Evaluation, Optimization, and Enhancement of Chaos Based Reconfigurable Logic Design

Md Sakib Hasan, Md. Badruddoja Majumder, Aysha S. Shanta, Mesbah Uddin and Garrett S. Rose

IEEE SoutheastCon 2019, Huntsville, AL, Apr 2019.

The online home for this paper may be found at: http://web.eecs.utk.edu/ grose4/

Citation Information (BibTex):

```
@INPROCEEDINGS{hasan2019evaluation,
  author=" M. S. Hasan and M. B. Majumder and A. S. Shanta and
M. Uddin and G. S. Rose",
  title="Evaluation, Optimization, and Enhancement of Chaos
Based Reconfigurable Logic Design",
  booktitle="{IEEE} SoutheastCon 2019",
  month="April",
  year="2019",
  address="Huntsville, AL, USA"
}
```

Evaluation, Optimization, and Enhancement of Chaos Based Reconfigurable Logic Design

Md Sakib Hasan, Md Badruddoja Majumder, Aysha S. Shanta, Mesbah Uddin and Garrett S. Rose
Department of Electrical Engineering and Computer Science
University of Tennessee, Knoxville
Knoxville, Tennessee 37996 USA
Email: {mhasan4, mmajumde, ashanta1, muddin6, garose}@vols.utk.edu

***Abstract*—Nonlinear dynamics and chaos can be utilized to build reconfigurable and adaptive logic gates. The inherent reconfigurability of these gates has promising security applications. The existing designs of chaos based logic gates have high overhead compared to CMOS gates. In order to make chaos based logic competitive, we need performance metrics to enable optimization of the design. In this work, we propose a metric that will enable circuit designers to optimize the logic gate design based on application specifications. We focus on a particular topology and study how different factors influence its performance and how they can be optimized to reduce overhead. However, the basic idea can be used for chaotic map circuit using other topologies as well. We also propose an enhancement technique to significantly expand the design space which is desirable for security applications.**

I. Introduction

Chaos has attracted a lot of attention in areas such as physics, chemistry, biology, financial systems and ecology [1]. The work on chaos has gained popularity since Lorenz's seminal work on chaotic motion on a strange attractor in 1963 [2]. Nonlinear systems have been effectively used to model engineering and natural systems [3], [4]. Over the years, nonlinear dynamics in chaotic systems has become an active field of research due to advancements in chaos communications and chaotic neural networks [5], [6]. Innovative researchers have found application of chaos theory in different branches of engineering such as lasers, cryptography, secure communication and plasma technologies [7], [8].

Another vibrant research area is computing based on chaotic progression of state in non-linear circuits. Researchers have worked on various aspects of chaos computing exploring flexibility, reconfigurability and security in the nonlinear systems [9]–[11]. Chaos based computation enables a single circuit to perform a large number of functions by changing the parameters in the circuit topology. A single operation can be performed in many ways by changing the control input, threshold, initial state, iteration number, etc [12]. Researchers have come up with different chaos based implementations of basic logic gates. Higher input functions are possible to implement using chaos [13] in addition to two input logic functions. Chaos computing can also be used in secure and confidential computing. Chaos gates are used in the field of hardware security as a means of obfuscating power profile in processing units and hence mitigating power based side channel attack [14], [15].

One of the significant limitations of chaos computing is its overhead compared to traditional CMOS based design. In order to leverage its unique reconfigurability, specially for security applications, we want a large functionality space i.e. many different ways to reconfigure the gate to implement numerous logic functions without excessive overhead. With this goal in mind, we propose a design metric with adjustable weights depending on application specifications to evaluate the performance of a chaotic map circuit. This, in turn, will enable designers to optimize the circuit by utilizing the metric. We also propose a novel method to expand the design space by orders of magnitude by utilizing the body terminals of MOS transistors.

The paper is organized as follows. Section II provides background and necessary details about chaos computing and its application to perform reconfigurable and flexible logic operations. Design and working principles of a chaotic map circuit is described in Section III. Section IV describes the proposed metric to evaluate the performance of a chaotic circuit. Optimization of the circuits based on application requirement is described in V. A significant enhancement in design space utilizing the body terminals of MOSFETs is proposed in VI. Finally, the paper is concluded in Section VII.

II. Chaos Computing Preliminaries

There are areas where digital systems do not meet the demands and specifications of the modern world despite its huge success in the past few decades. Conventional computing units are built from Boolean circuits. A Boolean circuit contains transistors that act as switches. The switches open and close based on the the incoming input to the gates of the transistors. Digital systems require millions of transistors because different topologies of CMOS circuits need to be designed in order to implement different logic functions [13]. Huge number of transistors lead to excessive power consumption and heat production in a chip.

Sinha *et al.* proposed that chaos circuits can be used to build computing systems [16]. Bohl *et al.* proposed the use of chaos gates in the field of hardware security [17] for logic obfuscation and power analysis mitigation. The power profile of the logic computations in chaos gates helps to eliminate side channel attacks.

978-1-7281-0137-8/19/$31.00

The nonlinear dynamics of CMOS circuits and their inherent computational capability is being explored in chaotic systems. The nonlinear system can develop in discrete time or in continuous time. Continuous time chaotic systems are highly complex and have low efficiency compared to discrete time chaotic systems. A discrete time chaotic oscillator is designed by connecting the output of the map to the input of the circuit creating a feedback path [18]. The map circuit translates the initial state of the circuit to future states.

A major challenge in using chaotic logic gate is that they require more supplemental circuits in addition to the chaotic map which increases the area overhead compared to CMOS logic gates. One way to overcome this limitation is to ensure that each chaotic gate is able to generate increased functionality. The number of functions that a single chaotic circuit can implement increases exponentially with the number of iterations and other parameters. The functions generated by the chaotic circuit can be dynamically chosen to implement different logic functions in each clock cycle. In reality, all the functions may not be accessible or usable due to instability or noise [10].

In previous days, arithmetic operations were performed using arrays of chaotic elements. Chaos gates are now used to implement logic gates which are capable of implementing $A + B$, $\overline{A + B}$, $\overline{AB}$, AB, $A \oplus B$, $\overline{A \oplus B}$, ON and OFF. The one-dimensional system is able to implement only eight of the sixteen possible functions because the initial state is a function of the sum of the inputs [11]. This problem can be eliminated if each of the inputs has its own state variable.

Chaotic circuits are made flexible and reconfigurable by changing the control parameters in the circuit topology. Murali *et. al.* used a thresholding mechanism to implement a NOR gate with continuous-time chaotic system [19]. Chua's circuit has been used to design a flip-flop which is a building block for memory devices [20]. Rose constructed a chaos based arithmetic logic unit where different functions are selected by controlling the control input and iteration number [14]. Rizk *et. al.* also applied a threshold mechanism to Chua's circuit in order to obtain all the functions such as AND, NAND, OR, NOR and NOT [21]. In summary, chaos computing is an emerging field with plenty of opportunities for creative ideas and research.

III. Discrete Time Chaotic Map Circuit

A non linear map $f : [a; b] \rightarrow [a; b]$ can generate discrete time chaotic behavior in the form of iterated function such as

$$x_{n+1} = f(x_n) \tag{1}$$

Some of the well known chaotic maps are logistic map, tent map, circle map. All of these map functions exhibits unimodality which is the simplest condition to show chaotic behavior in discrete time. Any function that has the same basic characteristics can essentially produce chaotic behaviors. Periodicity of discrete time series of an iterated map progresses towards infinity through period doubling by changing specific paramters of the function. This progression of period doubling is known as bifurcation and the parameters controlling such progression is called bifurcation parameter. Universality of iterated map's route from periodicity towards chaos was studied in a greater detail in the classic work of Feigenbaum [22].

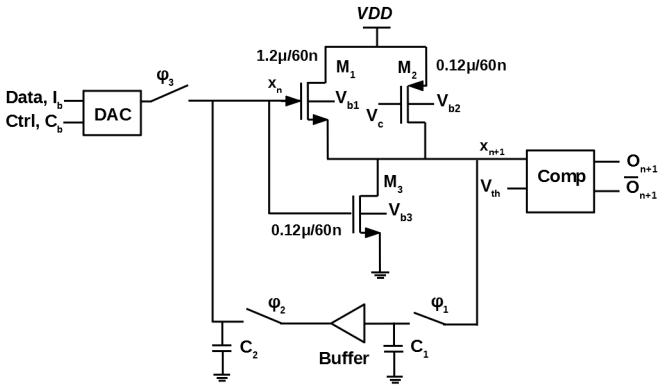


Fig. 1: Logic gate built from a discrete time chaotic map circuit.

Well known map functions such as logistic maps, tent maps etc. have been implemented in circuit level. In this paper, we focus on 3 transistor ($M1; M2; M3$) based chaotic map circuit shown in Fig. 1 for generating discrete time chaotic signals [10], [23], [24]. A scaled down implementation in a 65 nm process is used for this well-studied chaotic map circuit topology. For this circuit, bias voltage V_c works as the bifurcation parameter. For now, the body terminals of n-MOS and p-MOS transistors are connected to GND and VDD, respectively. Later, we will see how the body terminals can be utilized to expand the design space. The transistor sizes are chosen initially among other candidate geometries based on a performance metric. The performance metric aims to maximize the chaotic behavior and minimizes circuit overhead from the perspective of area, power and delay. This is going to be described in greater detail in the following section. Chosen size for the transistors are as follows: M_1 = 1.2 µm/60 nm, M_3= 0.12 µm/60 nm and M_2 = 0.12 µm/60 nm. Performance of other geometries are also explored later. Fig. 2 shows the transfer characteristics of the map circuit for different values of the bifurcation parameter, V_c. It can be seen from the figure that the transfer curve is a V-shaped which is a common characteristic found in many well known non linear map functions generating chaos.

Two sample-and-hold circuits are used with the map circuit in order to implement logic functions as shown in Fig. 1. Each iteration takes place in two phases of a clock cycle. In the first phase, $_1$ is enabled and the output of the map circuit is sampled onto C_1. $_2$ is enabled in the second phase where the output is transferred to C_2. Output of the map circuit is thereby fed back as the input of the next iteration. For simplicity, we are using a 3-bit digital-to-analog converter (DAC) to convert the binary input signal made of 2 data (I_b) bits and 1 control (C_b) bit into analog input of the map. More control bits can be used to enhance the design space and more data bits can be used to implement multi-input logic functions. After a certain number of iterations(n), sampled output is compared against

TABLE I: Evolution of chaotic output (V_C=0.68 V, C_b = 1).

X_0 (V)	X_{n+1} (V)				
	i = 1	i = 2	i = 3	i = 4	i = 5
0:171(00)	1:19	0:52	0:40	0:93	0:36
0:514(01)	0:41	0:88	0:34	1:15	0:49
0:857(10)	0:33	1:16	0:50	0:49	0:52
1:2(11)	0:52	0:40	0:94	0:37	1:06

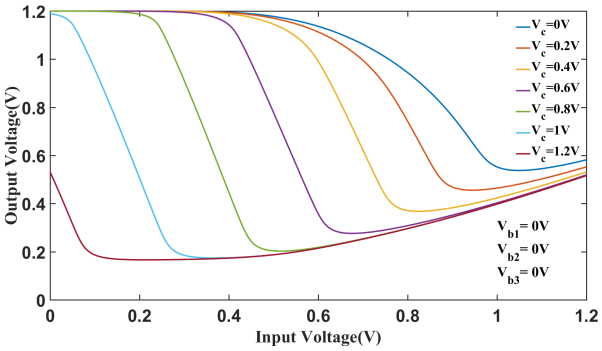


Fig. 2: Transfer curve of map circuit.

a threshold V_{th} using a comparator and a binary output, O is generated. In addition to V_c and C_b, the functionality also depends on the choice of n and V_{th}. Reconfigurabilty can be obtained by tuning the control parameters [12]. Table I shows example analog output voltage sequences for five iterations for all possible input combinations with V_c=0.68 V and C_b=1.

IV. Design Evaluation

Bifurcation diagram of a chaotic map circuit represents all the possible outputs with respect to a bifurcation parameter. Bifurcation diagram for the map circuit of this paper is shown in Fig. 3. Steady state value of the map circuit from 1000 iterations are plotted in the y-axis corresponding to each value of V_c in the range of 0 V to 1.2 V. Period doubling for this map circuit can be observed from the bifurcation diagram. Output of the map circuit oscillates with a period of 2 when V_c < 0:52 V. First period doubling occurs at V_c = 0:52 V where the output oscillates with a period of 4. The map circuit bifurcates again at V_c = 0:637 V for which the period is 8. At this point, the period

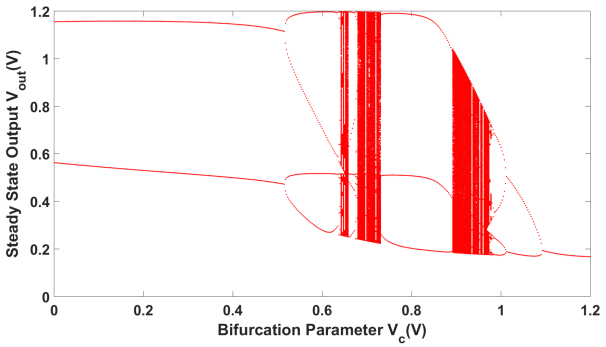


Fig. 3: Bifurcation diagram.

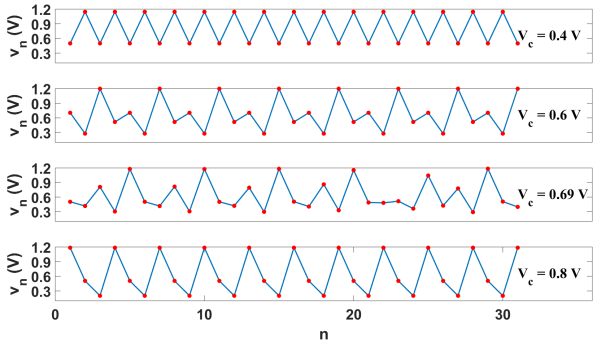


Fig. 4: Evolution through successive iterations for different values of bifurcation parameter.

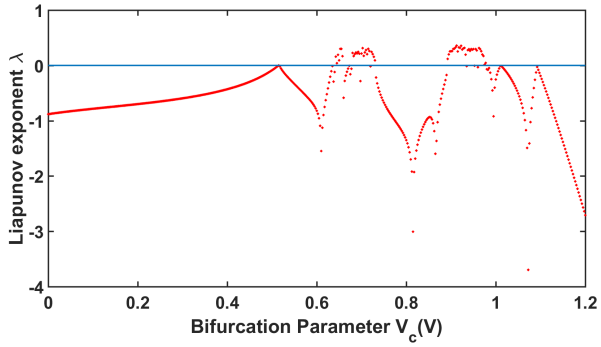


Fig. 5: Lyapunov exponent against bifurcation parameter for geometry 6.

doubling occurs at a progressively faster rate with the increase of V_c. At V_c = 0:642 V, the map circuit demonstrates chaotic behavior and starts oscillating aperiodically. Fig. 4 shows the transient response of the output voltage for 30 iterations with 4 different values of V_c. The period of the output voltage sequences are 2, 4 and 3 for Vc =0.4 V, 0.6V and 0.8 V, respectively. For Vc = 0:69 V, the sequence is aperiodic.

In order to be considered "chaotic", a system should show sensitive dependence on initial conditions. It means that if there is a small perturbation on the initial condition, the neighboring

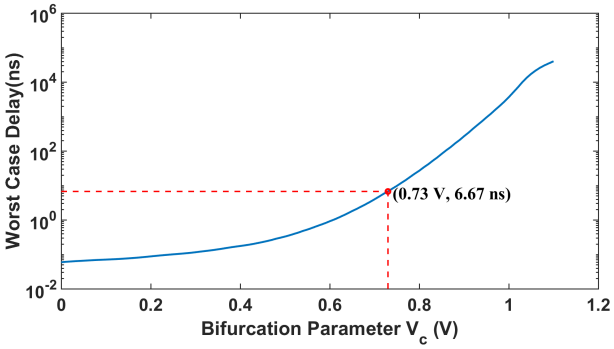


Fig. 6: Worst case delay.

orbits should separate exponentially fast, on average. Lyapunov exponent, is used to quantify sensitive dependence on initial condition. For discrete-time chaotic maps, it is defined as,

$$= \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} ln|f'(x_i)| \quad (2)$$

For stable fixed points and cycles, it is negative; for chaotic attractors, it is positive [1]. Fig. 5 shows the Lyapunov exponent against bifurcation parameter. As expected, we see that positive Lyapunov exponent in Fig. 5 corresponds to chaotic regions in Fig. 3.

For the chosen geometry, there are 2 distinct regions of V_c when the oscillator is in chaotic region. The first one is 0.62 V to 0.71 V (with intermittent periodic region) and the second region is 0.88 V to 0.96 V. Investigating the circuit, it is found that the delay is maximum when the input makes a transition from VDD to 0. It can be seen in the semi-logarithmic plot in Fig. 6 that the worst case delay increases very fast as the value of V_c increases. The first region of V_c results in considerably less delay than the second one. Therefore, we constrain the V_c to the first region only.

A closed form analytical solution of the chaotic map function for sub-micron transistors is highly intractable and is out of the scope of this work. For this reason, the choice of geometry is based on an empirical analysis. In order to show the influence of geometries on device performance, we have defined a performance metric considering most relevant factors for reconfigurable logic gate implementation. we have simulated 7 different test geometries and ranked them based on the design metric. For all these geometries, the basic shape of bifurcation diagram and the relationship between delay and bifurcation parameter is similar. To illustrate this, we show the bifurcation diagram and delay plot for geometry 1 and geometry 3 in Fig. 7 and Fig. 8, respectively. The performance metric (PM) is defined as

$$PM = \frac{C^{w_c}}{A^{w_a} \cdot P^{w_p} \cdot D^{w_d}} \quad (3)$$

where C= width of desired chaotic region in bifurcation diagram, A= area of the three transistor chaotic map, P= average power consumption in the chaotic region and D= worst case delay. w_c, w_a, w_p and w_d are corresponding weight for these factors which are considered 1 i.e. $w_c = w_a = w_p = w_d = 1$. However, depending on applications, we may modify these weights to prioritize particular factors in the performance metric. As width of chaotic region in the bifurcation diagram contributes to the large functionality space of the chaotic gate, it is kept as the numerator in the performance metric. On the other hand, quantities contributing to design overhead such as area, power and delay are kept in the denominator. Higher value of PM implies better performance of the chaotic circuit.

Table II shows the performance metric for each of the 7 geometries. W_1 , W_2, W_3 represent the widths of transistors M_1, M_2 and M_3 in Fig. 1. Geometry 6 which has the highest performance metric is chosen in the final design used in this work. Length of the transistors are chosen as $60nm$ which is the minimum length of the $65nm$ process.

TABLE II: Performance metric for different geometry of chaotic oscillator.

Index	Geometry			Metric
	W_1(m)	W_2(m)	W_3(m)	$\frac{mV}{fJ:\ m^2}$
1	4.8	0.48	0.48	0.85
2	4	0.15	0.15	5.09
3	4	0.12	0.12	7.08
4	6	0.15	0.15	3.54
5	2	0.15	0.15	4.73
6	1.2	0.12	0.12	7.15
7	2.4	0.24	0.24	2.04

TABLE III: Components of performance metric for different geometry of chaotic oscillator.

Index	C(s)	A(m^2)	P(W)	D(ns)
1	90	0.35	63.6	4.79
2	72.5	0.26	22.15	2.49
3	95	0.25	18.44	2.86
4	85	0.38	23.52	2.7
5	62.5	0.14	20.8	4.61
6	65	0.09	16.35	6.43
7	67.5	0.17	32.68	5.86

we can clearly see from Table III, that as we increase the widths of transistors, the area and power consumption increase and the delay decreases.It gives the designers a sense about general trend of the influence of geometry on different components and performance metrics. For example, if the priority is to make the circuit fast even at the expense of significant power/area overhead, then the designers should choose bigger transistors.

V. Design Optimization

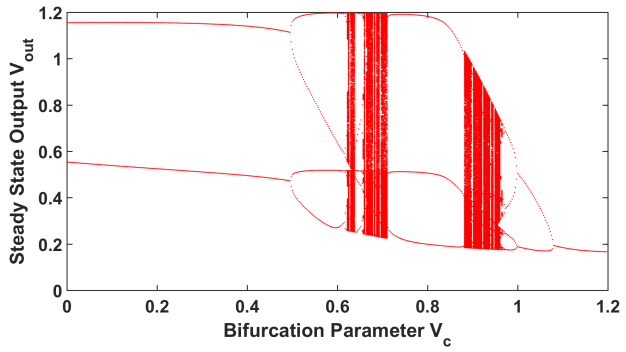
We already saw design optimization at work when we chose to operate inside the first chaotic region to minimize delay. Geometry selection based on performance metric can help us to optimize design. Here, it is important to discern the importance of application. For example, if we want to use chaos based logic for bitwise operation in larger system with security applications in mind, then all single bit instances do not require to be implemented using chaos logic. Instead, we can use a mixed implementation using a combination of CMOS and chaos gates which significantly reduces the overhead of the entire system. Overhead reduction using mixed implementation can be estimated as follows.

$$OV_{sys} = \frac{(1-k)G_{CMOS} + kG_{CHAO}}{G_{CMOS}}; \quad (4)$$

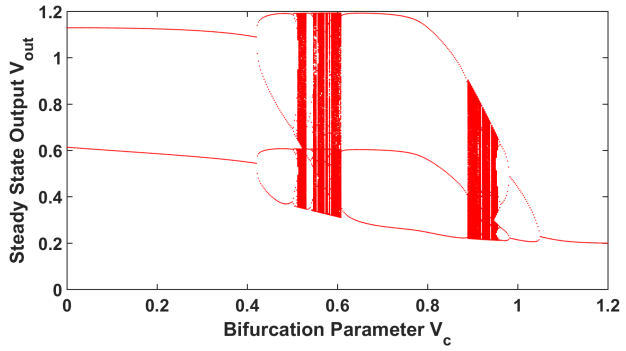
where k is the proportion of chaos gates in the entire system, G_{CMOS} and G_{CHAO} are the implementation cost associated with each standard CMOS gate and chaos gate, respectively. Eq. 4 can be further simplified in terms of per gate overhead of chaos gate compared to the CMOS gates.

$$OV_{sys} = 1 + k(OV_{gate} - 1) \quad (5)$$

where $OV_{gate} = G_{CHAO}/G_{CMOS}$.

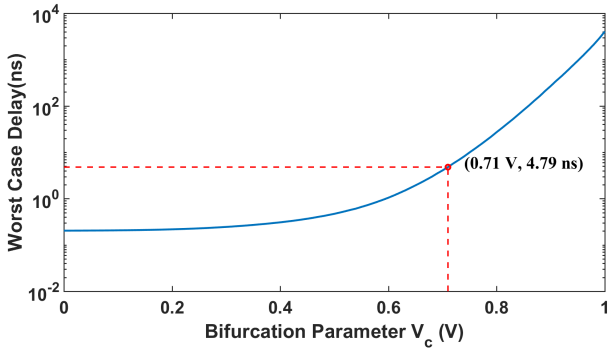


(a)

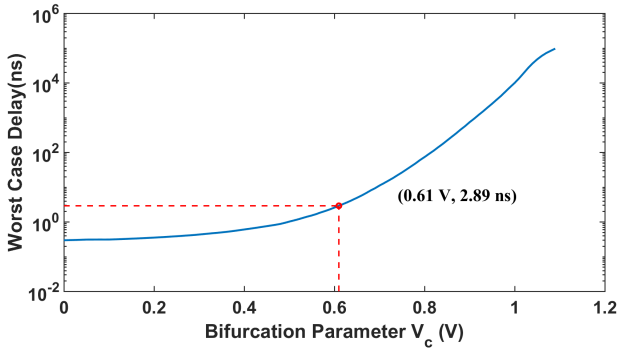


(b)

Fig. 7: Bifurcation Diagram for two geometries: a) Geometry1, b) Geometry 3



(a)



(b)

Fig. 8: Worst case delay against bifurcation parameter for two geometries: a) Geometry1, b) Geometry 3

TABLE IV: Performance metric for different geometry of chaotic oscillator (asymmetric weight).

Index	Geometry			Metric
	$W_1(m)$	$W_2(m)$	$W_3(m)$	$\frac{mV}{fJ\ m^2}$
1	4.8	0.48	0.48	0.18
2	4	0.15	0.15	2.04
3	4	0.12	0.12	2.48
4	6	0.15	0.15	1.31
5	2	0.15	0.15	1.03
6	1.2	0.12	0.12	1.11
7	2.4	0.24	0.24	0.35

Equation 5 holds for area and power overhead, but does not hold for delay overhead. In case of multi-bit operation, the delay of the longest cell determines the delay of the whole block. As a result, even if a 64 bit implementation has only one chaos gate, its delay will be the delay of the entire block i.e. $Delay_{sys} = Delay_{gate}$. Consequently, it is desirable that the delay is assigned larger weight compared to other factors in the performance metric as shown in Eq. 3. For example, if we choose $w_c = w_a = w_p = 1$ and $w_d = 2$, we get Table IV. We can see that after the weight modification, geometry 3 now has the best metric instead of geometry 6. Similarly, based on requirements, we can readjust the weights and then determine the best geometry for the application at hand.

VI. Design Enhancement

The functionality space of the reconfigurable chaos gate is already quite large as demonstrated in [12]. Here, four parameters are used to control functionality and the number of functions in n iterations can be written as,

$$f(n) = N_{V_{th}} \times 2^c \times b \times n: \tag{6}$$

Here, $N_{V_{th}}$ is the number of threshold voltage levels, c is the number of control bits used, b is the number of bias voltage levels, n is the number of iteration.

In earlier works of the same circuit topology, only one bifurcation parameter, V_c was used. The building block of this chaotic map circuit is MOSFET which is a four terminal device. The body terminal was not used as the bifurcation parameter in the earlier designs. If we use the body terminal of the three transistors in the map circuit, then the voltage applied at those terminals V_{b1}, V_{b2} and V_{b3} will act as added control parameters. For any bifurcation parameter, V_c in chaotic region, we can choose different combination of body bias voltages to get different functions. Fig. 9 shows the transfer curve for the same circuit in Fig.1 with the body terminal of p-MOS (M_2) connected to 0.8 V, instead of VDD. The difference in transfer characteristics between Fig. 2 and Fig. 9 is caused by the $0.8V$ body potential in p-MOS transistor.