacher signals for $s <$

M . Examples are shown in Fig.2 (c), right side past winner a_r and left side past winner a_l are shown.

Step 4: Let $s = s + 1$. Go to **Step 2** and repeat until the maximum step limit $s = N$.

Figs.2 and 3 show shapes of the DSM and DST in the learning process where the teacher signal is generated by DSM in Fig 1. We can see that the DST tends to mimic the teacher signal as the learning step s increases.

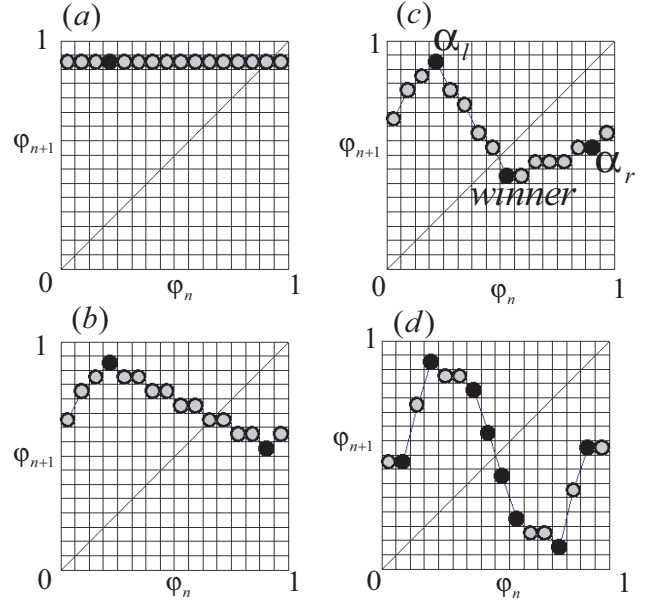


Figure 2: Learning process and the DSM for $M = 16$. The black circles denote learned points "winner" and grey circles denote interpolation points. (a) $s=1$, (b) $s=2$, (c) $s=3$, a_r and a_l denote the right side past winner and the left side past winner, respectively. (d) $s=15$.

4. Numerical experiments

In order to evaluate the algorithm efficiency, we have performed a basic numerical experiment. The teacher signal is constructed by the spike-train of the Izhikevich neuron whose dynamics is described by

$$\begin{aligned} \dot{v} &= 0.04v^2 + 5v + 140 - u + I \\ \dot{u} &= a(bv - u) \end{aligned} \quad (9)$$

with the auxiliary after-spike resetting

$$\text{if } v = 30 \text{ mV, then } \begin{cases} v \leftarrow c \\ u \leftarrow u + d \end{cases} \quad (10)$$

where, v and u are dimensionless variables. After trial-and-errors, we fix the dimensionless parameters $a = 0.1, b = 0.2, c = -53, d = 4$ and $I = 10$. Fig. 5 (a) shows an example of spike-trains: this is between "Fast Spiking (FS)"

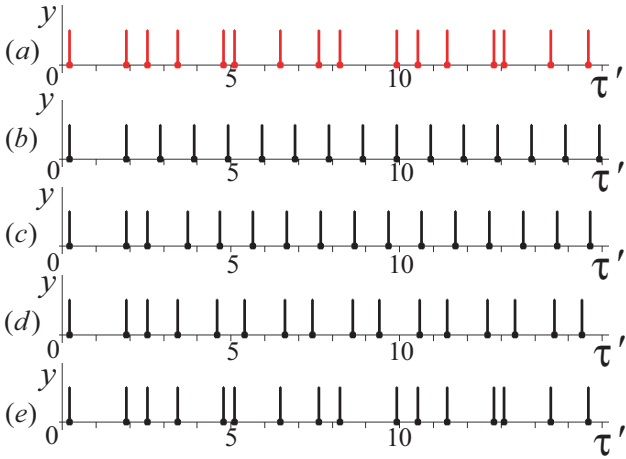


Figure 3: Learning process and the DST. The teacher signal is from DSM for $M=16$. (a) teacher signal is from DSM in Fig.1, (b) $s=1$, (c) $s=2$, (d) $s=3$, (e) $s=15$.

and "Intrinsically Bursting (IB)". The neuron fires and a spike is generated if $v \geq 30$. Let p_n be the n -th spike position, and let Δp_n be the n -th inter-spike-interval (ISI): $\Delta p_n = p_{n+1} - p_n$ where $n = 1 \sim N - 1$. We use average of *ISI* of the teacher signal: ISI_{ave} for normalization of position. The teacher signal is normalized as $\tau_n = p_n / ISI_{ave}$ and the phase is extracted $\theta_n = \tau_n \pmod{1}$ where time-axis is adjusted to satisfy $\tau_1 \in [0, 1)$. In order to make the teacher signal, we quantize the spike phase: a sequence of spike phases: $\{\theta_1, \dots, \theta_N\}$ is converted to a digital spike phases: $\{\theta'_1, \dots, \theta'_N\}$: if the n -th teacher spike phase θ_n is included in the i -th subinterval Δ_i , the n -th teacher spike phase is $\theta'_n = \frac{2i-1}{2M}$. As mentioned in the algorithm, the s -th pair of spike-phases is presented at learning step s :

$$(\theta'_s, \theta'_{s+1}), s = 1 \sim N - 1 \quad (11)$$

For simplicity, we consider a DST of 16 ($= N$) spikes for $0 < \tau < N$. Let φ_n denote the n -th spike phase of the DST $y(\tau)$. The distance between the DST and the teacher signal is measured by

$$S_{TD} = \frac{1}{M} \sum_{n=1}^M |\theta'_n - \varphi_n| \quad (12)$$

Note that $S_{TD}=0$ if $\theta'_n = \varphi_n$ for all n .

Figs.4 and 5 show shapes of the DSM and DST in the learning process where the teacher signal is generated by the Izhikevich neuron. Fig.6 shows the approximation characteristic. Note that, even if the leaning is not terminated for $s < 15$, we have used the DSM in the learning process at step s to generate a DST of 16 spikes. We have measured the distance between the teacher signal and DST. In Fig.6, we can see that the closest distance between teacher signal and DST decreases as s increases. For $s > 9$, the distance seems to converge to a small value:

$s = 9$ seems to be sufficient for the DSM to approximate the teacher signal. Our algorithm is applicable to many other systems including the BN [19].

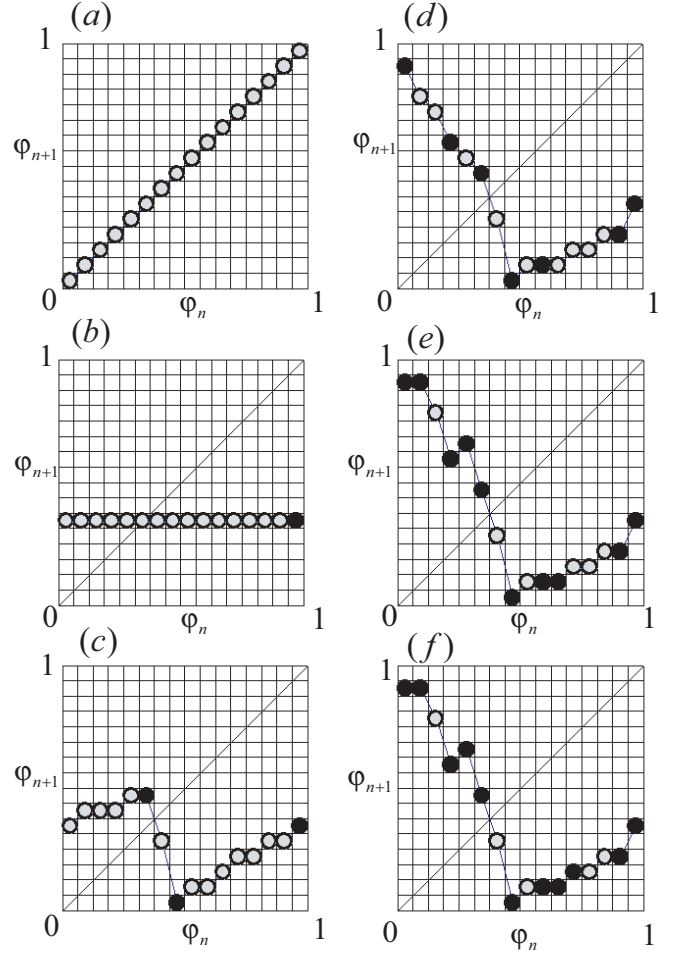


Figure 4: Learning process and the DSM for $M = 16$. The teacher signal is constructed by the Izhikevich neuron for $a = 0.1, b = 0.2, c = -53, d = 4, I = 10$. (a) $s=0$, (b) $s=1$, (c) $s=3$, (d) $s=7$. (e) $s=11$, (f) $s=15$.

5. Conclusions

DSM and its basic learning algorithm are studied in this paper. The learning algorithm is simple and includes a self-organizing function. Performing basic numerical experiments, we have confirmed the DSM can approximate a typical class of the Izhikevich neuron even if the number of spikes is not sufficient.

Future problems are many including

- (1) detailed analysis of learning process,
- (2) learning wider class of spike-trains,
- (3) relation between DSM and CA,
- (4) engineering application and
- (5) building hardware.

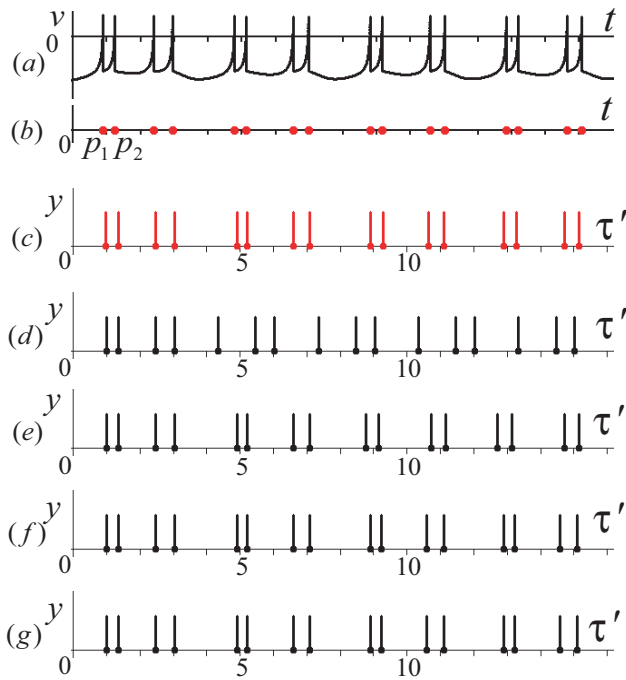


Figure 5: Learning process and the spike-trains. The teacher signal is constructed by the Izhikevich neuron (a) teacher signal is constructed by the Izhikevich neuron, (b) analog spike position of the Izhikevich neuron (a), (c) digital teacher signal is based on (b), (d) $s=3$, the distance between DST and the teacher signal is $S_{TD} = 0.357$, (e) $s=7$, $S_{TD} = 0.177$, (f) $s=11$, $S_{TD} = 0.072$, (g) $s=15$, $S_{TD} = 0.021$.

References

- [1] E. M. Izhikevich, Weakly Pulse-coupled oscillators, FM Interactions, Synchronization, and oscillatory associative memory, IEEE Trans. Neural Networks, 10, 3, pp. 508-526, 1999.
- [2] E. M. Izhikevich, Simple Model of Spiking Neurons, IEEE Trans. Neural Networks, 14, 6, pp. 1569-1572, 2003.
- [3] T. Toyozumi, K. Aihara and S. Amari, Fisher information for spike-based population coding, Phys. Rev. Lett., 97, 098102, 2006.
- [4] R. Perez and L. Glass, Bistability, period doubling bifurcations and chaos in a periodically forced oscillator, Phys. Lett., 90A, 9, pp. 441-443, 1982.
- [5] G. Lee and N. H. Farhat, The bifurcating neuron network 1, Neural networks, 14, pp. 115-131, 2001.
- [6] E. D. M. Hernandez, G. Lee and N. H. Farhat, Analog Realization of Arbitrary One-Dimensional Maps. IEEE Trans. Circuits Syst. I, 50, 12, pp. 1538-1547, 2003.
- [7] Y. Kon'no, T. Saito and H. Torikai, Rich dynamics of pulse-coupled spiking neurons with a triangular base signal, Neural Networks, 18, pp. 523-531, 2005.
- [8] A. Patel and B. Kosko, Stochastic resonance in continuous and spiking neuron models with Levy noise, IEEE Trans. Neural Networks, 19, 12, pp. 1993-2008, 2008.
- [9] S. R. Campbell, D. Wang and C. Jayaprakash, Synchrony and desynchrony in integrate-and-fire oscillators, Neural computation, vol. 11, pp. 1595-1619, 1999.

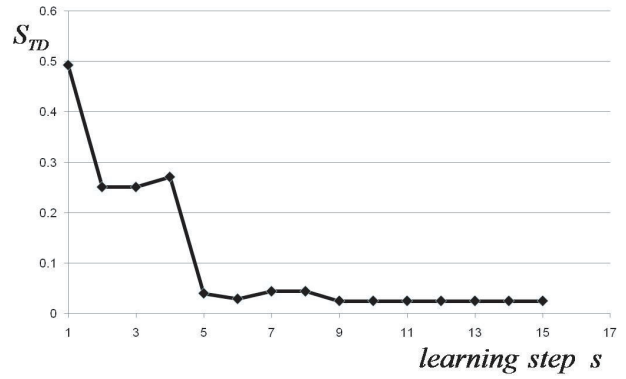


Figure 6: Approximation characteristics for learning step s . The plot mean the distance between the teacher signal is constructed by the Izhikevich neuron and DST.

- [10] H. Hamanaka, H. Torikai and T. Saito, Quantized spiking neuron with A/D conversion functions, IEEE Trans. Circuits Syst. II, 53, 10, pp. 1049-1053, 2006.
- [11] H. Torikai, T. Saito and W. Schwarz, Synchronization via multiplex pulse-train, IEEE Trans. Circuits Syst. I, 46, 9, pp.1072-1085, 1999.
- [12] M. Sushchik, N. Rulkov, L. Larson, L. Tsimring, H. Abarbanel, K. Yao, and A. Volkovskii, Chaotic pulse position modulation: a robust method of communicating with chaos, IEEE Comm. Lett., 4, pp. 128-130, 2000.
- [13] S. Hashimoto, T. Hishiki and H. Torikai, Approximation of neuron dynamics by discrete state dynamics: an approach toward reconfigurable hardware based neural prosthesis, Proc. NOLTA, pp. 254-257, 2009
- [14] H. Torikai, A. Funew, T. Saito, Digital spiking neuron and its learning for approximation of various spike-trains, Neural Networks 21, 2-3, 140-149, 2008.
- [15] H. Torikai, T. Nishigami, An artificial chaotic spiking neuron inspired by spiral ganglion cell: Paralleled spiking encoding, theoretical analysis, and electronic circuit implementation, Neural Networks 22, 5-6, 164-173, 2009.
- [16] L. O. Chua, S. Yoon and R. Dogaru, A nonlinear dynamics perspective of Wolfram's new kind of science. Part I: Threshold of complexity, Int'l J. Bifurcation and Chaos, 12, 12, 2655-2766, 2002.
- [17] M. Wada, J. Kuroiwa and S. Nara, Errorless reproduction of given pattern dynamics by means of cellular automata, Phys. Rev. E 68,036707, pp. 1-8, 2003.
- [18] H. Hamanaka, H. Torikai and T. Saito, Analysis of composite dynamics of two bifurcating neurons, IEICE Trans. Fundamentals, E88-A, 2, pp. 561-567, 2005.
- [19] T. Ogawa and T. Saito, Digital Spike Maps and Learning of Spike Signals, Proc. of IEEE-INNS/JCNN, 2010