

Soft Error-Aware Power Optimization using Gate Sizing

Foad Dabiri, Ani Nahapetian, Miodrag Potkonjak, Majid Sarrafzadeh
Computer Science Department, University of California Los Angeles
Los Angeles, CA 90095 USA
{dabiri,ani,miodrag,majid}@cs.ucla.edu

Abstract

Power consumption has emerged as the premier and most constraining aspect in modern microprocessor and application specific designs. Gate sizing has been shown to be one of the most effective methods for power (and area) reduction in CMOS digital circuits. Recently, as the feature size of logic gates (and transistors) is becoming smaller and smaller, the effect of soft error rates caused by single event upsets (SEU) is becoming exponentially greater. As a consequence of technology feature size reduction, the SEU rate for typical microprocessor logic at the sea level will go from one in hundred years to one every minute. Unfortunately, the gate sizing requirements of power reduction and resiliency against SEU can be contradictory.

1) We consider the effects of gate sizing on SEU and incorporate the relationship between power reduction and SEU resiliency to develop a new method for power optimization under SEU constraints. 2) Although a non-linear programming approach is a more obvious solution, we propose a convex programming formulation that can be solved efficiently. 3) Many of the optimal existing techniques for gate sizing deal with an exponential number of paths in the circuit, we prove that it is sufficient to consider a linear number of constraints. As an important preprocessing step we apply statistical modeling and validation techniques to quantify the impact of fault masking on the SEU rate. We evaluate the effectiveness of our methodology on ISCAS benchmarks and show that error rates can be reduced by a factor of 100% to 200% while, on average, the power saving is simultaneously decreased by less than 7% to 12% respectively, compared to the optimal power saving with no error rate constraints.

1 Introduction

Single event upsets (SEUs) from transient faults have emerged as a key challenge in logic circuitry design [14]. Recent studies indicate that from 1992 to 2011 SUEs rate for logic will increase by more than a billion times and will surpass the soft error rates of unprotected memory. As a consequence of technology feature size reduction, the SEU rate for typical microprocessor logic at the sea level will go from one in hundred years to one every minute [14] resulting in a clear need for addressing the problem in systematic way. SEU faults arise from energetic particles such as neutrons from cosmic rays and alpha particles from packaging material, generating electron-hole pairs as they pass through a semiconductor device [18]. During ICs' normal operation, these

faults can be caused by electromagnetic interference (EMI). Transistor source and diffusion nodes can collect these charges, and a sufficient amount of accumulated charge may invert the state of a logic device, such as an SRAM cell, a latch or a gate, thereby introducing a logical fault into the circuit's operation. Because this type of fault does not reflect a permanent failure of the device, it is termed soft error (SE) or transient fault (TF) .

Advances in microelectronic technology, which shrink IC size to the nanometer range while also reducing the power supply, are making electronic circuits increasingly susceptible to transient faults (TFs). In fact, the reduction of the charge stored on circuit nodes, along with the decrease in noise margins, greatly increases the probability of voltage glitches temporarily altering nodes' voltage values [4]. Meanwhile, the continuous increase in ICs' operating frequencies makes the sampling of such glitches increasingly probable. Consequently, TFs will become a frequent cause of failure in many applications, while technology advances.

Power consumption has been recognized as the critical constraint in modern microprocessor and application specific designs and gate sizing has been one of the most effective methods for power minimization in CMOS digital circuits. Unfortunately, gate sizing requirements for power reduction and resiliency against SEU are contradictory. We consider the effects of gate sizing on SEU and incorporate the relationship between power reduction and SEU resiliency, and we have developed a new method of power optimization under SEU constraints that leverages convex programming to obtain provably optimal solutions. As an important preprocessing step and consideration we apply statistical modeling and validation techniques.

Gate sizing is a timing optimization process in high performance VLSI circuit design. In this design process, the size of each gate in a combinational circuit is properly tuned so that circuit area and/or overall power dissipation are minimized under specified timing constraints. Gate sizing or the similar problem of transistor sizing has been an active research topic in recent years. Many approaches have been proposed before [13] [2] [11] [17]. Previous approaches that have taken power considerations into account during transistor sizing include [1] [10] [16]. The approach in [10] utilizes linear dependency between power and gate sizes, however, since it optimizes one path at a time, the approach may lead to suboptimal solutions.

A linear programming approach for exploring the power-delay-area tradeoff for a CMOS circuit is presented in [1]; we use more accurate nonlinear logical effort delay models in this work. Another linear programming based approach is presented in [16]. Power optimization with convex programming is proposed by Menezes, Baldick, and Pillegi [13]. In their work, timing constraint are constructed for every path in the circuit which can potentially generate a very large (exponential) number of constraints. In order to capture the effects of logic fault masking, we introduce resubstitution-based statistical methodology and techniques [6], [5] for quantifying error propagation through logic circuitry. We also introduce a new formulation for gate sizing problem using convex programming. Our approach is different from previous ones because the number of constraints in our formulation is linear with respect to circuit size as opposed to the exponential size of constraints in previous work. At the same time, we impose a bound on soft error rates and evaluate the performance of gate sizing considering single event upset.

The rest of the manuscript is organized in the following way. First, in Section I we go over the preliminaries and cover models we have used for delay and soft errors. In Section II we introduce the statistical methodology that we have incorporated to calculate logical masking probabilities. Our formulation and problem transformation is presented in Section III and Section IV illustrates the simulation results on ISCAS85 benchmarks.

2 Preliminaries

2.1 Power and Delay Models

Power dissipation of gates in digital CMOS circuits is composed of dynamic and static components. Dynamic power corresponds to the power dissipated in charging and recharging internal capacitors in every gate, given by Equation (1).

$$P_{dynamic} = \sum_{i=1}^N C_i f_{clock} V_{DD}^2 = \sum_{i=1}^N \Phi_i \cdot W_i \quad (1)$$

where C_i is the effective switching capacitance of the gate i which is a linear function of the size of the gate. f_{clock} is the clock frequency, V_{DD} is the power supply voltage. Φ_i simply represents the linear dependency of the power to the size since the capacitance of a gate is a linear function of the size (width of the gate). The above sum is taken over all the gates in the circuit. Gate delay can be represented as a function of internal capacitors of logic gates. We use the logical effort method to model the delay [15]. The delay d_i of gate i can be written as:

$$d_i = p_i + g_i h_i \quad (2)$$

where p_i is the parasitic delay of the gate and is independent of size. g_i is the logical effort of the gate which is intuitively the driving capability of the gate, and h_i is the electrical effort (gain). h_i is the size dependent term in the delay:

$$h_i = \frac{\sum_j C_j}{C_i} \quad (3)$$

The above sum is taken over all the loads that gate i drives. As stated previously, gate capacitance is linearly dependent to the size of the gate and therefore it is a function of size. To represent the dependency of delay to size, we rewrite Equation 2 using a new function κ :

$$d_i = \kappa_i(W_i, W_j, \dots) = p_i + g_i \frac{\sum_j k_j W_j}{W_i} \quad (4)$$

where $\frac{C_j}{C_i} = \frac{k_j W_j}{W_i}$.

Logical effort model is a simplified gate delay model which may not be very accurate for current circuit technologies. We have chosen this model to illustrate the concept of soft error rate impact on power optimization and how we can address this issue. Our method can be generalized for more complicated delay models as well.

2.2 Single Event Upset

A single event upset (SEU) is an event that occurs when a charged particle deposits some of its charge in a micro-electronic device, such as a CPU, memory chip, or power transistor. This happens when cosmic particles collide with atoms in the atmosphere, creating cascades or showers of neutrons and protons. At deep sub-micrometer geometries, this affects semiconductor devices at sea level. In space, the problem is worse in terms of higher energies. Similar energies are possible on a terrestrial flight over the poles or at high altitude. Trace amounts of radioactive elements in chip packages also lead to SEUs. Frequently, SEUs are referred to as bitflips.

A method for estimating soft error rate (SER) in CMOS SRAM circuits was recently developed by [9]. This model estimates SER due to atmospheric neutrons (neutrons with energies $> 1MeV$) for a range of submicron feature sizes. It is based on a verified empirical model for the 600nm technology, which is then scaled to other technology

generations. The basic form of this model is:

$$SER = F \times A \times e^{-\frac{Q_{crit}}{Q_S}} \quad (5)$$

Where F is the neutron flux with energy $\geq 1MeV$, in particles/(cm^2s), A is the area of the circuit sensitive to particle strikes (the sensitive area is the area of the source of the transistors which is a function of gate size), in cm^2 , Q_{crit} is the critical charge, in fC, and Q_S is the charge collection efficiency of the device, in fC. ¹

[4] presents a very accurate model for Q_{crit} and its dependency on gates sizes. In the following model, Q_{crit} for the gate i is dependent on gate sizes as below:

$$Q_{crit_i} = Q_{crit_{min}} + a'_i(W_i - W_{i_{min}}) + \sum_j b'_j(W_j - W_{j_{min}}) \quad (6)$$

where $Q_{crit_{min}}$ is the critical charge for minimum driver conductance, minimum diffusion capacitances, and minimum fan-out gate input capacitance. Coefficients a' and b'_j s constant parameters that weigh the contribution to Q_{crit} . The sum is taken over all gates driven by gate i . As seen in Equation 5, gate sizing has an effect on Q_{crit} , therefore we use a function, Θ , to represent this dependency:

$$Q_{crit_i} = \Theta_i(W_i, W_j, \dots) \quad (7)$$

Furthermore, the sensitive area to SEU, A , is linearly dependant of size:

$$A_i = \alpha_i W_i \quad (8)$$

Substituting Q_{crit} and A_i in Equation 5 gives us a nonlinear relationship between error rate and gate sizes for a given logic gate. It is important to notice that even if a soft error is generated in a logic gate, it does not necessarily propagate to the output. Soft error can be masked due to following factors:

- Logical masking occurs when the output is not affected by the error in a logic gate due to subsequent gates whose outputs only depend on other inputs.
- Temporal masking (Latching-window masking) occurs in sequential circuits when the pulse generated from particle hit reaches a latch but not at the clock transition, therefore the wrong value is not latched.
- Electrical masking occurs when the pulse resulting from SEU attenuates as it travels through logic gates and wires. Also pulses outside the cutoff frequency of CMOS elements will be faded out [7][14].

Therefore we assign a probability ρ to each logic gate indicating how likely a pulse resulted from SEU can survive to the end and cause an error in the output. The final error rate (λ) assigned to each gate i would be $\lambda_i = SER_i \cdot \rho_i$. In section 3 we introduce a methodology for statistically computing these probabilities.

2.3 System Lifetime and MTTF

In this paper we are considering soft error rates as a measure for system failure. If the error rate in a system is λ , the mean time to failure is $MTTF = \frac{1}{\lambda}$. Therefore, if an MTTF greater than a given value, such as Υ , is desired, it implies that $\lambda \leq \frac{1}{\Upsilon}$.

In digital circuits, since all gates are potentially prone to soft errors, the total error rate of the circuit (Λ) is $\Lambda = \sum_{\forall gate} \lambda_i$. Using equation 5 we can derive the following equation for the total error rate of a digital circuit:

$$\Lambda = \sum_i \rho_i SER_i = \sum_i \rho_i \cdot F \cdot A_i e^{-\frac{Q_{crit_i}}{Q_{S_i}}} \quad (9)$$

¹the term "gate" is used to represent both "logic gates" and "gate terminal" of a CMOS transistor which can be misleading

3 Statistical analysis of gate masking

3.1 General approach

Extensive statistical analysis was done on the circuits from the ISCAS 85 and 89 benchmarks to determine the impact that gate masking can have on the circuit level soft error rate. The first approach was to observe statistically what the impact of an error in a specific gate would have on the observed error in the circuit. In other words, we compared the global output that we observed with and without soft errors in gates. From this analysis we were able to determine the probability that an error in a specific gate could result in an error in the circuit.

The analysis was conducted by simulating the output values of the circuits for randomly selected input values. First, we simulated the circuit a statistically large number of times, using random independently generated input values for all the inputs. Then we compared the output results of the proper functioning circuit, with that of the circuit where a single event upset had occurred, or in other words where a bit had been flipped. As would be expected, because of gate masking, the effect of the flipped bit was not always realized at any of the global outputs. We carried out this simulation for every gate in the circuit, for all the benchmark circuits.

3.2 Reliability of results

In our experimentation we specifically considered 2000 independently random input values. Of course, this is a small fraction of the actual number of possible input values, but experimentation over various runs with different input instances revealed a large correlation among runs. We verified our results by running various instances of the experiments to verify that indeed the results we were obtaining were consistent. One such instance, benchmark c432, is shown in the graph in Figure 1. The graph shows the fifteen different runs on the same benchmark, for a statistically significant number of different randomly selected input values. The results obtained are compared with the total sum of all of these fifteen runs. The graph shows that fifteen runs each deviate by less than 3% from the values obtained with the fifteen times larger test case. The results help to demonstrate the reliability of the statistical analysis conducted. And they give statistical evidence of the correlation between the results obtained and the actual characteristics of the circuit.

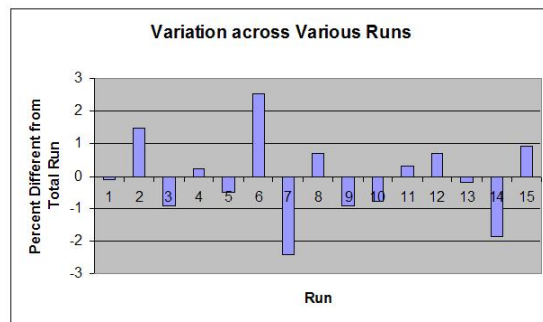


Figure 1: Shown here for a single benchmark, c432, across 15 different runs, we see less than a 3% variation from the values obtained using the total iterations across all 15 of the runs. This provides evidence that our results are statistically close to the actual circuit characteristics.

4 Problem formulation

In logic synthesis, circuits are usually modeled as a directed acyclic graph $G = (V, E)$ (see Figure 2 a. In this model, nodes represent the logic gates and edges stand for the precedence relation between them. We transform the given graph G into G' in such a way that, each node v in G is split into two nodes v_1 and v_2 and an edge connecting v_1 to v_2 . In the transformed graph, the new edges are basically the logic gates from the original graphs. Figure 2 a shows an example of such transformation. In order to have a single input and a single output, nodes s and t have been added to the graph; s is connected to all primary inputs and all primary outputs are connected to t .

The delay of a path $p = \langle s, v_1, v_2, \dots, t \rangle$ from node s to node t is equal to the summation of the delays of each edge along the path. We use the terms 'delay of a path' and 'the distance between nodes s and t ', interchangeably. Where the sum is taken over all the edges in path p .

The problem is defined as: Given a DAG $G = (V, E)$ and a timing constraint T and an error rate constraint Υ .

$$\text{minimize } \sum_{\forall e_{ij} \in E} P_{ij} \tag{10}$$

such that the delay of every path from s to t is less than or equal to T and the error rate (caused by SEU) is less than Υ . P_{ij} is the power consumption of ij^{th} edge in the DAG which is a function of the capacitance of the gate². The timing constraint can be stated as: $\sum_{e_{ij} \in p_k} d_{ij} \leq T$ for every path p_k from s to t . Note that the number of paths in a DAG is exponential in terms of the number of edges in the graph, therefore this formulation is not efficient. Throughout the rest of this section, we will convert it to a formulation with the same objective function that has a linear number of constraints.

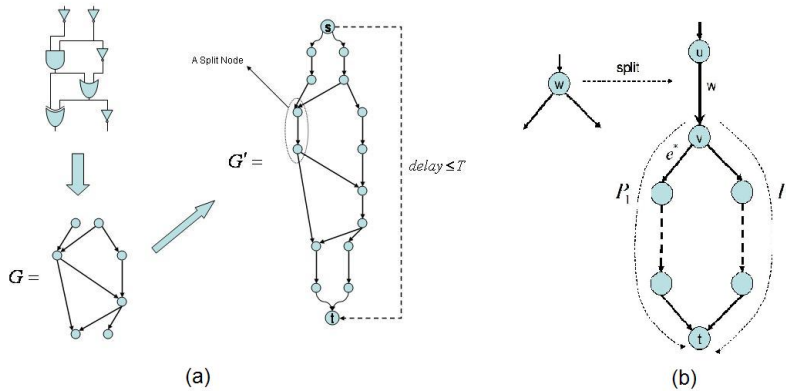


Figure 2: (a) DAG representation of the circuit up-left and its transformation. Each node (gate) in the original DAG is replaced by an edge in G' . (b) Figure for Theorem 1

Theorem 1: There is an optimal gate sizing solution on a DAG such that the distance between any node u and the output t is independent of the choice of the path taken between them and this distance is unique.

Proof: Suppose the claim is not true, i.e. there exists a node v where its distance to t through path P_1 is less than P_2 (see Figure 2 b). Before getting to the proof, it is important to note that the edge e_{uv} is a split node. In other words it represents the

²Indices for gate parameters such as power (P) and delay (d), is changed starting from this section of the paper because every gate is represented by an edge in the transformed graph (see Figure 2). For example instead of P_i we use P_{ij} for the power consumption of a gate

gate w in the original graph and therefore the outgoing edges from v are representing actual edges from the original graph, not any gates.

Without loss of generality we can assume P_1 is shorter than P_2 . We claim that there exists an edge (e^*) in P_1 that can be slowed down and still not violate the timing constraint because P_2 is on the critical path from v to t . One immediate candidate for e^* is the first edge in P_1 . Increasing the delay of e^* by $d_{P_2} - d_{P_1}$ will not cause a timing violation and since it does not contribute in the cost function nor error rate constraint, the total power dissipation and error rate remain constant. This increase is made by assigning a "dummy" delay to e^* . Therefor we can maintain the same objectives by equalizing the delay of P_1 and P_2 and since this optimization problem has only one global minimum, there should exist an optimal solution in which the statement in Theorem 1 holds.

Theorem 1 still holds for more complex delay-power model. As long as the power dissipation of a gate is a nondecreasing function of gate size (which indeed is) the theorem holds. The following observation is immediately inferred from the above theorem:

Now that the delay between every node to the destination in the optimal solution is independent of the path taken, let t_i be a variable assigned to each node v_i that represents its distance to t . A similar technique was proposed in [8] which has resulted in an efficient integer delay budgeting algorithm. We call t_i the distance variable of node v_i . In other words, t_i is the delay of the system from node v_i to the output. Therefore, the delay and power consumption of each edge (node in the original graph) is represented by:

$$\begin{aligned} d_{ij} &= t_i - t_j = p_{ij} + g_{ij}h_{ij} \\ P_{ij} &= \phi_{ij}W_{ij} \\ &\forall e_{ij} \in E \end{aligned}$$

Thus, instead of having a constraint for each path from s to t , we construct the following constraints:

$$t_i - t_j \geq p_i + g_i h_i, \forall e_{ij} \in E(G') - E(G) \quad (11)$$

$$t_i - t_j \geq 0, \forall e_{ij} \in E(G') \cap E(G) \quad (12)$$

$$t_s - t_t \leq T \quad (13)$$

$$W_{ij} \geq W_{min} \quad (14)$$

Equation 11 enforces that the delay assigned to each gate is greater or equal to its minimum delay (parasitic delay) while Equation 12 assigns a non-negative delay to those edges in the original graph that represent connection between gates. Equation 13 guarantees that the distance from s to t is less than or equal to the timing constraint T . Equation 13 can be interpreted as a minimum delay required for the virtual edge between s and t . This edge is also shown in Figure 2 a. The constraint in Equation 14 is simply the lower bound on transistor width. Error rate can be bounded by the following constraint:

$$\sum_i \rho_i \cdot F \cdot \alpha_i W_i e^{-\frac{\Theta_i(W_i, W_j, \dots)}{Q S_i}} \leq \Upsilon \quad (15)$$

in which Υ is the desired upper bound on soft error rate. All the timing constraints on paths are reformulated as edge constraints and the optimization problem can be restated as:

$$\text{minimize } f(\vec{W}) = \sum_{\forall e_{ij} \in E} \phi_{ij} W_{ij} \quad (16)$$

subject to constraints in Equations 11 - 15. The objective in 16 along with constraints stated in Equations 11 through 15 form a nonlinear optimization problem with linear number of constraints in terms of the size of the graph. In the next section we modify and solve this problem using convex programming method.

4.1 Convexity of the Optimization Problem

Convex optimization problems are far more general than linear programming problems, but they share the desirable properties of LP problems: They can be solved quickly and reliably even in very large cases. A convex optimization problem is a problem where all of the constraints are convex functions and the objective is a convex function to be minimized, or a concave function to be maximized. With a convex objective and a convex feasible region, there can be only one optimal solution, which is globally optimal. Several proposed methods – notably the Interior Point method – can either find the globally optimal solution, or prove that there is no feasible solution to the problem.

The objective defined in Equation 16 is a linear function of the variable W_i and is therefore convex. To state the convexity of proposed formulation, it is required to show that the feasible solution space created by the constraints is in fact convex. The constraints in 11 can be rewritten as:

$$\frac{p_i}{t_i} + \frac{g_i h_i}{t_i} + \frac{t_j}{t_i} \leq 1 \quad (17)$$

Since all the variables in the above equation are positive, the constraint in 17 is a posynomial expression. Posynomials are not convex in this format but with a change of variables, they can be mapped to a convex space. If each variable x in a posynomial expression is substituted with e^z , the resulting expression becomes an exponential convex function [3]. The constraint presented in Equation 15 is a nonlinear function which generally is very hard to handle. This function has both linear and exponential dependency on variable which results in a non-monotone function. Each variable W_i , is linearly contributing in only one term of 15 (through sensitive are corresponding to A_i) but is exponentially included few terms of 15 (in Q_{crit} for gate i and all fan-in gates). Therefore, not only are there more terms that include W_i in the exponent but their effect compared to the linear dependency of A to W_i is much more significant. On the other hand, shrinking a gate size both reduces the power consumption and the sensitive area to SEU. Therefore the exponential contribution of gate sizes and power dissipation are the two contradictory factors, not the sensitive area. This observation led us to modify Equation 15 such that we assume an average value for each A_i and change the constraint to exponential form which is convex. This modification and exponential transformation in the posynomial constraints keeps the objective and other constraints convex and the solution space, which is the intersection of all subspaces created by convex constraints, is itself convex. In the simulation section, after calculation gate sizes, we use the actual resulting gate sizes to recompute total error rate, and report the correct numbers.

5 Simulation Results

We used the MOSEK convex optimization tool [12] to solve the proposed formulation on the ISCAS benchmarks. Although ISCAS benchmarks do not include very large circuits, but have been proven to be a proper set of test cases to show the validity of a proposed methodology and concept. For each benchmark, we calculate the total delay of the circuit, T , with initial size values and then use that delay as the timing constraint for the optimization problem. Each benchmark has a total error rate with the initial size, Λ , which we used a constraint on the error rate. For each circuit, we minimize the power consumption with four different bounds on the error rate, Υ , starting with the no bound on error rate, error rate of the initial circuit and up to reduced rate by a factor of 200%. Figures 3 and 4 illustrates the power consumption reduction versus the bound on error rate for combinational circuits. A point (x, y) in these graphs means that the power dissipation has been reduced by $y\%$ while the total error rate has been decreased by at least a factor of $x\%$ compared to the error rate in the initial circuit.

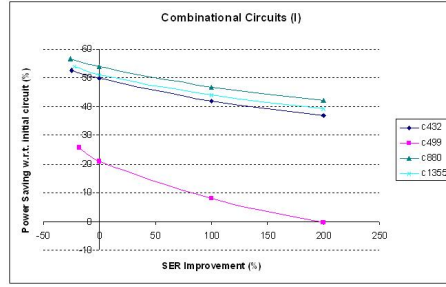


Figure 3: Soft error rate can be reduced by huge factor without compromising the power saving in these combinational circuits

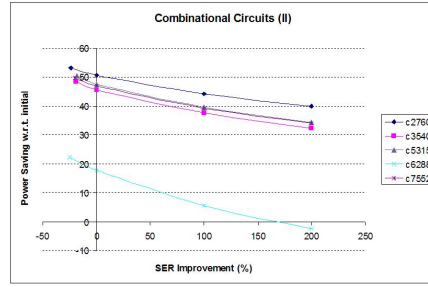


Figure 4: Simulation results for 4 larger combinational circuits: power saving in c6288 benchmark is more defandant to bounds on error rates

It can be observed that an average of 63% power saving can be achieved without any constraint on the error rate for these bench marks. Obviously, power saving percentage depends on the initial design which we compare our results with. If the original design is far from optimal, then the power saving ratio becomes larger. The significance of the results are that we can achieve *optimal* power reduction in the presence of *soft error rate constraints*. This is the reason why we have not compared our results with any other power optimization method. We ran the same optimization on sequential circuits as well. Figures 5 and 6 summarize these results. Average power reduction is about 61% for these benchmarks. Comparing the results of combinational circuits to sequential ones, we observe that the error rate bounds on sequential circuits as less restrictive in the power optimization process. The one immediate reason could be the fact that temporal masking reduces the affect of soft errors in sequential circuits whereas no such masking is present in combinational circuits.

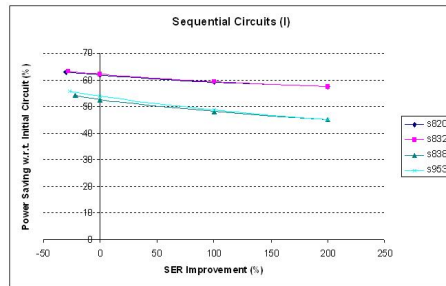


Figure 5: It can be observed that error rate in these sequential circuits can be improved dramatically without compromising power savings.

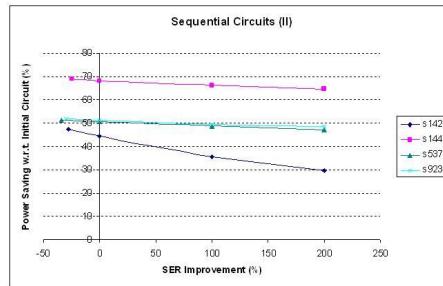


Figure 6: Simulation results for 4 larger sequential circuits including s1423 which is more defandant to bounds on error rates compared to other benchmark.

6 Conclusion

In this paper, we have introduced a new formulation for gate sizing that targets power optimization, resiliency against SEUs, and timing constraints simultaneously. As a preprocessing step for the optimization, we have developed a statistical modeling and validation technique that quantifies the impact of fault masking in combinational logic. We formulated the problem as a convex optimization problem of linear size, as opposed to the previous convex programming approaches which could potentially be exponen-

tial in size. The MOSEK convex optimization tool was used to evaluate the proposed approach on ISCAS benchmarks. We were able to minimize the power dissipation for a given timing constraint and various upper bounds on error rates caused by a single event upset. An important practical result was that convex programming-based gate sizing can simultaneously reduce power consumption and improve SEU resiliency. Our simulation showed that different circuits from the various benchmark behave differently when carrying out power optimization while considering soft errors. As future work, we propose examining the question of designing circuits such that through gate sizing the most power savings can be achieved while enforcing low error rates.

References

- [1] Michel R. C. M. Berkelaar and Jochen A. G. Jess. Gate sizing in mos digital circuits with linear programming. In *EURO-DAC '90: European design automation conference*, pages 217–221, Los Alamitos, CA, USA, 1990. IEEE Computer Society Press.
- [2] Manjit Borah, Robert Michael Owens, and Mary Jane Irwin. Transistor sizing for minimizing power consumption of cmos circuits under delay constraint. In *ISLPED '95, International Symposium on Low Power Design*, pages 167–172, New York, NY, USA, 1995. ACM Press.
- [3] Stephen Boyd and Lieven Vandenbergh. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.
- [4] J. M. Cazeaux, D. Rossi, M. Omana, C. Metra, and A. Chatterjee. On transistor level gate sizing for increased robustness to transient faults. In *IOLTS '05, International On-Line Testing Symposium*, pages 23–28, Washington, DC, USA, 2005. IEEE Computer Society.
- [5] B. Efron. *The Jackknife, the Bootstrap, and Other Resampling Plans*. S.I.A.M, Philadelphia, 1982.
- [6] Tibshirani R.J Efron, B. *An Introduction to the Bootstrap*. Chapman & Hall/CRC, New York, NY, USA, 1994.
- [7] Subhasish Mitra et.al. Logic soft errors in sub-65nm technologies design and cad challenges. In *DAC '05, Design Automation Conference*, pages 2–4, New York, NY, USA, 2005. ACM Press.
- [8] S. Ghiassi, E. Bozorgzadeh, S. Choudhuri, and M. Sarrafzadeh. A unified theory of timing budget management. In *ICCAD '04, International conference on Computer-aided design*, pages 653–659, Washington, DC, USA, 2004. IEEE Computer Society.
- [9] C.; Wender S.A Hazucha, P.; Svensson. Cosmic-ray soft error rate characterization of a standard 0.6- μ m cmos process. In *IEEE Journal of Solid-State Circuits*, pages 1422–1429, Washington, DC, USA, 2000.
- [10] K. S. Hedlund. Aesop: a tool for automated transistor sizing. In *DAC '87: Proceedings of the 24th ACM/IEEE conference on Design automation*, pages 114–120, New York, NY, USA, 1987. ACM Press.
- [11] Noel Menezes, Ross Baldick, and Lawrence T. Pileggi. A sequential quadratic programming approach to concurrent gate and wire sizing. In *ICCAD '95: Proceedings of the 1995 IEEE/ACM international conference on Computer-aided design*, pages 144–151, Washington, DC, USA, 1995. IEEE Computer Society.
- [12] MOSEK ApS, Denmark. *The MOSEK optimization tools manual (<http://www.mosek.com>)*, 2002.
- [13] Sachin S. Sapatnekar and Weitong Chuang. Power-delay optimizations in gate sizing. *ACM Trans. Des. Autom. Electron. Syst.*, 5(1):98–114, 2000.
- [14] Premkishore Shivakumar, Michael Kistler, Stephen W. Keckler, Doug Burger, and Lorenzo Alvisi. Modeling the effect of technology trends on the soft error rate of combinational logic. In *DSN '02, Dependable Systems and Networks*, pages 389–398, Washington, DC, USA, 2002. IEEE Computer Society.
- [15] Ivan E. Sutherland and Robert F. Sproull. Logical effort: designing for speed on the back of an envelope. In *Proceedings of the 1991 University of California/Santa Cruz conference on Advanced research in VLSI*, pages 1–16, Cambridge, MA, USA, 1991. MIT Press.
- [16] Yutaka Tamiya, Yusuke Matsunaga, and Masahiro Fujita. Lp based cell selection with constraints of timing, area, and power consumption. In *ICCAD '94, International Conference on Computer-Aided Design*, pages 378–381, Los Alamitos, CA, USA, 1994. IEEE Computer Society Press.
- [17] Hiran Tennakoon and Carl Sechen. Efficient and accurate gate sizing with piecewise convex delay models. In *DAC '05, Design Automation Conference*, pages 807–812, New York, NY, USA, 2005. ACM Press.
- [18] Christopher Weaver, Joel Emer, Shubhendu S. Mukherjee, and Steven K. Reinhardt. Techniques to reduce the soft error rate of a high-performance microprocessor. In *ISCA '04: Proceedings of the 31st annual international symposium on Computer architecture*, page 264, Washington, DC, USA, 2004. IEEE Computer Society.