On the Accurate Reliability Analysis of Combinational Circuits using Theorem Proving

Osman Hasan
School of Electrical Engineering and Computer Science
National University of Science and Technology
Sector H-12, Islamabad, Pakistan
Email: osman.hasan@seecs.nust.edu.pk

Jigar Patel and Sofiène Tahar
Department of Electrical and Computer Engineering
Concordia University
Montreal, Quebec, H3G 1M8, Canada
Email: {ji_p,tahar}@ece.concordia.ca

Abstract— **Reliability analysis of combinational circuits has become imperative these days due to the extensive usage of nanotechnologies in their fabrication. Traditionally, reliability analysis is done using simulation or paper-and-pencil proof methods. But, these techniques do not ensure accurate results and thus may lead to disastrous consequences when dealing with safety critical applications. In this paper, we mainly tackle the accuracy problem of reliability analysis by presenting a formal approach that is based on higher-order-logic theorem proving. The paper presents formal definitions of gate fault and reliability and utilizes them to formally verify some key reliability properties in a theorem prover. This formal infrastructure can be used to formally analyze the reliability of any combinational circuit. For illustration purposes, we utilize the proposed framework to analyze the reliability of a comparator and a full adder.**

I. Introduction

Nowadays, the ability to efficiently analyze the reliability of combinational circuits has become very challenging since they are being fabricated at the nanoscale level and are thus not only humongous in size but are also more prone to errors because of the inherent variability in the fabrication processes. A number of reliability analysis approaches have been recently proposed that tend to somewhat meet the above mentioned challenges. One approach is based on representing the erroneous behavior of a gate as a matrix, referred to as the probabilistic transfer matrix (PTM) [9]. The PTM evaluation is based on the exhaustive listing of all input and output probabilities. Therefore, a circuit with i inputs and j outputs is represented by a PTM with $2^{(i+j)}$ entries. Thus, the main problem with this approach is that, as circuits grow bigger in size, their PTMs require a significant amount of memory for storage and computational time for their reliability evaluation. Algebraic decision diagrams have been utilized to minimize these requirements but still the scalability remains a big issue for the PTM based approach. A similar, but more efficient, approach [5] is based on developing von-Neumann models, called the probability gate models (PGMs), for unreliable logic gates. It uses these models to manually analyze the reliability for a single output and an input pattern combination. Such a capability has been found to be particularly useful for the reliability modeling of certain critical paths in a circuit.

Despite their practical effectiveness, the main limitation of the above mentioned techniques is their approximate nature. This is the case because these approaches are primarily based either on paper-and-pencil proof methods or simulation. The paper-and-pencil proof methods have always some risk of an erroneous analysis due to the lengthy nature of computations involved in the reliability analysis of present age combinational circuits coupled with the human-error factor. Whereas in computer simulations, the fundamental idea is to approximately answer a query by analyzing a large number of samples and thus by its inherent nature the results cannot be termed as accurate.

The accuracy of hardware system reliability analysis results has become imperative these days because of their extensive usage in safety critical areas, like medicine, military and transportation, where an erroneous analysis could even result in the loss of human lives. Formal methods [4], which analyze a system based on pure mathematical techniques, are capable of conducting precise system analysis and thus overcome the above mentioned limitations. Given the dire need of accuracy in the area of reliability analysis of combinational circuits, probabilistic model checking, which is a state-based formal technique for analyzing random systems, has been recently used for their analysis as well [1]. Due to the inherent nature of model checking, the worst case space and time complexity for the reliability analysis of a combinational circuit with i inputs and j outputs is $O(2^{(i+j)})$. This limits the applicability of probabilistic model checking for such an analysis due to the well-known state-space explosion problem [2]. Similarly, to the best of our knowledge, it has not been possible to precisely reason about most of the commonly used reliability related statistical quantities, such as averages and variances, using probabilistic model checking so far.

We believe that due to its high expressiveness nature, higher-order-logic theorem proving [7]can be utilized to overcome the above mentioned limitations of probabilistic model checking in the domain of accurate reliability analysis of combinational circuits. This paper illustrates the practical effectiveness of this idea and thus,presents the first theorem proving based approach for the reliability analysis of combinational circuits. Our approach is primarily based on the PGM approach. We formalize the behavior of an erroneous combinational logic gate and the notion of reliability for such a gate in higher-order logic. These definitions exhibit random and probabilistic

978-1-4244-6804-1/10/$26.00 ©2010 IEEE

behaviors, due to the random nature of gate-faults, and thus have been formally defined by building upon the methodology for higher-order-logic formalization and verification of probabilistic algorithms, given in [8]. These definitions are then utilized to verify key properties associated with the reliability evaluation of combinational circuits in the PGM approach. These properties include a generalized form of von-Neumann equation, which allows us to evaluate the probability of getting a logical 1 for any combinational gate, and a generic expression that allows us to evaluate the reliability of a combinational circuit. Due to their generic nature, these formally verified results can be utilized to reason about the reliability of any combinational circuit. In order to illustrate the practical effectiveness of the proposed reliability analysis approach, we utilize it to assess the reliability of a simple comparator and a full adder. The foremost motivation of selecting these circuits is the ability to compare our results with the ones available using the PGM approach. The work described in this paper is done using the HOL theorem prover [3], which is based on higher-order logic. This choice was made to leverage upon its available probabilistic analysis framework.

The rest of the paper is organized as follows. In Section II, we present an overview of the infrastructure for the probabilistic analysis of algorithms. Section III presents the formalization details of the reliability properties that allow us to conduct the reliability analysis of combinational circuits in a theorem prover. The experimental results are given in Section IV. Finally, Section V concludes the paper.

II. Probabilistic Analysis in HOL

The foremost criteria for implementing a theorem proving based reliability analysis framework is to be able to formalize random variables in higher-order logic and verify their probabilistic properties. Random variables are fundamentally probabilistic functions that can be modeled in higher-order logic as deterministic functions with access to an infinite Boolean sequence $\mathbb{B}^{\infty}$; a source of infinite random bits [8]. These deterministic functions make random choices based on the result of popping the top most bit in the infinite Boolean sequence and may pop as many random bits as they need for their computation. When the functions terminate, they return the result along with the remaining portion of the infinite Boolean sequence to be used by other programs. Thus, a random variable which takes a parameter of type α and ranges over values of type β can be represented by the function.

$$\mathcal{F} : \alpha \rightarrow B^{\infty} \rightarrow \beta \times B^{\infty}$$

Consider the following formalization of the Bernoulli($\frac{1}{2}$) random variable that returns 1 or 0 with equal probability $\frac{1}{2}$:

```
bit=(λs.if shd s then 1 else 0,stl s)
```

where *s* is the infinite Boolean sequence and `shd` and `stl` are the sequence equivalents of the list *'head'* and *'tail'*.

Now, by formalizing a probability space of infinite Boolean sequences in higher-order logic, where the probability function $\mathbb{P}$ maps from sets of infinite Boolean sequences to real numbers between 0 and 1, we can formally prove probabilistic properties for random variables in a theorem prover [8]. For example, the following Probability Mass Function (PMF) property can be verified for the function `bit`.

$\mathbb{P}$`{s | fst(bit s)=1}=`$\frac{1}{2}$

The HOL function `fst` returns the first component of its argument, which is a pair.

The above approach has been successfully used to formalize most of the commonly used random variables and verify them based on their corresponding probability distribution properties. In this paper, we utilize the model for the Bernoulli random variables, formalized as the function `ber_rv`, and verified using the following PMF relation [8]:

Lemma 1: *PMF of Bernoulli(p) Random Variable*

```
∀ p. 0 ≤ p ∧ p ≤ 1 ⇒
  ℙ {s | fst (ber_rv p s)} = p
```

The function `ber_rv` for the Bernoulli(*p*) random variable models an experiment with two outcomes; *True* and *False*, whereas p represents the probability of obtaining a *True*.

III. Reliability Analysis in HOL

The first step in the proposed approach is to formally express the behavior of a faulty component.

Definition 1: *von-Neumann Faulty Component*

```
∀ f P e. faulty_comp f P e =
  bind(bern_rv e) (λx. bind(indep_rv_l P)
  (λy.unit(if x then ¬(f y) else (f y))))
```

where ¬ denotes the logical negation and $(\lambda x.t)$ denotes the lambda abstraction function that maps its argument x to $t(x)$. The function `indep_rv_l` accepts a list of random variables and returns the list of the same random variables such that the outcome of each one of these random variables is independent of the outcomes of all the others. The function `faulty_comp` accepts three variables, i.e., a function `f` that represents the Boolean logic functionality of the given component with data type *bool list* → *bool*, where the *bool list* represents the list of Boolean values corresponding to the inputs of the component and the return type *bool* corresponds to the output of the component, a list of Boolean random variables `P`, which corresponds to the values available at the input of the component, and the probability `e` of error occurrence in the component. The function `faulty_comp` returns a Boolean value corresponding to the output of the component with parameters `f` and `e`, when its inputs are modeled by calling the random variables in the list of random variables `P` independently. It is important to note here that the output of such a faulty component, which follows the von-Neumann model [5], is an unpredictable quantity, which is dependent on the error probability `e` and the input random variable list `P`. Therefore, this function is formally modeled using the Bernoulli random variable function, which is in turn based on the infrastructure explained in the previous section.

Gate	Theorem
AND	$\mathbb{P}\{s\|fst(faulty\_comp\ and\_gate[X_1;X_2]e\ s)\} =$ $X_1X_2 + e(1-2X_1X_2)$
NAND	$\mathbb{P}\{s\|fst(faulty\_comp\ nand\_g[X_1;X_2]e\ s)\}$ $= (1-e) + (2e-1)X_1X_2$
NOR	$\mathbb{P}\{s\|fst(faulty\_comp\ nor\_g[X_1;X_2]e\ s)\}$ $= 1 - X_2 - X_1 + X_1X_2(1-2e)+$ $e(2X_1 + 2X_2 - 1)$
Inverter	$\mathbb{P}\{s\|fst(faulty\_comp\ not\_g[X]e\ s)\}$ $= 1 - X - e + 2eX$
Majority	$\mathbb{P}\{s\|fst(faulty\_comp\ maj\_g[X_1;X_2]e\ s)\}$ $= X_1X_2 + X_1X_3 + X_2X_3 - 2X_1X_2X_3+$ $e(4X_1X_2X_3 - 2X_1X_2 - 2X_1X_3 - 2X_2X_3 + 1)$

TABLE I
PROBABILITY OF OUTPUT EQUAL TO 1 FOR COMMONLY USED GATES

Next, we verify a general expression for the probability of obtaining a $True$ or a logical 1 at the output of the von-Newmann model of a component.

Theorem 1: *General Expression for Gate Reliability*

```
∀ e f P. (0 ≤ e ≤ 1) ⇒
  (ℙ {s|fst(faulty_comp f P e s)} =
    e (1 - ℙ {s|f(fst (rv_list P s))}) +
    (1 - e) (ℙ {s|f(fst (rv_list P s))}))
```

The theorem is verified under the assumption that the error probability of the component e is bounded in the closed interval $[0, 1]$. The right-hand-side (RHS) of the theorem represents the given probability in terms of the probability of obtaining a $True$ from an error-free component, which is much easier to reason about. The HOL proof is primarily based on the independence of error occurrence and PMF of the Bernoulli random variable, given in Lemma 1.

Theorem 1 can now be used to formally reason about the probability of obtaining a logical 1 from any logical gate with functionality f. The formally verified theorems corresponding to such probabilities for some commonly used 2-input logical gates are given in Table I, which are verified under the assumption that e lies in the interval $[0, 1]$. In this table, the probability of an input x_i being equal to 1, i.e., ℙ{s|fst(xi s)}, is represented as X_i. For illustration purposes, consider the theorem for a 2-input AND gate, given in the first row of Table I. The function and_g accepts a list of Boolean values and recursively returns their logical conjunction. The variables x1 and x2 are Boolean random variables and [x1;x2] is a list containing them, which represents the inputs of the 2-input AND-gate. The proof of this theorem is based on Theorem 1 along with the fact that the probability of obtaining a logical 1 at the output of an error-free AND-gate is equal to the product of the probabilities of obtaining all logical 1's at its inputs.

Next, we formally define the reliability of a gate as the probability that the gate produces the error free result [5].

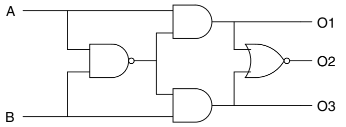

Fig. 1. A 2-bit Comparator

Definition 2: *Reliability*

```
∀ f L e. rel f L e=
  ℙ {s|fst(faulty_comp f (L e) e s) =
       fst(faulty_comp f (L 0) 0
         (snd (faulty_comp f (L e) e s)))}
```

The function rel accepts three parameters. The variables f and e represent the Boolean logic functionality of the given component and the probability of error occurrence in the component, respectively. The third variable L is a function that accepts an error probability as a *real* number and returns a list of Boolean random variables with the same type as the variable P in the function faulty_comp. The function rel returns the desired reliability of the component with functionality f and error probability e. The left-hand-side (LHS) term in the set represents the output of the component while considering the effect of error and the the RHS term represents the error free output of the given component.

Using our reliability definition, we verified the following alternative expression for it [10].

Theorem 2: *Alternate Expression for Reliability*

```
∀ f L e. 0 ≤ e ≤ 1 ⇒
  (rel f L e =
    ℙ{s|fst(faulty_comp f (L e) e s)}
    ℙ{s|fst(faulty_comp f (L 0) 0 s)} +
    (1-ℙ{s|fst(faulty_comp f (P e) e s)})
    (1-ℙ{s|fst(faulty_comp f (P 0) 0 s)}))
```

The main advantage of the above expression is that it can be used to evaluate the reliability of a logical gate in terms of the probability of attaining a logical 1 at its output, which we have already verified for the common gates (Table I).

IV. EXPERIMENTAL RESULTS

For illustration purposes, consider the comparator circuit of Figure 1. The reliability for its output $O1$ or $O3$ for an input pattern (pA,pB) can be formally expressed as follows:

Theorem 3: *Reliability for Comparator Output O1/O3*

```
∀ pA pB e. (0 ≤ e ≤ 1) ∧
  (0 ≤ pA ≤ 1) ∧ (0 ≤ pB ≤ 1) ⇒
    (rel and_g
    (λx. [ber_rv pA; (faulty_comp nand_g
       [ber_rv pA;ber_rv pB] x)]) e =
(pA(1-e+(2e-1)pApB)+
  e(1-2pA(1-e+(2e-1)pApB)))(
pA(1-(pApB)))+(1-(pA(1-e+(2e-1)pApB)+e(1
-2pA(1-e+(2e-1)pApB))))(1-pA(1-(pApB))))
```

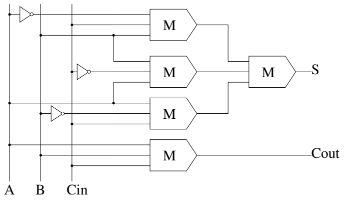

Fig. 2. Majority Gate based Full Adder

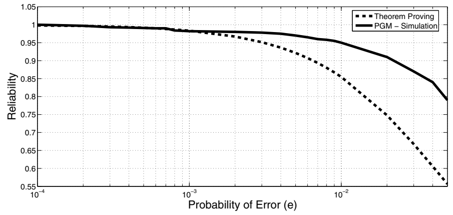

Fig. 3. Reliability for Majority Gate based Full Adder

The assumptions ensure that probability variables, `pA`, `pB` and `e` lie within the interval $[0, 1]$. The LHS of the proof goal represents the reliability of the given comparator circuit, using the function `rel` given in Definition 2, and the RHS gives the reliability in terms of the variables `pA`, `pB` and `e`. The function `and_g` represents the AND-gate in Figure 1, the output of which is the one that we are interested in finding the reliability for. It is a two input gate and its list of random variables, which corresponds to the inputs of the gate, contains two random variables. The first input is coming from a primary port and therefore we use the Bernoulli random variable function `ber_rv` with input probability `pA` of getting a logical 1 at this input for its input random variables list. The second input of the AND-gate is coming from a 2-input NAND-gate, for which the inputs are in turn connected to the primary ports A and B and these connections can be observed in the input random variable list for the function `nand_g`. The reasoning process for Theorem 3 is primarily based on the theorems given in Table I and Theorem 2. The distinguishing feature of the above theorem is its generic nature, i.e., it is true for all values of `e`, `pA` and `pB`. In other words, once this theorem is verified, it can be readily used to evaluate the reliability of outputs $O1$ or $O3$ for any values of `e`, `pA` and `pB`.

We now assess the reliability of a majority gate based full adder, given in Figure 2. Both outputs are independent so we get the overall reliability by simply multiplying the individual reliabilities of the two outputs. Since the expression that we verify in a theorem prover is generic, i.e., it is valid for any value of the gate error probability `e`, we used it to evaluate the reliability values for different values of `e` and the results are summarized in Figure 3. Reliability analysis for the same full adder circuit was done in [6] using the simulation based PGM approach and the results obtained are quite different from what we get and the difference gets more prominent as the probability of gate error increases ($> 10^{-3}$). In our opinion, the source of this discrepancy is the usage of the approximate random variable models and the inherent nature of simulation. This clearly demonstrates the effectiveness of the proposed approach because if we are getting inaccurate results for such small circuits, the impact of approximations in the simulation based analysis would be most likely greater as the circuit sizes increase.

V. CONCLUSIONS

The paper presents the first theorem proving based infrastructure for the reliability analysis of combinational circuits. Due to the formal nature of the approach, the reliability results are 100% accurate and thus can be very useful for the analysis of combinational circuits used in safety critical applications. The results of the formal reliability analysis in the paper form the main core of the proposed infrastructure and were interactively verified in HOL. These formally verified theorems then in turn can be used to assess the reliability of any combinational circuit, which has been illustrated in the paper by a couple of examples.

This work opens the doors of many new areas in the direction of theorem proving based reliability analysis of combinational circuits. First of all, we are in the process of analyzing some benchmark combinational circuits in order to further illustrate the practical effectiveness of our approach. Besides that, we are also working to develop an automatic reliability analysis tool based on the proposed approach. The reasoning process regarding the reliability theorems of combinational circuits is decidable as it utilized the theorems given in Table I and Theorem 2 only and thus can be automated. Another important aspect to work on is the scalability of our approach. We are investigating the option to partition the given circuits in smaller modules and using the reliability of these small modules to assess the reliability of the complete circuit.

REFERENCES

[1] D. Bhaduri, S. Shukla, P. Graham, and M. Gokhale. NANOPRISM: A Tool for Evaluating Granularity versus Reliability Trade-offs in Nano Architectures. In *Great Lakes Symp. VLSI,*, pages 109–112. ACM, 2004.

[2] E.M. Clarke, O. Grumberg, and D.A. Peled. *Model Checking*. The MIT Press, 2000.

[3] M.J.C. Gordon and T.F. Melham. *Introduction to HOL: A Theorem Proving Environment for Higher-Order Logic*. Cambridge University Press, 1993.

[4] A. Hall. Realising the Benefits of Formal Methods. *J. Universal Computer Science*, 13(5):669–678, 2007.

[5] J. Han, E. Taylor, J. Gao, and J. Fortes. Faults, Error Bounds and Reliability of Nanoelectronic Circuits. In *Application-Specific Systems, Architecture Processors*, pages 247–253. IEEE Computer Society, 2005.

[6] J. Han, E. Taylor, J. Gao, and J. Fortes. Reliability Modelling of Nanoelectronic Circuits. In *Conference on Nanotechnology*, pages 104–107. IEEE, 2005.

[7] J. Harrison. *Handbook of Practical Logic and Automated Reasoning*. Cambridge University Press, 2009.

[8] J. Hurd. *Formal Verification of Probabilistic Algorithms*. PhD Thesis, University of Cambridge, Cambridge, UK, 2002.

[9] S. Krishnaswamy, G. F. Viamontes, I.L. Markov, and J.P. Hayes. Accurate Reliability Evaluation and Enhancement via Probabilistic Transfer Matrices. In *Design, Automation and Test in Europe*, pages 282–287. IEEE Computer Society, 2005.

[10] E. Taylor, J. Han, and J. Fortes. Towards the Accurate and Efficient Reliability Modeling of Nanoelectronic Circuits. In *Nanotechnology Conference*, pages 395–398, 2006.