Paper

A pulse-width-modulation mode CMOS integrated circuit implementation of threshold-coupled map

*Seiji Uenohara^{1,2a)}, Takashi Morie^{1b)}, Hakaru Tamukoh¹,
and Kazuyuki Aihara²*

¹ Graduate School of Life Science and Systems, Kyushu Institute of Technology
2-4 Hibikino Wakamatsu-ku, Kitakyushu 808-0196, Japan

² Institute of Industrial Science, The University of Tokyo
4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan

^{a)} *s.uenohara@gmail.com*

^{b)} *morie@brain.kyutech.ac.jp*

Received July 23, 2017; Revised December 5, 2017; Published April 1, 2018

Abstract: We propose a coupled-map-lattice complementary-metal-oxide-semiconductor very-large-scale-integration (CMOS VLSI) circuit based on the threshold-coupled map (TCM) that has been proposed previously as a unidirectional connected network model exhibiting different spatiotemporal patterns according to its underlying nonlinear map and update scheme. We introduce mutual connections and arbitrarily valued connection weights into the TCM to realize cellular automata. In this study, we design, fabricate, and evaluate a CMOS integrated circuit with which to implement this extended TCM (ETCM). The ETCM is a universal Turing machine as confirmed in circuit experiments using the fabricated circuit, which can achieve Rule110 of a one-dimensional cellular automaton.

Key Words: coupled map lattice, CMOS integrated circuit, cellular automata, universal Turing machine, spatiotemporal pattern

1. Introduction

Among previously proposed coupled-map lattices (CMLs) [1–4], the threshold-coupled map (TCM) [5] is a simple model that exhibits various spatiotemporal patterns. Generating such patterns from simple dynamics is a useful way to consider the applications of such a system.

A supervised learning model constructed by a recurrent neural network with fixed weights and a classifier is known as a “reservoir computing (RC) model” [6, 7]. In an RC model, a recurrent neural network known as a “reservoir” memorizes the short-term history of the input signal as spatiotemporal patterns generated from the reservoir driven by the input. The first RC model was echo state networks [6], which use a random recurrent neural network as a reservoir. Other reservoir models have since been proposed that use a ring network [8] or cellular automata (CA) [9] as the reservoir.

We have previously proposed circuits for implementing CMLs in the form of voltage- and current-

waveform-sampling-mode circuits that can achieve arbitrary analog nonlinear dynamics in the time domain using pulse-width/phase modulation [10, 11]. Moreover, we have proposed a core circuit for a CML that is robust to parameter mismatches and variations, and fabricated it using complementary-metal-oxide-semiconductor (CMOS) technology [12, 13].

In this paper, we propose a CMOS array circuit constructed using the core circuits for a CML, and show experimentally that it can realize a TCM. We also show experimentally that a TCM with mutual connections and arbitrarily valued connection weights can realize a one-dimensional CA (1D-CA).

This paper is organized as follows. In Section 2, we explain the TCM model and how to implement 1D-CA using a TCM. In Section 3, we show a core circuit for a TCM and explain its operation. A CMOS array circuit very-large-scale-integration (VLSI) chip that consists of the core circuits is shown in Section 4. We present measurement results of a fabricated circuit in Section 5, and we conclude the paper in Section 6.

2. Threshold-coupled map and its extension

The TCM was proposed as a coupled-map model as shown in Fig. 1(a). This model performs the following nonlinear transformation:

$$x_i(n) = f_{non}(x_i(n-1)), \quad (1)$$

where i is the site index in the lattice, n is the discrete time, $x_i(n)$ is the state variable of the i th lattice site at time step n , and $f_{non}(\cdot)$ is the nonlinear transformation function, such as a logistic map or a circle map [14]. An excess $\delta_i(n) (= x_i(n) - x_{th})$ is then transported to the neighboring lattice site(s) if $x_i(n)$ is larger than a threshold value x_{th} , and $x_i(n)$ is reset to x_{th} . This process is expressed by

$$\begin{aligned} x_i(n) &\rightarrow x_{th}, \\ x_{i+1}(n) &\rightarrow x_{i+1}(n) + \delta_i(n). \end{aligned} \quad (2)$$

Equation (2) is referred to as a “relaxation.” In the relaxation process, no nonlinear transformation is performed and only $\delta_i(n)$ is transported. The time step n does not increase in the relaxation process. This model can use several updating schemes. For example, multiple relaxation process are performed for one iteration of nonlinear transformation. In this scheme, various spatiotemporal patterns are observed that depending on the number l of relaxation operations [15, 16].

In the TCM, $\delta_i(n)$ is weighted by the reciprocal of the number of neighboring cells. We introduce here an arbitrary value g for the connection weight, and mutual connections into the TCM (see Fig. 1(b)). We refer to this new TCM as the “expanded threshold-coupled map (ETCM).” The nonlinear transformation of the ETCM is expressed as follows:

$$x_i(n) = f_{non}(x_i(n-1)). \quad (3)$$

Fig. 1. (a) Unidirectional connections and (b) mutual connections.

Relaxation of the ETCM is expressed by

$$\begin{aligned} x_i(n) &\rightarrow x_{th} + \delta_{sum,i}(n) & \text{if } x_i(n) > x_{th}, \\ x_i(n) &\rightarrow x_i(n) + \delta_{sum,i}(n) & \text{if } x_i(n) \leq x_{th}, \end{aligned} \quad (4)$$

where $\delta_{sum,i}(n)$ is expressed by

$$\delta_{sum,i}(n) = g_R \delta_{i+1}(n) + g_L \delta_{i-1}(n). \quad (5)$$

The coefficients g_R and g_L are the connection weights from the $(i+1)$ th and $(i-1)$ th lattice sites, respectively, to the i th lattice site. The ETCM can realize a CA if we choose $f(\cdot)$, g_R , g_L , and x_{th} appropriately.

2.1 Implementation of a one-dimensional cellular automaton

A CA was proposed as a self-reproducing model of life by von Neumann et al [17]. Each CA consists of a regular grid of cells that is updated according to rules depending on the relations between the state of a cell and those of its neighbors. Although various CA models have been proposed, in this paper we discuss binary synchronous CAs.

A cell in a one-dimensional CA (1D-CA) has a binary state, namely either 0 or 1. Cell-update rules are defined by combining the states of a cell and its neighbors. There are 8 ($= 2^3$) such combinations (000, 001, 010, ..., 111) in a 1D-CA. These combinations are considered to be 3-bit strings and are labeled as rules P0 to P7 in ascending order. There are 256 ($= 2^8$) rule sets, each of which consists of eight update rules (P0, P1, P2, ..., and P7) and the binary state after updating because the state of a cell updated by these eight rules at the next time step ($n+1$) is 0 or 1. This set is referred to by the resulting 8-bit binary number (read from the left) as a decimal. For example, because 01011010 is 90 as a decimal, it is referred to as Rule90.

When a 1D-CA is realized by the ETCM, we express $\delta = 1 - x_{th}$ because $x_i(n)$ after nonlinear transformation is 0 or 1 for $x_{th} > 0$, and we set l and $x_i(n)$ to one and a binary value, respectively. The 256 rule sets are achieved by selecting $f_{non}(\cdot)$, x_{th} , g_L , and g_R with $0 \leq g_R \leq 1$, and $0 \leq g_L \leq 1$.

To clarify the design procedure for rules P0, P1, P2, ..., and P7 using the ETCM, we formulate $x_i(n)$ after relaxation and nonlinear transformation. For P0, δ_{i-1} , δ_{i+1} , and $x_i(n)$ after relaxation are zero. By using this condition, $x_i(n)$ after relaxation and nonlinear transformation is expressed by

$$P0 : f_{non}(x_i(n) + g_L \delta_{i-1} + g_R \delta_{i+1}) = f_{non}(0 + g_L \cdot 0 + g_R \cdot 0) = f_{non}(0).$$

For P1, δ_{i-1} and $x_i(n)$ after relaxation are zero. By using this condition, $x_i(n)$ after relaxation and nonlinear transformation is expressed in a similar manner for P0 as follows:

$$P1 : f_{non}(0 + g_L \cdot 0 + g_R \cdot \delta) = f_{non}(g_R \delta).$$

We summarize the formulated rules P0, P1, P2, ..., P7 as follows:

$$\begin{aligned} P0 &: f_{non}(0) \rightarrow \{0, 1\}, \\ P1 &: f_{non}(g_R \delta) \rightarrow \{0, 1\}, \\ P2 &: f_{non}(x_{th}) \rightarrow \{0, 1\}, \\ P3 &: f_{non}(x_{th} + g_R \delta) \rightarrow \{0, 1\}, \\ P4 &: f_{non}(g_L \delta) \rightarrow \{0, 1\}, \\ P5 &: f_{non}(g_L \delta + g_R \delta) \rightarrow \{0, 1\}, \\ P6 &: f_{non}(x_{th} + g_L \delta) \rightarrow \{0, 1\}, \\ P7 &: f_{non}(x_{th} + g_L \delta + g_R \delta) \rightarrow \{0, 1\}. \end{aligned} \quad (6)$$

The value of $x_i(n)$ after relaxation and nonlinear transformation is set independently by selecting $f_{non}(\cdot)$, $x_{th}(= 1 - \delta)$, g_L , and g_R as shown in Eq. (6). Therefore, the ETCM realizes all the rules by using the network shown in Fig. 1. We show the procedures for achieving Rule90 and Rule110 below as examples. The ETCM can realize not only 1-D CA but also more complicated CAs such as 2-D CA by using more connection weights and/or more complicated nonlinear function.

Fig. 2. An example of $f_{non}(x(n))$ for achieving Rule90.

2.1.1 Rule90

Rule90 shows interesting spatiotemporal patterns such as the Sierpinski gasket. According to Eq. (6), this rule is expressed as follows:

$$\begin{aligned}
 P0 &: f_{non}(0) \rightarrow 0, \\
 P1 &: f_{non}(g_R\delta) \rightarrow 1, \\
 P2 &: f_{non}(x_{th}) \rightarrow 0, \\
 P3 &: f_{non}(x_{th} + g_R\delta) \rightarrow 1, \\
 P4 &: f_{non}(g_L\delta) \rightarrow 1, \\
 P5 &: f_{non}(g_L\delta + g_R\delta) \rightarrow 0, \\
 P6 &: f_{non}(x_{th} + g_L\delta) \rightarrow 1, \\
 P7 &: f_{non}(x_{th} + g_L\delta + g_R\delta) \rightarrow 0.
 \end{aligned} \tag{7}$$

We consider the condition of g_R and g_L satisfying Eq. (7). If $g_R = 0$, P2 and P3, P6 and P7, and P2 and P5 are mutual contradictions. Also, P0 and P4, P2 and P6, and P3 and P7 are mutual contradictions when $g_L = 0$. Hence, we obtain $g_R \neq 0$ and $g_L \neq 0$ from these the conditions, which requires a network with mutual coupling. There is no contradiction when $g_R = g_L$, which can realize Rule90. It is noted that Rule90 is achieved when the connection is a mutual coupling ($g_R \neq g_L$). Figure 2 shows an example of $f_{non}(\cdot)$ for achieving Rule90 when $g_L < g_R$.

2.1.2 Rule110

Rule110 is a model known to having Turing completeness [18]. According to Eq. (6), this rule is given by

$$\begin{aligned}
 P0 &: f_{non}(0) \rightarrow 0, \\
 P1 &: f_{non}(g_R\delta) \rightarrow 1, \\
 P2 &: f_{non}(x_{th}) \rightarrow 1, \\
 P3 &: f_{non}(x_{th} + g_R\delta) \rightarrow 1, \\
 P4 &: f_{non}(g_L\delta) \rightarrow 0, \\
 P5 &: f_{non}(g_L\delta + g_R\delta) \rightarrow 1, \\
 P6 &: f_{non}(x_{th} + g_L\delta) \rightarrow 1, \\
 P7 &: f_{non}(x_{th} + g_L\delta + g_R\delta) \rightarrow 0.
 \end{aligned} \tag{8}$$

Fig. 3. An example of $f_{non}(x(n))$ for achieving Rule110.

We consider the condition of g_R and g_L satisfying Eq. (8) in a similar manner to that for Rule90. Rules P2 and P3, and P6 and P7, are contradict when $g_R = 0$. Also, P0 and P4 are contradictory when $g_L = 0$. Therefore, g_R and g_L should be non-zero. Moreover, $g_R \neq g_L$ because P1 and P4 are contradictory when $g_R = g_L = g$. Figure 3 shows an example of $f_{non}(\cdot)$ for achieving Rule110 when $g_L < g_R$.

3. CMOS core circuit for extended threshold-coupled-map

A CMOS core circuit for the ETCM is shown in Fig. 4 [12, 13]. The core circuit consists of capacitors, source-follower-type analog buffers, a switched current source (SCS), a comparator (CMP), and logic gates. The core circuit holds $x_i(n)$ and δ as the node voltages of P_x and P_δ , namely V_x and V_δ , respectively. The weight g is implemented by PMOS transistor M_δ biased by V_{cc} ; the value of g is varied by adjusting V_{cc} . By using this core circuit to design the array circuit, we assume that the coupling can be set freely in the four neighbors. Therefore, the single core has four SCSs and gate bias voltages.

The core circuit expresses variables using the voltage and the time domain. The nonlinear transformation is executed by sampling and holding a nonlinear voltage waveform $V_{non}(t)$ to C_x using a pulse-width modulation (PWM) signal S_{out} with pulse width $T_{out}(=\propto V_x)$. The PWM signal S_{out} is generated by comparing V_x with a referenced ramp voltage $V_{rmp}(t)$. The threshold processing in relaxation is executed by comparing T_{out} with T_{th} in the time domain using a logic circuit, where T_{th} is the pulse width of the threshold signal S_{th} . In the relaxation operation, we use $V_{rmp}(t)$ as $V_{non}(t)$. We define $V_{non_bt0}(t)$ and $V_{rmp_bt0}(t)$ as the base voltages of $V_{non}(t)$ and $V_{rmp}(t)$, respectively. We define V_{non_bt1} and V_{rmp_bt1} at the output stages of analog buffers SF_{non} and SF_{rmp} as the base voltages of $V_{non}(t)$ and $V_{rmp}(t)$, respectively. In the nonlinear transformation operation, S_{iniA} is kept at “ON” to fix the voltage at node P_δ to V_{iniA} , and the SCS circuit is not operated.

Figure 5 shows the timing diagram of the control signals and voltages for nonlinear transformation and relaxation. Here, we define V_{x0} as the initial output voltage of SF_x .

1. When S_{set} and S_x are turned on, nodes P_{st} and P_{cmp} are set to V_{x0} and the threshold voltage V_{inv_th} of the first-stage inverter in the CMP, respectively.
2. After S_{set} is turned off and S_{non} is turned on, node P_{st} is varied from V_{x0} to V_{non_bt} . At the same time, node P_{cmp} is varied from V_{inv_th} to $(V_{inv_th} - [V_{x0} - V_{non_bt}])$. Capacitor C_{DC} holds

achieved by comparing $(V_{inv_th} - [V_{x0} - V_{non_bt}])$ with the ramped reference voltage $V_{rmp}(t)$. The nonlinear voltage waveform $V_{non}(t)$ is sampled and the resultant voltage at C_x is S_{out} , as shown in Fig. 5.

4. Repeat steps 1 and 2. The PWM signal is generated in a similar manner to step 3 and is compared with T_{th} . S_{iniA} is turned off at the same time as the trailing edge of T_{th} . If T_{out} is larger than T_{th} , the difference signal $S_{\delta,out}$ with pulse width $T_{\delta,out} (= T_{out} - T_{th})$ is transported to the neighboring core circuits. The difference signal $S_{\delta,in}$ charges up C_δ by turning on the SCSs.

The nonlinear transformation and relaxation are executed in steps 1–3 and 4, respectively. Here, step 4 is repeated l times. In the relaxation process, a ramp voltage waveform is sampled to C_x as $V_{non}(t)$ using S_{out} . This achieves calculation of the excess and its addition to the nonlinear transformation result in single PWM transformation because the nonlinear transformation result is held in C_x before transporting the excesses. The nonlinear function can be changed by changing the nonlinear voltage waveform $V_{non}(t)$. The core circuit has a circuit that compensates for variations in current and capacitance; see [12, 13] for details.

4. CMOS array circuit constructed from 400 core circuits

4.1 VLSI chip architecture

The VLSI chip architecture is shown in Fig. 6. The core circuits in the VLSI chip share $V_{rmp}(t)$ and $V_{non}(t)$, and operate in parallel. The output signal S_{out} is transmitted outside the VLSI chip through a common bus line in the row, and is read out for each column. A shift-register circuit selects a column. Signals $sftX$ and sft_clk shown in Fig. 6 are the control and clock signals, respectively, of the shift register. The coupling can be set freely in the four neighbors. Therefore, the VLSI chip can achieve a one- or two-dimensional network as shown in Fig. 7.

The four SCSs that implement g are biased commonly by gate voltage V_{cc} that is held in capacitor C_{cc} . When mutual asymmetric coupling is required, as in realizing Rule110, it is necessary to set different gate voltages on the left and right, which is achieved by the current/capacitance-variation compensation (CCC) circuit [12, 13]. The CCC circuit not only compensates for current/capacitance variations but also sets g arbitrarily; see [12, 13] for further details about the CCC circuit.

4.2 LSI circuit design

We designed and fabricated a CMOS array circuit VLSI chip consisting of 400 ($= 20 \times 20$) core circuits

Fig. 6. VLSI chip architecture.

Fig. 7. Examples of the network topology that can be implemented with the array circuit VLSI chip: (a) 1D and (b) 2D.

Fig. 8. Circuit design: (a) layout and (b) microphotograph.

using TSMC 0.25- μm (1-Poly, 5-Metal) CMOS technology. The layout result and a microphotograph of the array circuit VLSI chip are shown in Fig. 8.

5. Measurement and evaluation results

We experimented with the TCM and ETCM by using the array VLSI chip shown in Fig. 8. We employed the logistic map as the nonlinear function for chaotic pattern generation, and achieved 1D-CA operation of Rule90 and Rule110 by using a digital map as nonlinear functions. The network topology was set to the one-dimensional mutual connections shown in Fig. 7(a) in both experiments. We set $V_{rmp}(t) = 0.414 \times 10^6 t + 1.6$ [V], and the maximum pulse width of the output PWM signal to 2,650 ns in the circuit experiments.

Fig. 9. Spatiotemporal patterns from the array circuit for threshold-coupled map.

5.1 Threshold-coupled map for logistic map

We set V_{cc} , T_{th} , and $V_{non}(t)$ to 2.35 V, 1,856 ns, and $1.385t(1-t) \times 10^6 + 0.6$ V, respectively. A resulting spatiotemporal pattern is shown in Fig. 9 for the logistic map, where the color bar represents the pulse width of S_{out} . The array VLSI chip outputs a random spatiotemporal pattern for $l = 1$, and non-random patterns as l was increased. Disappearing oblique lines and passing patterns were observed after some lines crossed.

5.2 Extended threshold-coupled map for one-dimensional cellular automaton

Figure 10 shows the digital voltage waveform for achieving Rule90 and Rule110. In these experiments, we set l and T_{th} to 1 and 1,308 ns, respectively. We experimented to achieve 1D-CA with Rule90 and Rule110 using the waveforms shown in Fig. 10.

(a)

(b)

Fig. 10. Digital voltage waveforms for achieving (a) Rule90 and (b) Rule110.

5.2.1 Rule90

We set V_{cc} to 2.6 V, which corresponds to $g_L = g_R = 0.4$, and used 400 cores. The initial voltage $V_x(0)$ of only the 200th core circuit was set by the PWM signal with a pulse width of 600 ns; the initial voltages of the other cores were set to the base voltage of $V_{non}(t)$. Figure 11 shows the spatiotemporal patterns obtained from the numerical simulation and the circuit experiment. In the latter, we observed a fractal pattern (Sierpinski gasket) similar to that obtained from the former; the pattern in panel (b) is noisy because it was not binarized.

5.2.2 Rule110

To achieve Rule110, g_L and g_R must have different values, as shown in Section 2. We set these values using the CCC circuit. The CCC circuits adjusts V_{cc} to minimize the difference in pulse widths

Fig. 11. Spatiotemporal patterns for Rule90 obtained from (a) the numerical simulation and (b) the circuit experiment.

between the target pulse width T_{tgt} and T_δ [12, 13], which is executed separately from the left capacitor $C_{cc,L}$ and the right capacitor $C_{cc,R}$. We set T_δ and the initial voltage of C_{cc} to 1,040 ns and 2.3 V, respectively. We set T_{tgt} to 308 ns or 580 ns when $C_{cc,L}$ or $C_{cc,R}$ was adjusted, respectively. Here, values of g_L and g_R correspond to 0.296 ($= 308/1040$) and 0.558 ($= 580/1040$), respectively.

In this experiment, 20 core circuits were used. The initial voltage $V_x(0)$ of only the 20th core circuit was set by the PWM signal with a pulse width of 600 ns; the initial voltages of the other cores were set to the base voltage of $V_{non}(t)$. Figure 12 shows spatiotemporal patterns obtained from the numerical simulation and the circuit experiment. The pattern obtained from the array VLSI chip is similar to that obtained from the numerical simulation. From this result, the ETCM is found to be Turing complete because Rule110 can construct a universal Turing machine.

6. Conclusions

We proposed that threshold-coupled maps with mutual connections, which are referred to as extended threshold coupled maps (ETCM), can achieve CA, and we showed how to implement a 1D-CA using this map. We designed and fabricated this circuit using TSMC 0.25- μm (1-Poly, 5-Metal) CMOS technology, and evaluated an ETCM circuit. The ETCM circuit showed distinctive spatiotemporal patterns as seen in a 1D-CA with Rule90 or Rule110. Therefore, the ETCM and its circuit are Turing complete because the 1D-CA with Rule110 is a universal Turing machine. In this experiment, we

Fig. 12. Spatiotemporal patterns for Rule110 obtained from (a) the numerical simulation and (b) the circuit experiment.

used digital waveforms as the nonlinear map to achieve 1D-CA operation, but our circuit could also use analog waveforms. It is unique that discrete dynamics and continuous dynamics are achieved by the same chip without changing its architecture. This is important for realizing hybrid dynamical systems [19]. The proposed model may stimulate ideas relating to applications involving spatiotemporal patterns. In particular, dynamically changeable hardware such as the circuit proposed in this paper should be important for nonlinear science and engineering.

The theoretical novelty in this paper is that we have developed a model that changes its dynamics from discrete to continuous models or vice versa without resetting its internal state variable at arbitrary time. We have designed discrete maps as shown in Figs. 2 and 3 for achieving Rule90 and Rule110, respectively. We can also consider these discrete maps as binarized quartic and quadratic functions. From this point of view, the class (e.g. Class 1, Class 2, Class 3, and Class 4) of a spatiotemporal pattern obtained from a coupled map lattice model might be identified by the shape of its binarized nonlinear map.

Acknowledgments

This research was partially supported by the Aihara Project, the FIRST program from JSPS initiated by CSTP, JSPS KAKENHI (No. 22240022), and NEC Corporation. It was also supported by the VLSI Design and Education Center (VDEC), the University of Tokyo in collaboration with Cadence Designs

References

- [1] K. Kaneko, "Period-doubling of kink-antikink patterns, quasiperiodicity in anti-ferro-like structures and spatial intermittency in coupled logistic lattice," *Prog. Theor. Phys.*, vol. 72, no. 3, pp. 480–486, 1984.
- [2] K. Kaneko, "Spatiotemporal intermittency in coupled map lattices," *Prog. Theor. Phys.*, vol. 74, no. 5, pp. 1033–1044, 1985.
- [3] K. Kaneko, "Chaotic but regular posi-nega switch among coded attractors by cluster-size variation," *Phys. Rev. Lett.*, vol. 63, no. 3, pp. 219–223, 1989.
- [4] T. Yanagita and K. Kaneko, "Coupled map lattice model for convection," *Phys. Lett. A*, vol. 175, no. 6, pp. 415–420, 1993.
- [5] S. Sinha, "Unidirectional adaptive dynamics," *Phys. Rev. E*, vol. 49, no. 6, pp. 4832–4842, 1994.
- [6] H. Jaeger, "The "echo state" approach to analysing and training recurrent neural networks-with an erratum note," GMD Report159, German National Research Institute for Computer Science, Tech. Rep., 2001.
- [7] W. Maass, T. Natschläger, and H. Markram, "Real-time computing without stable states: A new framework for neural computation based on perturbations," *Neural comput.*, vol. 14, no. 11, pp. 2531–2560, 2002.
- [8] L. Appeltant, G. Van der Sande, J. Danckaert, and I. Fischer, "Constructing optimized binary masks for reservoir computing with delay systems," *Scientific reports*, vol. 4, p. 3629, 2014.
- [9] Ö. Yilmaz, "Machine learning using cellular automata based feature expansion and reservoir computing," *J. Cellular Automata*, vol. 10, no. 5-6, pp. 435–472, 2015.
- [10] D. Atuti, T. Morie, and K. Aihara, "A current-sampling-mode CMOS arbitrary chaos generator circuit using pulse modulation approach," *IEICE trans. Fundamentals.*, vol. 92, no. 5, pp. 1308–1315, 2009.
- [11] T. Morie, D. Atuti, K. Ifuku, Y. Horio, and K. Aihara, "A CMOS nonlinear-map circuit array for threshold-coupled chaotic maps using pulse-modulation approach," in *European Conf. on Circuit Theory and Design (ECCTD)*, pp. 126–129, 2011.
- [12] S. Uenohara, D. Atuti, K. Matsuzaka, H. Tamukoh, T. Morie, and K. Aihara, "A PWM-mode CMOS threshold-coupled-map circuit robust to device mismatches," in *Proc. NOLTA*, pp. 503–506, 2013.
- [13] S. Uenohara, D. Atuti, K. Matsuzaka, H. Tamukoh, T. Morie, and K. Aihara, "A CMOS circuit for pwm-mode nonlinear transformation robust to device mismatches to implement coupled map lattice models," *NOLTA*, vol. 6, no. 4, pp. 570–581, 2015.
- [14] S. Sinha, "Adaptive dynamics on circle maps," *Phys. Lett. A*, vol. 199, no. 5, pp. 365–374, 1995.
- [15] A. Mondal and S. Sinha, "Spatiotemporal consequences of relaxation time scales in threshold-coupled systems," *Phys. Rev. E*, vol. 73, no. 2, pp. 026 215–1–8, 2006.
- [16] M.D. Shrimali, S. Sinha, and K. Aihara, "Asynchronous updating induces order in threshold coupled systems," *Phys. Rev. E*, vol. 76, no. 4, pp. 046 212–1–10, 2007.
- [17] J. Von Neumann and A.W. Burks, *Theory of self-reproducing automata*, University of Illinois Press, 1966.
- [18] M. Cook, "Universality in elementary cellular automata," *Complex Systems*, vol. 15, no. 1, pp. 1–40, 2004.
- [19] R. Goebel, R.G. Sanfelice, and A.R. Teel, "Hybrid dynamical systems," *IEEE Control Systems Magazine*, vol. 29, no. 2, pp. 28–93, 2009.