Chaos, Solitons & Fractals 69 (2014) 74–80

Contents lists available at ScienceDirect

Chaos, Solitons & Fractals

Nonlinear Science, and Nonequilibrium and Complex Phenomena

journal homepage: www.elsevier.com/locate/chaos

Reconfigurable chaotic logic gates based on novel chaotic circuit

S. Behnia [a,*], Z. Pazhotan [a], N. Ezzati [b], A. Akhshani [c]

[a] Department of Physics, Urmia University of Technology, Orumieh, Iran
[b] Department of Electronic, Urmia University of Technology, Orumieh, Iran
[c] School of Physics, Universiti Sains Malaysia, 11800 USM, Penang, Malaysia

ARTICLE INFO

Article history:
Received 10 June 2014
Accepted 21 August 2014
Available online 3 October 2014

ABSTRACT

The logical operations are one of the key issues in today's computer architecture. Nowadays, there is a great interest in developing alternative ways to get the logic operations by chaos computing. In this paper, a novel implementation method of reconfigurable logic gates based on one-parameter families of chaotic maps is introduced. The special behavior of these chaotic maps can be utilized to provide same threshold voltage for all logic gates. However, there is a wide interval for choosing a control parameter for all reconfigurable logic gates. Furthermore, an experimental implementation of this nonlinear system is presented to demonstrate the robustness of computing capability of chaotic circuits.

1. Introduction

Everything begins, when Charles Babbage's presented the first "mechanical computer". The basic part of this machine called a "logic gate". Logic gates, provide ingredients of universal dynamic computers. A logic gate is an electronic device that provides logic 0 or logic 1 depending on the input conditions. In computer architecture, it is common to use 0 for false and 1 for true. Recently, there has been a great interest in developing new approaches and directions in the computer architectures. In 1998, initiated by the pioneer work of Sinha and Ditto [1], the study of chaos computing was the subject of many recent papers [32,2–5]. For instance, Murali et al. reported the experimental realization of fundamental logic gates consisting of *AND*, *OR*, *XOR* and *NOT* gates in discrete-time and continuous-time chaotic circuits [4,5].

On the other hand, as far as bridging dynamics and computation are concerned, examples of dynamic logic architecture were provided with chaotical [7–9], optical [10], chemical [11–14], physical [15], mechanical [16], biological [17–19], molecular [20,21] and many other systems [22–27] for emulating different logic gates. These gates are used as basic unites of a general-purpose device that is more flexible than statically wired hardware.

Murali et al. performed sequences of logic operations in time based on the dynamical evolution of a single logistic map [28]. The dynamical logic architectures by using piecewise-linear systems is obtained by Peng et al. [29,30]. In this scheme, the intrinsic properties of piecewise-linear dynamical systems, enable the system to discriminate the two input signals. A dynamical multiple-input multiple-output logic gates based on window threshold method are proposed by Peng et al. [31].

In this work, a novel method for obtaining fundamental logic gates based on a hierarchy of one-parameter family of chaotic maps is introduced. Referring to rich behavior of the maps, all threshold voltages are equal, and we have a wide interval for choosing a control parameter to make the system chaotic. It seems that, new dynamical systems are able to create an opportunity for finding new ways to improve Sinha and Ditto's brilliant method for morphing logic gates. Such a universal programmable logic gate could

* Corresponding author.
E-mail address: s.behnia@sci.uut.ac.ir (S. Behnia).

http://dx.doi.org/10.1016/j.chaos.2014.08.011
0960-0779/

potentially provide the starting point for more mature approaches to faster, robust and reconfigurable computational platforms based on the principle of large numbers of identical, reconfigurable and programmable dynamic logic gates. First, we review the theoretical scheme for obtaining all basic logic gates and then go on to introduce the hierarchy of family of one-parameter maps. As a primary example of logic gates based on the introduced hierarchy, a novel *NOR* gate is built and introduced in this paper. To understand the advantages of this map, a numerical example is considered. We have supported our analytical calculation by experimental implementation. Additionally, the bifurcation and Lyapunov exponent diagrams of the one-parameter map are compared with the logistic map. The map has an interesting property of having the wide range of chaotic behavior that relates to the threshold voltage is investigated.

2. Theoretical scheme for obtaining all basic logic gates

In this section, we review a straightforward scheme for morphing mechanism in discrete systems. By considering the inputs and output of logic gate as x_1, x_2 and O respectively. The scheme consists of following three steps:

1. Inputs states:
   - $x = x_0 + x_1 + x_2$ for the AND, OR, XOR and *NAND* operations.
   - $x = x_0 + x$ for the *NOT* operation x_0 can be thought of, as the initial state of the system. For $I = 0$, logic inputs x_1, x_2 and x have value 0 and for $I = 1$, they have value δ (δ is interpreted as a positive constant (see Table 1).
2. Chaotic update ($x \longrightarrow \Phi(x)$).
3. Outputs (using Threshold mechanism)

$$O = 0, \text{ if } \Phi(x) \leqslant x^* \text{ and } O = \Phi(x) - x^*, \text{ if } \Phi(x) > x^*,$$

where O and x^* are the output and threshold voltages respectively. $O = 0$ is considered as a logical 0 and $O = \delta$ is considered as a logical 1. Table 1 shows essential conditions for obtaining fundamental logic gates.

Recently, a number of discrete-time nonlinear maps have been proposed for emulating logic gates [5,6] and the most well-known is based on the logistic map which is defined as:

$$F(x) = rx(1 - x). \tag{1}$$

For emulating logic gate, the control parameter of the map is considered in fully chaotic area ($r = 4$). In this study, for a complete comprehension of this system (Eq. (1)), a numerical example which satisfy essential conditions in Table 1 have been demonstrated in Table 2. We repeat the numerical example as shown in Table 3 by using new chaotic map and the same initial condition x^* for all fundamental logic gates.

Table 2
Numeric example for logistic map $\delta = 0.25$.

Operation	NOT	XOR	OR	AND
X_0	1/2	1/4	1/8	0
X^*	0.7	0.7	0.6	0.7

Table 3
Numeric example for new map $\delta = 1$.

Operation	NOT	XOR	OR	AND
X_0	−1.01	−1.03	−1.02	0.012
X^*	1	1	1	1

3. One-parameter families of chaotic maps

In this paper, we present new one-parameter families of chaotic maps of interval $[0, 1]$ to illustrate the capability of chaotic systems for designing new logic gates. The map governed by Eq. (2):

$$\Phi_N(x, \alpha) = \frac{\alpha^2 \left(T_N\left(\sqrt{x}\right)\right)^2}{1 + (\alpha^2 - 1)\left(T_N\left(\sqrt{x}\right)^2\right)} \tag{2}$$

as an example:

$$\Phi_2(x, \alpha) = \frac{\alpha^2 (2x - 1)^2}{4x(1 - x) + \alpha^2 (2x - 1)^2}. \tag{3}$$

The control parameter is represented by a value α. The map has only a fixed-point attractor ($x = 1$) provided that their parameter belongs to interval $(2, \infty)$. While, at $\alpha \geqslant 2$ they bifurcate to the chaotic regime without having any period doubling or period-n-tupling scenario and remain chaotic for all $\alpha \in (0, 2)$ [33].

The invertible map, $h(x) = \frac{1-x}{x}$, maps $I = [0, 1]$ into $[0, \infty)$ and transform maps $\Phi_N(x, \alpha)$ in to $\tilde{\Phi}_N(x, \alpha)$ defined as:

$$\begin{aligned} \tilde{\Phi}_N(x, \alpha) &= h \circ \Phi_N(x, \alpha) \circ h^{(-1)} \\ &= \frac{1}{\alpha^2} \tan^2 (2 \arctan \sqrt{x_n}). \end{aligned} \tag{4}$$

Table 1
Essential conditions for obtaining fundamental logic gates.

Operation	OR	AND	XOR	NOT
Condition (1)	$f(x_0) \leqslant x^*$	$f(x_0) \leqslant x^*$	$f(x_0) \leqslant x^*$	$f(x_0) - x^* \sim \delta$
Condition (2)	$f(x_0 + \delta) - x^* \sim \delta$	$f(x_0 + \delta) \leqslant x^*$	$f(x_0 + \delta) - x^* \sim \delta$	$f(x_0 + \delta) \leqslant x^*$
Condition (3)	$f(x_0 + 2\delta) - x^* \sim \delta$	$f(x_0 + 2\delta) - x^* \sim \delta$	$f(x_0 + 2\delta) \leqslant x^*$	