Bifurcations and Fundamental Error Bounds for Fault-Tolerant Computations

J. B. Gao, Yan Qi, and José A. B. Fortes, *Fellow, IEEE*

***Abstract*—In the emerging nanotechnologies, faulty components may be an integral part of a system. For the system to be reliable, the error of the building blocks has to be smaller than a threshold. Therefore, finding exact error thresholds for noisy gates is one of the most challenging problems in fault-tolerant computations. Under the von Neumann's probabilistic computing framework, we show that computation by circuits built out of noisy NAND gates with an arbitrary number of K inputs under worst case operation can be readily described by nonlinear discrete maps. Bifurcation analysis of such maps naturally gives the exact error thresholds above which no reliable computation is possible. It is further shown that the maximum threshold value for a K-input NAND gate is obtained when $K = 5$. This implies that if one chooses NAND gate as basic building blocks, then the optimal number of inputs for the NAND gate may be very different from the conventional value of 2. The analysis technique generalizes to other types of gates and circuits that use voting to improve reliability, as well as a network built out of the so-called para-restituted NAND gates recently proposed by Sadek *et al.* Nonlinear dynamics theory offers an interesting perspective to study rich nonlinear interactions among faulty components and design nanoscale fault-tolerant architectures.**

***Index Terms*—Bifurcation, error threshold, noisy NAND gate, probabilistic/reliable computation.**

I. Introduction

RELIABLE computation using unreliable components is a central issue in emerging nanotechnologies. Conventional fault-tolerant computation aims to optimize reliability and efficiency of redundant systems which are assumed to be initially built from defect-free components, but later develop faults [1], [2]. However, in systems built out of nanoscale devices, a significant percentage of components may be faulty and impossible to replace by, or isolate from, operational ones. Such device defects may be generated during manufacturing, induced by interactions with environments or due to fundamental physical laws (e.g., occasional "violation" of the second law of thermodynamics in microscopic systems [3]–[5] and quantum decoherence [6]). If defects cannot be worked around, unlike, for example, in the case of the programmable Teramac system [7], then the faulty components have to be an active part of the system to be built. In order to perform computational tasks, different faulty components may interact in a nonlinear fashion. How may the nonlinear interactions among faulty components be studied? We are interested in the value of the probability of errors by individual gates above which no circuit built out of such gates can compute reliably, hereon referred to as gate error threshold. May the study of nonlinear interactions among faulty components help us solve the fundamental problem of determining the exact error threshold values for noisy gates? We show in this paper that worst case operations of faulty gates can be readily described by discrete nonlinear maps, from which fundamental gate error thresholds can be derived using elementary bifurcation theory. The bifurcation diagram and its associated concept of convergence to attractors allow us to interpret probabilistic computations geometrically. In turn, the geometrical interpretation makes it clear that the error thresholds obtained are exact and cannot be further improved.

The study of gate error thresholds has a long and rich history. By defining reliable computation to be the situation that the probability of a function being correctly computed by a circuit is greater than 1/2, but not necessarily 1, it has been shown, in both classical [8] and quantum [9] computations, that, when the probability of error of individual gates of a logic circuit is smaller than a certain threshold value, reliable computation is possible. By this definition, noisy NAND gates consistently giving 80% correct answers is considered reliable in a probabilistic sense. Since 80% accuracy may be too low to be useful in practice, a better term for *reliable computation* may be *consistent computation*. To be consistent with the terminology of [14], however, we shall keep using the term *reliable computation* in the paper.

When von Neumann proposed to study faulty three-input majority gates and two-input NAND gates, he advocated to treat errors in the gates thermodynamically [8]. His vision was fulfilled by Pippenger [10] by studying the problem using an information theoretic approach. Since then, more accurate error bounds have been gradually worked out for three-input majority gate [11], arbitrary $2K + 1$-input majority gate [12], [13], and two-input NAND gate [14]. We show here that nonlinear map and bifurcation approach cannot only provide another powerful solution to this fundamental problem, but yield interesting new results and insights.

The remainder of this paper is organized as follows. In Section II, we study gate error threshold values for arbitrary K-input NAND gates and derive the probability of reliable computation for a two-input NAND gate. In Section III, we briefly show that the general nonlinear map and bifurcation approach can be extended to the study of majority gates as well

Manuscript received August 7, 2004; revised December 1, 2004. This work was supported in part by the National Science Foundation under ITR Grant 0 135 946 and by the National Aeronautics and Space Administration under Award NCC 2-1363.

J. B. Gao and J. A. B. Fortes are with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611 USA (e-mail: gao@ece.ufl.edu; fortes@ufl.edu).

Y. Qi was with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611 USA. She is now with the Department of Biomedical Engineering, The Johns Hopkins University, Baltimore, MD 21218 USA (e-mail: yanqi@bme.jhu.edu).

Digital Object Identifier 10.1109/TNANO.2005.851289

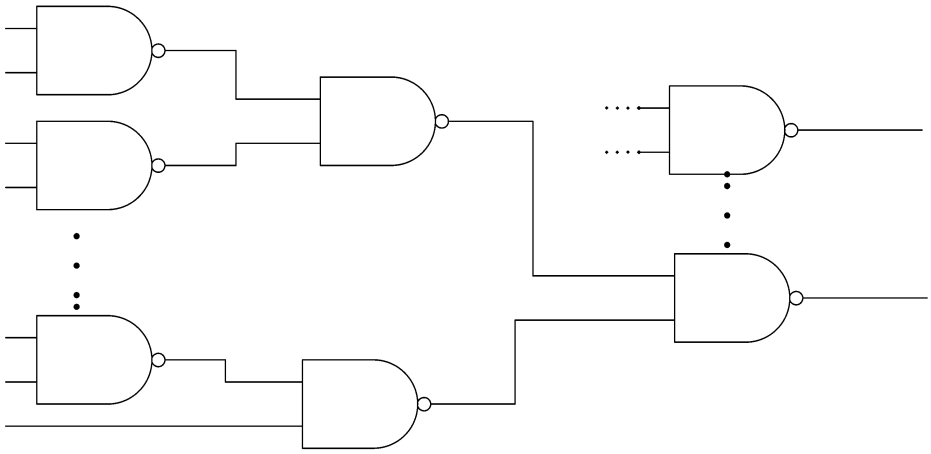


Fig. 1. Schematic of a circuit built out of two-input NAND gates.

as para-restituted NAND gates recently proposed by Sadek *et al.* [15]. In Section IV, we briefly summarize our findings and make a few concluding remarks.

II. Error Threshold Values for Arbitrary K-Input NAND Gates

Recently, a few fault-tolerant techniques for nanocomputers based on redundancy have been proposed [16]–[18]. With a device density of 10^{12} per chip and faulty components pervasive in space and time, these techniques, however, do not yield sufficiently high fault tolerance. Thus, the classic von Neumann multiplexing technique, which essentially treats faulty components as an integral part of the system, has received a revival [19]–[25]. Noisy NAND gates are building blocks of the von Neumann multiplexing technique. Understanding the nonlinear interactions among noisy NAND gates may considerably improve our understanding of the von Neumann multiplexing scheme as well as give us new insights into how one may reduce the redundancy of the scheme while maintaining the reliability of the system. In this section, we first derive the error threshold value and the probability of reliable computation for a two-input NAND gate. We then extend the analysis to the study of arbitrary K-input NAND gate.

A. Two-Input NAND Gates

1) Error Threshold: The basic computation in the von Neumann multiplexing scheme involves the circuit schematic shown in Fig. 1, where it is assumed that there are no feedback loops and the output of each gate is connected to an input of only one other gate in the circuit. For the basic gate error, we adopt the simple von Neumann model, which assumes that the gate flips the output with a probability $\epsilon \leq 1/2$, while the input and output lines function reliably. For a single NAND gate, let us denote the probabilities of the two inputs being "1" by X and Y. Since the circuits considered have neither closed loops nor fan-out, the two inputs can be treated as independent. Thus, the probability Z of the output being a "1" is

$$Z = (1-\epsilon)(1-XY) + \epsilon XY = (1-\epsilon) + (2\epsilon - 1)XY. \quad (1)$$

We observe a few interesting properties of Z: 1) when $\epsilon = 0$ and X, Y, Z take on values ether 1 or 0, (1) reduces to the standard definition of an error-free NAND gate, $Z = 1 - XY$; 2) $Z_{\max} = 1-\epsilon, Z_{\min} = \epsilon$; 3) when X (or Y) is fixed, Z linearly decreases with Y (or X); and 4) Z decreases most rapidly along $X = Y$, and hence $X = Y$ constitutes the worst case scenario.

In von Neumann's multiplexing scheme, duplicates of each output randomly become the inputs to another NAND gate [19], [23]. This motivates us to first consider the case $X = Y$. Furthermore, we label a sequence of NAND gates by index $i, i = 1, 2, \ldots, n, \ldots$, where the output of gate i becomes the input to gate $i+1$. Noticing that a NAND gate is a universal gate, in an actual computation, i may also be considered equivalent to the ith step of the computation. Equation (1) thus reduces to a simple nonlinear map

$$X_{n+1} = (1-\epsilon) + (2\epsilon - 1)X_n^2. \quad (2)$$

In such a map, ϵ is called a bifurcation (or controlling) parameter. The dynamic behavior of the map can be examined by the so-called bifurcation analysis. Computationally, this involves the following procedure. For any fixed $0 \leq \epsilon \leq 1/2$, one arbitrarily chooses an initial value for X_0 and then iterates (2). After throwing away a sufficient number of iterates so that the solution of the map has converged to some attractor, one retains, say, 100 iterates and plots those 100 points against each ϵ. Nonlinear maps may contain attractors such as fixed point solutions, periodic motions, and chaos. If the map has a globally attracting fixed point solution, then, after the transients die out, the recorded value of X_n all becomes the same. One then only observes a single point for that specific ϵ. When the solution is periodic with period m, then one observes m distinct points for that specific ϵ. When the motion becomes chaotic, one then observes as many distinct points as one records (which is 100 in our example). If one knows *a priori* the largest period of the motion M, one may only record M transient-free iterates for each ϵ, to save storage. Such a bifurcation analysis of (2) reveals that a period-doubling bifurcation occurs at

$$\epsilon_* = (3 - \sqrt{7})/4 = 0.08856 \quad (3)$$

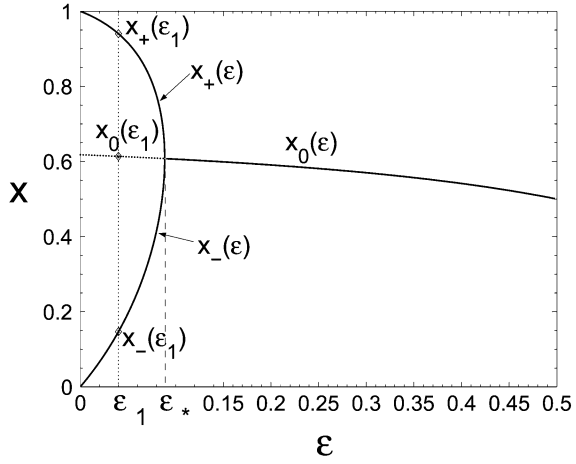


Fig. 2. Bifurcation diagram for (2). ϵ is the individual gate error probability and X is the probability of a gate output being "1." $x_0(\epsilon)$, given by (4), is unstable when $0 < \epsilon < \epsilon_*$ and stable when $\epsilon_* < \epsilon < 1/2$. In the interval $0 < \epsilon < \epsilon_*, x_\pm(\epsilon)$ are given by (8) and form the upper and lower branches of the bifurcation.

(see Fig. 2). When $\epsilon_* < \epsilon \leq 1/2$, the system has a stable fixed point solution

$$x_0 = \frac{-1 + \sqrt{4(1-\epsilon)(1-2\epsilon)+1}}{2(1-2\epsilon)}. \tag{4}$$

By stable, we mean that, if one chooses an initial condition of $x_0+\delta$, where δ is small, then x_n converges to x_0 when n is large. Mathematically, this is ensured by the condition $|f'(x_0)| = |(4\epsilon-2)x_0| = |1-\sqrt{1-4(1-\epsilon)(2\epsilon-1)}| < 1$. Exactly at $\epsilon = \epsilon_*, f'(x_0) = -1$, from which one can readily find ϵ_*, as given by (3). When $0 \leq \epsilon < \epsilon_*, x_0$ loses stability, and the motion is periodic with period 2. These two periodic points have been labeled by x_+ and x_- in Fig. 2. For convenience of our later discussions, we explicitly write down the following three equations:

$$x_0 = (1-\epsilon) + (2\epsilon-1)x_0^2 \tag{5}$$

$$x_+ = (1-\epsilon) + (2\epsilon-1)x_-^2 \tag{6}$$

$$x_- = (1-\epsilon) + (2\epsilon-1)x_+^2. \tag{7}$$

Noticing that x_0 is also a solution to (6) and (7), after factoring out the terms involving x_0, one obtains the two periodic points on the limit cycle

$$x_\pm = \frac{1 \pm \sqrt{4(1-\epsilon)(1-2\epsilon)-3}}{2(1-2\epsilon)}. \tag{8}$$

For a NAND gate to function reliably, two identical inputs of "1" or "0" should output a "0" or "1," respectively. We thus see that $0 \leq \epsilon < \epsilon_*$ is the parameter interval where the NAND gate can function. When $\epsilon > \epsilon_*$, the output x_0 can be interpreted as either "1" or "0" and, hence, is what von Neumann called a state of irrelevance. Equations (3), (4), and (8) are identical with expressions derived in [14] through a much more complicated approach. In what follows, it will become clear that bifurcation analysis provides additional insights and generalizes to K-input NAND gates.

2) Probability of Reliable Computation: Let us now try to understand why any circuit built out of NAND gates with $\epsilon < \epsilon_*$ can compute reliably with probability greater than $1/2$. Without loss of generality, we consider an individual gate which satisfies (1) and whose inputs could be outputs of other gates. Let us interpret an output larger than $x_0(\epsilon)$ (or x_0 for notational simplicity) to be "1" and an output smaller than x_0 to be "0." (More formally [14], we can interpret an output belonging to either intervals $[0, x_0-\alpha_1]$ or $[x_0+\alpha_2, 1]$ to be "0" or "1," where $\alpha_i, i = 1, 2$, are infinitesimally small positive real numbers. Since all equations studied here are continuous, whether a small open interval containing x_0 is removed or not does not affect the results of our analysis, since such infinitesimal open intervals have measure zero.) Since x_+ and x_- are points on the attracting limit cycle, we may first assume $X = Y = x_+$ or $X = Y = x_-$. Then we find, by (6) and (7), that the output Z is indeed in a desired state of "0" and "1." Next, we take $X = x_+, Y = x_-$. Will $Z = x_0 + \delta z > x_0$? To find out if $\delta z > 0$, we write $x_+ = x_0 + \delta_+, x_- = x_0 - \delta_-$. Then it follows from (1) that $\delta z = (2\epsilon-1)x_0(\delta_+ - \delta_-) - (2\epsilon-1)\delta_+\delta_-$. Notice that, when $\epsilon < 1/2, \delta_+$ and δ_- can either be analytically evaluated using (4) and (8) or determined from Fig. 2. From either approach, we find $0 < \delta_+ < \delta_-$, hence we indeed have $\delta z > 0$ and the output is again correct. The constraints $X = x_+$ and $Y = x_-$ can be relaxed and replaced by small closed intervals which include either x_+ or x_- and assuming that X and Y have converged to those small intervals. The latter approach was actually adopted by Evans and Pippenger [14] to establish that the NAND gate can compute reliably.

One may ask whether the NAND gate can still compute reliably if the inputs X and Y are very different from x_+ and/or x_-. To answer this question, let us consider the following four cases.

Case 1) $X > x_0, Y > x_0$.
Case 2) $X < x_0, Y < x_0$.
Case 3) $X > x_0, Y < x_0$.
Case 4) $X < x_0, Y > x_0$.

(with the assumption that X and Y are uniformly distributed on the interval $[0, 1]$). By (5) and recalling that Z described by (1) is a decreasing function of both X and Y, we immediately see that we have correct outputs for cases 1) and 2). Due to symmetry, cases 3) and 4) are equivalent. Hence, we only need to consider one of them, say, case 3). We write $X = x_0 + \eta_1, Y = x_0 - \eta_2$. Will we always have the correct output $Z > x_0$? The answer is not given by a simple yes or no. Instead, we need to evaluate both probabilities of $\eta_z > 0$ and $\eta_z < 0$, where η_z is defined by $Z = x_0 + \eta_z$. Again, using (1), we find $\eta_z = (2\epsilon-1)[(\eta_1 - \eta_2)x_0 - \eta_1\eta_2]$. Since $0 \leq \epsilon < 1/2$, the sign of η_z is determined by $(\eta_1 - \eta_2)x_0 - \eta_1\eta_2$. Noticing $0 \leq \eta_1 \leq 1 - x_0, 0 \leq \eta_2 \leq x_0$, we can compute the probabilities for the two cases by computing the area bounded by the curves $\eta_1\eta_2 = (\eta_1 - \eta_2)x_0, \eta_1 = 1 - x_0$, and $\eta_2 = 0$, as schematically depicted in Fig. 3. We thus find that the probability P_e for $\eta_z < 0$ is

$$P_e = x_0[1 - x_0 + x_0 \ln x_0] < x_0(1-x_0)^2$$

while the probability P_c for having a correct output is

$$P_c = x_0(1-x_0) - P_e > x_0^2(1-x_0).$$

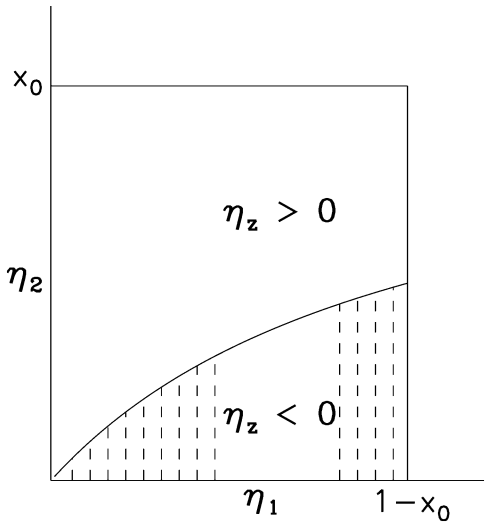


Fig. 3. Schematic for evaluating the error probability.

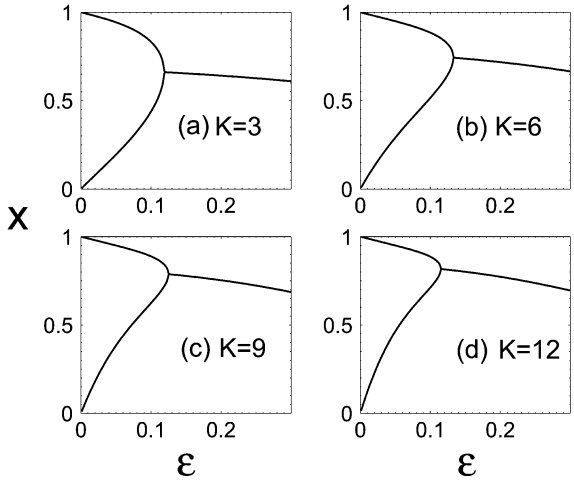


Fig. 4. Bifurcation diagrams for (10) with $K = 3, 6, 9$, and 12.

Noting $x_0 > 1/2$, we see that $P_c > P_e$. Combining the four cases considered, we find that the total probability of the gate making an error is

$$P_e(\text{overall}) = 2x_0[1 - x_0 + x_0 \ln x_0].$$

When $0 \leq \epsilon \leq \epsilon_*$, using (4), we find that $P_e(\text{overall}) \approx 10.9\%$. Hence, in this case, the probability of reliable computation is about 89.1%.

Note that our discussion on P_e and P_c is valid for the entire range of $\epsilon < 1/2$. When the gate operates under conditions of 3) and 4), we always have the probability of correct computation being larger than $1/2$, which means that, however noisy gates may be, they can always compute reliably in a probabilistic sense. On the other hand, when the gate operates under conditions 1) and 2), then, when $\epsilon > \epsilon_*$, the (conditional) error probability is 1. It is interesting to note that these discussions once again justify that $X = Y$ is the worst case scenario.

We have studied von Neumann's NAND multiplexing scheme with a high degree of redundancy [8], [19], [23]. Mathematically, the scheme is described by a Markov chain. We have found that the stationary distribution of the probability transition matrix is characterized by a unimodal distribution when $\epsilon_* \leq \epsilon \leq 1/2$, where the median of the distribution is given by x_0, while, for $\epsilon < \epsilon_*$, the stationary distribution is bi-modal peaking around x_+ and x_-, with its median again given by x_0. Hence, the basic structure of the bifurcation completely determines the multiplexing scheme. A detailed discussion appears in [23].

B. Arbitrary K-Input NAND Gate

Next we consider K-input NAND gates. Generalizing (1) and (2), we come up with the following two basic equations:

$$Z = (1-\epsilon) + (2\epsilon - 1)Y_1 Y_2 \cdots Y_K \tag{9}$$

$$\begin{aligned} X_{n+1} &= (1-\epsilon) + (2\epsilon - 1)X_n^K \\ &= 1 - X_n^K + \epsilon(2X_n^K - 1) \\ &= f(X_n). \end{aligned} \tag{10}$$

The bifurcation diagrams for (10) with $K = 3, 6, 9$, and 12 are shown in Fig. 4(a)–(d). We observe that the fixed point x_0 monotonically increases with K and that the bifurcation involved is always period-doubling. The latter makes perfect sense, since, for a sequence of NAND gates to function, the states of the outputs must flip. Such behavior can only be described by a period-2 motion.

Let us now find the bifurcation point ϵ_*. Let x_0 again be a fixed point of the map. We have

$$x_0 = 1 - x_0^K + \epsilon(2x_0^K - 1). \tag{11}$$

This is stable when $\epsilon > \epsilon_*$. When K is large, the explicit expression for x_0 may be hard to obtain. Let us not bother to do so. Instead, let us examine when x_0 may lose stability when ϵ is varied. This can be found by requiring the derivative of $f(x)$ evaluated at x_0 to be -1, $f'(x_0) = -1$ [26]. Hence

$$(2\epsilon_* - 1)Kx_0^{K-1} = -1. \tag{12}$$

Combing (11) and (12), we have

$$\left(1 + \frac{1}{K}\right)\left[\frac{1}{K(1-2\epsilon_*)}\right]^{\frac{1}{K-1}} = 1 - \epsilon_*. \tag{13}$$

For example, let us solve (13) for $K = 2$ and 3. We obtain (3) for a two-input NAND gate and

$$\epsilon_* = \frac{5}{6} - \frac{1}{6(33 - 8\sqrt{17})^{\frac{1}{3}}} - \frac{1}{6}(33 - 8\sqrt{17})^{\frac{1}{3}} \approx 0.1186$$

for a three-input NAND gate. For $K \geq 4$, analytical expressions for ϵ_* are messy or hard to obtain; however, we can still obtain the threshold values by numerically solving (13). Fig. 5 shows the variation of ϵ_* with K for $K \leq 20$. We observe that the error threshold value ϵ_* of 0.1186 for $K = 3$ is almost 34% larger than that for $K = 2$. We also note that ϵ_* assumes a maximum value of 0.1330 at $K = 5$, which is more than 50% larger than that for $K = 2$. We thus see that in situations where a two-input NAND gate is too noisy to compute reliably, it may be advantageous to use gates with more than two (but less than six) inputs.

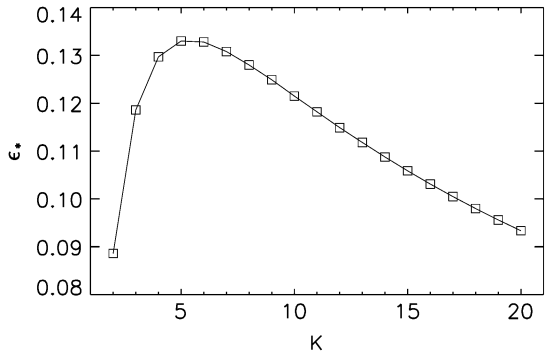


Fig. 5. Variation of ϵ_* versus K.

To end this section, let us briefly examine if the K-input NAND gate can compute reliably. We consider three cases: 1) all $Y_i, i = 1, 2, \ldots, K$ in (9) are larger than x_0; 2) all $Y_i, i = 1, 2, \ldots, K$ in (9) is smaller than x_0; and 3) otherwise. Noticing that Z decreases with $Y_i, i = 1, 2, \ldots, K$, and using (11), we find that cases 1) and 2) give desired output "0" and "1," respectively. Case 3) gives both correct and incorrect outputs, depending on actual values of $Y_i, i = 1, 2, \ldots, K$. Since $x_0 > 1/2$ and monotonically increases when K is increased, we can be sure that more than $1/2$ of the combinations in case 3) should give correct output of "1." Combining all three cases, we thus can conclude that the probability of reliable computation is larger than $1/2$, hence, the gate computes reliably in a probabilistic sense. However, when $K > 2$, the exact probability of reliable computation is harder to derive, since one has to consider a lot of different combinations for the inputs. We shall not go into all the details to derive the exact probability of reliable computation for $K > 2$ here.

III. Error Thresholds for Noisy Majority Gates and Para-Restituted NAND Gates

Here, we apply our nonlinear map and bifurcation approach to the study of noisy majority and para-restituted NAND gates. Majority gates have been widely used to construct neural network based fault-tolerant computations [24], while para-restituted NAND gates are building blocks of the so-called para-restituted network [15]. Our analysis may offer additional insights into the working mechanisms of those types of networks.

A. Noisy Majority Gate

Here we shall consider noisy majority gates first proposed by von Neumann, with $2K+1$ inputs. Let ϵ be the probability that the noisy gate makes a von Neumann error (flipping the output logic value). Assume the inputs are independent, and their probabilities of being "1" are all the same, which we shall denote by x. Let θ represent the probability of the output being "1" when the gate is fault-free. Notice that the majority gate outputs "1" when the number of excited inputs is larger than the number of unexcited inputs. Otherwise, the gate outputs "0." Thus, θ is given by

$$\theta = \sum_{i=K+1}^{2K+1} \binom{2K+1}{i} x^i (1-x)^{2K+1-i}. \tag{14}$$

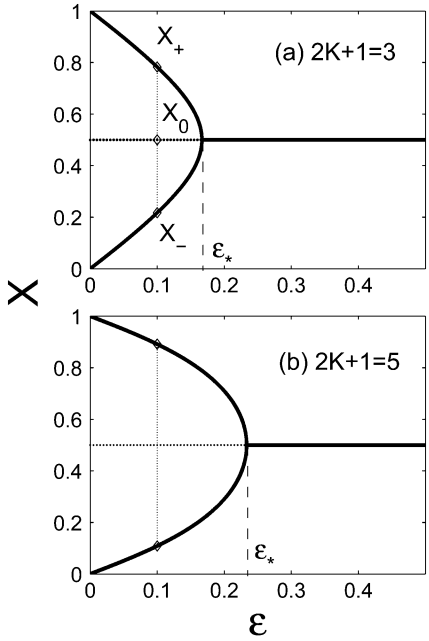


Fig. 6. Bifurcation diagrams for $2K + 1 = 3$ and 5.

It follows that the probability Z that the output is "1" is

$$\begin{aligned} Z &= \epsilon(1-\theta) + (1-\epsilon)\theta \\ &= \epsilon + (1-2\epsilon)\left[\sum_{i=K+1}^{2K+1} \binom{2K+1}{i} x^i (1-x)^{2K+1-i}\right]. \end{aligned} \tag{15}$$

Often there are situations (such as in the case of a network of gates) where one needs to consider a sequence of gates, where the output from one gate becomes the input to the next gate. This motivates us to denote the probability of the output from gate n being "1" as x_n, while the probability of the output from gate $n+1$ being "1" by x_{n+1}. We thus obtain the following iterative equation:

$$\begin{aligned} x_{n+1} &= \epsilon + (1-2\epsilon)\left[\sum_{i=K+1}^{2K+1} \binom{2K+1}{i} \cdot x_n^i \cdot (1-x_n)^{2K+1-i}\right]. \end{aligned} \tag{16}$$

Alternatively, we may interpret x_n as the probability of the input to gate n being "1," while x_{n+1} as the probability of the input to gate $n+1$ being "1." We identify that ϵ is a bifurcation parameter of the nonlinear map described by (16).

Let us now numerically investigate the bifurcations of (16) for $2K + 1 = 3$ and 5. For any fixed ϵ, we arbitrarily choose an initial x_0 and then iterate (16). We find that, when $\epsilon < \epsilon_*$, there are two fixed point solutions: one larger than 1/2 and the other smaller than 1/2. When $x_0 > 1/2$, the fixed point solution with value larger than 1/2 is eventually approached. Otherwise, the fixed point solution with value smaller than 1/2 is approached. When $\epsilon \geq \epsilon_*$, then there is only one fixed point solution of value 1/2. These verbal descriptions are depicted in Fig. 6. Mathematically, such a bifurcation is called a pitchfork bifurcation.

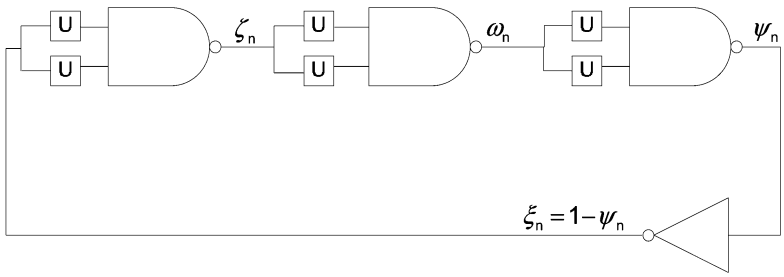


Fig. 7. Para-restituted NAND gate. It consists of three sequential NAND gates, followed by a NOT gate, which feeds back to the input of the first NAND gate and closes the loop.

To better understand the above bifurcation behavior, let us intuitively discuss how faulty majority gates function. When the gate is faulty, there are two distinct cases.

Case 1) When the error is small enough, we anticipate that we can assign unambiguously a logic "1" or "0" to the output of the gate, depending on whether the number of excited inputs is larger than that of the unexcited inputs. This is the case with $\epsilon < \epsilon_*$.

Case 2) When the error is very large, we will no longer be able to decide what logic value of the output of the gate is in. This is the case with $\epsilon \geq \epsilon_*$. This was referred to as a state of irrelevance by von Neumann.

It is interesting to note that the ϵ_* obtained by the bifurcation approach is the same as that obtained by Evans and Shulman [12] by an information theoretic approach

$$\epsilon_* = \frac{1}{2} - \frac{2^{2K-1} \cdot (K!)^2}{(2K+1)!}. \tag{17}$$

The bifurcation approach makes it evident that ϵ_* is the threshold value and cannot be further improved. It would be very interesting to prove (17) by a bifurcation approach.

B. Para-Restituted NAND Gate With and Without an Inverter

Recently, Sadek *et al.* [15] have proposed an interesting fault-tolerant scheme called para-restituted network. Its building blocks are the para-restituted units each consisting of three sequential NAND gates (see the upper part of Fig. 7). The worst case operation of a para-restituted unit can be described by iterating the nonlinear map of (2) three times. Therefore, the error threshold values for such para-restituted NAND gates is the same as a single NAND gate, i.e., 0.0886. By assuming very large bundle size N and very small gate error probability ϵ, von Neumann [8] derived an error bound of 0.0107 for ϵ. In deriving this bound, von Neumann employed a simple decision rule: a bundle is assigned "1" or "0" if the portion of excited wires is greater than $1 - \Delta$ or smaller than Δ. It is worth emphasizing that this decision rule, leaving $[\Delta, 1 - \Delta]$ as an uncertainty interval, does not depend on the nonlinearity of NAND gates and is different from that employed in our analysis. Such a decision at the output bundle of each multiplexing unit in a multistage system amounts to adding a restoration device at each step of computation. von Neumann's analysis inspired Sadek *et al.* [15] to feed the output of the third NAND gate back into the first NAND gate through a NOT gate, i.e., an inverter, as schematically shown in Fig. 7. Mathematically, the operation of the gates in Fig. 7 can be expressed by the following four equations:

$$\zeta_n \leftarrow (1-\epsilon) + (2\epsilon - 1)\xi_n^2 \tag{18}$$

$$\omega_n \leftarrow (1-\epsilon) + (2\epsilon - 1)\zeta_n^2 \tag{19}$$

$$\psi_n \leftarrow (1-\epsilon) + (2\epsilon - 1)\omega_n^2 \tag{20}$$

$$\xi_n \leftarrow 1 - \psi_n. \tag{21}$$

The first three equations are equivalent to (2). While one may relate the iterations of (18)–(21) to the operation of a network, a para-restituted NAND gate itself is in fact already a dynamical system: the states will be updated so long as the solution has not converged to a fixed point solution. To emphasize the dynamic nature of such a gate, in (18)–(21), we have used an "assign" symbol $\leftarrow$ instead of "=." The latter was used by Sadek *et al.* [15]. It was found that the error bound for such a gate is 0.010 770 9 [15]. It is interesting to note that this bound is much smaller than the error threshold value for a noisy gate, which is 0.088 56. Let us use the bifurcation approach to refind this error bound of 0.010 770 9 and try to understand why this bound is much smaller than that for a noisy NAND gate.

To iterate (18)–(21), let us follow Sadek *et al.* [15] and use $\xi_0 = 1$ as an initial condition. We find that, for each fixed ϵ, the attractor is a fixed point solution, as depicted in Fig. 8(a). It is interesting to note that around $\epsilon_* = 0.0107709$, the variation of the fixed point solution with ϵ has a huge discontinuity. Since when a para-restituted NAND gate functions, the input and output state have to be the same, beyond ϵ_*, one has to conclude that the gate no longer works, and ϵ_* is the error threshold value of the gate. However, we can understand more by iterating (18)–(21) by using other initial conditions. Another case is shown in Fig. 8(b), with $\xi_0 = 0.8$. It is interesting to note that now the variation of the fixed point solution with ϵ is continuous. In fact, one always obtains Fig. 8(b) so long as $\xi_0 \leq 0.8$. With an initial condition such as $\xi_0 = 0.89$, one obtains a bifurcation diagram which is similar to Fig. 8(a), but has a shorter upper branch. Such behavior shows that the fixed point solution in the upper branch can be reached only when ξ_0 falls within a small interval enclosing the fixed point solution. Mathematically, such an interval is called basin of attraction (BOA). The BOA for the fixed point solution in the upper branch is $(0.8, 1]$ at $\epsilon = 0$ and keeps shrinking with increasing ϵ. When ϵ is only slightly smaller than ϵ_*, the BOA for the fixed point solution in the upper branch is tiny. With ϵ close to 0, the point $\xi_0 = 0.89$ falls in the BOA for the attractor in the upper branch. However, the same point would belong to the BOA for the attractor in the

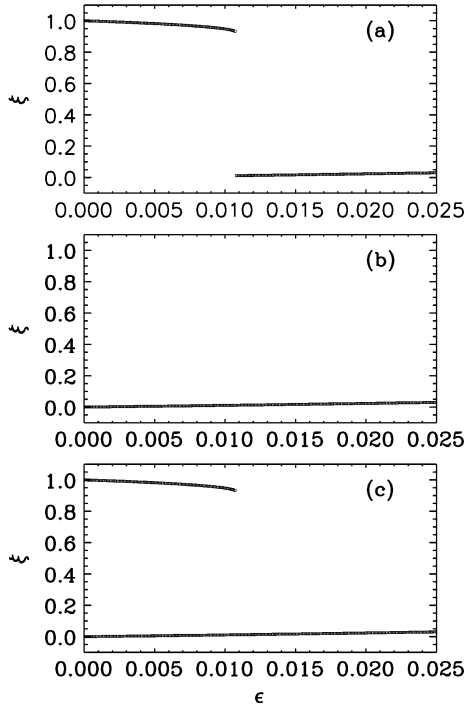


Fig. 8. Bifurcation diagrams for the para-restituted NAND gate (a) is obtained with an initial condition $\xi_0 = 1$, (b) is obtained with $\xi_0 = 0.8$, and (c) is simply the combination of (a) and (b).

lower branch with larger ϵ. Thus, the upper branch is shorter when $\xi_0 = 0.89$ is chosen as the initial condition.

It is interesting to note that, when $\epsilon \geq \epsilon_*$, the system only has one fixed point solution. That is, curves in Fig. 8(a) and (b) become identical when $\epsilon \geq \epsilon_*$. To clearly indicate this feature, we have plotted curves in Fig. 8(a) and (b) together in Fig. 8(c). Again, as is the case for the majority gate, the bifurcation involved is a pitchfork bifurcation.

Why does the error threshold for a para-restituted NAND gate correspond to a pitchfork bifurcation, while the threshold for a noisy NAND gate corresponds to a period-2 bifurcation? The answer lies in the fact that, for a para-restituted NAND gate to function, the input and output state has to be the same. However, for a NAND gate to function desirably, the input and output state has to flip—hence a period-2 oscillation.

Finally, let us understand why the error threshold for the para-restituted NAND gate is so much smaller than that for a noisy NAND gate. There are two fundamental reasons.

1) The bifurcation diagram for a noisy two-input NAND gate is asymmetric (Fig. 2). In fact, by (8), $x_+ + x_- = 1/(1-2\epsilon) \geq 1$. When $\epsilon \to 0$, the equality holds better.
2) The NOT gate requires $x_+ = 1 - x_-$. It is now clear that this is possible only when ϵ is very small. One could say that the ultimate "culprit" is the NOT gate: negation of the NOT gate has imposed such a strong constraint that the resulting error threshold is almost one order of magnitude smaller than that of a noisy NAND gate.

We should emphasize that, so long as a network only consists of NAND gates but not inverters, the error threshold value is the same as that for a single NAND gate, 0.088 56. When $\epsilon < \epsilon_* = 0.08856$, one can make the output error arbitrarily small by using a sufficiently large amount of redundancy. This follows from the observation that, when $\epsilon < \epsilon_*$, the stationary distribution of the system is bi-modal [23].

IV. Concluding Remarks

To summarize, we have found by using elementary bifurcation theory that arbitrary K-input NAND gates can be characterized by a nonlinear discrete map that has a period-doubling bifurcation. The bifurcation point gives the exact error threshold value above which circuits cannot compute reliably. It is further shown that the maximum threshold value for a K-input NAND gate is obtained when $K = 5$. This implies that, if one chooses NAND gate as basic building blocks, then the optimal number of inputs for the NAND gate may be very different from the conventional value of two. The analysis technique generalizes to other types of gates and circuits that use voting to improve reliability, as well as a network built out of the so-called para-restituted NAND gates recently proposed by Sadek *et al.* [15]. The bifurcation approach proposed is fairly technology-independent. The analysis technique is versatile in studying fundamental error thresholds for various types of gates or computing modules. It is particularly useful for the burgeoning nanotechnologies, which not only bring about novel devices such as carbon nanotube transistors [27], [28] and single electron transistors [29]–[31], but also call for unconventional system architectures and integration solutions [24], [32], [33]. The reliability of a particular nanoelectronic system is ultimately technology-dependent, since the components may have different faulty rates and dynamical behavior and may interact among themselves in different ways in different architectures. However, a good understanding of the error behavior of the basic building blocks and how errors are suppressed or aggregated through their interactions should be very useful for the design of nanoelectronic systems. These aspects can be conveniently tackled by bifurcation approach in particular and nonlinear dynamics theory in general.

While our analysis has yielded a number of interesting new results, under the framework developed here, it may be possible to gain considerable insights into the following problems: 1) analytically derive the error threshold values for arbitrary $2K+1$-input noisy majority gates; 2) find the error threshold values for noisy gates when the gate error ϵ itself is a random variable; and 3) in fault-tolerant networks using noisy NAND gates, is it possible to reduce the redundancy of the system by using K-input NAND gates, with $3 \leq K \leq 5$, so that the reliability of the system is comparable with the network using two-input NAND gates with certain redundancy?

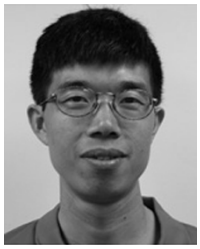
Acknowledgment

The authors would like to thank Dr. V. Roychowdhury, University of California at Los Angeles (UCLA), for many stimulating discussions.

References

[1] B. W. Johnson, *Design and Analysis of Fault Tolerant Digital Systems*. New York: Addison-Wesley, 1989.
[2] P. C. Kandellakis and A. A. Shvartsman, *Fault-Tolerant Parallel Computation*. Boston, MA: Kluwer, 1997.

[3] D. J. Evans and D. J. Searles, "Equilibrium microstates which generate 2nd law violating steady-states," *Phys. Rev. E*, vol. 50, pp. 1645–1648, 1994.
[4] G. M. Wang *et al.*, "Experimental demonstration of violations of the second law of thermodynamics for small systems and short time scales," *Phys. Rev. Lett.*, vol. 89, 2002. Art. no. 050 601.
[5] J. Liphardt *et al.*, "Equilibrium information from nonequilibrium measurements in an experimental test of Jarzynski's equality," *Science*, vol. 296, pp. 1832–1835, 2002.
[6] W. H. Zurek, "Decoherence and the transition from quantum to classical," *Phys. Today*, vol. 44, pp. 36–44, 1991.
[7] J. R. Heath, P. J. Kuekes, G. S. Snider, and R. S. Williams, "A defect-tolerant computer architecture: Opportunities for nanotechnology," *Science*, vol. 280, pp. 1716–1721, 1998.
[8] J. von Neumann, "Probabilistic logics and the synthesis of reliable organizms from unreliable components," in *Automata Studies*, C. E. Shannon and J. McCarthy, Eds. Princeton, NJ: Princeton Univ. Press, 1956, pp. 43–98.
[9] E. Knill, R. Laflamme, and W. H. Zurek, "Resilient quantum computation," *Science*, vol. 279, pp. 342–345, 1998.
[10] N. Pippenger, "Reliable computation by formulas in the presence of noise," *IEEE Trans. Inf. Theory*, vol. 34, no. 2, pp. 194–197, Mar. 1988.
[11] B. Hajek and T. Weller, "On the maximum tolerable noise for reliable computation by formulas," *IEEE Trans. Inf. Theory*, vol. 37, no. 2, pp. 388–391, Mar. 1991.
[12] W. Evans and L. J. Schulman, "Signal propagation and noisy circuits," *IEEE Trans. Inf. Theory*, vol. 45, no. 7, pp. 2367–2373, Nov. 1999.
[13] W. Evans, "Information theory and noisy computation," Ph.D. dissertation, Dept. Comput. Sci., Univ. California, Berkeley, CA, 1994.
[14] W. Evans and N. Pippenger, "On the maximum tolerable noise for reliable computation by formulas," *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 1299–1305, May 1998.
[15] A. S. Sadek, K. Nikoliæ, and M. Forshaw, "Parallel information and computation with restitution for noise-tolerant nanoscale logic networks," *Nanotechnol.*, vol. 15, pp. 192–210, 2004.
[16] M. Forshaw, K. Nikoliæ, and A. S. Sadek, "Fault-tolerant techniques for nanocomputers: Third year report," Univ. College London, London, U.K., Tech. Rep. MEL-ARI 28 667, 2001.
[17] K. Nikoliæ, A. S. Sadek, and M. Forshaw, "Architectures for reliable computing with unreliable nanodevices," in *Proc. IEEE Nanotechnology*, Piscataway, NJ, Oct. 2001, pp. 254–259.
[18] ——, "Fault-tolerant techniques for nanocomputers," *Nanotechnol.*, vol. 13, pp. 357–362, 2002.
[19] J. Han and P. Jonker, "A system architecture solution for unreliable nanoelectronic devices," *IEEE Trans. Nanotechnol.*, vol. 1, no. 4, pp. 201–208, Dec. 2002.
[20] ——, "A defect- and fault-tolerant architecture for nanocomputers," *Nanotechnol.*, vol. 14, pp. 224–230, 2003.
[21] ——, "A fault-tolerant technique for nanocomputers: NAND multiplexing," in *Proc. ASCI 8th Annu. Conf. Advanced School for Computing and Imaging*, Lochem, The Netherlands, Jun. 2002, pp. 59–66.
[22] ——, "A study on fault-tolerant circuits using redundancy," in *Proc. VLSI Multiconf. Computer Science and Engineering*, Las Vegas, NV, Jun. 2003, pp. 65–69.
[23] Y. Qi, J. B. Gao, and J. A. B. Fortes, "Markov chain and probabilistic computation: A general framework for fault-tolerant system architectures for nanoelectronics," *IEEE Trans. Nanotechnol.*, to be published.
[24] V. Beiu, "A novel highly reliable low-power nano architecture when von Neumann augments Kolmogorov," in *Proc. IEEE 15th Int. Application-Specific Systems, Architectures, Processors Conf.*, Sep. 2004, pp. 167–177.
[25] S. Roy and V. Beiu, "Multiplexing schemes for cost effective fault tolerance," in *Proc. IEEE Nanotechnology Conf.*, Aug. 2004, pp. 589–592.
[26] S. Wiggens, *Introduction to Applied Nonlinear Dynamical Systems and Chaos*. Berlin, Germany: Springer-Verlag, 1990.
[27] A. Bachtold, P. Hadley, T. Nakanishi, and C. Dekker, "Logic circuits with carbon nanotube transistors," *Science*, vol. 294, pp. 1317–1320, 2001.
[28] K. Keren, R. S. Berman, E. Buchstab, U. Sivan, and E. Braun, "DNA-templated carbon nanotube field-effect transistor," *Science*, vol. 302, pp. 1380–1382, 2003.
[29] K. Likharev, "Single-electron devices and their applications," *Proc. IEEE*, vol. 87, no. 4, pp. 606–632, Apr. 1999.
[30] S. Mahapatra, V. Pott, S. Ecoffey, A. Schmid, C. Wasshuber, J. Tringe, Y. Leblebici, M. Declercq, K. Banerjee, and A. Ionescu, "SETMOS: A novel true hybrid SET-MOS high current coulomb blockade cell for future nanoscale analog ICs," *Int. Electron Devices Meeting Tech. Dig.*, pp. 1–4, 2003.
[31] K.-W. Song, G. Baek, S.-H. Lee, D. H. Kim, K. R. Kim, D.-S. Woo, J. S. Sim, J. D. Lee, and B.-G. Park, "Realistic single-electron transistor modeling and novel CMOS/SET hybrid circuits," in *Proc. IEEE Nanotechnology Conf.*, 2003, pp. 119–121.
[32] M. B. S. C. Goldsteinc, "Nanofabrics: Spatial computing using molecular electronics," in *Proc. 28th Annu. Int. Computer Architecture Symp.*, 2001, pp. 178–191.
[33] Z. H. Zhong, D. L. Wang, Y. Cui, M. W. Bockrath, and C. M. Lieber, "Nanowire crossbar arrays as address decoders for integrated nanosystems," *Science*, vol. 302, pp. 1377–1379, 2003.



J. B. Gao received the Ph.D. degree from the University of California at Los Angeles (UCLA), in 2000.

He is currently an Assistant Professor with the University of Florida, Gainesville. He has been involved in numerous fields including nanotolerant and fault-tolerant computing, nonlinear time-series analysis, traffic modeling in communications networks, nonlinear dynamics of the Internet, characterization of non-Gaussian noise in radio communications channels, and bioinformatics. Some of the tools he has developed have been applied to study a wide range of real-world signal-processing problems arising from diverse fields such as physics, geophysics, biology, economics, and engineering.

Yan Qi received the B.Sc. degree in electrical engineering from Zhejiang University, Hangzhou, China, in 2002, the M.Sc. degree in electrical and computer engineering from the University of Florida, Gainesville, in 2003, and is currently working toward the Ph.D. degree in biomedical engineering at The Johns Hopkins University, Baltimore, MD.

Her research interests include fault-tolerant computation, nanoelectronic system architectures, statistical modeling, and network dynamics analysis in computational biology and bioinformatics.

José A. B. Fortes (S'80–M'83–SM'92–F'99) received the Ph.D. degree in electrical engineering from the University of Southern California, Los Angeles, in 1984.

From 1984 to 2001, he was on the faculty of the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN. In 2001, he joined both the Department of Electrical and Computer Engineering and the Department of Computer and Information Science and Engineering, University of Florida, Gainesville, as a Professor and BellSouth Eminent Scholar. His research interests are in the areas of network computing, distributed information-processing systems, advanced computing architectures, and nanocomputing.

Dr. Fortes was a Distinguished Visitor of the IEEE Computer Society from 1991 to 1995.