

A Novel Model for Injecting Error in Probabilistic Gates

Mohamed A. K. Abuelala
EECE Department
Cairo University
Giza, Egypt
mohamed.abuelala@ieee.org

Amr Wassal
Computer Engineering
Cairo University
Giza, Egypt
wassal@eng.cu.edu.eg

Ahmed Khattab
EECE Department
Cairo University
Giza, Egypt
akhattab@ieee.org

Hossam A. H. Fahmy
EECE Department
Cairo University
Giza, Egypt
hfahmy@alumni.stanford.edu

Abstract—Modeling inaccurate gates shows new challenges for electronic design automation (EDA) tools which mainly utilize deterministic approaches in their methods. This paper presents an abstract probabilistic model which depends mainly on the probability of error in the device. We add uniform and normal noise distributions on the probabilistic model for simple gates to assess our proposed model. Also, four different topologies of XOR gates and chain of inverter gates are used to investigate the reliability of simple probabilistic gates. This model shows accurate performance for all the gates by evaluating new metric referred to output percentage error (OPE). Finally, this metric is proved to be as the cumulative distribution function (CDF) for the fitted probability distribution function (PDF) of the injected noise.

Index Terms—EDA tools, probabilistic computing

I. INTRODUCTION

There is a vital concern about the future computing system that can solve difficult problems in an energy-efficient manner. Many features need to be granted in the next computing system according to the target application. To get the better usage of the system, it is important to understand the application's requirements as some applications can tolerate some error in their arithmetic blocks to get low-power consumption. For example, allowing about 5% loss in classification accuracy for k-means clustering algorithm can lead to 50X improvement in energy saving compared to the fully accurate classification [1]. In many fields as machine learning, pattern recognition and signal processing, accurate results are not necessary and they can accept some error in the results.

The error in logical gate may be deterministic or non-deterministic depending on its source. In fact, This is the fundamental concept of inaccurate computing which relies on the ability of many systems and applications to accept some loss of quality in the computed result. Therefore, design methodologies using unreliable devices to assemble reliable circuits and systems are developed [2], [3]. This development in the computing systems has a great impact on electronic design automation (EDA) tools, as they are used through the design flow for any system by modeling, designing, verifying and testing the specifications of this system.

The conventional EDA tools employ deterministic methods to simulate the circuits by controlling the inputs and observing the outputs. However, the evolution in the integrated circuit

(IC) industry and computing systems adds more complexity and challenges for EDA as there are many sources of error such as: the probability of switching in device, or the signal modeling in probabilities form. There are many probabilistic methods used in different areas of EDA tools to measure the circuit reliability [2], and the process variation on ICs manufacturing [4].

This paper is organized as follows: Section 2 discusses the proposed abstract model for inaccurate logic gates which can be useful in modeling the error effect of their devices in EDA tools and the assessment metric for this model. Section 3 presents the probabilistic gates using the model as well as different implementations of simple and complex gates. Investigation of simulation results for these gates in simple and complex designs is provided in Section 3 with detailed analysis of these results, and section 4 concludes the paper.

II. THE PROPOSED INACCURATE MODEL

In this section, a new model of inaccurate gates is introduced. Also, we introduce a simple metric to assess the proposed model referred to output percentage error (OPE) which is similar to error distance ED metric proposed in Liang's work [5]. OPE relates outputs of n random inputs for the inaccurate gate to the relative outputs from the accurate gate, then accumulates the number of differences between the generated outputs in the gates for same input combinations, and calculates the percentage error as shown in figure 1. In general, OPE for any compared outputs is given by:

$$OPE = \left(\frac{\sum_{i=0}^N |out(i)_{acc} - out(i)_{imprecise}|}{N} \right) * 100\% \quad (1)$$

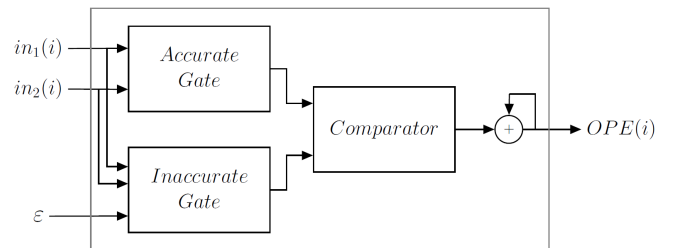


Figure 1: OPE Model

In figure 1, ϵ is a metric for gate error which depends on the injected data to represent the error. It could be correlated or uncorrelated to the inputs of gate. In this model, the injected error is uncorrelated to the applied inputs on the gates. Also, ϵ can be dependent or independent on the gates' type. For probabilistic gates, the injected error is independent on gate type and it has a known probability distribution as uniform, normal, exponential, ... etc. Table I shows the effect of ϵ when it is 0 and 1 on different gates. It is clear that in case of 0 the gate will behave normally and its functionality will be correct. In case of 1 for all inputs combination, it will behave normally too but with inverted functionality. ϵ may be 0 or 1 through the time space which will affect different input combinations with various probability.

Table I: Truth Table for AND, OR, XOR gates with $\epsilon = 0$ and 1

	Inputs		AND	OR	XOR
	ϵ	$B \ A$			
Accurate	0	0 0	0	0	0
	0	0 1	0	1	1
	0	1 0	0	1	1
	0	1 1	1	1	0
Inaccurate	1	0 0	1	1	1
	1	0 1	1	0	0
	1	1 0	1	0	0
	1	1 1	0	0	1

The probability distribution function (PDF) shows the number of times a specific discrete probability value p_i of the random variable A occurs. On the other side, $F(x)$ is the cumulative distribution function (CDF) that shows the probability of the random variable being less than a specified value x . From the definitions of PDF and CDF, the distribution of device error can be represented as a PDF, and accordingly, the gate's OPE is supposed to be equivalent to the CDF.

III. PROBABILISTIC MODEL

Applying the probabilistic gate on the proposed model is shown in figure 2. Our probabilistic model is analogous to stochastic computational models (SCMs) that is proposed in Han's work [6] but with a difference that the injected error in our model is as a result of the physical behavior of the implementing device, such as probabilistic CMOS (PCMOs) [7], or memristor. We assume this behavior can be modeled as a probability distribution function that is between $[0, 1]$. Within a specific time t , the logic gate has a finite number of inputs and outputs n . According to the error distribution and the probability of error, our model forces the output to be correct or incorrect.

The model consists of three main components: (1) Accurate gate which defines the main required functionality of the model based on its inputs, (2) Comparator between the desired probability of error in our model and the estimated physical noise. The result of comparator depends on the probability distribution function of the noise, and (3) Multiplexer selects between result of the accurate gate or its inversion based on the output of the comparator.

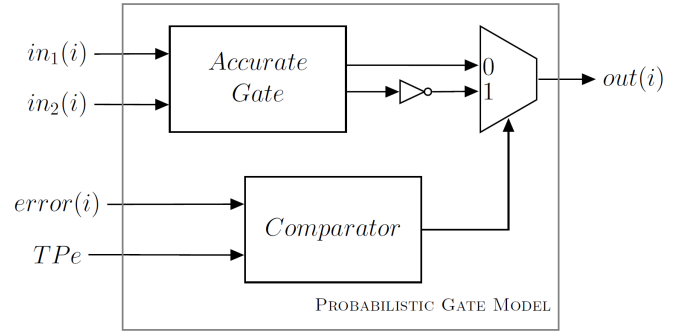


Figure 2: Probabilistic Gate Model

Unlike the traditional stochastic approaches that use the numbers which are represented as a bit-streams and process these numbers as probabilities [8], the source of error in our probabilistic gate model (ϵ) is a function of *error* and *Target Probability Error (TPE)* which are characteristics of the physical error of the device. For example, the probability of error in PCMOs is the intersection of two normal distributions [7]. Term *Error* declares the probability distribution function of the physical error in the device, and *TPE* is related to the target number of outputs forced to be incorrect for a set of inputs. Therefore, OPE of logic circuit will depend on the injected distribution. Algorithm 1 clarifies the probabilistic gate model.

Algorithm 1 Probabilistic Gate Model

```

n ← Number of Samples within Time t
Inputs: in1(i), in2(i) ∈ [0, 1] ← Time Space Inputs
error(i) ∈ [0, 1] ← Time Space Physical Error
Pe ∈ {0, 1} ← Probability of Error
inth ∈ [0, 1] ← Input Threshold
for (i ≤ n) do
    if (in1(i) ≤ inth) then
        crispin1(i) ← 0
    else
        crispin1(i) ← 1
    end if
    if (in2(i) ≤ inth) then
        crispin2(i) ← 0
    else
        crispin2(i) ← 1
    end if
    correct(i) ← crispin1(i) logic operation crispin2(i)
    incorrect(i) ← NOT(correct(i))
    if error(i) ≤ Pe then
        out(i) ← incorrect(i).
    else
        out(i) ← correct(i).
    end if
end for

```

A. Simple and Complex Gates

We assume that there is a single device that can assemble all different simple logic gates, as: NOT, AND, OR, NAND, NOR, ... etc, for example, one memristor can give 14 out of 16 logical operations [9]. We apply the probabilistic models on these gates to investigate the effect of incorrect outputs in the gate's functionality. Also, we investigate four different topologies for complex XOR gate to check the effect of simple gates from aspects as: gates count, structure, and logic levels on complex XOR gate. Figure 3 and table II show these topologies with their characteristics.

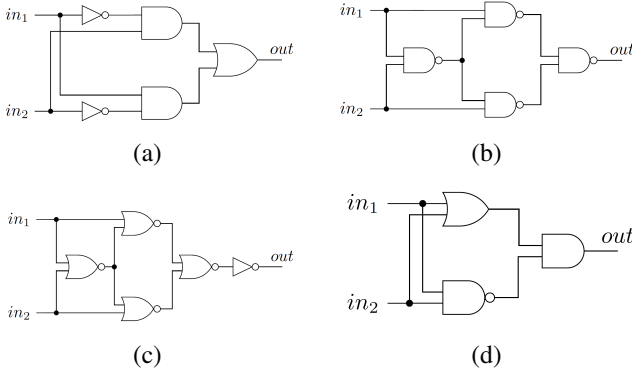


Figure 3: Different Topologies for XOR gate

Table II: Characteristics of XOR Topologies

Topology	Gate Counts	Structure	levels
a	5	Symmetric	3
b	4	Symmetric	3
c	5	Symmetric	4
d	3	Non-Symmetric	2

IV. SIMULATION RESULTS

In this section, we assess our proposed models for probabilistic gates using Matlab. We present three sections of results: (1) OPE for six simple probabilistic gates as: NOT, OR, AND, NOR, NAND, and XOR, and (2) OPE for four complex XOR gates based on the simple probabilistic model.

We apply uniform and normal distributions by generating sufficient number of random data and fit the probability distribution function to these data. For the normal distribution, the applied data has $\mu = 0.5$ and $\sigma = 0.125$. As there is no control on the physical error in the real device, we sweep P_e in the range $[0 : 1]$ to get the result of OPE. The generated random data are fitted to its corresponding probability distribution function. Then, OPE is calculated over P_e range to check the effect of the added noise on simple and complex gates. OPE is compared with the cumulative distribution function to assess the evaluation metric.

A. Simple Gates

For any probabilistic gates, $P(0)$ is the probability of correct '0' output, and $P(1)$ is the probability of correct '1' output.

These probabilities are proved to be exact to the accurate gate in case of $P_e = 0$, and inverted for $P_e = 1$. The results of OPE are found to be identical with *CDFs* of fitted *PDFs* for the injected noise either as uniform or normal distribution as shown in figure 4. OPE for simple gates is independent of the functionality of these gates, as it is only effected by the noise distribution in the gate. This proof is very useful to check the effect of different noise distributions in less reliable components in complex systems. OPE equation can be re-written as:

$$OPE = \left(\sum_{i \leq P_e} \frac{|error_i|}{N} \right) * 100\% \quad (2)$$

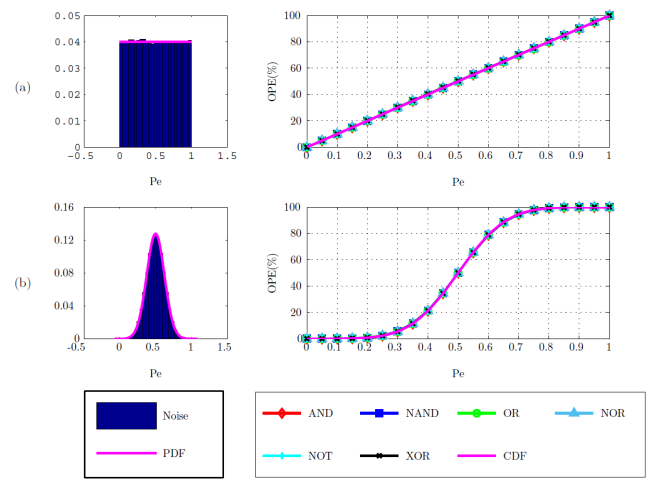


Figure 4: OPE for simple gates, and CDF of the injected noise (on the right) and Histogram of the injected noise with the fitted PDF of it (on the left) are plotted for, (a) Uniform Distribution, and (b) Normal Distribution

B. Complex Gates

The uniform and normal distributions of the injected noise are applied on each gate in the four XOR topologies to check the effect of simple unreliable gates on complex ones. The last gate relates its output probability to the input which is affected by the previous gates and error distribution.

Figure 5 shows different OPE for the XOR topologies. The results for complex probabilistic XOR gates show great improvement in OPE for P_e higher than 0.5 and little degradation in the performance for P_e less than 0.5. The analysis of the results can show that

- First and second topologies have the same OPE despite the difference constellation cause of OPE is identical between each level in first topology with its analogous level in second topology as shown in both distributions.
- Third topology which has the largest level of gates and the highest count has the best OPE among other topologies because of the last NOT gate in the constellation which has the great impact on OPE improvement.

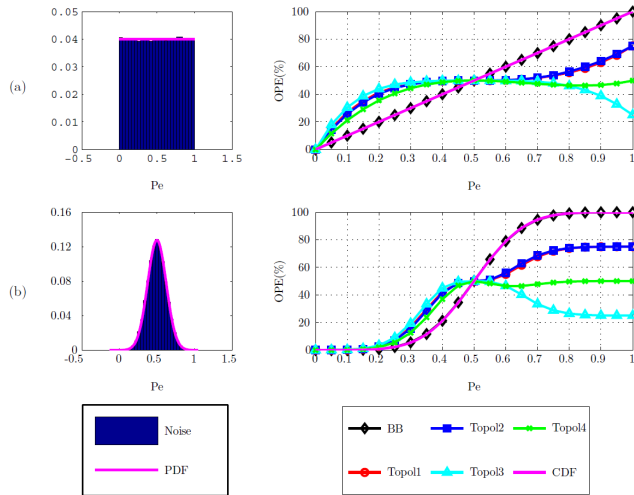


Figure 5: OPE for different XOR topologies, and CDF of the injected noise (on the right) and Histogram of the injected noise with the fitted PDF of it (on the left) are plotted for, (a) Uniform Distribution, and (b) Normal Distribution

- All levels of gates in the XOR topologies have the same OPE. Topologies with even levels as (3, 4) show better OPE rather than that with odd levels (1, 2).

V. CONCLUSION

The primary purpose of this paper is developing an algorithm to model probabilistic gates which is implemented by a device with an error that can be extracted as a probability distribution function. This can be useful in modeling probabilistic gates in EDA tools. This model is investigated for simple and complex gates by applying uniform and normal random distribution as error for these gates. Using the output gate percentage error for numbers of inaccurate samples as a metric to measure our model's efficiency proves validation of the model, as OPE for simple probabilistic gates matches CDF of injected noise for uniform and normal distributions. Integrating these simple gates to implement four different complex topologies of XOR gate, and study the effect of simple probabilistic gates on XOR functionality shows OPE improvement in topologies with even gate levels.

REFERENCES

- [1] S. Mittal, "A survey of techniques for approximate computing," *ACM Computing Surveys (CSUR)*, vol. 48, no. 4, p. 62, 2016.
- [2] J. Von Neumann, "Probabilistic logics and the synthesis of reliable organisms from unreliable components," *Automata studies*, vol. 34, pp. 43–98, 1956.
- [3] E. F. Moore and C. E. Shannon, "Reliable circuits using less reliable relays," *Journal of the Franklin Institute*, vol. 262, no. 3, pp. 191–208, 1956.
- [4] J.-J. Liou, A. Krstic, L.-C. Wang, and K.-T. Cheng, "False-path-aware statistical timing analysis and efficient path selection for delay testing and timing validation," in *Proc. of IEEE Design Automation Conference*, 2002, pp. 566–569.
- [5] J. Liang, J. Han, and F. Lombardi, "On the reliable performance of sequential adders for soft computing," in *Proc. of IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems*, 2011, pp. 3–10.

- [6] J. Han, H. Chen, J. Liang, P. Zhu, Z. Yang, and F. Lombardi, "A stochastic computational approach for accurate and efficient reliability evaluation," *IEEE Transactions on Computers*, vol. 63, no. 6, pp. 1336–1350, 2012.
- [7] B. E. Akgul, L. N. Chakrapani, P. Korkmaz, and K. V. Palem, "Probabilistic cmos technology: A survey and future directions," in *Proc. of IEEE-IFIP international conference on very large scale integration*, 2006, pp. 1–6.
- [8] A. Alaghi and J. P. Hayes, "Survey of stochastic computing," *ACM Transactions on Embedded computing systems (TECS)*, vol. 12, no. 2s, p. 92, 2013.
- [9] E. Linn, R. Rosezin, S. Tappertzhofen, U. Böttger, and R. Waser, "Beyond von neumann—logic operations in passive crossbar arrays alongside memory operations," *Nanotechnology*, vol. 23, no. 30, p. 305205, 2012.