Fast-Yet-Accurate Statistical Soft-Error-Rate Analysis Considering Full-Spectrum Charge Collection

Hsuan-Ming Huang and Charles H.-P. Wen
National Chiao Tung University

Editor's notes:
Soft errors are a growing concern in highly scaled CMOS technologies; estimating error rates for a given design remains very challenging. This article presents a fast statistical soft-error-rate analysis approach that is nearly as accurate as computationally complex Monte Carlo SPICE simulation.

—Adit Singh, Auburn University

A SOFT ERROR is a transient fault, induced by one particle striking the sensitive region of a device with deposited charges, and latched by a memory element after propagation. These errors can change the state of the circuit and result in various system failures. Traditionally, soft errors are a major concern in memory circuits, such as SRAM and DRAM. However, technology scaling has recently reduced device size and logic depth. Together with increasing operating frequencies, this has led to exponential growth in soft-error rates (SERs) for combinational circuits. Because the soft-error issue of combinational circuits can no longer be ignored, it is critical for circuit reliability in very deep submicron (VDSM) technologies.

Several works characterize and analyze SERs for combinational circuits. Rao et al. [1] present a linear-time algorithm for SER analysis of combinational circuits using parameterized descriptors. MARS-C [2] and FASER [3] propose efficient estimation frameworks with high level accuracy for soft-error analysis using two symbolic techniques, binary decision diagrams (BDDs), and algebraic decision diagrams (ADDs), respectively. SERA [4] computes SER by combining the graph theory, fault simulation, probability theory, and circuit simulation. AnSER [5] investigates a signature-based SER framework, while considering the timing masking effect to enhance circuit reliability. SEAT-LA [6] models the propagation of a pulse and estimates SER by particular characterized cell libraries and analytical equations. However, all previous approaches do not address fluctuation in gate performance that is induced by process variation in very deep submicron technologies. Thus, the SER accuracy is typically unsatisfactory.

With the continuous technology scaling, especially in nanometer technologies, process variation significantly influences the performance of fabricated chips and is a key issue for advanced CMOS designs. More specifically, process variation makes SER estimation more challenging as the electrical characteristic (e.g., delay) of a gate is no longer a fixed value, but a random variable from a

Digital Object Identifier 10.1109/MDT.2012.2194471
Date of publication: 11 April 2013; date of current version: 29 May 2013.

probability distribution. As a result, SER analysis must shift from a deterministic framework to a statistical one.

The impact of process variation on soft error in state-holding elements, such as SRAM and DFF, are first studied in [7] and [8]. Variations such as, hot carrier injection (HCI), negative bias temperature instability (NBTI), and interdie channel length variation, on combinational circuits are also analyzed for soft errors in [9] and [10], which have not yet been exercised on large-scale designs. Later, Peng et al. [11] propose an accurate statistical soft-error rate (SSER) framework built on learning-based statistical models for transient-fault distributions. Using statistical tables for cell models in Monte Carlo simulation, the work [12] investigates an alternative SSER approach that is more accurate, but runs slower than the previous method [11]. However, both works [11], [12] simplify the SER estimation by injecting only four levels of electrical charges. Therefore, this study poses a simple, yet important, question, ‘‘*Are four levels of electrical charges enough to converge SER correctly and properly address the process variation effect?*’’

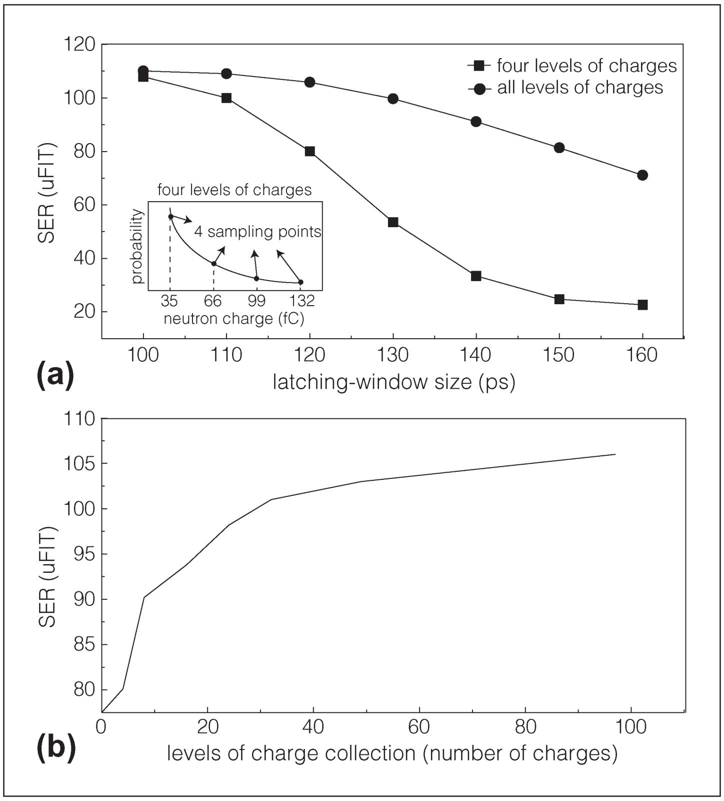

Figure 1. (a) Statistical SER comparison using four-level and full-spectrum charge collection w.r.t. different latching-window sizes by SPICE simulation on c17. (b) Statistical SER computation w.r.t. different levels of charge collection (using different number of charges), indicating that all levels of deposited charges should be considered.

Figure 1a compares of SERs from Monte Carlo SPICE simulations. These SERs had different levels of charges when collected onto a sample circuit (c17 from ISCAS'85) with different latching-window sizes. The line with square symbols and the line with circle symbols represent the SERs induced by four-level and full-spectrum charge collection, respectively. Moreover, the Y-axis denotes SERs in failure-in-time (FIT), which is defined as one failure in 10^9 hours. This study applies process variation for the Monte Carlo SPICE simulation by perturbing the gate width (W) and channel length (L) of each device. As the latching-window size was set to 100 ps, the SERs obtained from four-level and full-spectrum analyses were the same. However, as the latching-window size grew to 150 ps, the effective range of charge collection for SSER analysis increased from 35 to 132 fC. Therefore, the SER difference between four-level and full-spectrum analyses grew to 69%. Another question naturally arises, “*If four levels of charge collection are not sufficient to derive accurate SERs, how many levels are sufficient*?”

Figure 1b suggests the answer. All levels of deposited charges should be considered because SERs increase with charge collections. SER difference using different levels of deposited charges is further illustrated (see Figure 2), where the upper and lower parts show SER estimation by only four levels of charges and by all levels of charges, respectively. The X- and Y-axis denote the pulsewidth of transient faults and the effective frequency for a particle strike of different levels of deposited charges. For the analysis using four-level deposited charges, only four transient-fault (TF) distributions were generated and could contribute to the final soft error rate. In other words, soft errors can only be generated from four concentrated distributions, and therefore may result in mistakes on SER integration. As the latching-window size of one flip-flop was far from the first TF distribution, soft errors from such TF distributions were entirely masked due to the timing-masking effect [5]. For example, the biggest pulsewidth distribution in the upper part of Figure 2 is excluded from SER estimation. But, only part of them (those smaller decomposed TF distributions) were masked during analysis using all levels of deposited charges (see Figure 2, lower part). As a result, SER estimation was no longer valid with analysis using only four levels of charges and instead should comprehensively consider full-spectrum charge collection.

This study presents a fast-yet-accurate framework that integrates the process-variation effect and considers full-spectrum charge collection during SSER analysis of combinational circuits. In addition, a technique of automatic bounding-charge selection is incorporated for accelerating SER computation and determining the least required set of deposited charges to apply statistical analysis. Advanced learning technique [i.e., support vector machine (SVM)] is also used to derive a quality cell model for facilitating SER computation. The rest of the paper is organized as follows: first we describe the background of transient faults and the overview of statistical SER (SSER) estimation. Then, we describe the techniques including intensified data-learning and automatic bounding-charge selection. We follow that with a description of the experiments on ISCAS'85 circuits, a series of multipliers, and an AES chipher from IWLS 2005 benchmark, with the result of a $10^7 X$ runtime speed-up and 0.8% accuracy loss in SERs on average when compared to the Monte Carlo SPICE simulation. Finally, we draw the conclusion.

Background

Radiation-induced transient faults

A neutron particle generates electron-hole pairs upon striking the silicon bulk of a device. These freeing electron hole pairs result in transient faults and may cause system failures. However, three masking mechanisms affect transient fault propagation through an arbitrary path to a flip-flop. These masking mechanisms collectively help prevent the soft errors caused by such transient glitches in the circuit. The following discussion briefly introduces each of three masking mechanisms [1], [11], [12].

1) **Logical masking**: a transient fault disappears because one of the side-inputs for a gate on its propagation has the controlling value (0 for AND-type gates and 1 for OR-type gates) and stops the propagation of such transient fault.

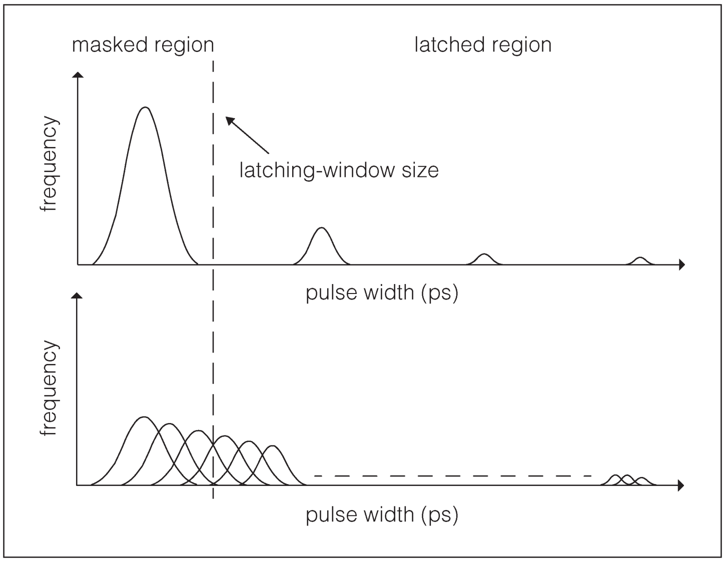

Figure 2. Transient-fault distributions induced by four-level and full-spectrum charge collection.

2) **Electrical masking**: a transient fault is attenuated, and has a weaker pulsewidth in voltage, after propagating through a gate. This is because of the electrical properties (charging/discharging) of the gate. If the attenuation effect is strong enough, such transient fault may disappear after propagation.
3) **Timing masking**: a transient fault will not be latched and not become a soft error because its arrival time falls outside the latching window (setup time + hold time) of one flip-flop.

Additionally, a transient fault induced by a particle strike can be modeled as a current source injected into the drain of a transistor as shown in Figure 3. The transient fault can be induced by two

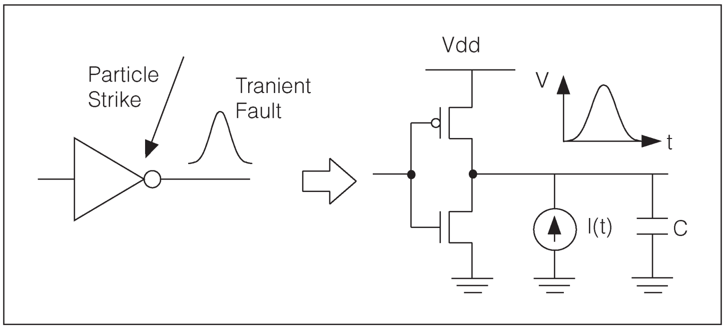

Figure 3. Current source model of a particle strike at a circuit node.

kinds of particles, alpha particles and neutron particles. Generally, the current source model for an alpha particle induced transient fault is typically expressed into a double-exponential term [7], [8], [10] while the current source model for a neutron particle which is our focus in this paper can be formulated into a single exponential pulse [1], [3], [4], [13] as

$$I(t) = \frac{Q}{\tau}\sqrt{\frac{t}{\tau}}e^{-t/\tau} \tag{1}$$

where Q is the total amount of deposited charges. τ is charge collection time constant that depends on the process-related factors and can be calibrated through TCAD simulation [14]. However, not every energy level of such particle can result in a transient fault. Transient faults induced by low energy levels of particles (i.e., < 35 fC in this paper) may disappear due to its resulting output voltage less than VDD/2. Some high energy levels of particles (i.e., > 132 fC in this paper) can be ignored because of extremely low flux of neutrons (10× less than low energy levels) [15]. According to the current source model in (1) and extensive SPICE simulation, the range of injected charge becomes [35 fC, 132 fC].

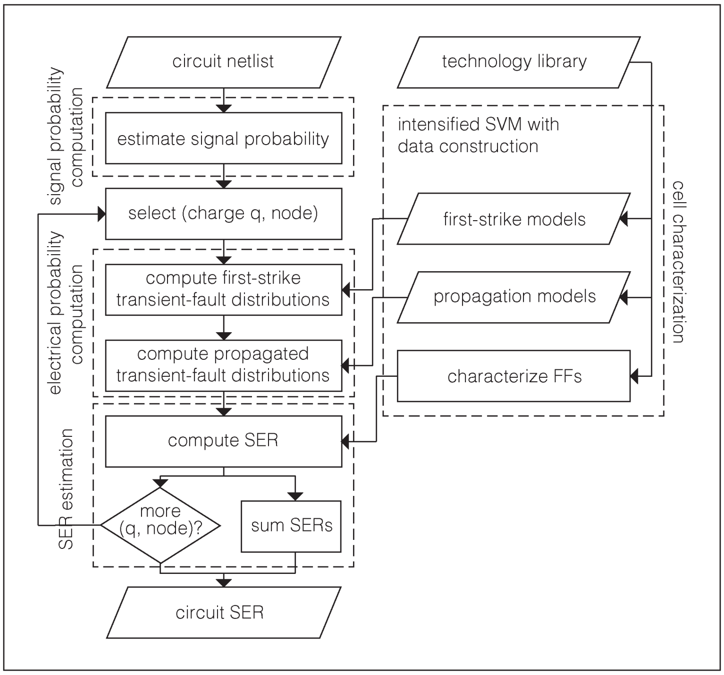
circuit netlist
technology library
signal probability computation
estimate signal probability
intensified SVM with data construction
select (charge q, node)
first-strike models
cell characterization
electrical probability computation
compute first-strike transient-fault distributions
propagation models
compute propagated transient-fault distributions
characterize FFs
SER estimation
compute SER
more (q, node)?
sum SERs
circuit SER

Figure 4. Proposed SSER analysis flow modified from [11].

Overview of statistical SER

This section provides an overview of the statistical SER analysis presented in [11]. Figure 4 shows the modified statistical analysis flow appended and considers full-spectrum charge collection, mainly involving **cell characterization**, **signal probability computation**, **electrical-probability computation**, and **SER estimation**.

1) ***SER Estimation***: SER induced by a neutron-particle strike on a gate i in the circuit under test (CUT) is first denoted as SER_i with the following definition:

$$\text{SER}_i = \int_{q=0}^{Q_{\max}} (R(q) \times P_{\text{soft-err}}(i,q))dq \tag{2}$$

where $P_{\text{soft-err}}(i,q)$ is the probability of soft-error occurrence. This indicates that a transient fault originated from a particle striking at gate i and the deposited charge q results in a soft error at an arbitrary flip-flop. Frequency $R(q)$ is the striking rate of deposited charge q in unit time. That is

$$R(q) = F \times K \times A \times \frac{1}{Q_s} \times e^{\frac{-q}{Q_s}} \tag{3}$$

where F is the neutron flux with energy, K is a technology independent fitting parameter, A is the susceptible area in cm^2, and Q_s is the charge-collection slope. Finally, total SER for the CUT is represented as the summation of all SER_i

$$\text{SER}_{CUT} = \sum_{i=0}^{N_g-1} \text{SER}_i \tag{4}$$

where N_g is the total number of gates in the CUT.

2) ***Signal-Probability Computation***: The term $P_{\text{soft-err}}(i,q)$ in (2) includes the computation for the logic probability $P_{logc}(i,j)$ and the electrical probability $P_{\text{elec}}(i,j,q)$ to reflect the three masking mechanisms. This term can be further defined as

$$P_{\text{soft-err}}(i,q) = \sum_{j=0}^{N_{ff}-1} P_{logc}(i,j) \times P_{\text{elec}}(i,j,q) \tag{5}$$

where N_{ff} is the total number of flip-flops in the CUT. Logic probability $P_{logc}(i,j)$ is the probability of a transient fault not being masked by the logical-masking mechanism through path $(i \rightarrow j)$ from gate i to flip-flop j. This probability is computed by the signal probability (P_{sig}) for the designated logic value on the strike node and multiplied by the accumulated signal probability (P_{side}) for noncontrolling values on all side-inputs along the target path as follows:

$$P_{logc}(i,j) = P_{\text{sig}} \times \prod_{k \in i \rightarrow j} P_{\text{side}}(k) \qquad (6)$$

where k denotes one of the gates along the target path $(i \rightarrow j)$ starting from node i towards flip-flop j.

3) ***Electrical-Probability Computation***: Electrical Probability $P_{\text{elec}}(i,j,q)$ accounts for the electrical- and timing masking mechanisms. Its definition is as follows:

$$\begin{aligned} P_{\text{elec}}(i,j,q) &= P_{\text{err-latch}}(pw_j, w_j) \\ &= P_{\text{err-latch}}\left(\lambda_{\text{elec-mask}}(i,j,q), w_j\right) \qquad (7) \end{aligned}$$

where $P_{\text{err-latch}}(pw_j, w_j)$ is the latching probability with the following definition:

$$P_{\text{err-latch}}(pw, w) = \frac{1}{t_{clk}} \int_{0}^{\mu_x + 3\sigma_x} x \times P(x > 0)dx. \qquad (8)$$

Here, the pulsewidth (pw) of a transient fault, and the latching-window size (w) of the flip-flop, are random variables. $x = pw - w$ is a new random variable with μ_x and σ_x as the mean and variance.

Note that $\lambda_{\text{elec-mask}}(i,j,q)$ in (7) is the electrical masking function used to reflect the electrical-masking mechanism, and can be formulated as

$$\begin{aligned} &\lambda_{\text{elec-mask}}(i,j,q) \\ &\quad = \underbrace{\delta_{\text{prop}}(\ldots(\delta_{\text{prop}}(\delta_{\text{prop}}}_{m\ times}(pw_0, 1), 2), \ldots), m). \qquad (9) \end{aligned}$$

First, $pw_0 = \delta_{\text{strike}}(q,i)$ denotes the initial pulsewidth for a particle strike with charge q deposited at gate i. Then, this transient fault propagates to the next gate, resulting in the respective pulsewidth (pw_1), according to the electrical properties of such gate. Finally, pw_j is derived when this transient fault propagates along the propagation path $(i \rightarrow j)$ from node i through m gates to the flip-flop j. In (9), δ_{strike} and δ_{prop} represent the first-strike function and the propagation function, respectively. They are explained in the following section.

An intensified SSER analysis framework

Based on the approach in [11], to enable a better SSER analysis, we need to find more accurate, but efficient, first-strike function δ_{strike} and propagation function δ_{prop} to encompass the process-variation effect. Hence, this study uses a **learning with data construction** method for statistical model extraction. The proposed algorithm also incorporates an **automatic bounding-charge selection** technique to remove unnecessary charges for facilitating SER estimation.

Intensified learning with data reconstruction

Although the lookup-table (LUT) method [12] is eminent for providing adequate model accuracy, it is not cost-effective to compute SER, especially when considering full-spectrum charges. Therefore, this method uses a state-of-the-art computational learning technique, called support vector machine (SVM) [16], for cell characterization, instead of the LUT method. SVM also provides two additional merits: 1) SVM models can be generalized to predict unseen samples and 2) SVM models are highly compact. For more details on the statistical learning theory and SVM, see [16].

Although SVM provides accurate and compact models to estimate SER in [11], two problems remain unsolved: 1) the training time for data preparation and 2) parameter search for high-quality models. For these two problems, this framework incorporates a metaheurisitc, ***particle swarm optimization*** (**PSO**), to facilitate the search for the optimal setting within short training time.

PSO is one of the evolutionary computation techniques developed by Kennedy and Eberhart in 1995 [17]. PSO adopts a strategy to search for potential solutions based on the behavior of particle swarms which are inspired by swarm intelligence from insects, birds and fish. Initially, PSO generates a set of random particles in a multidimensional search space. The position and velocity are each represented by a particle. The position indicates a possible solution of the optimization problem and

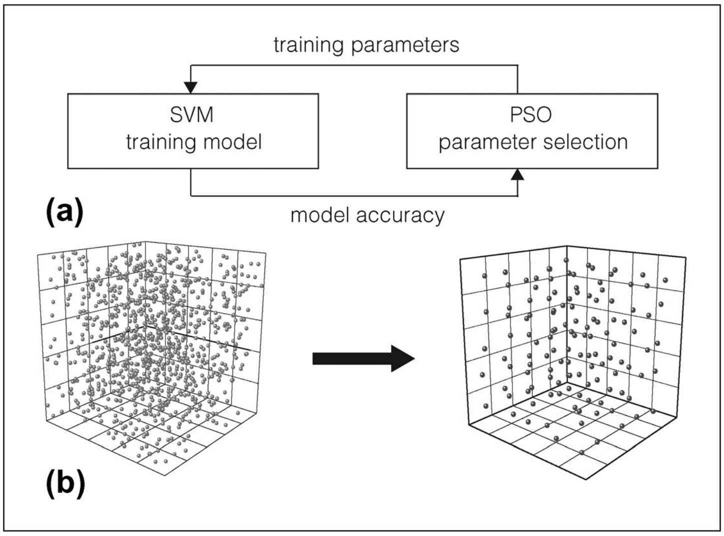

Figure 5. (a) Intensified SVM learning with PSO. (b) Example for data construction.

the velocity is used to determine the search direction. At each iteration, particles change their positions by tracking the best position of all particles (*Gbest*) and their best positions (*Pbest*). The velocity and position of particle *i* is updated according to the following equation:

$$V_i^{k+1} = wV_i^k + \left(Pbest - X_i^k\right) + c_2 r_2 \left(Gbest - X_i^k\right)$$
$$X_i^{k+1} = X_i^k + V_i^{k+1} \tag{10}$$

where *k* is the iteration index, *w* is the inertia weight, c_1 and c_2 are the learning factors, and r_1 and r_2 are both random numbers from [0,1].

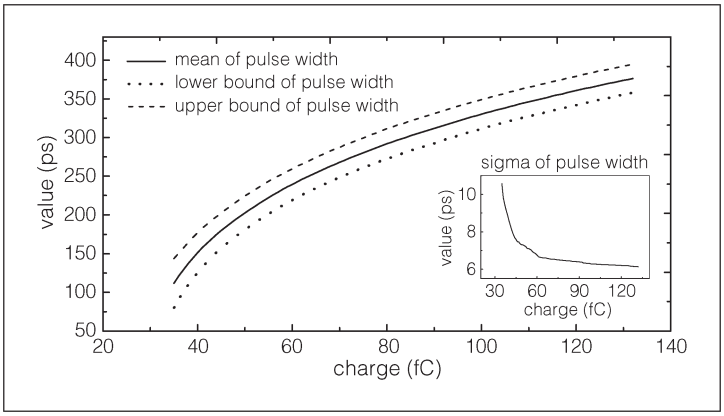

Figure 6. The mean, sigma, lower bound (mean − 3 ∗ sigma), and upper bound (mean + 3 ∗ sigma) of TF distribution which are induced by different electrical charges.

The advantages of PSO are easy implementation, it requires only a few setting parameters to be adjusted, and it is capable of avoiding being trapped in a local optimum solution when compared with other evolutionary algorithms, such as the genetic algorithm (GA).

Figure 5a illustrates the interaction between our intensified SVM-learning and PSO. First, PSO generates a set of training parameters required for SVM to build behavioral models. After building the training models, SVM reports model's accuracy to PSO as its fitness value. Based on the model's accuracy, PSO will breed new generations and generate better parameters for training. This process iterates for a specific number of generations or until achieving a stopping criteria.

Besides PSO, this study uses a data reconstruction technique to reduce the size of the training data and greatly improve the training time and the compression ratio of models. This data reconstruction calculates the average value of training data in each block (see Figure 5b). The red points represent the raw data from the extensive SPICE simulation. The blue points illustrate the average values of each block. After reconstruction, the size of the training data is greatly reduced. Combining the intensified learning with data reconstruction, the framework can systematically find a set of high quality parameters to build accurate models. Furthermore, training time significantly reduces from the order of **months** to the order of **hours**.

Automatic bounding-charge selection

Computing SER with full-spectrum charge collection is still challenging, even using the new models. Therefore, to save time from too many rounds of statistical analysis, a technique of ***automatic bounding-charge selection*** is further proposed to discover charges that only need to be computed by traditional static analysis.

Figure 6 shows the mean, sigma, lower bound (mean − 3 ∗ sigma), and upper bound (mean+ 3 ∗ sigma) of TF distributions, which are induced by different levels of deposited charges. Results show that the mean of pulsewidths increases monotonically as the deposited charge increases. The larger deposited charge also leads to a smaller sigma of its pulsewidth. Hence, larger lower- and upper-bounds of the TF distribution can be observed when the level of charge collection increases.

Based on this finding, a technique of automatic bounding charge selection is proposed to accelerate the overall SER estimation. For computing overall SERs, this study only needs to consider the distribution of a pulsewidth, which overlaps the latching-window (see Figure 7). The pulse-width distributions in dotted lines are entirely masked. The pulse-width distributions in solid lines undoubtedly results in soft errors. In other words, when the lower bound of a TF distribution exceeds than the latching-window size, the SER from such distribution can be replaced by the corresponding static results. On the contrary, the SER from a distribution in the dotted line induced by a weaker deposited charge (its upper bound is smaller than the latching-window size), will be masked completely and can be ignored. Only distributions in dash lines require statistical analysis.

Algorithm 1 shows the pseudocode for automatic bounding charge selection. First, it chooses deposited charge q to strike a gate in the circuit and then it derives the upper and lower bounds of TF distributions from each Flip-Flop. After estimating the upper and lower bounds, the maximum upper bound and minimum lower bound can be found. If the maximum upper bound is smaller than the latching-window size, then the minimum charge ($Q_{\min}$) is obtained. On the other hand, the maximum charge ($Q_{\max}$) is decided when the minimum lower bound of TF distribution is greater than the latching-window size. As a result, the algorithm only considers deposited charges in the range of $[Q_{\min}, Q_{\max}]$ for SER estimation.

ALGORITHM 1: Automatic Bounding Charge Selection()

```
while $Q_{min}$ or $Q_{max}$ are undecided
  pick a charge $q$
  compute each TF dist. latched by each FF
  MaxUpperBound = max(upper bound of
    TF dist.)
  MinLowerBound = min(lower bound
    of TF dist.)
  if MaxUpperBound < latching-window size
    $Q_{min} = q$
  if MinLowerBound > latching-window size
    $Q_{\max} = q$
end
```

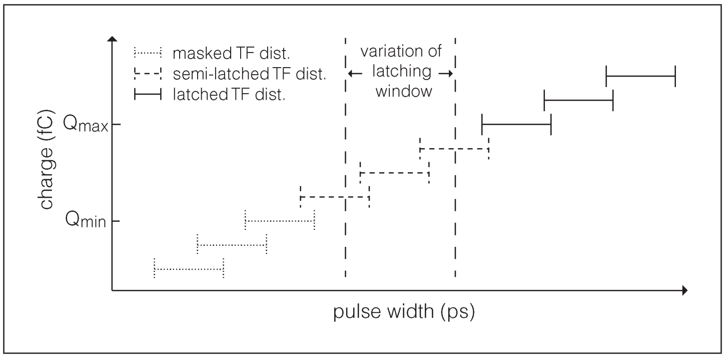

Figure 7. Different pulsewidth distributions versus a latching-window size.

Experimental results

First, this section verifies the accuracy of statistical cell models from the intensified SVM learning with data reconstruction and then compares the results with [11]. Second, the SERs on four sample circuits from Monte Carlo SPICE simulation are compared with SERs from the proposed framework. Last, SERs for large benchmark circuits are evaluated using this approach. Note that the proposed framework was implemented in C++ and run on a Linux machine with a Pentium Core Duo (2.4 GHz) processor and 4 GB RAM. The technology used was 45 nm, Predictive Technology Model (PTM) and the neuron flux rate at sea-level was assumed as 56.5 $\mathrm{m}^{-2}\mathrm{s}^{-1}$. In addition, the size of the latching window was set at 120 ps. Table 1 shows the accuracy of the built models, including three types of cells under full-spectrum charge collection. The error rates of all proposed models are less than those from [11] (see Table 1). The error rates of sigma values for the generated models reduced

Table 1 Comparison of model accuracy.

	Error Rate (%)			
Cell	μ_{first}	σ_{first}	μ_{prop}	σ_{prop}
INV	0.38%	4.45%	1.66%	2.42%
AND	0.39%	3.91%	1.09%	2.27%
OR	0.44%	3.95%	1.51%	2.05%

(a)

	Error Rate (%)			
Cell	μ_{first}	σ_{first}	μ_{prop}	σ_{prop}
INV	2.0%	12.0%	3.7%	3.8%
AND	2.8%	12.9%	2.4%	3.9%
OR	2.6%	11.9%	3.3%	3.7%

(b)

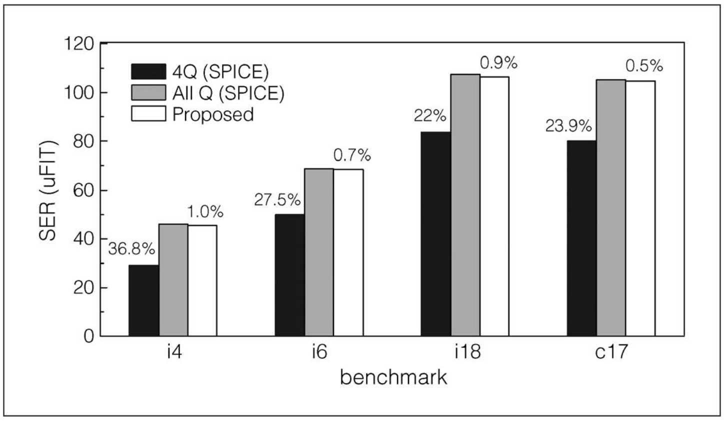

Figure 8. Soft error rate comparison between SPICE simulation and the proposed approach.

significantly from 12% to 4%. Such results state that, the effectiveness of the intensified SVM learning and data reconstruction collectively provide better quality models for further SER estimation.

The c17, and the other three sample circuits from [11], were used to perform Monte Carlo SPICE simulation to validate the accuracy and efficacy of this current method. These four, small-sized circuits could only be affordable on our machines when considering the extremely long Monte Carlo SPICE simulation time for SSER analysis. For example, c17 had only seven gates, 12 striking nodes, and five inputs and took more than three days to finish Monte Carlo SPICE simulation.

Figure 8 visualizes SER comparison between Monte Carlo SPICE simulation (All Q) and the proposed approach (Proposed) on four benchmark circuits. The SER using only four levels of charges (4Q) is also shown. Based on these results, two observations are concluded: 1) differences between the SERs induced by four levels of charges and SERs induced by all levels of charges on i4, i6, i18, and c17 are 36.8%, 27.5%, 22%, and 23.9%, respectively. This result, again, proves that SERs evaluated by only four levels of charges is underestimated and not accurate enough. 2) The SERs differences between Monte Carlo SPICE simulation (All Q) and this full-spectrum-charge SSER framework (Proposed) on i4, i6, i18, and c17 are 1.0%, 0.7%, 0.9%, and 0.5%, respectively. This result proves that the proposed framework results in accurate SERs with the average error rate of 0.8% on benchmark circuits.

Finally, this framework also applies to ISCAS'85 circuits, a series of multipliers (m4 to m32) and an AES cipher from IWLS 2005 benchmark. Table 2 shows the corresponding SERs and also includes results from our implementations of [11] (Column SVR impl. [11]) and [12] (Column Monte Carlo impl. [12]), the selected charges range (Column Q_r),

Table 2 Experimental results of each benchmarks.

		SVR impl. [11]		Monte-Carlo impl. [12]		Our approach					Runtime comparison	
Circuit	Node	SER (FIT)	$T_{[11]}$ (s)	SER (FIT)	$T_{[12]}$ (s)	SER (FIT)	Q_r (fC)	T_r (s)	T_a (s)	T_a/T_r (s)	[11]/Our spd (X)	[12]/Our spd (X)
i4	4	2.52E-5	<0.1	2.18E-5	0.2	4.54E-5	(35,39)	0.002	0.06	30.0	-	100.0
i6	6	4.06E-5	<0.1	4.73E-5	0.3	6.84E-5	(35,39)	0.004	0.1	25.0	-	75.0
i18	12	7.21E-5	<0.1	6.05E-5	0.7	1.06E-4	(35,39)	0.012	0.3	25.0	-	58.3
c17	12	6.66E-5	<0.1	5.31E-5	0.7	1.05E-4	(35,40)	0.016	0.4	25.0	-	43.7
c432	233	1.48E-4	5.9	1.25E-4	114.4	1.72E-3	(35,41)	5.6	113.9	20.3	1.0	20.4
c499	638	1.59E-4	42.9	1.15E-4	870.6	1.77E-3	(35,42)	30.1	692.3	23.0	1.4	28.9
c880	443	2.18E-4	6.1	1.52E-4	173.2	1.93E-3	(35,42)	4.3	138.0	32.1	1.4	40.2
c1355	629	1.36E-4	43.5	1.19E-4	891.8	2.24E-3	(35,42)	29.7	779.1	26.2	1.4	30.0
c1908	425	2.27E-4	18.4	2.12E-4	365.1	1.78E-3	(35,41)	15.6	307.2	19.7	1.1	23.4
c2670	841	3.40E-4	9.6	3.48E-4	401.0	3.95E-3	(35,42)	8.4	193.2	23.0	1.1	47.7
c3540	901	6.67E-4	39.8	7.41E-4	1070.6	4.10E-3	(35,41)	35.5	753.7	21.2	1.1	30.1
c5315	1806	1.09E-3	35.1	1.15E-3	818.2	1.23E-2	(35,41)	30.6	628.0	20.5	1.1	26.7
c6288	2788	8.45E-4	501.5	6.86E-4	15703.1	5.18E-3	(35,42)	628.3	11778.0	18.7	0.8	24.9
c7552	2114	8.89E-4	97.4	1.04E-3	1406.7	5.92E-3	(35,44)	53.1	1041.1	19.6	1.8	26.4
m4 (4bit)	158	1.79E-4	2.4	1.58E-4	98.8	1.48E-3	(35,40)	2.0	39.2	19.6	1.2	49.4
m8 (8bit)	728	6.06E-4	45.1	4.14E-4	710.2	3.80E-3	(35,40)	33.3	699.2	21.0	1.3	21.3
m16 (16bit)	3156	1.47E-3	784.7	1.48E-3	9565	7.92E-3	(35,42)	572.0	11656.4	20.4	1.3	16.7
m24 (24bit)	7234	2.35E-3	3553.2	2.63E-3	39628.5	1.21E-2	(35,42)	3599.3	66330.9	18.4	0.9	11.0
m32 (32bit)	13017	3.21E-3	11142.1	2.82E-3	131535.6	1.64E-2	(35,42)	9049.1	156606.9	17.3	1.2	14.5
AES Cipher	22881	-	-	-	-	1.42E-1	(35,42)	682.2	16978.7	24.9	-	-
Average										22.6	1.2	36.2

runtime using only Q_r (Column T_r), runtime using all levels of charges (Column T_a), runtime speed-up between T_r and T_a (Column T_a/T_r spd), and runtime comparison between [11], [12] and our approach (Column Runtime comparison). The experimental results show that the levels of deposited charges used for analysis were reduced from 98 to 10 at the most because of automatic bounding-charge selection. Therefore, SER estimation is accelerated with a 22.6× speedup, on average, for all circuits. Compared with the Monte Carlo SPICE simulation, the runtime results of i4, i6, i18, and c17 are all less than 0.1 s in the proposed framework, where the speed-up is on the order of 10^7.

Moreover, the runtime comparison from our approach to [11] and [12] are also shown in Table 2. The results indicate that our approach is faster than these two previous approaches by 1.2× and 36.2×, on average, for all circuits. In other words, our approach demonstrates better efficiency than these two approaches even if our approach considers all levels of charge collection instead of using only four levels of charge collection in [11] and [12]. Moreover, if full-spectrum charges are considered in [11], our approach can run approximately 30× faster while maintaining comparable (or even better) SER accuracy.

For accurate SSER analysis, all levels of deposited charges should be considered instead of only four levels. This paper proposes a fast-yet-accurate SSER framework with full spectrum charge-collection analysis. High-quality models (with only 0.8% error rate) were built from the proposed intensified SVM learning and data reconstruction technique. Automatic bounding-charge selection is also integrated into this framework and enables a 22.6X speedup, on average, for benchmark circuits by intelligently filtering out charges that do not need statistical analysis. ■

References

[1] R. R. Rao, K. Chopra, and D. T. Blaauw, ‘‘Computing the soft error rate of a combinational logic circuit using parametrized descriptors,’’ *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 26, no. 3, pp. 468–479, 2007.

[2] N. Miskov-Zivanov and D. Marculescu, ‘‘MARS-C: Modeling and reduction of soft errors in combinational circuits,’’ in *Proc. Design Autom. Conf.*, 2006, pp. 767–772.

[3] B. Zhang, W.-S. Wang, and M. Orshansky, ‘‘FASER: Fast analysis of soft error susceptibility for cell-based designs,’’ in *Proc. Int. Smyp. Quality Electron. Design*, 2006, pp. 755–760.

[4] M. Zhang and N. Shanbhag, ‘‘A Soft Error Rate Analysis (SERA) methodology,’’ in *Proc. Int. Conf. Comput.-Aided Design*, 2004, pp. 111–118.

[5] S. Krishnaswamy, I. Markov, and J. P. Hayes, ‘‘On the role of timing masking in reliable logic circuit design,’’ in *Proc. Design Autom. Conf.*, 2008, pp. 924–929.

[6] R. Rajaraman, J. S. Kim, N. Vijaykrishnan, Y. Xie, and M. J. Irwin, ‘‘SEATLA A soft error analysis tool for combinational logic,’’ in *Proc. Int. Conf. VLSI Design*, 2006, pp. 499–502.

[7] D. Qian, L. Rong, and X. Yuan, ‘‘Impact of process variation on soft error vulnerability for nanometer VLSI circuits,’’ in *Proc. Int. Conf. ASIC*, 2005, pp. 1117–1121.

[8] X. Fu, T. Li, and J. A. B. Fortes, ‘‘Soft error vulnerability aware process variation mitigation,’’ in *Proc. Int. Symp. High Perform. Comput. Arch.*, 2009, pp. 93–104.

[9] K. Ramakrishnan, R. Rajaraman, S. Suresh, N. Vijaykrishnan, Y. Xie, and M. J. Irwin, ‘‘Variation impact on SER of combinational circuits,’’ in *Proc. Int. Symp. Quality Electron. Design*, 2007, pp. 911–916.

[10] Z. Chong and S. Dey, ‘‘Modeling soft error effects considering process variations,’’ in *Proc. Int. Conf. Comput. Design*, 2007, pp. 376–381.

[11] H.-K. Peng, C. H.-P. Wen, and J. Bhadra, ‘‘On soft error rate analysis of scaled CMOS designs—A statistical perspective,’’ in *Proc. Int. Conf. Comput.-Aided Design*, 2009, pp. 157–163.

[12] Y.-H. Kuo, H.-K. Peng, and C. H.-P. Wen, ‘‘Accurate Statistical Soft Error Rate (SSER) analysis using a Quasi-Monte Carlo framework with quality cell models,’’ in *Proc. Int. Symp. Quality Electron. Design*, 2010, pp. 831–838.

[13] W. Sootkaneung and K. K. Saluja, ‘‘On techniques for handling soft errors in digital circuits,’’ in *Proc. Int. Conf. Test*, 2010, pp. 1–9.

[14] R. Naseer, Y. Boulghassoul, J. Draper, S. DasGupta, and A. Witulski, ‘‘Critical charge characterization for soft error rate modeling in 90 nm SRAM,’’ in *Proc. Int. Symp. Circuits Syst.*, 2007, pp. 1879–1882.

[15] Appendix A.2: Reference neutron spectrum in JEDEC JESD89, *Measurement and Reporting of Alpha*

Particles an Terrestrial Cosmic Ray-Induced Soft Errors in Semiconductor Devices, Joint Electron Device Eng. Council, Solid State Technol. Assoc., 2001, pp. 55–58.

[16] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge, U.K.: Cambridge Univ. Press, 2002.

[17] J. Kennedy and R. Eberhart, ''Particle swarm optimization,'' in *Proc. Int. Conf. Neural Netw.*, 1995, pp. 1942–1948.

Hsuan-Ming Huang is currently pursuing the PhD in electrical engineering at National Chiao Tung University (NCTU), Hsinchu, Taiwan. His research interests include data mining, design reliability and automatic test pattern generation in computer-aided design of electronic circuits and systems. He has an MS in communication engineering from NCTU.

Charles H.-P. Wen is an assistant professor with the Department of Electrical and Computer Engineering and the Institute of Communications Engineering at National Chiao Tung University (NCTU), Hsinchu, Taiwan. His research interests include testing and verification of VLSI designs, multicore and cloud computing, and applications of data mining and machine learning. He has a PhD in electrical and computer engineering from the University of California, Santa Barbara. He is a member of the IEEE.

■ Direct questions and comments about this article to Hsuan-Ming Huang, Department of the Electrical Engineering, National Chiao Tung University, Hsinchu 300, Taiwan; hmhuang.eed00g@g2.nctu.edu.tw.