Sensitivity of a Chaotic Logic Gate
Noeloikeau F. Charlot and Daniel J. Gauthier, Member, IEEE

Abstract—Chaotic logic gates or ‘chaogates’ are a promising mixed-signal approach to designing universal computers. However, chaotic systems are exponentially sensitive to small perturbations, and the effects of noise can cause chaotic computers to fail. Here, we examine the sensitivity of a simulated chaogate to noise and other parameter variations (such as differences in supply voltage). We find that the regions in parameter space corresponding to chaotic dynamics coincide with the regions of maximum error in the computation. Further, this error grows exponentially with a maximum likelihood rate constant of 3.5 iterations of the chaotic map. As such, we discuss the fundamental challenges of chaotic computing, and suggest potential improvements. Our Python simulation methods are open-source and available at https://github.com/Noeloikeau/chaogate.

Index Terms—Chaos, chaogate, chaotic circuit, chaotic computation.

I. INTRODUCTION

CHAOTIC systems have been shown to be able to implement any logical function [1]. Therefore, chaos can be used for computation [2]. In fact, a chaotic computer can perform faster and/or require fewer resources than a typical digital computer in certain cases [3], [4]. Moreover, the extreme sensitivity that characterizes chaotic systems may function as an inherent encryption mechanism, serving to obfuscate circuits and information [5]. As a result, chaotic circuits have the potential to offer faster, more secure, and more compact methods of computation than traditional hardware.

Here, we consider an established class of mixed-signal digital-analog circuits built around a chaotic ‘core’ (an analog nonlinear circuit) that interfaces with digital control logic [6], [7]. This control logic is used to specify the initial condition of the chaotic core and the amount of time that it is allowed to operate or ‘iterate.’ After the specified amount of time or number of iterations, the analog state of the core is binarized and converted into a digital value by a comparator circuit (register, Schmitt trigger, etc.), the result of which is used in the output of the computation.

In practice, different measurement times are used to select different logical functions. This is because the state of the chaotic core changes over time, with a tendency to eventually visit the full range of states that the system can occupy—a property known as ergodicity [8]. Hence, as time changes, a different logical function tends to describe the mapping from the initial condition to the current Boolean state. As a result, a chaotic computer is effectively described by a look-up-table (LUT) listing which function is implemented given an iteration number, initial condition, and control signal.

However, if repeated measurements of the chaotic system vary for a fixed input (such as due to thermal noise, fluctuating power supply, etc.) then the computation is non-deterministic, as the same input may yield different outputs. This changes the LUT entries describing the computer from functions to distributions of functions, each occurring with some probability dependent upon the characteristics of the device and its environment. As such, accounting for the sensitivity of the chaotic core to its operational parameters is necessary for achieving fault-tolerant chaotic computation in practice [9]. Moreover, recent work has shown that implementation-specific design decisions such as order of operations in the addition of binary numbers can affect the cryptographic and local dynamical properties of chaotic systems, while maintaining their global properties in both hardware and software [10]. Therefore, reproducible chaotic computers must take into account the sensitivity of the chaotic logic gates of which they are composed. However, to the best of our knowledge a chaogate sensitivity analysis is absent from the literature.

We address this information gap by analyzing the sensitivity properties of a simulated chaogate, and comparing our findings with modern theoretical results and experimental data. The core of the chaogate is composed of three coupled transistors forming an analog nonlinear circuit [11]–[13]. Recently, we introduced a comprehensive simulation framework for this system [14]. Using this framework, we calculate the Lyapunov exponent, parametric sensitivity, and noise sensitivity (error rate). Our unique contributions follow:

- We find that the chaogate is an ergodic system, and that its noise sensitivity, parameter sensitivity, and positive Lyapunov exponent share the same support. This suggests a fundamental trade-off between chaos and error rate.
- We find that chaogate Lyapunov times are well fit to an F-distribution peaked at 3.5 iterations, with a range of 3-10 iterations similar to reported experimental timescales.

This brief is organized as follows. In Section II we describe the chaogate simulation and precisely define ‘chaos’ and ‘sensitivity’. In Section III we analyze these properties across an optimal region of parameter space and discuss the results. In Section IV we conclude with future research.

II. DESIGN AND METHODOLOGY

Each chaogate (Fig. 1 (a)) implements a function \( f \) that maps an input voltage \( V_n \) to an output voltage \( V_{n+1} \), given a fixed set

---

1549-7747 © 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
of parameters $p$. Control logic initially holds a chaotic circuit (Fig. 1 (b)) to the input voltage $V_0$. The circuit equilibrates and produces an output voltage $f(V_0) = V_1$. This is sampled by the control logic and used as the next input (producing an output $f(V_1) = V_2$, and so on). The ‘sample and hold’ design repeats this process $N$ times in a procedure known as iteration. At each iteration $n$, the output voltage is passed through a comparator $F$, which produces a bit used in the result of the computation.

The chaotic circuit simulation is split into two parts. In the first part, a DC voltage sweep is performed using the standard NgSPICE simulation tool, which takes a set of parameters $p$ (Table I) describing a chaogate netlist, and returns the DC transfer function of that chaogate over the input voltage range $[0, 1.2, 0.001] \text{V}$. This yields a 1200 element array that is interpolated to obtain the chaogate input-output mapping at the given parameter set $f_p$. We find that voltage steps $< 0.001 \text{V}$ do not improve performance, and stress that NgSPICE is only used to obtain DC transfer functions.

The second step implements the sample-and-hold iteration procedure of the digital control logic by recursively evaluating the DC transfer function of an individual chaogate. This yields an $N$-element timeseries with unit timestep via the 1-D map

$$V_{n+1}(p) = f(V_n, p).$$

Parameters are assumed to be held fixed during iteration. This assumption is reasonable due to the DC operating point asymptotically suppressing transient fluctuations in all but the final output voltage supplied to the next stage. As such, the logical output of the chaogate at each iteration is a function of only this voltage and the Booleanization threshold

$$F(V_n, p) = \begin{cases} 0 & f(V_n, p) < V_\text{bool}, \\ 1 & f(V_n, p) \geq V_\text{bool}. \end{cases}$$

where $V_\text{bool} = 0.6 \text{V}$ is the intermediary logic threshold, set to halfway between the range of $V_n \in [0, 1.2 \text{V}]$. See [14] and the ‘chaogate’ Python package documentation for details.

The behavior of $F$ in both time ($n$) and parameter space ($p$) forms the basis of any subsequent computation. Whether or not $F$ is chaotic depends heavily on the particular value of the parameters, and in practice control logic is introduced which modifies these parameters and implements the sample-and-hold/iteration process. Additionally, multiple chaogates are typically combined to form higher-order logical functions. However, the implementation of this control and combinational logic is design-dependent. For generality we study the behavior of $F$ by varying $p$ directly, without extraneous logic. The output of $F$ at different $p$ can then be combined as needed to compose logical functions and design chaotic computers.

### A. Lyapunov Exponent

Chaos in a one-dimensional map is defined by a positive Lyapunov exponent $\lambda > 0$ [15]. $\lambda$ is an exponential measure of the rate at which a sequence diverges in time. The local Lyapunov exponent $\lambda_n$ at iteration $n$ of a chaogate with fixed parameters $p$ and initial condition $V_0 = 0$ is the logarithm of the absolute value of the derivative

$$\lambda_n(p) = \ln \left| \frac{\partial f}{\partial V_n}(V_n, p) \right|. \quad (3)$$

The (asymptotic) Lyapunov exponent $\lambda$ is then defined as

$$\lambda(p) = \lim_{N \to \infty} \frac{1}{N} \sum_{n=1}^{N} \lambda_n(p). \quad (4)$$

We find the limit is well approximated by $N = 1000$ and fix this quantity throughout. Further, we find that $\lambda$ is asymptotically independent of $V_0$, indicating that the chaogate is an ergodic system [9]. As such, we fix $V_0 = 0$ V and discard the first 100 transient iterations in our calculations of $\lambda$, which we find yields representative results.

### B. Parameter Sensitivity

The capacity for the chaogate to act as a deterministic computer is directly related to whether its logical output changes in the presence of small fluctuations on the control parameters. In reality, parameters fluctuate under experimental operating conditions, and this can lead to two bitstreams produced by the same chaogate with the same input diverging over time. Thus, the parametric sensitivity $\sigma_n$ of the n’th iteration can be
characterized by the average distance between Boolean states at nearby points in parameter space

\[
s_n(p, \hat{p}) = \sqrt{ \frac{1}{||\hat{p}||} \sum_{i \in \hat{p}} \left( \frac{\partial F}{\partial p_i}(V_n, p) \Delta p_i \right)^2 },
\]

(5)

where \( \hat{p} \) is the set of parameters over which the sensitivity is calculated, \( ||\hat{p}|| \) is the number of elements in this set, and \( \Delta p_i \) is the increment value of parameter \( i \) (see Table I). In analogy to the Lyapunov exponent, we define the asymptotic parametric sensitivity \( \sigma = (1/N) \sum \sigma_n \).

**C. Noise Sensitivity**

In addition to parametric sensitivity, the chaogate also possesses noise sensitivity. Here we consider noise to be an additive voltage fluctuation on the output terminal, modifying Eq. 1 to read

\[
V_{n+1}(p) = f(V_n, p) + N(0, \nu^2),
\]

(6)

where \( \nu \in p \) is the standard deviation of the output voltage fluctuation in volts, drawn from the Normal distribution \( N \) with zero mean at each iteration. The noise sensitivity \( \varepsilon_n \) of the \( n \)'th iteration is obtained by averaging the pairwise differences between \( N_T = 1000 \) repeated measurements

\[
\varepsilon_n(p) = \frac{2}{N_T(N_T + 1)} \sum_{r=1}^{N_T} \sum_{r' = r+1}^{N_T} \left| F^{n^r}(p) - F^{n'r}(p) \right|,
\]

(7)

where \( F^{n^r} \) is the \( n \)'th iterate of the \( r \)'th repetition, i.e., the \( n \)'th bit of Boolean string \( r \) obtained by iterating the chaogate map with parameters \( p \), where each string is specified by the unique noise vector drawn over all timesteps. Hence, \( \varepsilon_n \) is a local measure of the probability that an error occurs due to noise. In our calculations of \( \lambda \) and \( \sigma \), we take \( \varepsilon(\nu = 0) = 0 \) so that the results are purely deterministic. Similarly, we define the asymptotic noise sensitivity as \( \varepsilon = (1/N) \sum \varepsilon_n \).

Here, we use the value \( \nu = 0.001 \) V, obtained via the 'noise' function in NgSPICE, which calculates the total output noise from all elements of the chaogate circuit on the output voltage terminal \( V_{n+1} \) (see Fig. 1). To estimate the total noise, we superimpose an arbitrarily small AC sinusoid on the supply voltage \( V_DD \). This noise source affects the entire circuit, and we integrate over the frequency range \([1, 10^{10}]\) Hz. This provides a conservatively large estimate for the output noise in the sample-and-hold circuit design.

**D. Optimization and Chaotic Sensitivities**

In order to maximize the design space for a chaotic computer, we search for the largest region of parameter space over which the dynamics are chaotic. In practice, the three MOSFET widths \( W \) are the only parameters required for tape-out after the technology node \( L \) has been chosen. Hence, we select an optimal width \( W_{opt} \) defined as maximizing the area of the chaotic region \( C^* \) in the \( V_{bias} - V_{DD} \) plane (two typical control parameters in an experiment)

\[
C^*(W) = \{(V_{bias}, V_{DD}, W) \in \mathcal{G} | \lambda(V_{bias}, V_{DD}, W) > 0 \},
\]

(8)

Fig. 2. Chaotic sensitivities vs. number of iterations. The black curves underneath \( \hat{\varepsilon}_n \), and \( \hat{\sigma}_n \) are fits to a saturating exponential function of the form \( y(n) = A(1 - e^{-n/\tau}) \). The fit coefficients are \( A_\varepsilon, C_\varepsilon = (0.401, 0.004) \) and \( A_\sigma, C_\sigma = (0.244, 0.003) \). The early fluctuations of \( \lambda_n \) have been truncated for visibility.

\[
W_{opt} = \arg \max_{W \in \mathcal{G}} \left( ||C^*(W)|| \right),
\]

(9)

where \( \mathcal{G} \) is the Cartesian product of the parameter values defined by Table I (i.e., a grid search). In practice we find that approximately half the width space is non-chaotic, emphasizing the importance of parameter optimization on building a chaotic computer. As a result of this search, we find \( W_{opt} = (725, 65, 1365) \) nm and fix this quantity throughout. However, we find that other chaotic regions of parameter space (obtained through random selection and linear combinations of the sensitivity measures \( a\lambda - b\sigma - c\varepsilon \)) display similar chaotic sensitivities, defined as follows.

We develop a characteristic measure of the chaogate sensitivity at each time by averaging over the chaotic region of parameter space \( C^* \). This gives an expected value for the local sensitivity measures \( \chi_n \in \{\lambda_n, \sigma_n, \varepsilon_n\} \) of a chaotic timeseries.

In practice we find these data are noisy, and so employ a moving average to improve the quality of the signals, which we then fit analytically to a saturating exponential. We define the chaotic sensitivities \( \hat{\chi}_n \) at each iteration as

\[
\hat{\chi}_n(C^*(W_{opt})) = \frac{1}{(N_T + 1)||C^*||} \sum_{p \in C^*} \sum_{n'=n-N_T/2}^{n+N_T/2} \chi_{n'}(p),
\]

(10)

where \( N_T = 2 \) is the size of the window (nearest-neighbor).

**III. RESULTS AND DISCUSSION**

Shown in Figure 2 are the chaotic sensitivities of the chaogate over the first 100 iterations. As can be seen, \( \hat{\varepsilon}_n \) and \( \hat{\sigma}_n \) are well fit to a saturating exponential function. These results show that chaogates are exponentially sensitive to perturbations of any kind - whether they are dynamical (\( \lambda \)), parametric (\( \sigma \)), or stochastic (\( \varepsilon \)).

From Fig. 2, we note the following two observations: 1. \( \hat{\sigma}_n \approx \hat{\lambda}_n \) (the parametric sensitivity approximates the Lyapunov exponent), and 2. \( \hat{\varepsilon}_n > \hat{\lambda}_n \) (the noise sensitivity
Fig. 3. Heat maps of the asymptotic (top) and local (bottom) chaotic Lyapunov exponent $\lambda > 0$, and parametric $\sigma$ and noise $\varepsilon$ sensitivities. Here $\sigma$ is calculated over $P = \{V_{bias}, V_{DD}\}$. Similar results ($\varepsilon \sim \lambda \sim \sigma$) are observed over other parameter slices.

is greater than the Lyapunov exponent). The first observation indicates that the average rate at which the chaotic map divergences in time ($\lambda$) is equivalent to the way in which it diverges in parameter space ($\sigma$). Simply put, a variation of the parameters manifests as a dynamical perturbation of the function value, which is magnified by subsequent iterations wherever $\lambda > 0$. See [16] for a more thorough analysis of this sensitivity property in ergodic 1-D chaotic maps.

The second observation indicates that the effects of noise eventually overwhelm the system, becoming the dominant source of error ($A_\varepsilon > A_\sigma$ in Fig. 2). This is also demonstrated by the slightly shorter saturation time for the noise sensitivity (defined as the time required for the function to reach $1/e$ of its maximum value $n_x = -\ln(1 - e^{-1})/C_x$, here given by $n_\varepsilon = 11.29 < n_\sigma = 12.57$). Thus, the transient saturation timescale is on the order of $10^1$ iterations for both sensitivities. As a lower-bound, we repeat this analysis using nanovolt noise amplitude $\nu \approx 10^{-9}$ V, and observe similar functional behavior of $\bar{\varepsilon}_n$, but shifted forward in time by 40 iterations. Hence, error eventually saturates with nonzero noise.

Figure 3 depicts heat maps of the chaogate measures over the supply-bias voltage plane (top), and the iteration-bias voltage plane (bottom). As can be seen, the chaotic regions $\lambda > 0$ correspond with the regions of maximum noise and parametric sensitivity. Additionally, certain non-chaotic regions still possess nonzero sensitivity, especially when surrounded by chaos. This demonstrates that chaogates existing on the ‘edge of chaos’ accrue a nonzero error rate that approaches its maximum inside the fully chaotic regions. Hence, these error rates are inseparable from the chaotic dynamics, and there exists a design trade-off between the two.

Shown in Fig. 4 are the distribution of Lyapunov times $1/\lambda$ for the chaotic regions of parameter space $\lambda > 0$. The most probable error rate is $e^{-1}/n_{max}$. Probability densities are constructed in Python using ‘numpy.histogram(…, density = True)’ and fit using ‘scipy.stats.f.fit’.

Fig. 4. Lyapunov times $1/\lambda$ for the chaotic regions of parameter space $\lambda > 0$. The most probable error rate is $e^{-1}/n_{max}$. Probability densities are constructed in Python using ‘numpy.histogram(…, density = True)’ and fit using ‘scipy.stats.f.fit’.
iterations; saturation time in Fig. 2), meaning that errors are expected to grow exponentially within 3-10 iterations of the chaotic map. This range matches the reported bounds on the experimental error rates of the chaotic configurations of the chaogate reported in [7, Table 1]. A more comprehensive comparison to experiment is the subject of future work.

IV. CONCLUSION AND FUTURE RESEARCH

In summary, our simulations predict that 3-transistor chaotic logic gates demonstrate an ergodic sensitivity property, diverging similarly in response to dynamical, parametric, and stochastic variables. As a result, errors are predicted to grow exponentially anywhere that the map is both chaotic and subject to noise or fluctuation. These results confirm the ergodicity of the chaogate and quantify its sources of error, while addressing the lack of a sensitivity analysis in the chaogate literature. We conclude by suggesting how to utilize this ergodic sensitivity property and minimize the error rate.

The observed exponential scaling of the parametric sensitivity is useful in the context of Physically Unclonable Functions (PUFs) [17]. PUFs act as ‘digital fingerprints’ by transforming the physical differences between devices (such as differences in transistor width) into bits used for identification. Hence, chaogates may be applicable as PUFs due to their exponential parametric sensitivity. We note that $\epsilon$ is mathematically identical to the typical measure of reliability $\mu_{\text{intra}}$ in PUF literature, and a similar extension exists for $\sigma$ and the uniqueness measure $\mu_{\text{inter}}$.

More robust error correction mechanisms could improve chaogate reliability (reduce noise error) by selecting for outliers in parameter space, and attempting to preserve long-lived chaotic transients ($\lambda^{-1} > 10$ iterations) robust to noise. These outliers possess small but nonzero positive Lyapunov exponents, yielding large Lyapunov times with potential applications in computing. However, noise can dramatically reduce the lifetime of these transients [18], and balancing these effects is a challenging optimization problem for future research.

Alternatively, using an autonomous circuit as the chaotic core could potentially leverage these sensitivities as properties of the dynamics. Rather than a sample-and-hold circuit, a transmission line could be used to directly connect the input terminal to the output terminal. This produces an autonomous system in which the output of the computation is mapped directly onto the analog attractor of the AC circuit dynamics. The effects of noise could potentially hasten the approach of the chaotic transient toward the attractor, converging on a solution within a probabilistically bounded timeframe as in stochastic computing [19].

ACKNOWLEDGMENT AND DISCLAIMER

The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of Air Force Research Lab (AFRL) or the U.S. Government. The authors would also like to acknowledge helpful discussions with Domenic Forte and Rabin Acharya.

REFERENCES

[1] B. Kia, J. F. Lindner, and W. L. Ditto, “A simple nonlinear circuit contains an infinite number of functions,” IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 63, no. 10, pp. 944–948, Oct. 2016.
[2] T. Munakata, S. Sinha, and W. L. Ditto, “Chaos computing: Implementation of fundamental logical gates by chaotic elements,” IEEE Trans. Circuits Syst. I, Fundam. Theory Appl., vol. 49, no. 11, pp. 1629–1633, Nov. 2002.
[3] B. Molnár and M. Ercey-Ravasz, “Asymmetric continuous-time neural networks without local traps for solving constraint satisfaction problems,” PLoS ONE, vol. 8, Sep. 2013, Art. no. e73400.
[4] B. Kia, J. F. Lindner, and W. L. Ditto, “Nonlinear dynamics as an engine of computation,” Philos. Trans. Roy. Soc. A, Math. Phys. Eng. Sci., vol. 375, Mar. 2017, Art. no. 20160222.
[5] G. S. Rose, “A chaos-based arithmetic logic unit and implications for obfuscation,” in Proc. IEEE Comput. Soc. Annu. Symp. VLSI, 2014, pp. 54–58.
[6] V. Kohar, B. Kia, J. F. Lindner, and W. L. Ditto, “Implementing boolean functions in hybrid digital-analog systems,” Phys. Rev. Appl., vol. 7, Apr. 2017, Art. no. 44006.
[7] B. Kia, K. Mobiely, and W. L. Ditto, “An integrated circuit design for a dynamics-based reconfigurable logic block,” IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 64, no. 6, pp. 715–719, Jun. 2017.
[8] C. C. Moore, “Ergodic theorem, ergodic theory, and statistical mechanics,” Proc. Nat. Acad. Sci., vol. 111, pp. 1907–1911, Feb. 2015.
[9] M.-R. Motlagh, B. Kia, W. L. Ditto, and S. Sinha, “Fault tolerance and detection in chaotic computers,” J. Bifurcation Chaos Appl. Sci. Eng. Appl., vol. 17, no. 6, pp. 1955–1968, Jun. 2007.
[10] W. S. Sayed, A. G. Radwan, H. A. Fahmy, and A. L. El-Sedeek, “Software and hardware implementation sensitivity of chaotic systems and impact on encryption applications,” Circuits Syst. Signal Process., vol. 39, no. 11, pp. 5658–5655, 2020.
[11] W. L. Ditto, A. Miliotis, K. Murali, S. Sinha, and M. L. Spano, “Chaogates: Morphing logic gates that exploit dynamical patterns,” Chaos Interdiscipl. J. Nonlinear Sci., vol. 20, Sep. 2010, Art. no. 37107.
[12] P. Dudek and V. Juncu, “Compact discrete-time chaos generator circuit,” Electron. Lett., vol. 39, no. 20, pp. 1431–1432, 2003.
[13] V. D. Juncu, M. Rafiei-Naeini, and P. Dudek, “Integrated circuit implementation of a compact discrete-time chaos generator,” Analog Integr. Circuits Signal Process., vol. 46, pp. 275–280, Mar. 2006.
[14] R. Y. Acharya, N. F. Charlot, M. M. Alam, F. Ganji, D. Gauthier, and D. Forte, “Chaogate parameter optimization using bayesian optimization and genetic algorithm,” in Proc. 22nd Int. Symp. Qual. Electron. Design (ISQED), Apr. 2021, pp. 426–431.
[15] H. G. Schuster and W. Just, Deterministic Chaos: An Introduction, 4 ed. Weinheim, Germany: Wiley-VCH, 2005.
[16] A. A. Śliwiak, N. Chandramoorthy, and Q. Wang, “Ergodic sensitivity analysis of one-dimensional chaotic maps,” Theor. Appl. Mech. Lett., vol. 10, no. 6, pp. 438–447, 2020.
[17] Y. Gao, S. F. Al-Sarawi, and D. Abbott, “Physical unclonable functions,” Nat. Electron., vol. 3, pp. 81–91, Feb. 2020.
[18] J. Lohmann, O. D’Huys, N. D. Haynes, E. Scholl, and D. J. Gauthier, “Transient dynamics and their control in time-delay autonomous boolean ring networks,” Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top., vol. 95, Feb. 2017, Art. no. 22211.
[19] O. Camps, S. G. Stavrinides, and R. Picos, “Stochastic computing implementation of chaotic systems,” Mathematics, vol. 9, no. 4, p. 375, 2021.