Complexity-Theoretic Obstacles to Achieving Energy Savings with Near-Threshold Computing

Antonios Antoniadis*, Neal Barcelo†, Michael Nugent†, Kirk Pruhs†, and Michele Scquizzato†
*Max-Planck Institut für Informatik
E-mail: aantonia@mpi-inf.mpg.de
†Department of Computer Science, University of Pittsburgh
E-mails: ncb30@pitt.edu, mpn1@pitt.edu, kirk@cs.pitt.edu, scquizza@pitt.edu

***Abstract*—In the traditional approach to circuit design the supply voltages for each transistor/gate are set sufficiently high so that with sufficiently high probability no transistor fails. One potential method to attain more energy-efficient circuits is *Near-Threshold Computing*, which simply means that the supply voltages are designed to be closer to the threshold voltage. However, this energy saving comes at a cost of a greater probability of functional failure, which necessitates that the circuits must be more fault tolerant, and thus contain more gates. Thus achieving energy savings with Near-Threshold Computing involves properly balancing the energy used per gate with the number of gates used. We show that if there is a better (in terms of worst-case relative error with respect to energy) method than the traditional approach then $P = NP$, and thus there is a complexity theoretic obstacle to achieving energy savings with Near-Threshold Computing.**

I. Introduction

The threshold voltage of a transistor is the minimum supply voltage at which the transistor starts to conduct current. However, if the designed supply voltage was exactly the ideal threshold voltage, some transistors would likely fail to operate as designed due to manufacturing and environmental variations. In the traditional approach to circuit design the supply voltages for each transistor/gate are set sufficiently high so that with sufficiently high probability no transistor fails, and thus the designed circuits need not be fault-tolerant. One potential method to attain more energy-efficient circuits is *Near-Threshold Computing*, which simply means that the supply voltages are designed to be closer to the threshold voltage. As the power used by a transistor/gate is roughly proportional to the square of the supply voltage [4], Near-Threshold Computing can potentially significantly decrease the energy used per gate. However, this energy savings comes at a cost of a greater probability of functional failure, which necessitates that the circuits must be more fault-tolerant, and thus contain more gates. As an example from [8], the circuit shown in Figure 1 is the traditional 6-transistor design for an SRAM cell, while the circuit shown in Figure 2 is a more fault-tolerant, and thus more suited for Near-Threshold Computing, 10-transistor design for an SRAM cell.

K. Pruhs was supported, in part, by NSF grants CCF-1115575, CNS-1253218, CCF-1421508, and an IBM Faculty Award. M. Scquizzato was supported, in part, by a fellowship of "Fondazione Ing. Aldo Gini", University of Padova, Italy.

978-1-4799-6177-1/14/$31.00

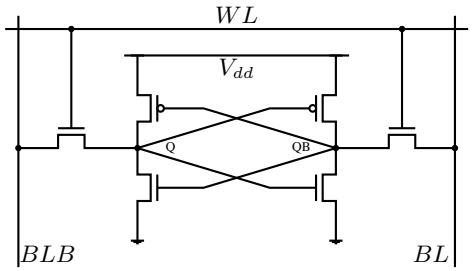

Fig. 1: Standard 6-transistor SRAM cell design.

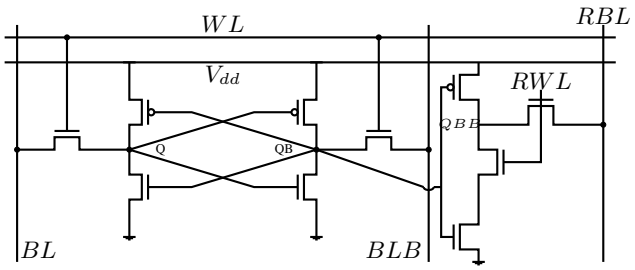

Fig. 2: A more fault-tolerant 10-transistor SRAM cell design from [5].

To understand the relationship between the supply voltage, energy/power, and error probability, consider the semi-log plot of voltage-to-failure for an SRAM cell from [8] in Figure 3. Since the relationship between voltage and the logarithm of the failure rate is approximately linear, we conclude that the error as a function of supply voltage v is approximately of the form of $\epsilon(v) = c^{-v}$, for some positive constant c. Using the fact that the energy is proportional to the square of the supply voltage [4], we conclude that the failure-to-energy function for a 65nm SRAM cell is approximately $f(\epsilon) = \Theta(\log^2(1/\epsilon))$.[1] To be a bit more general, we will assume in this paper that $f(\epsilon) = \Theta(\log^\alpha(1/\epsilon))$ for some constant $\alpha > 1$. In particular, note that error and energy are inversely related.

As the total energy used by a circuit is approximately the energy used per gate times the number of gates, achieving energy savings with Near-Threshold Computing involves prop-

[1]Throughout this paper, and unless otherwise specified, log denotes the natural logarithm.

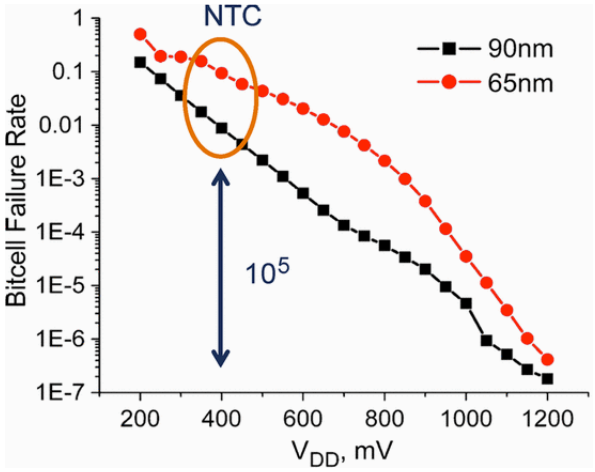

Fig. 3: Semi-log plot of voltage-to-failure for an SRAM cell from [8].

erly balancing the energy used per gate with the number of gates used. The optimization problem that a circuit designer most naturally would like to solve is:

Definition 1. Minimum Energy Circuit Problem: *Given a function F, and an error bound δ, output a circuit C and a setting v of the supply voltage such that C uses minimal energy, subject to the constraint that C computes F with probability at least $1-\delta$.*

But the ability to even approximately bound optimal circuit sizes is essentially at least as hard as the P vs. NP question,[2] and is untouchable with current mathematical knowledge. Thus in this paper we consider the following more limited optimization problem:

Definition 2. Minimum Circuit Energy Problem (MCE): *Given a circuit C, and an error bound δ, output a setting v of the supply voltage such that C uses minimal energy, subject to the constraint that C computes correctly (what C would compute if there were no errors) with probability at least $1-\delta$.*

We show in Section V that this problem is NP-hard, even in the special case that the input to the circuit is fixed. Thus if $P \neq NP$ then there is no efficient method for computing the optimal supply voltage setting. The standard fallback approach for NP-hard optimization problems is to seek algorithms that are guaranteed to produce solutions with optimal/good relative error compared to the optimal solution. In our case, an algorithm A has approximation ratio c (or equivalently worst-case relative error $c-1$) if for all inputs the energy used by the circuit with the supply voltage setting given by A is at most c times the optimal minimum energy. We show in Section III that the approximation ratio of the traditional algorithm, which sets the supply voltages for each gate sufficiently high so that with the desired probability no gate fails, is $O(\log^{\alpha} n)$. In contrast, we show in Section IV that it is NP-hard to approximate the energy within a factor of $O(\log^{\alpha-\gamma} n)$ for any constant $\gamma > 0$. Putting these two results together, we can see that there is a complexity-theoretic obstacle to achieving more energy-efficient circuits by using lower supply voltages than one obtains by the traditional high supply voltage approach. More precisely, if one could find a computationally efficient algorithm for setting supply voltages that has better worst-case relative error than the traditional approach, then $P = NP$. Hence, assuming $P \neq NP$, any proposed algorithm would either not have worst-case relative error better than the traditional approach, or would take super-polynomial time on some circuits. But of course the standard caveat applies here: as NP-hardness is a worst-case concept, this does not mean that one can not beat the energy used by the traditional approach for particular circuits of interest.

A natural question is whether one can restrict the type of circuits to some class that both contains the type of circuits that one cares about in practice, and for which (maybe nearly) energy optimal supply voltages can be efficiently computed. As a small step in this direction, we show in Section VI that there is an efficient algorithm to verify whether a particular setting of the supply voltage achieves the desired error bound if the circuit is a tree. This hints at the hardness of the Minimum Circuit Energy problem coming from "cycles" in the circuit. Finally, in Section VII we make the curious observation that there are circuits where the reliability of the output is not monotone in the reliability of the gates. Understanding this non-monotonicity seems to be the key to being able to solve the Minimum Energy Circuit problem for circuits that are trees.

We next briefly discuss related theoretical work, and then in Section II we introduce the formal models and definitions necessary to make the above discussion mathematically rigorous. It is important to note that our conclusion, that there is a complexity-theoretic obstacles to achieving energy savings with Near-Threshold Computing, does not seem to be particularly sensitive to our modeling choices. In particular, these results hold for several natural ways to model faults, and for all failure-to-energy functions that are roughly of the same form as those observed in current technologies.

A. Related Work

The paper [8] gives an excellent survey on Near-Threshold Computing.

Some of our formal models are inherited from [1]. The four main results in [1] are: (1) to compute a function with sensitivity s requires a circuit that uses energy $\Omega(s \log s)$, (2) if a function can be computed by a circuit with n reliable gates, then it can be computed by a circuit with energy $O(n \log n)$, (3) there are circuits where there is a feasible heterogeneous setting of the supply voltages which uses much less energy than any feasible homogeneous setting of the supply voltages, and (4) there are functions where there are nearly optimal energy circuits that have a homogeneous setting

[2] If one could prove that your favorite NP-complete problem required super-polynomially many gates to compute, this would prove $P \neq NP$.

of the supply voltages. A recent paper shows that almost all functions require exponential-energy circuits [3].

The study of fault-tolerant circuits started with the seminal paper by von Neumann [17]. Several subsequent papers [6], [7], [14], [15], [16], [10], [9], [12] have considered the question of how many faulty gates, each (independently) having a (small) fixed probability of failure, are required to mimic the computation of an ideal circuit with some desired probability of correctness.

II. Models and Definitions

A. Models

In this subsection we formally define the models that we will use throughout the paper. The difference in the two models described is how we model functional failures in a circuit. We first formally define Boolean functions and circuits.

A *Boolean function* h is a function from $\{0,1\}^n$ to $\{0,1\}$. A *gate* is a function $g : \{0,1\}^{n_g} \to \{0,1\}$, where n_g is the number of inputs (i.e., the *fan-in*) of the gate. We assume that the maximum fan-in of the circuit, $\max_{g \in C} n_g$, is at most a constant. A *Boolean circuit* C with n inputs is a directed acyclic graph in which every node is a gate. A *wire* is an edge of this graph. Every circuit has n gates with fan-in zero, each of which outputs one of the n inputs of the circuit. One gate is designated as the output gate, which has out-degree zero. Any Boolean function can be represented by a Boolean circuit, and every Boolean circuit computes a unique Boolean function. For any $I \in \{0,1\}^n$, we denote by $C(I)$ the output of the Boolean function computed by circuit C. The circuit is supplied with a voltage v. A voltage-to-failure function $\epsilon(v) : \mathbb{R}^+ \to (0, 1/2)$ maps a supply voltage to a probability of functional failure. We study two models of functional failures in circuits.

von Neumann Failure Model: In the *von Neumann failure model* each non-input gate g fails independently with some probability $\epsilon(v)$. When a gate fails on input $x \in \{0,1\}^{n_g}$, the output of the gate is the complement of $g(x)$, and otherwise it is $g(x)$. Equivalently, if g receives input x then with probability $1-\epsilon$ the output of g is $g(x)$, and with probability ϵ the output of g is the complement of $g(x)$, and these probabilities are independent of any other gate failures in the circuit. There is a voltage-to-energy function $E(v)$ mapping the supply voltage to the energy used by a gate with that supply voltage. The energy required by a circuit C is simply the aggregate energy used by the gates, $\sum_{g \in C} E(v)$. For convenience, we define a failure-to-energy function $f(q) := E(\epsilon^{-1}(q))$, where ϵ^{-1} denotes the inverse of the function ϵ. Thus the energy of a circuit C can be rewritten as $\sum_{g \in C} f(\epsilon(v))$.

0-default Failure Model: In the *0-default failure model* each input wire to a gate g is associated with a probability of failure ϵ, and when a wire fails it sends the default value of 0 (e.g., the wire by default carries a low voltage). More formally, for a given input $x = (b_1, b_2, \ldots, b_{n_g}) \in \{0,1\}^{n_g}$, the i^{th} input wire carries bit b_i. If $b_i = 0$, then with probability 1 gate g receives 0 as the i^{th} input bit. If $b_i = 1$, then with probability ϵ the wire fails and g receives 0 as the i^{th} input bit, and with probability $1-\epsilon$ gate g receives 1 as the i^{th} input bit. (Note that a failure can only change a wire from carrying a 1 to carrying a 0.) There is a voltage-to-energy function $E(v)$ mapping the supply voltage to the energy used by a wire with that supply voltage. The energy required by a circuit C is simply the aggregate energy used by the wires, $\sum_{w \in C} E(v)$. For convenience, we define a failure-to-energy function $f(q) := E(\epsilon^{-1}(q))$, where ϵ^{-1} denotes the inverse of the function ϵ. Thus the energy of a circuit C can be rewritten as $\sum_{w \in C} f(\epsilon(v))$.

Since the two quantities we are most interested in are failure probability and energy, and the failure-to-energy function describes a direct relationship between the two, henceforth we drop all reference to the supply voltage (e.g., we denote $\epsilon(v)$ by ϵ).

B. Definitions

We now formally define what it means for a circuit to reliably compute a function. Note that this definition could apply to either failure model described above.

Definition 3. *Given a circuit C, a probability of failure $\epsilon \in (0, 1/2)$, a value $\delta \in (0,1)$, and an input I, C is said to be* (ϵ, δ)-reliable on input *I in the von Neumann failure model (resp., 0-default failure model) if the probability that it computes the correct output $C(I)$ for input I, when each of its gates (resp., wires) fails with probability ϵ, is at least $1-\delta$. A circuit C is said to be* (ϵ, δ)-reliable *if it is (ϵ, δ)-reliable on every input I.*

Since gate error and voltage are inversely related, we can restate the Minimum Circuit Energy problem in terms of reliability as follows.

Definition 4. *Minimum Circuit Energy Problem (MCE): Given a circuit C and $\delta \in (0,1)$, output the maximum ϵ such that C is (ϵ, δ)-reliable.*

We will also consider bi-criteria approximations on energy and circuit failure.

Definition 5. *For any circuit C and $\delta \in (0,1)$, let $\epsilon^*_{C,\delta}$ be the solution to MCE(C,δ). An algorithm is a (c, d)-approximation for MCE if on any input (C, δ) it outputs a value ϵ such that C is $(\epsilon, d\delta)$-reliable and $f(\epsilon) \leq c \cdot f(\epsilon^*_{C,\delta})$.*

Note that a $(c, 1)$-approximation for MCE means that the approximation is only on energy, i.e., the algorithm outputs an ϵ such that the circuit is (ϵ-δ)-reliable and the circuit uses at most c times the energy of the circuit with the optimal choice of ϵ.

III. Polynomial-Time Approximation of the Minimum Circuit Energy Problem

In this section we show in Theorem 6 that the approximation ratio achievable by the traditional algorithm, which sets $\epsilon \approx \delta/n$, is $O(\log^{\alpha} n)$. We can actually prove a slightly more general bi-criteria approximation bound, in Theorem 7, that shows the trade-off on approximation between energy and reliability for a generalization of the traditional approach. For

the 0-default failure model, we require that the circuit is *non-trivial* in the sense that there is at least one input that causes the output to be 0, and at least one input that causes the output to be 1.

Theorem 6. *In both the von Neumann and 0-default failure models, the traditional approach is an* $(O(\log^\alpha n), 1)$*-approximation for the MCE problem on non-trivial circuits.*

Theorem 7. *Let* w *denote the total number of wires of the circuit* C*, that is,* $w = \sum_{g \in C} n_g$*, and let* φ *denote the fan-in of the output gate of the circuit. In the* 0*-default failure model, setting* $\epsilon = \delta/(\beta w)$*, for any* $\beta \geq 1$*, yields a* $((2\varphi^2/\log 2)^\alpha \log^\alpha(\beta w), 3/(2\beta))$*-approximate solution for the MCE problem on non-trivial circuits. In the von Neumann failure model, setting* $\epsilon = \delta/(\beta n)$*, for any* $\beta \geq 1$*, yields a* $((2/\log 2)^\alpha \log^\alpha(\beta n), 3/(2\beta))$*-approximate solution for the MCE problem.*

Proof. We first prove Theorem 7 for the 0-default failure model. We will choose a "high" value of ϵ for which we can prove that the probability that no wire in the circuit C fails is at least $1 - 3\delta/(2\beta)$. Since the probability that no wire in the circuit fails is $(1-\epsilon)^w$, it is sufficient to set ϵ such that $(1-\epsilon)^w \geq 1 - \frac{3}{2\beta}\delta$, that is, $\log(1-\epsilon) \geq \frac{\log\left(1-\frac{3}{2\beta}\delta\right)}{w}$. This inequality is satisfied by setting $\epsilon = \delta/\beta w$, since we can obtain $\log(1-\epsilon) = \log\left(1 - \frac{\delta}{\beta w}\right) > -\frac{3}{2}\frac{\delta}{\beta w} > \frac{\log\left(1-\frac{3}{2\beta}\delta\right)}{w}$ by applying the standard calculus inequalities $\log(1-x) > -\frac{3}{2}x$ for $0 < x \leq 0.5828$, and $\log(1-x) < -x$ for $x < 1$ and $x \neq 0$.

Now we have to show that with this choice of ϵ the energy E used by the circuit is at most a factor of $(2\varphi^2/\log 2)^\alpha \log^\alpha(\beta w)$ of the energy E^* used in an optimal solution. As for the preceding theorem, to do this we determine an upper bound to the optimal solution ϵ^*, that is the maximum value of ϵ for which the circuit is (ϵ, δ)-reliable, from which it follows a lower bound for the energy used in an optimal solution. We have two cases, depending on whether the output gate g_o of the circuit outputs 0 or 1 on input $(0, 0, \ldots, 0)$. Consider first the case, that is, $g_o(0, 0, \ldots, 0) = 0$. The other case is symmetric. Since by hypothesis the circuit is non-trivial, then the circuit does not represent the constant function $f' = 0$. Hence, there must be at least one input I to the circuit C for which $C(I) = 1$. Let q denote the probability that all the φ wires entering the output gate g_o receive value 0 when the input to the circuit is I. If we denote with p the probability that the circuit outputs the correct bit when each of its wires fails with probability ϵ, then it holds that

$$\begin{aligned}
1 - p &= \mathbf{Pr}[\text{circuit } C \text{ outputs the wrong bit}] \\
&\geq \mathbf{Pr}[\text{circuit } C \text{ outputs the wrong bit on input } I] \\
&= \mathbf{Pr}[\text{circuit } C \text{ outputs 0 on input } I] \\
&= q \cdot 1 + (1-q) \cdot \\
&\qquad \cdot \mathbf{Pr}[g_o \text{ receives an input } x \text{ s.t. } g_o(x) = 0] \\
&\geq q + (1-q)\,\mathbf{Pr}[g_o \text{ receives input } x = (0, 0, \ldots, 0)] \\
&\geq q + (1-q)\,\mathbf{Pr}[\text{all the } \varphi \text{ input wires of gate } g_o \text{ fail}] \\
&= q + (1-q)\epsilon^\varphi \\
&\geq \epsilon^\varphi,
\end{aligned}$$

and therefore, $p \leq 1 - \epsilon^\varphi$. In an optimal solution it must be that $p \geq 1 - \delta$, and thus, combining the two previous inequalities, it must hold that $1 - \delta \leq 1 - (\epsilon^*)^\varphi$, that is, $\epsilon^* \leq \delta^{1/\varphi}$. This implies a lower bound of $n \log^\alpha(1/\delta^{1/\varphi})$ for the optimal energy consumption E^*.

For the same reason, the energy consumption E of our approximate solution is $n \log^\alpha(\beta w/\delta)$. Since $\delta < 1/2$ and $\beta \geq 1$, we have

$$\begin{aligned}
E &= n\left(\log(\beta w) + \log\frac{1}{\delta}\right)^\alpha \\
&\leq n 2^{\alpha-1}\left(\log^\alpha(\beta w) + \log^\alpha\frac{1}{\delta}\right) \\
&\leq \left(\frac{2\varphi^2}{\log 2}\right)^\alpha \log^\alpha(\beta w) \cdot E^*,
\end{aligned}$$

where the first inequality follows from Jensen's inequality.

The proof for the von Neumann failure model is similar. □

We can then prove Theorem 6, showing that the traditional approach is a $(O(\log^\alpha n), 1)$-approximation, by using the same analysis with $\beta = 3/2$,

IV. Hardness of Approximation for the Minimum Circuit Energy Problem

In this section we essentially prove that it is NP-hard to obtain a better approximation than the $O(\log^\alpha n)$ obtained from the traditional approach.

Theorem 8. *In both the von Neumann and 0-default failure models, it is NP-hard to* $(\log^{\alpha-\gamma} n, 1)$*-approximate the MCE problem for any constant* $\gamma > 0$*.*

Proof. The main idea of the proof is to show that for a satisfiable circuit and an unsatisfiable circuit there is a large gap between the probability they correctly compute their input. In particular, in the case of a satisfiable input, we show that it is very unlikely for the output of the circuit to be a 1. For technical reasons we restrict γ to $\gamma \in (0, \alpha)$. It is clear that the problem is only computationally harder as γ increases. The proof makes use of some technical facts stated after this proof.

Assume by contradiction that there exists a $(\log^{\alpha-\gamma} n, 1)$-approximate algorithm A. For notational convenience, let $c = \log^{\alpha-\gamma} n$. Furthermore, let ϕ be an arbitrary 3SAT formula with m clauses and n variables. Let S_ϕ be the natural circuit for ϕ that uses at most $3m$ NOT gates to represent the negated variables, m OR gates of fan-in 3 to represent the clauses, and a tree of $m-1$ AND gates of fan-in 2 that computes the conjunction of all clauses. See Figure 4 for an example.

We choose ϵ such that

$$\left(1 - \epsilon^{\sqrt[\alpha]{c}}\right)^{m+1} + 4\epsilon^{\sqrt[\alpha]{c}} < 1 - 8\epsilon \qquad (1)$$

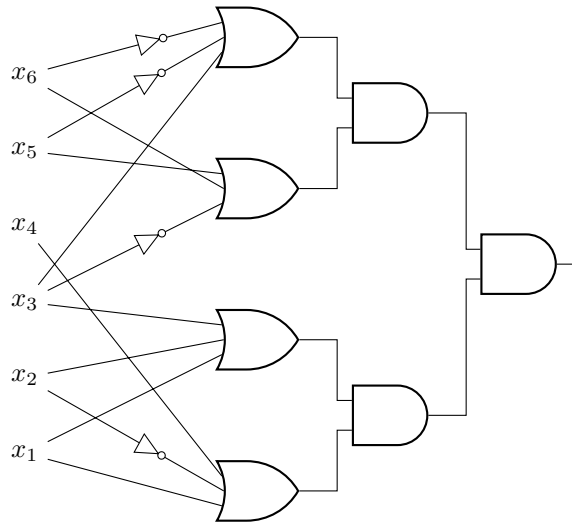

Fig. 4: The circuit S_ϕ where $\phi = (x_1 \vee \bar{x}_2 \vee x_4) \wedge (x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_3 \vee x_6 \vee x_5) \wedge (x_3 \vee \bar{x}_5 \vee \bar{x}_6)$.

and let $\delta = 8\epsilon$. We will show later that such an ϵ must exist. Now, consider the output of A, ϵ_A on S_ϕ with input δ. We claim that ϕ is satisfiable if and only if $f(\epsilon_A) > c\log^\alpha(\frac{1}{\epsilon})$. In the first case, assume ϕ is satisfiable. Consider S_ϕ where each gate fails independently with probability ϵ_A (or wire in the 0-default model), and the input x such that $\phi(x) = 1$. Let E_0 be the event that each of the OR gates receives at least 1 positive input, E_1 be the event that all of the OR gates output a 1 and E_2 be the event that S_ϕ outputs a 1. By Lemma 11, we know that $\mathbf{Pr}[E_2] \leq (1-\epsilon_A)^{m+1} + 4\epsilon_A$. Furthermore, by Inequality 1, if $\epsilon_A = \epsilon^{\sqrt[\alpha]{c}}$ we have

$$\mathbf{Pr}[E_2] \leq (1 - \epsilon^{\sqrt[\alpha]{c}})^{m+1} + 4\epsilon^{\sqrt[\alpha]{c}} < 1 - 8\epsilon = 1 - \delta. \quad (2)$$

Note that for $\epsilon_A \in [\epsilon^{\sqrt[\alpha]{c}}, 1/2)$, the quantity $(1-\epsilon_A)^{m+1} + 4\epsilon_A$ is maximized at $\epsilon_A = \epsilon^{\sqrt[\alpha]{c}}$. Therefore, we must have $\epsilon_A < \epsilon^{\sqrt[\alpha]{c}}$, otherwise by Inequality 2 the probability that S_ϕ is correct would not be within $1-\delta$, contradicting that A is a $(\log^{\alpha-\gamma}(n), 1)-$approximation. Further since f is decreasing, $f(\epsilon_A) > f(\epsilon^{\sqrt[\alpha]{c}}) = c\log^\alpha(\frac{1}{\epsilon})$.

Now, assume that ϕ is unsatisfiable. Consider S_ϕ with an arbitrary input x where each gate fails independently with probability ϵ_A (or wire in the 0-default model). Note since ϕ is unsatisfiable, $\phi(x) = 0$. Let the events E_0, E_1 and E_2 be defined in the same way as before. Using the bounds on $\mathbf{Pr}[E_2|E_1]$ and $\mathbf{Pr}[E_2|\neg E_1]$ from the proof of Lemma 11 we have,

$$\begin{aligned}
\mathbf{Pr}[E_2] &= \mathbf{Pr}[E_2|E_1]\,\mathbf{Pr}[E_1] + \mathbf{Pr}[E_2|\neg E_1]\,\mathbf{Pr}[\neg E_1] \\
&= (1-\epsilon_A)\,\mathbf{Pr}[E_1] + (4\epsilon_A)\,\mathbf{Pr}[\neg E_1] \\
&= (1-\epsilon_A)(\mathbf{Pr}[E_1|E_0]\,\mathbf{Pr}[E_0] \\
&\qquad + \mathbf{Pr}[E_1|\neg E_0]\,\mathbf{Pr}[\neg E_0]) + (4\epsilon_A)\,\mathbf{Pr}[\neg E_1] \\
&\leq (1-\epsilon_A)\left((1-\epsilon_A)^m(3\epsilon_A) + (1-\epsilon)^{m-1}(\epsilon_A)\cdot 1\right) \\
&\qquad + (4\epsilon_A)\,\mathbf{Pr}[\neg E_1] \\
&\leq 8\epsilon_A.
\end{aligned}$$

Therefore for all inputs, the probability that S_ϕ is correct is at least $1 - 8\epsilon_A$. So note that if $\epsilon_A = \epsilon$, the probability S_ϕ is correct is at least $1-\delta$. This shows that $\epsilon^* \geq \epsilon$. By the definition of A being $(c,1)$ approximate this means that $f(\epsilon_A) \leq cf(\epsilon) = c\log^\alpha(\frac{1}{\epsilon})$. This shows that we can determine the satisfiability of ϕ using A. If $f(\epsilon_A) > c\log^\alpha(\frac{1}{\epsilon})$, ϕ is satisfiable, and otherwise if $f(\epsilon_A) \leq c\log^\alpha(\frac{1}{\epsilon})$, ϕ is not satisfiable. The last thing to do is show the existence of an ϵ satisfying Inequality 1. Consider $\epsilon = \left(\frac{1}{m+1}\right)^{\frac{1}{\sqrt[\alpha]{c}}}$. Then, $(1-\epsilon^{\sqrt[\alpha]{c}})^{m+1} + 4\epsilon^{\sqrt[\alpha]{c}} \leq e^{-\epsilon^{\sqrt[\alpha]{c}}(m+1)} + 4\epsilon^{\sqrt[\alpha]{c}} = e^{-1} + \frac{4}{m+1}$. Also, $1 - 8\epsilon = 1 - 8\left(\frac{1}{m+1}\right)^{\frac{1}{\sqrt[\alpha]{c}}}$. Note that $e^{-1} + \frac{4}{m+1} < 1 - 8\left(\frac{1}{m+1}\right)^{\frac{1}{\sqrt[\alpha]{c}}}$ since $\lim_{m\to\infty} 8\left(\frac{1}{m+1}\right)^{\frac{1}{\sqrt[\alpha]{c}}} \leq \lim_{m\to\infty} 8\left(\frac{1}{n}\right)^{\frac{1}{\log^{1-\frac{\gamma}{\alpha}} n}} \to 0$. □

We now state some technical lemmas used in the above proof.

Lemma 9. *The recurrence $p_i = p_{i-1}^2(1-\epsilon) + (1-p_{i-1}^2)\epsilon$, $p_0 = 1$ satisfies $p_i \leq p_{i-1}$ for all i.*

Lemma 10. *Let $\epsilon = (1/(m+1))^{1/\sqrt[\alpha]{\log^{\alpha-\gamma}(n)}}$ and let $p_i = p_{i-1}^2(1-\epsilon) + (1-p_{i-1}^2)\epsilon$, with $p_0 = 1$. Then, for m bigger than some constant M_0, $p_{\log_2 m} \leq 3\epsilon$.*

Lemma 11. *Let ϕ be some satisfiable 3SAT formula with n variables and x be the input such that $\phi(x) = 1$. Then, in both the von Neumann model and the 0-default model, the probability that S_ϕ outputs a 1 is bounded above by $(1-\epsilon)^{m+1} + 4\epsilon$, where $\epsilon = (1/(m+1))^{1/\sqrt[\alpha]{\log^{\alpha-\gamma}(n)}}$.*

Proof. We first show this holds in the von Neumann failure model. Let g_o be the output gate of C (the root of the tree of AND gates). Further, let E_0 be the event that each of the OR gates receives at least 1 positive input, E_1 be the event that all of the OR gates output a 1 and E_2 be the event that g_o outputs a 1. We first calculate $\mathbf{Pr}[E_2|E_1]$. This is the probability that the tree of $n-1$ AND gates outputs a 1 when all the inputs to the leaves are 1. Let p_i be the probability that a gate on the i^{th} level outputs a 1. We define the input to the leaves to be at level 0. Note that $p_0 = 1$, and for $i > 0$, we can write p_i as a recurrence in the form, $p_i = p_{i-1}^2(1-\epsilon) + (1-p_{i-1}^2)\epsilon$. Further, since $p_1 = (1-\epsilon)$, and by Lemma 9, the sequence p_i is decreasing as $i \to \infty$ we have that $\mathbf{Pr}[E_2|E_1] \leq (1-\epsilon)$. Next, we bound $\mathbf{Pr}[E_2|\neg E_1]$. Let A denote the event that g_o receives two $1's$ as input. We have,

$$\begin{aligned}
\mathbf{Pr}[E_2|\neg E_1] &= \mathbf{Pr}[E_2|\neg E_1 \wedge A]\,\mathbf{Pr}[A|\neg E_1] \\
&\quad + \mathbf{Pr}[E_2|\neg E_1 \wedge \neg A]\,\mathbf{Pr}[\neg A|\neg E_1] \\
&\leq (1-\epsilon)\,\mathbf{Pr}[A|\neg E_1] + \epsilon.
\end{aligned}$$

The last thing to do is bound $\mathbf{Pr}[A|\neg E_1]$. Informally, we first argue that the probability of getting a 1 to the root of the tree is only increased if E_1 occurs, that is all leaves have value 1. After that, we can use the recurrence to show that for sufficiently large trees this probability is $O(\epsilon)$. More formally,

for some fixed gate g', let p_L be the probability the left input is 1 and p_R be the probability the right input is 1. Then, if $p_{g'}$ denotes the probability g' outputs a 1, we have $p_{g'} = (p_L p_R)(1-\epsilon) + (1 - p_L p_R)\epsilon$. Taking the partial derivative with respect to p_L or p_R shows that $p_{g'}$ will increase as p_L or p_R increase. This implies that $\mathbf{Pr}[A|\neg E_1] \leq \mathbf{Pr}[A|E_1]$, since for every leaf, the probability of having a 1 will not decrease, and therefore by induction on the levels of the tree, every gate will have an increased probability of outputting a 1. Let h be the height of the tree. Then, note that $\mathbf{Pr}[A|E_1] = p_h^2 \leq p_h$ as defined by the recurrence in Lemma 9. However, since $h = \log_2 m$ by Lemma 10 $p_{\log_2 m} \leq 3\epsilon$ and therefore $\mathbf{Pr}[A|\neg E_1] \leq 3\epsilon$ and futher, $\mathbf{Pr}[E_2|\neg E_1] \leq 4\epsilon$. We are now ready to calculate the probability that $S_\phi(x)$ outputs a 1. We have,

$$\begin{aligned}
\mathbf{Pr}[E_2] &= \mathbf{Pr}[E_2|E_1]\,\mathbf{Pr}[E_1] + \mathbf{Pr}[E_2|\neg E_1]\,\mathbf{Pr}[\neg E_1] \\
&= (1-\epsilon)\,\mathbf{Pr}[E_1] + 4\epsilon\,\mathbf{Pr}[\neg E_1] \\
&= (1-\epsilon)(\mathbf{Pr}[E_1|E_0]\,\mathbf{Pr}[E_0] \\
&\qquad + \mathbf{Pr}[E_1|\neg E_0]\,\mathbf{Pr}[\neg E_0]) + 4\epsilon\,\mathbf{Pr}[\neg E_1] \\
&\leq (1-\epsilon)\left((1-\epsilon)^m \cdot 1 + (1-\epsilon)^{m-1}(\epsilon)\cdot 1\right) + 4\epsilon \\
&\leq (1-\epsilon)^{m+1} + 4\epsilon.
\end{aligned}$$

To see that this holds in the 0-default model, note that $\mathbf{Pr}[E_2|\neg E_1] = 0 \leq 4\epsilon$ since a 0 wire will never flip to a 1. Using this we can make an identical calculation to the above to get that $\mathbf{Pr}[E_2] \leq (1-\epsilon)^{m+1} + 4\epsilon$. □

We end by noting that a slight modification of the proof of Theorem 8 can be used to prove the following more general theorem.

Theorem 12. *It is NP-hard to (c,d)-approximate the MCE problem in both the von Neumann and 0-default failure models for all $c > 1$ and d such that $\lim_{m\to\infty} 8d\left(\frac{1}{m+1}\right)^{\frac{1}{\sqrt[d]{c}}} \to 0$.*

V. Hardness of Determining (ϵ,δ)-reliability on Fixed Inputs

In this section we prove the following theorem.

Theorem 13. *In the 0-default failure model, given ϵ, δ, C, and I, it is NP-hard to determine if C is (ϵ,δ)-reliable on I.*

The section proceeds as follows. Our reduction is from the gap-3SAT problem, which is known to be NP-hard for certain parameters, so we begin by formally defining this problem. We then bound the probability that the natural 3SAT circuit, S_ϕ, outputs a 1 when given a random input both when ϕ is satisfiable, and when at most $15/16$ fraction of the clauses of ϕ are satisfiable. Finally, we introduce a circuit N_k that, in the presence of failures, can be used to randomize our input.

First we must introduce the gap-3SAT$[\alpha,\beta]$ problem (with $\alpha \leq \beta$), as the NP-hardness reduction will be from this problem. The problem is as follows: Given a 3SAT instance, output "YES" if at least a β fraction of the clauses are satisfiable, "NO" if at most an α fraction of the clauses are satisfiable, and either "YES" or "NO" otherwise (i.e., such inputs are not given). The hardness of this problem for certain values of α and β follows from the PCP Theorem [2], and in particular, Håstad proved the following theorem, giving the best possible values for α and β.

Theorem 14 (Håstad [11])**.** *Gap-3SAT[$7/8+\epsilon$, 1] is NP-hard for all $\epsilon > 0$.*

The reduction is from the hardness of gap-3SAT[15/16,1]. We use as our main circuit the standard 3SAT circuit S_ϕ used elsewhere in this paper (see Figure 4 and the related discussion). As we have seen, if the tree of AND gates does not receive all 1's, then with probability 1 the output is 0. Thus, intuitively, if we could give S_ϕ a random input, then (i) if ϕ is satisfiable, on the satisfying input S_ϕ is much more likely to output a 1 than on any other input, and (ii) if ϕ is not satisfiable, then any assignment satisfies a fraction of at most $15/16$ of the clauses, so a large number (for example, at least a $n/16$) of wires would have to fail for S_ϕ to be likely to output a 1. We first bound the probability that S_ϕ outputs a 1 when receiving an almost random input in the two cases when there exists a satisfying assignment and when at most a $15/16$ fraction of the clauses can be satisfied. We then show that it is possible with a polynomially sized circuit to create an almost random input from a fixed input, and use this to complete the reduction.

Lemma 15. *Let ϕ be a 3SAT formula and S_ϕ be the circuit for ϕ, where each wire fails independently with probability ϵ. Suppose that each input to S_ϕ is a 1 with probability at least $1/2-\gamma$ and at most $1/2+\gamma$. Then, in the 0-default failure model:*

1) *If ϕ is satisfiable, then $\mathbf{Pr}[S_\phi \text{ outputs a } 1] \geq \left(\frac{1}{2}-\gamma\right)^n (1-\epsilon)^{5m}$.*
2) *If at most a $15/16$ fraction of the clauses of ϕ are satisfiable, then $\mathbf{Pr}[S_\phi \text{ outputs a } 1] \leq (3\epsilon)^{m/16}$.*

Proof. Let O be the random output of the circuit S_ϕ and A be the random event that the tree of AND gates of S_ϕ receives all 1's as input. Then clearly $O = 1$ if A occurs and none of the wires within the tree of AND gates fail, and $O = 0$ otherwise. Therefore, $\mathbf{Pr}[O=1] = (1-\epsilon)^{2m-1}\,\mathbf{Pr}[A]$.

1) **ϕ is satisfiable.** Let E be the event that S_ϕ receives a satisfying assignment as input. The probability E occurs is at is at least $(\frac{1}{2}-\gamma)^n$, since this is a lower bound on S_ϕ recieving any fixed input. Further, if none of the wires in entering the OR gates fail (the wires entering NOT gates in the clauses can only fail and output 1, which only increases the probability that$O=1$), then A occurs, so

$$\mathbf{Pr}[A|E \wedge \phi \text{ is satisfiable}] \geq (1-\epsilon)^{3m}.$$

Clearly, the probability that A occurs if S_ϕ does not receive satisfying assignment as input is at least 0, so the first statement of the lemma follows since

$$\begin{aligned}
&\mathbf{Pr}[O=1|\phi \text{ is satisfiable}] \geq \\
&\quad (1-\epsilon)^{2m-1}\,\mathbf{Pr}[A|\phi \text{ is satisfiable}] \geq
\end{aligned}$$

$$(1-\epsilon)^{2m-1}\,\mathbf{Pr}[E|\phi \text{ is satisfiable}]\cdot \mathbf{Pr}[A|E\wedge\phi \text{ is satisfiable}] \geq \left(\frac{1}{2}-\gamma\right)(1-\epsilon)^{5m-1}.$$

2) **At most a $15/16$ fraction of the clauses of ϕ are satisfiable.** For this case, every assignment satisfies at most a $15/16$ fraction of the clauses. Thus we have that an upper bound on A occurring is if at least one of the wires associated with not gates in every clause that is not satisfied fails (if a wire entering an OR gate fails the gate will output 0), and all other gates do not fail. Thus we have that $\mathbf{Pr}[A|\phi \text{ is not satisfiable}] \leq (3\epsilon)^{m/16}$, and therefore $\mathbf{Pr}[O=1|\phi \text{ is not satisfiable}] \leq (3\epsilon)^{m/16}$. □

The following circuit will be useful in the reduction.

Definition 16. *N_k is the circuit consisting of one input bit connected to a single line of k NOT gates, i.e., the output of the ith NOT gate is the input to the $i+1$st NOT gate, for $i \in [k-1]$.*

If no gate in N_k fails, the output on input bit b is $(b+k) \mod 2$. However, if each of these gates fail independently with probability ϵ, then the output is random and, for k large enough, will be b with probability very close to $\frac{1}{2}$. Consider the Markov chain M with two states that correspond to the output bit after a certain number of NOT gates, and transitions with probabilities based on whether or not the wire entering the current NOT gate fails. If we label one state "1" and the other "0", then the output of N_k is identical to the output of starting M in state b and running for k steps. The transition from the 0 state to the 1 state happens with probability 1, since the wire cannot fail in this case. On the other hand, the transition from the 1 state to the 0 state only happens with probability $1-\epsilon$, and the chain stays in the 1 state with probability ϵ. It is easy to verify that this chain is irreducible, aperiodic, and reversible. The transition matrix is

$$M = \begin{bmatrix} 0 & 1-\epsilon \\ 1 & \epsilon \end{bmatrix}.$$

The eigenvalues of M are 1 and $\epsilon - 1$, and the stationary distribution of M is $\frac{1-\epsilon}{2-\epsilon}$ in state 0, and $\frac{1}{2-\epsilon}$ in state 1, so the number of steps $k(\rho)$ until we are ρ away from the stationary distribution is $k(\rho) \leq \frac{1}{\epsilon}\log\left(\frac{2-\epsilon}{\rho(1-\epsilon)}\right)$. For a more in depth discussion of Markov chains and mixing times, see, e.g., [13]. By setting $\rho = 0.05$, we obtain the following observation.

Observation 17. *Suppose each wire of N_k fails independently with probability $\epsilon < 1/10$. Then in the 0-default failure model if $k \geq \log(44)/\epsilon$, we have that $0.4 \leq \mathbf{Pr}[N_k(b)=b] \leq 0.6$.*

We can now finish the reduction.

Proof of Theorem 13. The reduction is from gap-3SAT[15/16,1]. Let ϕ be a 3SAT formula that is either satisfiable or at most a $15/16$ fraction of the clauses can be satisfied. Without loss of generality, we can assume the assignments of all 1's and all 0's do not satisfy ϕ, and that there are at least n clauses in ϕ. We set $\epsilon = 1.4 \times 10^{-7}$ (a constant). Construct a circuit S'_ϕ that is S_ϕ except that each input first passes through a N_k circuit, where $k = \lceil \log(44)/\epsilon \rceil$, and thus S'_ϕ is polynomial in size and logarithmic in depth. We fix the input to this circuit to be the input of all 1's, so the correct output of S'_ϕ is 0. By Observation 17, the output of each N_k circuit is 1 with probability at least $\frac{1}{2}-\gamma$ and at most $\frac{1}{2}+\gamma$ for $\gamma = 0.1$. We set $\delta = (3\epsilon)^{m/16}$. By Lemma 15 (since S'_ϕ is incorrect if it outputs 1), if we show that $(3\epsilon)^{m/16} < (0.4)^n(1-\epsilon)^{5m}$ then it is NP-hard to determine whether or not S'_ϕ outputs correctly with probability at least $1-\delta$. Rearranging the exponents and noting that $n \leq m$, we obtain that $3\epsilon < (0.4)^{16}(1-\epsilon)^{80}$, and it is easy to verify that this inequality holds for our choice of ϵ. □

In the von Neumann failure model, we were unable to prove that determining if a circuit is (ϵ,δ)-reliable is NP-hard. Intuitively, the difficulty stems from the fact that, in the von Neumann model, a tree of AND gates has a higher probability of outputting a 1 if it has high probability of recieving a $15/16$ fraction of 1's as input, than if it has high probability of recieving very few 1's and low probability of recieving all 1's as input.

VI. Tree Circuits

There are classes of circuits for which the problems discussed in this paper are much easier, namely circuits whose graph representation is a tree. The hardness results in this paper stem from the fact that, in general, the undirected version of the DAG representing a circuit C may contain cycles. When this is not the case, then the probability that a gate g outputs a 1 or a 0 is dependent only on the outcomes of the immediate predecessors of g in C, and thus the situation is much simpler. Given a circuit C that is a tree and where each gate has bounded fan-in, we describe below how to, in both the von Neumann and 0-default failure models, answer the question of whether C is (ϵ,δ)-reliable in time polynomial in the size of C (it can be seen that polynomial complexity can be achieved also in slightly more general settings, e.g., when the circuit's structure is "close to" a tree).

The algorithm is as follows: Each gate g stores four probabilities:

1) The highest probability that g is correct given that its correct output is 1.
2) The lowest probability that g is correct given that its correct output is 1.
3) The highest probability that g is correct given that its correct output is 0.
4) The lowest probability that g is correct given that its correct output is 0.

Let φ be the fan-in of g, and let $g_1,\ldots,g_\varphi$ be the parents of g. By choosing one of the stored probabilities from each of g's parents, we can in $O(2^\varphi)$ steps calculate the probability that g outputs a 1 in that case, and the correct output of g in that case can be computed from the correct outputs for the probabilities

chosen from g's parents. Since there are 4^{φ} ways to choose one stored probability from each of g's parents, we calculate all of these probabilities. Of those where the correct output of g is a 1, we find and store the highest and lowest probabilities that g does output a 1, and do the same for those where the correct output of g is a 0. At the output gate, we find the minimum of the lowest probability that g is correct given that its correct output is 1, and the lowest probability that g is correct given that its correct output is 0. This value determines the minimum value for δ given that functional failures occur with probability ϵ. It is straightforward to see how this algorithm could be modified slightly to find the input to the circuit that minimizes the probability of correctness when functional failures occur with probability ϵ.

To see why this algorithm is correct, consider the situation where all but the ith parent, g_i, of some gate g output a 1 with fixed probability. In this case, the probability that g outputs a 1 is linear in the probability that g_i outputs a 1, and thus this probability is monotonically increasing, monotonically decreasing, or constant, as the probability that g_i outputs a 1 increases. Further, since the circuit is a tree, changing the input to the subtree rooted at g_i does not affect the probability that any of the other parents of g output a 1. Thus we can compute the highest and lowest probabilities that g will output a 1 by some combination of the highest and lowest probabilities that its parents will output a 1. Since we do not know what the correct output for g should be on the input that causes the circuit to be incorrect with highest probability, we store these probabilities in the cases when the correct output of g is either 1 or 0.

VII. Non-Monotonicity of δ in ϵ

For any circuit C, let $\delta^*(\epsilon)$ be the smallest value such that C is $(\epsilon, \delta^*(\epsilon))$-reliable. A question that one might ask is whether $\delta^*(\epsilon)$ is, in general, a non-decreasing function of ϵ. In both the von Neumann and 0-default failure models, this is not the case. The circuit depicted in Figure 5 provides an example for the von Neumann failure model. For this circuit, it is easy to see that for $\epsilon \leq 1/2$, $\delta^*(\epsilon) = 3\epsilon(1-\epsilon)^2 + \epsilon^2(1-\epsilon)$, which is strictly decreasing on $(a, 1)$, where $a = (5-\sqrt{7})/6 \approx 0.39$. Intuitively, this happens because in such a circuit, when ϵ increases, it is more likely that the errors occurring at the two gates cancel out each other.

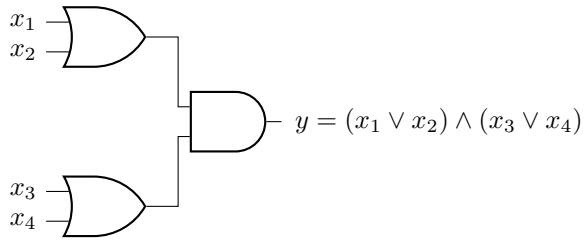

Fig. 5: A simple circuit where $\delta^*(\epsilon)$ is not monotone in ϵ in the von Neumann failure model, consisting of two OR gates and one AND gate.

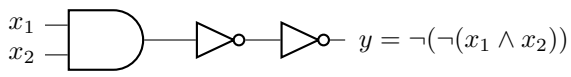

Fig. 6: A simple circuit where $\delta^*(\epsilon)$ is not monotone in ϵ in the 0-default failure model, consisting of an AND gate and two NOT gates.

Figure 6 depicts an example where $\delta^*(\epsilon)$ is not monotone in the 0-default failure model. For this circuit, we have that for $\epsilon \leq 0.45$, $\delta^*(\epsilon) = 1-\epsilon-(1-\epsilon)^4$, which is strictly decreasing on $(b, 1)$, where $b = 1 - 4^{-1/3} \approx 0.37$.

References

[1] A. Antoniadis, N. Barcelo, M. Nugent, K. Pruhs, and M. Scquizzato. Energy-efficient circuit design. In *Proceedings of the 5th conference on Innovations in Theoretical Computer Science (ITCS)*, pages 303–312, 2014.

[2] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *J. ACM*, 45(3):501–555, 1998.

[3] N. Barcelo, M. Nugent, K. Pruhs, and M. Scquizzato. Almost all functions require exponential-energy circuits. 2014. Manuscript.

[4] J. Butts and G. Sohi. A static power model for architects. In *Proceedings of the 33rd annual ACM/IEEE International Symposium on Microarchitecture (MICRO)*, pages 191–201, 2000.

[5] B. Calhoun and A. Chandrakasan. A 256-kb 65-nm sub-threshold SRAM design for ultra-low-voltage operation. *IEEE Journal of Solid-State Circuits*, 42(3):680–688, 2007.

[6] R. L. Dobrushin and S. I. Ortyukov. Lower bound for the redundancy of self-correcting arrangements of unreliable functional elements. *Problems of Information Transmission*, 13:59–65, 1977.

[7] R. L. Dobrushin and S. I. Ortyukov. Upper bound for the redundancy of self-correcting arrangements of unreliable functional elements. *Problems of Information Transmission*, 13:203–218, 1977.

[8] R. G. Dreslinski, M. Wieckowski, D. Blaauw, D. Sylvester, and T. N. Mudge. Near-threshold computing: Reclaiming Moore's law through energy efficient integrated circuits. *Proceedings of the IEEE*, 98(2):253–266, 2010.

[9] P. Gács. *Algorithms in Informatics*, volume 2, chapter Reliable Computation. ELTE Eötvös Kiadó, Budapest, 2005.

[10] P. Gács and A. Gál. Lower bounds for the complexity of reliable boolean circuits with noisy gates. *IEEE Transactions on Information Theory*, 40(2):579–583, 1994.

[11] J. Håstad. Some optimal inapproximability results. *Journal of the ACM*, 48(4):798–859, 2001.

[12] D. J. Kleitman, F. T. Leighton, and Y. Ma. On the design of reliable boolean circuits that contain partially unreliable gates. *Journal of Computer and System Sciences*, 55(3):385–401, 1997.

[13] R. Montenegro and P. Tetali. Mathematical aspects of mixing times in markov chains. *Foundations and Trends in Theoretical Computer Science*, 1(3), 2005.

[14] N. Pippenger. On networks of noisy gates. In *Proceedings of the 26th Symposium on Foundations of Computer Science (FOCS)*, pages 30–38, 1985.

[15] N. Pippenger, G. D. Stamoulis, and J. N. Tsitsiklis. On a lower bound for the redundancy of reliable networks with noisy gates. *IEEE Transactions on Information Theory*, 37(3):639–643, 1991.

[16] R. Reischuk and B. Schmeltz. Reliable computation with noisy circuits and decision trees–A general $n \log n$ lower bound. In *Proceedings of the 32nd Symposium on Foundations of Computer Science (FOCS)*, pages 602–611, 1991.

[17] J. von Neumann. Probabilistic logics and the synthesis of reliable organisms from unreliable components. In C. E. Shannon and J. McCarthy, editors, *Automata Studies*, pages 329–378. Princeton University Press, 1956.