CASSER: A Closed-Form Analysis Framework for Statistical Soft Error Rate

Austin C.-C. Chang, Ryan H.-M. Huang, and Charles H.-P. Wen

Abstract—CMOS designs in the deep submicrometer era require statistical methods to accurately estimate the circuit soft error rate (SER). However, process variation increases the complexity of statistical characteristics related to transient faults, leading to considerable uncertainty in the behavior of soft errors. Regardless of the methods used, current statistical SER (SSER) frameworks invariably involve a tradeoff between accuracy and efficiency. This paper presents accurate cell models in first-order closed form to overcome this problem, thereby enabling the analysis of SSERs in a block-based fashion similar to statistical static timing analysis. These cell models are derived as a closed form in the proposed framework named CASSER, and remain precise under the assumption of a normal distribution for the process parameters. Experimental results demonstrate the efficiency (> 2-order times faster than the latest framework) and accuracy ($< 3\%$ error) of CASSER in estimating circuit SERs, when compared with the Monte Carlo SPICE simulation.

Index Terms—Reliability, single event upset, statistical SER (SSER), statistical static timing analysis (SSTA), transient fault.

I. INTRODUCTION

WITH increased scaling in CMOS technology, the issue of reliability is becoming increasingly important for memory devices [1] as well as soft errors, which are a major failure mechanism for logic circuits. The cause of this type of error is radiation-induced transient faults, which are latched by state-holding elements causing nonpermanent damage to data. Soft error rates (SERs) are much higher than those typically associated with reliability mechanisms, and with recent increases in circuit speeds, soft errors occur even more frequently [2].

Behavioral analysis of soft errors depends on three masking effects [3]: logical, electrical, and timing. As shown in Fig. 1, logical masking occurs when transient faults are blocked along the propagation path by a controlling value on the side-input of one gate. Due to the electrical properties of gates, electrical masking leads to attenuation or amplification of transient faults, depending on the input value of the gates [4]. Timing masking occurs when a transient faults arrive at a state-holding
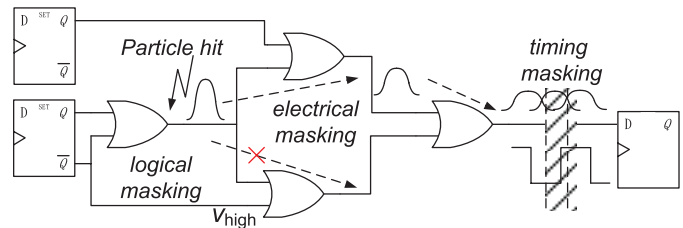
Fig. 1. Three masking mechanisms for soft errors.

element outside its latching time, or its pulse width is smaller than the clock transition window (i.e., setup time plus hold time of such state-holding element).

Several static methods have been devised to evaluate soft errors for combinational logic. FASER [5] and MARS-C [6] applied symbolic techniques to both logical and electrical maskings, scaling error probability according to the specified clock period. The SERA methodology [7] computes SER by evaluating the error-latching probability and the electrical-masking effect without considering logical masking. Krishnaswamy *et al.* [8] proposed static analysis for timing masking by retracing the propagation back from an error-latching window. SEAT-LA [9] and Rao *et al.* [10] achieved good SER estimation, compared to the results of SPICE simulation using pre-characterized models for gates, flip-flops, and the propagation of transient faults. Garg *et al.* [11] evaluated the propagation and electrical characteristics of changes in transient faults through each gate according to a logic function and an analytical model, which were incorporated with a nonlinear transistor current. Rossi *et al.* [12] presented an accurate linear model to estimate the critical charge and indicated that critical charge has stronger dependence on the driving strength of one node than its total capacitance. Furthermore, they also show the impact of aging phenomena on soft error susceptibility in [13]. As a result, SER has been extensively investigated and widely adopted as a key metric for circuit reliability.

In recent years, process variation has become an issue of concern, introducing new challenges in the accurate estimation of SER. Ramakrishnan *et al.* [14] and Natasa *et al.* [15] analyzed how the source of variation influences SER and discovered that traditional static approaches underestimate circuit SER in the presence of process variation. Fig. 2 illustrates the impact of process variation on SER [4] using 45-nm technology, where SERs are measured on a sample circuit with various process variation (σ_{proc}s). Thus, it is clear
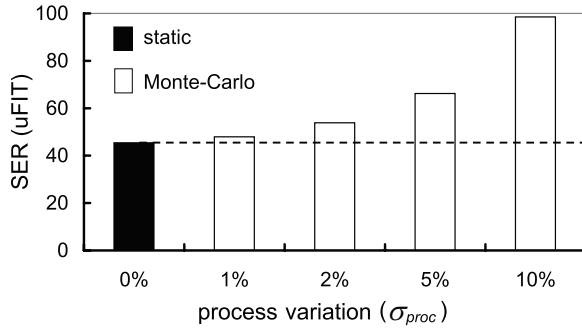
Fig. 2. Differences in SER between static and Monte Carlo SPICE simulation w.r.t. process variation [4].

that the simulation results of SER using static SPICE are underestimated compared to the statistical results. Peng *et al.* [4] applied a state-of-the-art statistical learning algorithm to tackle variation-induced uncertainty and built support vector machine (SVM) models to deal with transient faults. Kuo *et al.* [16] proposed quality table-based cell models to estimate SSER and customized the use of quasi-random sequences to shorten runtime. However, regardless of the approach used in current SSER frameworks, a compromise must be made between efficiency and accuracy.

This paper proposes a novel approach, similar to that of block-based statistical static timing analysis (SSTA) [17], for SSER in which a transient fault is decomposed into two transitions for analysis: a rising edge and a falling edge. Each edge is processed using an analytical approach and statistical static timing analysis [18], which is based on a first-order closed form [19]. Because the transient fault is analyzed using a mathematical method, the timing cost can be largely reduced and timing information can be preserved, which is helpful for describing the interactive behavior of transient faults. However, correlations are the main concern when applying a closed-form block-based approach to the estimation of SSER. Theoretically, all correlations between transition signals and corresponding gate delays must be considered; however, the correlation between transition signals can be overlooked because the difference in SER has been shown to be less than 1% according to our experiments. Thus, we devised a parameterized SSTA named CASSER framework that takes into account the timing correlation to derive more accurate SER. Experimental results demonstrate that CASSER is capable of providing reasonable results much more rapidly than those in previous works.

The remainder of this paper is organized as follows. In Section II, previous studies related to SSTA and SSER are reviewed. In Section III, we propose the outline of the closed-form framework for SSER analysis. A parameterized first-order closed form of transient faults is detailed in Section IV. Section V presents the experimental results, including the accuracy of our models, the SSERs, as well as the runtimes over a variety of ISCAS'85 benchmarks, a series of multipliers, and several industrial circuits. Section VI concludes this paper.

II. SSTA AND SSER REVIEW

In this section, we review the first-order closed-form statistical static timing analysis and the frameworks used to analyze statistical soft error rates (SSERs) in Sections II-A and II-B, respectively.

A. Statistical Static Timing Analysis

Visweswariah *et al.* [19] proposed a canonical first-order delay model that considers both correlated and independent random sources. By expressing timing quantities in closed form, the arrival time and required arrival time can be propagated through a timing graph using a linear-time block-based statistical timing algorithm. Moreover, the local and global criticality probabilities can be computed in a short time. In standard or first-order closed form, a timing quantity t for a gate or wire delay can be expressed as follows:

$$t \triangleq a_0 + \sum_{i=1}^{n} a_i \Delta X_i + a_{n+1} \Delta V_a$$

where a_0 is the nominal value of the delay, ΔX_i represents the variation of n global sources X_i from their nominal value, a_i is the sensitivity of each global source of variation, and $\forall i \in [1, n]$. ΔV_a is the variation of an independent random variable V_a from its nominal value and a_{n+1} is the sensitivity of the timing quantity to V_a.

To apply the first-order closed-form statistical static timing analysis, two operations, sum and max, are required. The procedure of the sum operation of two jointly-distributed random variables is described as follows.

Let $t' = t + d$, where t' is the sum of two mutually-correlated and normally-distributed variables t and d where μ_t, μ_d, σ_t, and σ_d are their means and variations, respectively. The mean and variance of t' can be derived as

$$\begin{aligned}
\mu_{t'} &= E(t') = E(t + d) \\
&= E(t) + E(d) = \mu_t + \mu_d \qquad (1) \\
\sigma_{t'}^2 &= E((t' - E(t'))^2) = E(t'^2) - (E(t'))^2 \\
&= E((t + d)^2) - (E(t + d))^2 \\
&= E(t^2) + 2E(td) + E(d^2) \\
&\quad - (E(t))^2 - 2E(t)E(d) - (E(d))^2 \\
&= E(t^2) - (E(t))^2 + E(d^2) - (E(d))^2 \\
&\quad + 2E(td) - 2E(t)E(d) \\
&= \sigma_t^2 + \sigma_d^2 + 2\rho_{td}\sigma_t\sigma_d \qquad (2)
\end{aligned}$$

where ρ_{td} denotes the correlation coefficient of t and d.

Visweswariah *et al.* [19] further used the concept of tightness probability to deduce the result of the max operation of two timing quantities in closed form. The definition of the max operation is described as follows.

Let $Z = \max(X, Y)$, where Z is the responsive random variable derived by taking a max operation between random variables X and Y. The moment of Z can be derived as

$$\begin{aligned}
\mu_Z &= E(Z) = E(\max(X, Y)) \\
&= \mu_X T_x + \mu_Y (1 - T_X) + \theta_\phi \left(\frac{\mu_X + \mu_Y}{\theta} \right) \qquad (3)
\end{aligned}$$

$$\sigma_Z^2 = \mu_2(Z) = \mu_2(\max(X, Y))$$
$$= (\sigma_X^2 + \mu_X^2)T_X + (\sigma_Y^2 + \mu_Y^2)(1 - T_X)$$
$$+(\mu_X + \mu_Y)\theta_\phi\left(\frac{\mu_X - \mu_Y}{\theta}\right) - \mu_Z^2. \qquad (4)$$

The definition of tightness probability T_X is the probability of random variable X being larger than random variable Y. θ is the intermediate notation used to compute T_X. More details regarding (3) and (4) can be found in [20] and [21].

Similarly, in our SSER framework, a transient fault is split into two transition signals, which are both timing quantities and expressed in closed forms. Thus, they can also be efficiently analyzed in a parameterized block-based method like SSTA. The only difference is that SSER considers changes in the pulse-width of a transient fault, whereas SSTA emphasizes the timing signal with the maximum delay.

B. SSER Analysis

Process variation often results in unpredictable transient fault behavior that cannot be accurately estimated using static approaches. Both learning-based and simulation-based methods of SSER analysis have been studied in the literature.

Peng et al. [4] re-examined the soft-error behaviors caused by radiation-induced particles under process variation, and found that transient faults do not monotonically diminish after propagation. In other words, both the amplification and attenuation of transient faults are possible. Moreover, they found that the traditional static methods underestimate the SER because the weak charge-induced soft errors are overlooked, and hence proposed a learning-based framework to cope with these complex issues. The main idea behind predicting the behavior of soft errors is to analyze three masking effects using machine learning. Although learning-based algorithms can be more effective than simulation-based methods, the accuracy and efficiency require further improvement.

According to [4], the components of a SSER problem can be divided into:

1) signal-probability computation;
2) electrical-probability computation.

where the electrical probability computation includes the timing-masking effect and the electrical masking-effect, and the signal probability computation corresponds to the logical-masking effect. More details of each component are provided in Section III.

Because the quality statistical model has been the bottleneck in all previous SSER frameworks, SSER results of satisfactory accuracy have not yet been achieved. For this reason, Kuo et al. [16] proposed accurate table-based cell models for transient fault distributions, according to which a Monte Carlo SSER analysis framework was built. By looking up pre-characterized table cells, both the sample points of hitting and the propagation transient faults can be obtained in each iteration, whereupon a new distribution of the hitting and propagation models is computed from these points. To shorten the runtime, the authors deployed a heuristic algorithm to customize the use of quasi-random sequences, which accelerated the convergence of simulation error. Despite the accuracy of
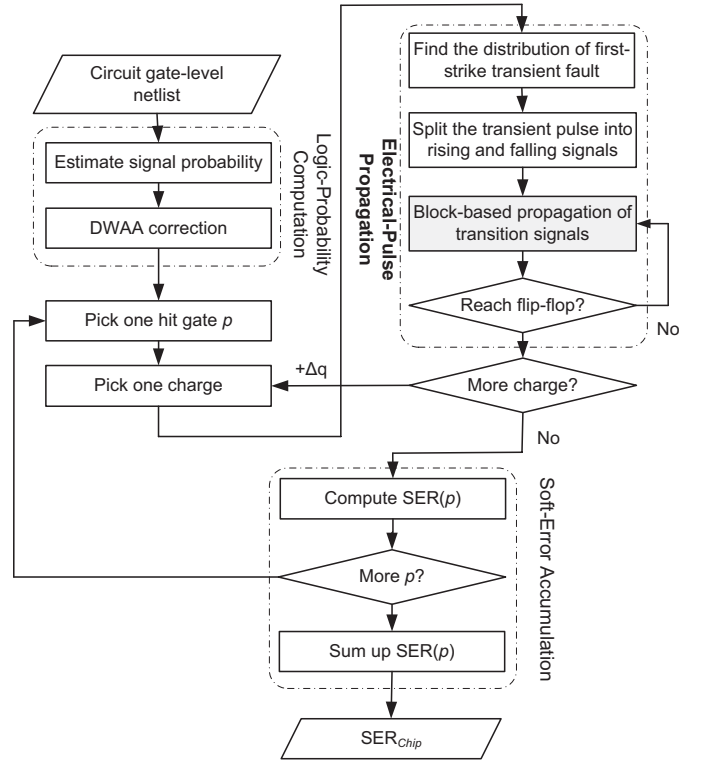

Fig. 3.   SSER analysis framework.

the SSER results, long simulation time was required, resulting in a simulation-based method that is inapplicable for industrial circuits.

Both of the studies described above deal with the computation of electrical probability, but differ in the computational methods used to derive the transient-fault distribution. The aim of this paper was to achieve high efficiency and accuracy simultaneously during the computation of the transient-fault distribution, through linear closed-form formulation, as shown in Fig. 3. After acquiring the distribution of transient faults, the occurrence of soft errors on the flip-flops can be determined by checking whether these transient faults are smaller than the error-latching window of the flip-flops. If a transient fault is wide enough, a soft error is captured; otherwise, it is masked.

III. SSER ANALYSIS

In this section, we review the analysis of soft error rate considering the impact of process-variation beyond the deep submicrometer era [4]. Overall analysis comprises three main components: 1) computation of logic probability; 2) electrical-pulse propagation; and 3) the accumulation of soft errors. A flowchart of the overall process is shown in Fig. 3. The following sections deal with each component in detail and the global view of such a linear closed-form formulation, respectively.

A. Accumulation of Soft-Errors

The overall SER can be defined as the accumulation of soft errors ($SE(\cdot)$) resulting from particle hits at each individual

gate (c_i) in the circuit. That is

$$\text{SER}_{\text{total}} = \sum_{i=1}^{\#_{\text{gate}}} \text{SE}(c_i) \tag{4}$$

where $\#_{\text{gate}}$ denotes the total number of gates susceptible to hits by radiation particles in the circuit. Note that the transient fault caused by a particle hit may propagate and be captured by different state-holding elements, resulting in numerous soft errors.

Each $\text{SE}(c_i)$ can be further formulated by integrating the products of the particle-hit rate and the error probability over the range of charge strength from $q_{\min}$ to $q_{\max}$ as

$$\text{SE}(c_i) = \int_{q=q_{\min}}^{q_{\max}} \text{R}_{PH}(q) \times \text{Pr}_{\text{err}}(c_i, q) \, dq \tag{5}$$

where $\text{Pr}_{\text{err}}(c_i, q)$ denotes the probability of a transient fault originating from a collection charge with strength q at hit node c_i being latched by one flip-flop.

In (5), $\text{R}_{PH}(q)$, the particle-hit rate, is the effective frequency at which particle with strength q hits the circuit in unit time, defined in [3] and [7] as

$$\text{R}_{PH}(q) = F \times K \times A(c_i) \times \frac{1}{q_s} \times \exp\left(\frac{-q}{q_s}\right) \tag{6}$$

where F, K, $A(\cdot)$, and q_s denote the constants for neutron flux (> 10 MeV), the technology-independent fitting parameter, the susceptible area in cm^2, and the slope of charge collection, respectively.

One key point that can be observed from (5) and (6) is that small charge collection occurs more frequently than large charge collection and accounts for the difference between static SER and SSER in [4]. Moreover, for a practical SSER analysis framework, the above continuous integration in (5) is often approximated by a sum of discretized charges. That is

$$\text{SE}(c_i) = \sum_{k=1}^{n} \text{R}_{PH}(q_k) \times \text{Pr}_{\text{err}}(c_i, q_k) \tag{7}$$

where $q_k = k \times (q_{\max} - q_{\min})/n$. According to [5] and [4], empirically, $n = 3$ or $n = 4$ is sufficient to attain a satisfactory level of accuracy in SER.

The error probability $\text{Pr}_{\text{err}}(c_i, q)$ depends on all three masking effects illustrated in Fig. 1, which can be further decomposed into

$$\text{Pr}_{\text{err}}(c_i, q) = \sum_{j=1}^{\#_{\text{FF}}} \text{Pr}_{\text{logc}}(c_i, d_j) \times \text{Pr}_{\text{elec}}(c_i, d_j, q) \tag{8}$$

where $\#_{\text{FF}}$ and d_j represent the total number of flip-flops in the circuit and the jth flip-flop, respectively. Pr_{logc} and Pr_{elec}, respectively, denote the logic-masking probability and the electrical probability related to the electrical-masking and timing-masking effects. Corresponding details are elaborated in the following sections.
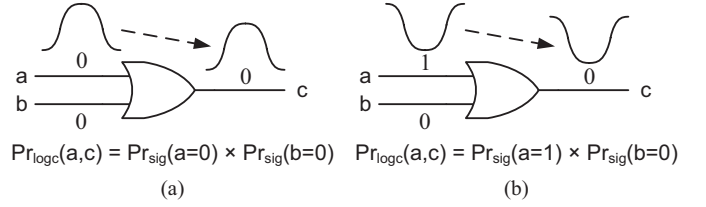

Fig. 4. Signal probability for one OR gate. (a) Positive transient fault. (b) Negative transient fault.

B. Computation of Logic Probability

$\text{Pr}_{\text{logc}}(c_i, d_j)$ represents the overall logic probability of successfully propagating the transient faults through a path from gate c_i to flip-flop d_j (denoted by $c_i \rightsquigarrow d_j$). $\text{Pr}_{\text{logc}}(c_i, d_j)$ can be computed by signal probabilities in which a transient fault can generate one c_i, which is multiplied by all signal probabilities of noncontrolling value of all gates on paths toward d_j, expressed as

$$\text{Pr}_{\text{logc}}(c_i, d_j) = \text{Pr}_{\text{sig}}(c_i^*) \times \prod_{c_k \in c_i \rightsquigarrow d_j} \text{Pr}_{\text{sig}}(c_k) \tag{9}$$

where $\text{Pr}_{\text{sig}}(c_i^*)$ is the probability of logic-0 (logic-1) when a positive (negative) transient fault is generated at c_i, and c_k, which is neither c_i nor d_j, is another gate along the path ($c_i \rightsquigarrow d_j$). $\text{Pr}_{\text{sig}}(c_k)$ represents the signal probability for a noncontrolling side-input that does not impede a transient fault propagating through gate c_k.

Take Fig. 4 as an example to compute Pr_{logc}. Assume that the probability of being 1 at input a is P_a, and is therefore P_b. The signal requirement for propagating a positive transient fault is both $a = 0$ and $b = 0$, as shown in Fig. 4(a). Hence, the probability of such an event occurring is $\text{Pr}_{\text{logc}} = (1 - \text{P}_a) \times (1 - \text{P}_b)$. To propagate a negative transient fault as shown in Fig. 4(b), the necessary conditions are $a = 1$ and $b = 0$, therefore the corresponding probability is $\text{Pr}_{\text{logc}} = \text{P}_a \times (1 - \text{P}_b)$. The probabilities for other types of gates can be similarly derived.

C. Electrical-Pulse Propagation

$\text{Pr}_{\text{elec}}(c_i, d_j, q)$ in (8) reflects the electrical-masking and timing-masking effects on the transient fault induced by a charge q along the path $c_i \rightsquigarrow d_j$, which can be further decomposed into

$$\begin{aligned} \text{Pr}_{\text{elec}}(c_i, d_j, q) &= \text{Pr}_{t\text{-mask}}(pw_j, w_j) \\ &= \text{Pr}_{t\text{-mask}}(f_{e\text{-mask}}(c_i, d_j, q), w_j) \end{aligned} \tag{10}$$

where $\text{Pr}_{t\text{-mask}}(\cdot)$ and $f_{e\text{-mask}}(\cdot)$ accounts for the timing-masking and electrical-masking effects, respectively.

In order to analyze the timing masking effect, the error-latching probability (PL) for one flip-flop is defined in [5] and [6] as shown in the following:

$$\text{PL} = \frac{pw - w}{t_{\text{clk}}} \tag{11}$$

where pw, w, and t_{clk} denote the pulse width of the arrival transient fault, the latching window ($t_{\text{setup}} + t_{\text{hold}}$) of the flip-flop, and the clock period, respectively. However, pw and w

become random variables under process variation. Therefore, we apply a new random variable v defined as $v = pw - w$ to compute $\text{Pr}_{t\text{-mask}}(\cdot)$ where μ_v and σ_v are its mean and standard deviation

$$\text{Pr}_{t\text{-mask}}(pw, w) = \frac{1}{t_{\text{clk}}} \int_0^{\mu_v + 3\sigma_v} v \times P(v > 0) dv.$$

On the other hand, the electrical-masking function, $f_{e\text{-mask}}()$, reflects the pulse-width change of transient faults passing through a gate and can be defined as the following.

Given gate c_i where a charge with strength q hits and causes a transient fault, and the flip-flop d_j at which the transient fault finally propagates to, and assuming that the transient fault propagates along the path $c_i \rightsquigarrow d_j$ through node $v_0, v_1, \ldots, v_n, v_{n+1}$ where v_0 and v_{n+1} denote the hit gate c_i and flip-flop d_j, respectively, the corresponding electrical masking function is

$$f_{e\text{-mask}}(c_i, d_j, q) = \underbrace{\psi_{\text{prop}}(\cdots(\psi_{\text{prop}}(\psi_{\text{prop}}(pw_0, 1), 2), \ldots), n)}_{n \text{ times}} \quad (9)$$

where $pw_0 = \psi_{\text{hit}}(c_i, q)$ is the initial pulse width induced by a particle with a charge q hitting at gate c_i and $\forall k \in [0, n)$, $pw_{k+1} = \psi_{\text{prop}}(pw_k, k+1)$ represents the resulting pulse width after propagating through v_{k+1}.

ψ_{hit} and ψ_{prop} in (9) represent the first-hit and propagation distribution functions, respectively, reflecting the behavior of transient faults during generations and propagations. Both functions are nondeterministic and can be used to approximate SER in our framework.

Accordingly, efficient and accurate models, ψ_{hit} and ψ_{prop}, become the most critical components due to the difficulty in integrating the impact of process variation on soft errors. In this paper, both ψ_{hit} and ψ_{prop} are derived in first-order closed forms; therefore, the deduction over ψ_{hit} and ψ_{prop} (to approximate ψ_{hit} and ψ_{prop}, respectively) can be conducted using the method of moment estimation (MME) [22]. Accordingly, the estimated electrical-masking function in (9) can be modified as

$$f_{e\text{-mask}}(c_i, d_j, q) \approx \underbrace{\psi_{\text{prop}}(\cdots(\psi_{\text{prop}}(\psi_{\text{prop}}(\widehat{pw}_0, 1), 2), \ldots), n)}_{n \text{ times}}$$

where $\widehat{pw}_0 = \psi_{\text{hit}}(c_i, q)$ and $\forall k \in [0, n)$, each $\widehat{pw}_{k+1} = \psi_{\text{prop}}(\widehat{pw}_k, k+1)$ is an estimator for the pulse width after propagating through v_{k+1} along the path $c_i \rightsquigarrow d_j$.

D. Algorithm of Transient-Fault Propagation

Because it is possible for a transient fault to occur at any gate on the circuit under test, all gates must be considered as candidates for hit gates. As soon as the hit gate c_i is determined, the transient fault induced by a particle hit at the output of c_i is generated and split into rising and falling transitions using the first-hit model ψ_{hit}, whereupon the propagation model ψ_{prop} is employed to propagate both transitions. Transitions appearing at one primary output (PO) or pseudo PO (PPO) are merged to reconstruct the transient

---

Algorithm 1 Transient_fault_at (hitGate c_i)

1: Split transient fault at c_i into t_r^0 and t_f^0
2: Mark propagation tree (G_{prop}) rooted at hit gate c_i
3: Sort G_{prop} topologically
4: **repeat**
5:    Gate Z = output of next gate c_j in G_{prop}
6:    D = Get_Moment(c_j)
7:    **if** Z is not a RFON **then**
8:      X = on-path input of c_j
9:      t_x = Get_moment(X)
10:     t_z = sum(D, t_x)
11:   **else**
12:     (X, Y) = inputs of c_j
13:     t_x = Get_moment(X)
14:     t_y = Get_moment(Y)
15:     t_x' = sum(D, t_x)
16:     t_y' = sum(D, t_y)
17:     t_z = **mix**(t_x', t_y')
18:   **end if**
19: **until** all gates in G_{prop} are VISITED
20: Merge transitions into transient faults
21: **return** transient-fault moments at one PO/PPO

---

faults, which are then used to compute SER. The pseudocode of the algorithm for electrical-pulse propagation is shown in Fig. 3 and described below.

In the generation stage, the first-hit model ψ_{hit} is used to deduce the distribution of the particle-induced transient fault on the output pin of the hit gate c_i. The initial transient fault is then split into a rising-transition signal and a falling-transition signal, denoted as t_r^0 and t_f^0, respectively, and their moments can also be deduced by ψ_{hit}.

The propagation stage starts after the generation stage and can be divided into three steps: in the first step, the breath-first search is employed to acquire the propagation tree G_{prop} of the transient fault starting from c_i and terminating at any PO or PPO. Once a gate is visited, it is added to G_{prop} and the flag is set as visited so that any gate on the reconvergent gates will not be added again. After G_{prop} is built, all gates in G_{prop} are ranked according to their topological orders.

In the second step, the initial transition signals t_r^0 and t_f^0 are propagated along G_{prop} using the propagation model ψ_{prop} in a block-based fashion. During propagation, the two conditions are handled in different ways. For the case in which the output pin of the current gate c_j is a reconvergent fanout node (RFON), sum and mix (introduced in Section IV) operations are deployed to deal with the issue of convolution of transient faults. For the opposite case, only the sum operation is required.

In the final step, the transient faults arriving at one PPO or PO are reconstructed by merging t_r and t_f, and combined pulse-width distributions are used to compute SER, accordingly. Details regarding ψ_{hit} and ψ_{prop} are described in the following section.

IV. FIRST-ORDER CLOSED FORMS FOR ψ_{hit} AND ψ_{prop} IN CASSER

Traditional Monte Carlo methods for SSER analysis are known to suffer from long simulation times when deriving the pulse-width distribution for particle hits and transient-fault propagation. Therefore, this paper employs a parameterized first-order closed form for these two distributions. We simply divide a transient-fault into two transition signals (rising and falling), and each signal can be analyzed individually. Accordingly, rising and falling transitions are modeled as two normally distributed random variables, t_r and t_f. Moreover, the first-hit and propagation distribution functions, ψ_{hit} and ψ_{prop}, can be expressed in the form of

$$\psi : \vec{x} \rightarrow \vec{y}$$

where $\vec{x}$ denotes a vector of input variables and $\vec{y}$ denotes a vector of target values. $\vec{x}$ provides guidance to find the target $\vec{y}$ in the models and includes several relationships of electrical and physical properties between gates and transient faults.

For example, the width of a transient pulse hitting the output of a gate decreases as the output load of the gate increases (because the charging/discharging time of capacitors increases). Another example is that a hitting charge with greater strength causes a wider transient pulse. Hence, for the first-hit model ψ_{hit}, $\vec{x}$ includes charge strength, the type of driving gate, and output loads; $\vec{y}$ contains the distribution of initial pulse width, correlation coefficients, and slopes of the two transitions. Similarly, for ψ_{prop}, $\vec{x}$ consists of the same components as $\vec{x}$ in ψ_{hit} with an additional component – the slope of the transition signal; $\vec{y}$ contains the transition slope, the distribution of gate delay, the correlation between transition signal and the corresponding gate delay, and the correlation between transition signals.

From the proposed idea, a random variable pw, denoting the width of a particle-induced transient pulse can be decomposed into two normal jointly-distributed random timing quantities, the rising edge of transition (t_r) and the falling edge of transition (t_f), expressed as

$$pw = \begin{cases} t_f - t_r, & \text{if the pulse is positive} \\ t_r - t_f, & \text{if the pulse is negative.} \end{cases} \quad (10)$$

Based on ψ_{hit} and ψ_{prop}, both t_r and t_f can be computed by a parameterized SSTA-like method where the approximated distribution of pw can be derived by replacing the statistical variables μ_{pw} and σ_{pw} with the estimators $\widehat{\mu}_{\mathrm{pw}}$ and $\widehat{\sigma}_{\mathrm{pw}}$.

The overall analysis is outlined as follows.

1) *Transient-Fault Generation and Decomposition:* Initially, the first-hit model ψ_{hit} is used to look up the distribution of the initial pulse width pw_0 from a pre-characterized table according to the output load of the hit gate and the strength of the hitting charge. Then, the estimated pulse width $\widehat{pw}_0$ is decomposed into two initial transitions t_r^0 and t_f^0 according to the ratio of their slopes.

2) *Block-Based Propagation:* Two timing signals are updated by ψ_{prop} whenever they are propagated through one gate, reflecting the gate delay. This step repeats until
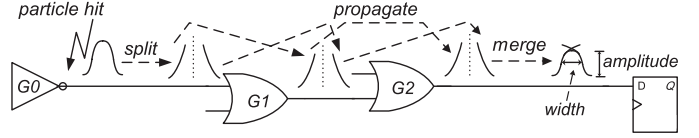

Fig. 5. SSTA-based method w/o considering the correlation between transition signals.

both the rising and falling signals arrive at one PO or PPO.

3) *Pulse-Width Reconstruction:* Once both signals reach PO or PPO, they are merged to reconstruct a new transient pulse to determine whether or not a soft error has occurred. The reconstruction step uses the idea proposed in (10).

Note that, when we split one transient fault into two transition signals, the related important information, such as its amplitude is also embedded implicitly in the timing models (t_r and t_f) to correctly estimate the behavior of a transient fault. To take Fig. 5 for example, the original transient pulse generated by a particle hit at the output of G0 is split into two transition signals, which then individually begin their propagation. Finally, both signals end at G2 and are merged to reconstruct the transient pulse based on t_r and t_f.

Details of each step are organized as follows. After introducing the first-hit model and propagation model in Section IV-A, the distributions of the width in a transient fault are estimated by the MME [21] in Section IV-B. The two issues related to correlation and reconvergence are discussed in Sections IV-C and IV-D, respectively.

A. Constructing Linear Timing Models

In the first step, ψ_{hit} is responsible for approximating the distribution of t_r^0 and t_f^0 and the corresponding computations can be enumerated as

$$\mu_{t_r^0} = 0$$
$$\mu_{t_f^0} = \mu_{\widehat{pw}_0}$$
$$\sigma_{t_r^0}^2 = \sigma_{t_f^0}^2 \times \tau_{r/f}^0$$
$$\sigma_{t_f^0}^2 = \sigma_{\widehat{pw}_0}^2 / (1 + (\tau_{r/f}^0)^2 - 2\tau_{r/f}^0 \times \rho_{t_r^0 t_f^0})$$

where the superscript is the corresponding topological order originating from hit gate G0, $\tau_{r/f}^0$ denotes the slope ratio defined as the slope of the rising signal to that of the falling signal, and $\rho_{t_r^0 t_f^0}$, pre-characterized into a table, is the correlation coefficient of t_r^0 and t_f^0.

After obtaining the distributions of the two initial transition signals, the linear timing model ψ_{prop} is deployed to propagate both signals toward the primary outputs. The derivation of the linear timing model ψ_{prop}, computed by typical statistical static timing analysis, is given as:

Transition signal t arrives at the input of a gate with delay d, where t and d can be expressed in linear closed form as

$$t = t_0 + \sum_{i=1}^{n} a_i \Delta X_i + a_{n+1} \Delta V_a$$

and

$$d = d_0 + \sum_{i=1}^{n} b_i \Delta X_i + b_{n+1} \Delta V_b.$$

Note that t_0 and d_0 are the nominal values of t and d, respectively. ΔX_i is the variation of n global sources from their nominal values; a_i and b_i represent the sensitivities of the transition signal and gate delay, respectively, of each ΔX_i. Both ΔV_a and ΔV_b are variations of the independent random variables V_a and V_b from their mean values, and their timing sensitivities are denoted as a_{n+1} and b_{n+1}, respectively.

After the timing signal t passes through the gate, the output timing signal t' is updated as $t + d$, enabling us to deduce t' by a sum operation of two normal jointly-distributed random variables, as described in Section II-A. Hence, a rising signal t_r^{in} and falling signal t_f^{in} at the gate input can be propagated to the gate output and modeled by ψ_{prop}. Accordingly, the two output timing signals become

$$t_r^{\text{out}} = t_r^{\text{in}} + d_r \qquad (11)$$
$$t_f^{\text{out}} = t_f^{\text{in}} + d_f \qquad (12)$$

where subscripts r and f represent rising and falling, respectively, and the superscripts (input or output) represent the pin locations.

Since we have deduced the first-hit model ψ_{hit} and the propagation model ψ_{prop}, the pulse width of a transient fault can be approximated using (10).

B. Estimating Pulse-Width Parameters

Given the first-hit model ψ_{hit} and the propagation model ψ_{prop}, the final distribution of $\widehat{pw}$ in Fig. 5 can be further expanded according to (10). That is

$$
\begin{aligned}
\widehat{pw} &= t_f^2 - t_r^2 \\
&= (t_f^1 + d_f^2) - (t_r^1 + d_r^2) \\
&= \left(t_f^0 + \sum_{i=1}^{2} d_f^i \right) - \left(t_r^0 + \sum_{i=1}^{2} d_r^i \right) \qquad (13)
\end{aligned}
$$

where the superscript is the corresponding topological order originating in the hit gate.

Thus, the distribution of $\widehat{pw}$ can be calculated by performing a series of sum operations over transition signals and corresponding gate delays. To derive the general form of a transient pulse, which is generated at one hit gate at the m-th level and propagated to one flip-flop at the n-th level where $n > m$, we can generalize (13) and rewrite it as

$$
\begin{aligned}
\widehat{pw} &= t_f^{n-m} - t_r^{n-m} \\
&= \left(t_f^0 + \sum_{i=1}^{n-m} d_f^i \right) - \left(t_r^0 + \sum_{i=1}^{n-m} d_r^i \right). \qquad (14)
\end{aligned}
$$

C. Determining Whether to Consider Transition Correlation

Correlation is a major concern when using a first-order closed-form method to approximate the behavior of transient pulses. This is because the pair of transition signals t_r and t_f are mutually dependent rather than completely uncorrelated.
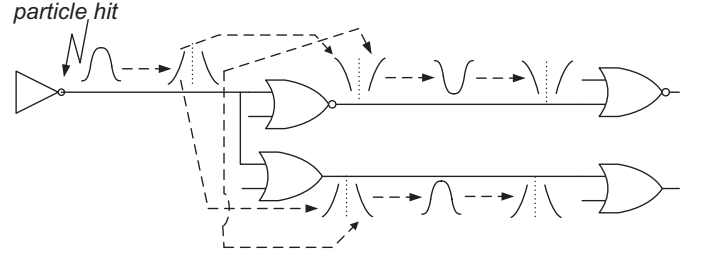

Fig. 6.   Process of iterative split and merge.

TABLE I
COMPARISON OF SER W/O AND W/CONSIDERING THE CORRELATION BETWEEN TRANSITION SIGNALS

Circuit	(a) SSER$_{\text{indep.}}$ (μFIT)	(b) SSER$_{\text{corr.}}$ (μFIT)	Difference (%) $\frac{|(b)-(a)|}{(a)} \times 100$
c17	180.91	180.92	5.52×10^{-5}
c432	2.28×10^5	2.28×10^5	1.82×10^{-3}
c2670	8.00×10^4	8.01×10^4	6.62×10^{-4}
c6288	8.10×10^7	8.10×10^7	9.88×10^{-8}
5-32 decoder	7.16×10^3	7.16×10^3	4.02×10^{-4}

Intuitively, the solution to this issue is to iteratively split and merge the transient faults during propagation. As illustrated in Fig. 6, a transient pulse is reconstructed by merging t_r and t_f after both transitions pass through a gate, and then splitting them again before they are propagated toward the succeeding gates.

Experimental results show that this process can be skipped because the impact of the correlation between transition signals on SSER is small. In Table I, the name of each circuit is listed in column 1; the remaining two columns show the results derived by the closed-form block-based SSER framework with independent transition signals (a) and with correlated transition signals (b), respectively. The last column computes the difference of the SSER results derived using these two methods. According to Table I, it is clear that the difference between the SER results derived by the two methods is negligible on four ISCAS'85 benchmark circuits (where the signal correlation is strong) and a 5-to-32 decoder circuit (where the correlation is weak). In other words, the correlation between transition signals is independent of SER estimation and thus can be overlooked in our framework.

D. Handling the Re-Convergence of Transient-Faults

The number of transient faults doubles if there is a reconvergent structure along the propagation path in the circuit, resulting in an exponential increase in the complexity of the SSER analysis. As shown in Fig. 7, a particle hits the output of G0 and induces a transient pulse. The transient faults then propagate along the paths in a block-based fashion, finally reconverging at the inputs of U0 and U1. Consequently, two positive transient faults appear on the output of U0, and two transient faults with different directions appear on the output of U1.

To resolve this problem of reconvergence, we propose a two-stage approach. In the first stage, transient faults are
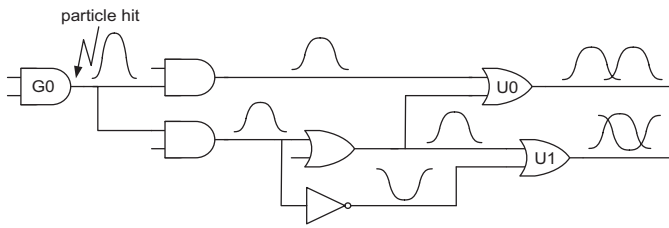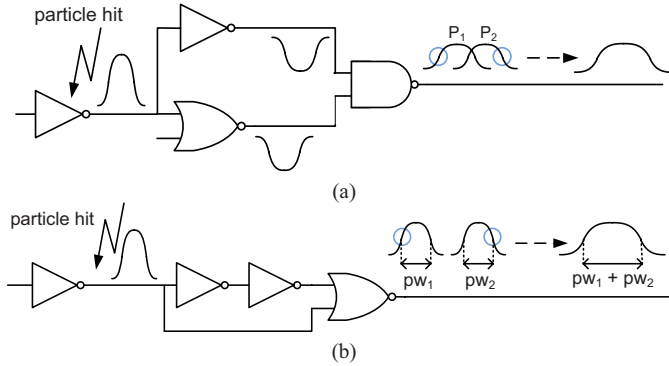
Fig. 7.   Reconvergent structure.

(a)

(b)

Fig. 8.   Illustration of mix operation in the same direction. (a) Overlapping case. (b) Non-overlapping case.

Fig. 9.   mix operation in opposite directions for AND and OR gates. (a) Non-overlapping. (b) Overlapping. (c) Non-overlapping. (d) Overlapping

classified into two groups according to their directions. The outcomes of the pulse width and the logic probability of these convoluted transient faults are then derived in the second stage. The pulse-width distribution of convoluted transient faults is derived using a newly-defined mix operation in which the logic probability is updated as the union of the logic probabilities associated with these transient faults.

1) Computing Re-Convergent Transient-Faults: The reason for defining a new mix operation for the two timing signals is that the pulse-width result of transient faults is underestimated and incorrect, if the traditional max operation is used to deduce the result of these convoluted timing signals.

The process for handling multiple positive transient faults can be expressed as

$$\text{mix}(pw_1, pw_2, \ldots, pw_n)$$
$$= \text{mix}(t_{f1}, \ldots, t_{fn}) + \text{mix}(t_{r1}, \ldots, t_{rn}). \quad (15)$$

The mix operation with multiple (>2) operands such as in (15) is computed by iteratively taking the two-operand mix. Let $t' = \text{mix}(t_1, t_2)$, t_1 and t_2 follow normal distributions, and so as t'

$$\text{mix}(t_1, \ldots, t_k) = \text{mix}(\text{mix}(t_1, t_2), \cdots, \text{mix}(t_k, t_{k+1}))$$
$$= \text{mix}(t_{1, \lfloor \frac{k}{2} \rfloor}, t_{\lceil \frac{k}{2} \rceil, k})$$
$$= t_{1,k}. \quad (16)$$

The two-operand mix can be further classified into two types to deduce convoluted pulses in the same directions and those in opposite directions.

To derive the pulse width of reconvergent transient faults in the same direction, we define the same-direction mix operation as a worst-case operation in which the new pulse comprises the latest transition signal and the earliest transition signal
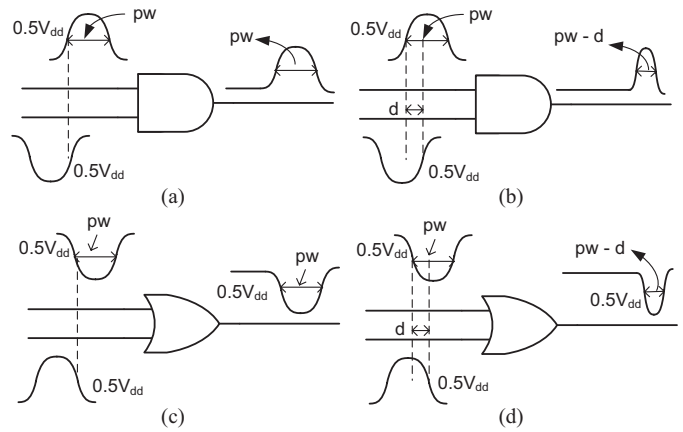
among these reconvergent transient faults. Before performing same-direction mix operations over two reconvergent transient faults, the existence of overlapping is checked. As shown in Fig. 8(a), in the event of overlapping, the earliest transition and the latest transition are selected to form a new pulse; otherwise, the width of the new transient fault is the sum of the widths of the two convoluted transient faults, as displayed in Fig. 8(b).

The results derived using the traditional max operation in SSTA may lead to an underestimation of the pulse-width associated with reconvergent transient faults. Taking Fig. 8(a) as an example, we denote the latter transient fault and former transient fault as P_1 and P_2, respectively. The result deduced by the same-direction mix operation performed on P_1 and P_2 should be the latest transition and the earliest transition among them, respectively, denoted as t_{r1} and t_{f2}. However, the results derived using the traditional max operation performed on P_1 and P_2 are t_{r2} and t_{f2}. Similarly, in Fig. 8(b), the pulse-width result deduced by SSTA's max operation is pw_2 rather than $pw_1 + pw_2$.

For reconvergent transient faults in opposite directions, the pulse width is determined according to interactive behavior. In Fig. 9, if the positive transient fault appearing at one input of an AND gate does not overlap with the negative transient fault appearing at the other input of the AND gate, the pulse-width result is the width of the positive transient fault pw, because the negative transient fault is completely masked by the controlling value on the side input. In the event of overlapping, the result is computed as the width of positive transient fault pw subtracted by the overlapping period (d) between the positive and negative transient faults due to the negative transient fault masking part of the positive transient fault. Other gate types can be derived in a similar manner.

It is worth noting that because the timing information of transition signals is preserved, the issue of reconvergence can be analyzed in a manner that would be impossible in traditional SSER methods [4], [16].

2) Updating Logic Probability: The logic probability at reconvergence fanout nodes should be updated to reflect the phenomenon of reconvergence. For convoluted transient faults, the result of logic probability is the union of the logic
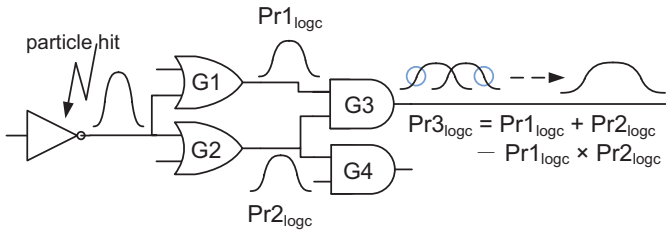
Fig. 10. Illustration of updating logic probability at a RFON.

TABLE II
SUMMARY OF MODEL ERROR

Cell	Error (%)			
	ψ_{hit}^{μ}	$\psi_{\text{hit}}^{\sigma}$	ψ_{prop}^{μ}	$\psi_{\text{prop}}^{\sigma}$
INV	−0.42	−1.29	0.15	−4.76
AND	−0.37	−0.96	1.96	−6.98
OR	−0.52	−3.46	1.85	−8.55
Average	−0.43	−1.90	1.32	−6.76

probabilities of input transient faults, because this condition is equivalent to all of these transient faults being able to pass through the reconvergent node. Taking Fig. 10 as an example, the logic probabilities of transient faults at the output pins of gate G1 and gate G2 are denoted as $\text{Pr}1_{\text{logc}}$ and $\text{Pr}2_{\text{logc}}$, respectively. The logic probability of a transient fault at the output pin of gate G3, denoted as $\text{Pr}3_{\text{logc}}$, as illustrated in Fig. 10.

V. EXPERIMENTAL RESULTS

In this section, the experiments are divided into two parts. In the first part, we examine the accuracy of pre-characterized models ψ_{hit} and ψ_{prop}, which are used extensively in the proposed framework. In the second part of the experiments, these pre-characterized models are integrated into the SSER analysis framework CASSER. Monte Carlo SPICE simulation results are compared with results from [4] and from CASSER to assess the characteristics of SER analysis.

A. Accuracy of ψ_{hit} and ψ_{prop} Models

To extract delay characteristics related to each type of gate in a 45-nm Nangate Open Cell Library [23], we performed extensive Monte Carlo SPICE simulation on randomly generated benchmark circuits. Training delay data for each gate type can be summarized in three steps. In step 1, all of the gates along the propagation path are randomly selected after the path is generated; in step 2, a number of output loads composed of randomly selected gates are arbitrarily selected for each gate along the propagation path; in step 3, the characteristics of the transient faults induced by radiation particles with various charge strength are extracted by performing Monte Carlo SPICE simulation. After obtaining these simulation results, data was grouped according to the charge strength of radiation particles, the transition slope, and the output loads. Details can be found in [4].

Figs. 11 and 12 compare the results from the probability density function (PDF) of transient faults induced by four
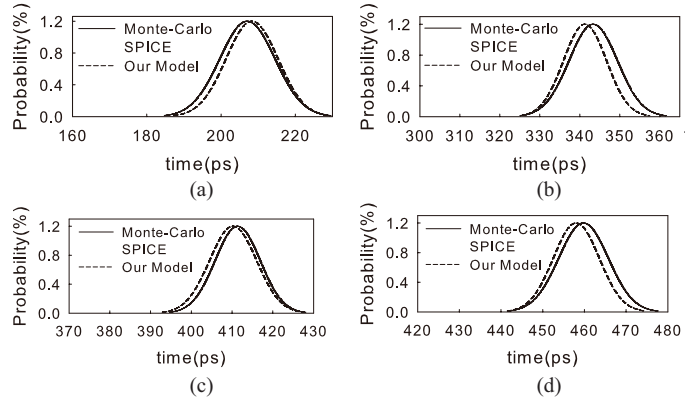

Fig. 11. Model accuracy of AND gates. (a) Under $34fC$ charge. (b) Under $66fC$ charge. (c) Under $99fC$ charge. (d) Under $132fC$ charge.
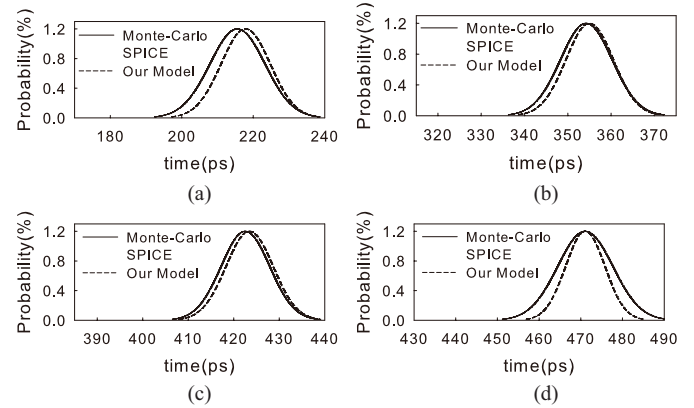

Fig. 12. Model accuracy of OR gates. (a) under $34fC$ charge. (b) under $66fC$ charge. (c) under $99fC$ charge. (d) under $132fC$ charge.
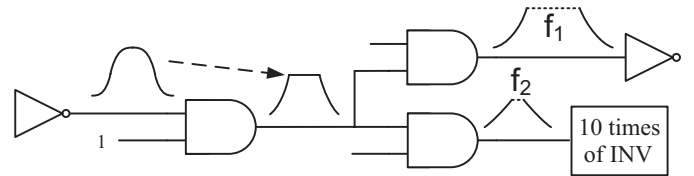

Fig. 13. Explanation for variance errors.

TABLE III
CIRCUIT INFORMATION

Circuit	#$_{\text{gate}}$	#$_{\text{PI}}$	#$_{\text{PO}}$	Lv$_{\text{max}}$
t1	4	1	1	4
t2	6	2	2	3
t3	12	5	2	5
c17	12	5	2	5

particles of different charge strength in the proposed models and those of Monte Carlo SPICE simulation for one AND gate and one OR gate, respectively. The solid line represents the PDF results of the Monte Carlo simulation while the PDF results from our models are denoted by a dotted line. The means by which PDF results are derived using our models are very close to those derived using Monte Carlo SPICE simulation, while the variances of PDF

TABLE IV
SSER MEASUREMENT OF VARIOUS BENCHMARK CIRCUITS

| Circuit	#gate	#PI	#PO	Lv_max	SVR-Learning [4]		Our Method		Speedup (X)
					SSER (μFIT)	Time (sec)	SSER (μFIT)	Time (sec)	
c432	233	36	7	30	5.85×10^3	24.00	2.28×10^5	0.08	300
c499	638	41	32	28	5.77×10^3	164.01	5.97×10^5	0.61	268
c880	433	60	26	33	7.26×10^3	24.05	7.30×10^4	0.11	218
c1355	629	41	33	30	6.19×10^3	164.11	7.26×10^5	0.60	273
c1908	425	33	25	39	9.18×10^3	68.00	2.63×10^5	0.34	200
c2670	872	157	64	38	1.22×10^4	40.12	8.00×10^4	0.17	235
c3540	901	50	22	52	2.58×10^4	180.02	2.98×10^6	0.68	265
c5315	1833	178	123	41	3.51×10^4	208.41	1.76×10^5	0.60	347
c6288	2788	32	32	122	3.74×10^4	3108.52	8.10×10^7	8.92	348
c7552	2171	207	108	60	3.33×10^4	308.31	1.56×10^6	0.64	481
mul_4	158	8	8	23	5.75×10^3	12.00	6.10×10^3	0.04	300
mul_8	728	16	16	50	2.48×10^4	164.93	6.73×10^4	0.56	293
mul_16	3156	32	32	105	7.79×10^4	3208.38	5.11×10^5	13.49	238
mul_24	7234	48	48	155	1.58×10^5	16132.57	1.45×10^6	70.59	228
mul_32	13017	64	64	194	–	–	2.74×10^6	304.37	–
bench2	110539	3975	3935	15	–	–	3.32×10^5	144.93	–
bench3	242347	5705	5661	20	–	–	8.84×10^5	449.12	–
bench4	49858	2429	2409	13	–	–	1.51×10^5	36.00	–
bench7	899618	17871	17823	22	–	–	4.38×10^6	7798.82	–
bench8	105334	4738	4718	19	–	–	3.04×10^5	162.61	–
								Average	286
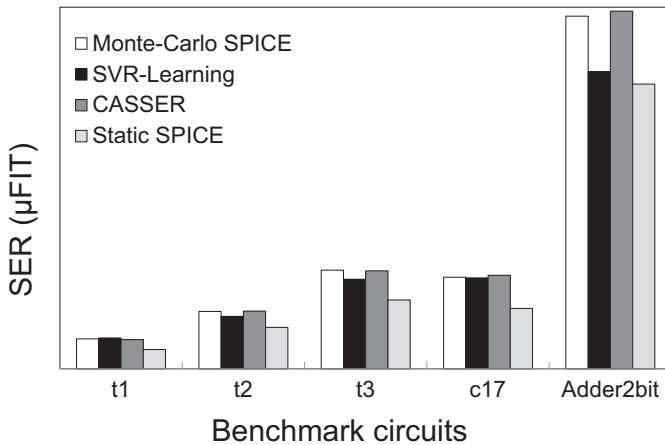
Fig. 14.   SER comparison between Monte Carlo SPICE simulation and CASSER.

results derived by our models are slightly smaller (6.76% on average).

Table II summarizes the accuracy of the first-hit models and propagation models. The first column lists the name of the cell libraries, and the following four columns denote the mean and variance errors of first-hit models and those of the propagation models, respectively. The average mean and variance errors of our first-hit model are all less than 2%, as is the average mean error of the propagation models. Moreover, except for the variance of the propagation model, both proposed models are more accurate than the SVM models

in the latest framework [4], and in particular, the variance of the first-hit model (1.90% versus 12.27%).

The reason that the variance error associated with the propagation models is worse is that the shape of the hitting pulse becomes irregular during propagation. As shown in Fig. 13, because the sinusoidal shape of a hitting pulse is transformed into a trapezoid, the variance of the flat part (like f_1 and f_2) of the trapezoid is hardly considered in the proposed framework, leading to an underestimation of variance.

The following section compares the results of SERs derived using CASSER and those obtained from Monte Carlo SPICE simulation. Furthermore, we compare the results of SSERs with those derived by the SVR-learning framework from [4] in terms of efficiency.

B. Comparison of the SSER

We implemented the proposed framework in C/C++ on a Linux equipped machine with an Intel Core i7 processor and 16G of RAM. The corresponding charge collection slope Q_s was set at 10.84 fC according to [24]. The neutron flux rate was set to at $F = 56.5$ m^{-2} s^{-1} at sea level [25]. For all circuits, each gate under every input pattern combination was injected with electrical charges of four levels: $q_0 = 34 fC$, $q_1 = 66 fC$, $q_2 = 99 fC$, $q_3 = 132 fC$, where q_0 is the weakest charge capable of generating a transient fault under the setting in the experiments. Overall, the SSERs of the circuits are built on ISCAS'85 circuits and a series of multipliers as well as five industrial benchmark circuits from the Industrial Technology Research Institute of Taiwan [26].

During Monte Carlo simulation, the pulse width of the arrival transient faults was measured at all PO/PPO for all input-pattern combinations. Due to the long runtime associated with Monte Carlo SPICE simulation (with 100 runs), we were only able to perform tests on small circuits of up to 26 gates, 31 hitting nodes, and five inputs. The runtime for such Monte Carlo SPICE simulation required more than one day.

Monte Carlo SPICE simulation, the SVR-learning approach, and CASSER were used to evaluate SER accuracy on five benchmark circuits (t1, t2, t3, c17, and Adder$_{2bit}$). Information related to the five benchmarks is listed in Table III. The name of each circuit is shown in column 1, and the following four columns denote the number of gates, the number of primary inputs (PI), the number of primary outputs, and the maximum topological level, respectively. Fig. 14 compares the SER analysis results of these five circuits. Our findings lead to two conclusions. 1) The SVR-learning framework does not typically yield results of satisfactory accuracy for SER compared to those using Monte Carlo SPICE simulation due to a lack of quality models. Moreover, the 16% difference in the result of two-bit adder (Adder$_{2bit}$) is due to reconvergence, which was not considered in that framework. 2) The proposed closed-form SSTA-based framework CASSER yields more accurate SERs with differences of less than 3%, demonstrating that the proposed idea is capable of achieving superior accuracy. The results of Adder$_{2bit}$ were quite accurate, despite the inclusion of many reconvergence fanout nodes, demonstrating the effectiveness of our reconvergence handling strategy. Moreover, Fig. 14 also shows that the SER obtained by Monte Carlo SPICE simulation are 19% $\sim$ 35% above that obtained by static SPICE analysis and proves that we need to consider the process variation for estimating SER again.

Information related to other benchmark circuits and their SSER results as well as runtimes derived using the two methods are listed in Table IV. Columns 1–5 denote the name of each circuit, the number of gates, the number of PI, the number of primary outputs, and the max topological level, respectively. The remaining four columns show more SSER results and runtimes derived by SVR-learning framework [4] and the proposed framework CASSER on a variety of circuits, respectively. The last column computes the improvement in timing cost. The last six test cases were aborted because the runtime exceeded one day. The runtime of each test case using CASSER was less than ten minutes except for bench7 and approximately half of the test cases were completed in one second. In addition, the timing cost grows slowly even if the circuit size grows rapidly, while that of the SVR-learning method increases rapidly as the circuit size increases. The runtime of CASSER was approximately 286 times faster than that of the SVR-learning method. Moreover, because the proposed idea is built upon a closed-form SSTA-like analysis, the longer logic depth will induce a longer runtime. For this reason, c6288 and some multipliers (mul_16 to mul_32) required a slightly longer runtime.

VI. CONCLUSION

Due to process variation beyond the deep submicrometer era, traditional static approaches are no longer effective for analyzing SERs. This is because soft errors originate from particle hits with small charges, which can easily be overlooked in traditional static analysis, resulting in an underestimation of SERs compared to Monte Carlo SPICE simulation. In recent years, numerous SSER frameworks have been proposed; however, simulation-based methods still suffer from extremely large timing costs, even when accurate SSER results were achieved. On the other hand, learning-based methods have been developed to overcome the problems of timing costs while sacrificing the accuracy of SSER.

To consider both efficiency and accuracy simultaneously, this paper proposed a framework named CASSER, which includes a novel idea for SSER analysis, in which a transient pulse was partitioned into two transition signals (one is rising transition and the other is falling transition). Because the two signals were expressed as timing quantities in closed form, they can be analyzed using a block-based SSTA-like method, which considers the correlation of timing. According to experimental results, the runtime of analysis using CASSER is small and SSER differences are within 3%, compared to Monte Carlo SPICE simulation. Moreover, the timing cost of CASSER is about 286 times faster than that of a previous SSER framework [4].

REFERENCES

[1] A. H. Johnston, "Scaling and technology issues for soft error rates," in *Proc. Design Autom. Conf.*, 2000, pp. 530–535.

[2] R. Baumann, "The impact of technology scaling on soft error rate performance and limits to the efficacy of error correction," in *Proc. Int. Electron. Dev. Meeting*, 2002, pp. 329–332.

[3] P. Shivakumar, M. Kistler, S. W. Keckler, D. Burger, and L. Alvisi, "Modeling the effect of technology trends on the soft error rate of combinational logic," in *Proc. Dependable Syst. Netw.*, 2002, pp. 389–398.

[4] 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.

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

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

[7] M. Zhang and N. R. Shanbhag, "Soft-error-rate-analysis (SERA) methodology," *IEEE Trans. Comput.-Aided Design*, vol. 25, no. 10, pp. 2140–2155, Oct. 2006.

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

[9] R. Rajaraman, J. S. Kim, N. Vijaykrishnan, Y. Xie, and M. J. Irwin, "SEAT-LA: A soft error analysis tool for combinational logic," in *Proc. Int. Conf. VLSI Design*, 2006, pp. 499–502.

[10] R. R. Rao, K. Chopra, D. Blaauw, and D. Sylvester, "An efficient static algorithm for computing the soft error rates of combinational circuits," in *Proc. Design Autom. Test Eur. Conf.*, 2006, pp. 164–169.

[11] R. Garg, C. Nagpal, and S.-P. Khatri, "A fast, analytical estimator for the SEU-induced pulse width in combinational designs," in *Proc. Design Autom. Conf.*, 2008, pp. 918–923.

[12] D. Rossi, J. M. Cazeaux, M. Omana, C. Metra, and A. Chatterjee, "Accurate linear model for SET critical charge estimation," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 17, no. 4, pp. 1161–1166, Aug. 2009.

[13] D. Rossi, M. Omana, C. Metra, and A. Chatterjee, "Impact of aging phenomena on soft error susceptibility," in *Proc. Int. Symp. Defect Fault Toler. VLSI Nanotechnol. Syst.*, 2011, pp. 18–24.

[14] 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,

[15] N. Miskov-Zivanov, K.-C. Wu, and D. Marculescu, "Process variability-aware transient fault modeling and analysis," in *Proc. Int. Conf. Comput.-Aided Design*, 2008, pp. 685–690.

[16] 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.

[17] D. Blaauw, K. Chopra, A. Srivastava, and L. Scheffer, "Statistical timing analysis: From basic principles to state of the art," *IEEE Trans. Comput.-Aided Design*, vol. 27, no. 4, pp. 589–607, Apr. 2008.

[18] C.-Y. Chuang and W.-K. Mak, "Accurate closed-form parameterized block-based statistical timing analysis applying skew-normal distribution," in *Proc. Int. Symp. Quality Electron. Design*, 2009, pp. 68–73.

[19] C. Visweswariah, K. Ravindran, K. Kalafala, S. G. Walker, S. Narayan, D. K. Beece, J. Piaget, N. Venkateswaran, and J. G. Hemmett, "First-order incremental block-based statistical timing analysis," in *Proc. Design Autom. Conf.*, 2004, pp. 331–336.

[20] C. E. Clark, "The greatest of finite set of random variables," in *Proc. Oper. Res.*, Mar.–Apr. 1961, pp. 145–162.

[21] M. Cain, "The moment-generating function of the minimum of bivariate normal random variables," *Amer. Stat.*, vol. 48, no. 2, pp. 124–125, May 1994.

[22] L. J. Bain and M. Engelhardt, *Introduction to Probability and Mathematical Statistics*, 2nd ed. Pacific Grove, CA: Duxbury, 2000.

[23] Nangate Inc. (2009). *Nangate 45 nm Open Library*, Sunnyvale, CA [Online]. Available: http://www.nangate.com/

[24] *Parameters of Low Power SoC Design*. (2003) [Online]. Available: http://strj-jeita.elisasp.net/pdf-nenjihoukoku-0303-roadmap/3-13_setsukei_task_force.pdf

[25] P. E. Dodd and L. W. Massengill, "Basic mechanisms and modeling of single-event upset in digital microelectronics," *IEEE Trans. Nucl. Sci.*, vol. 50, no. 3, pp. 583–602, Jun. 2003.

[26] *Industrial Technology Research Institute*. (2011) [Online]. Available: http://www.itri.org.tw/chi/

[27] O. A. Amusan, L. W. Massengill, B. L. Bhuva, S. DasGupta, A. F. Witulski, and J. R. Ahlbin, "Design techniques to reduce SET pulse widths in deep-submicron combinational logic," *IEEE Trans. Nucl. Sci.*, vol. 54, no. 6, pp. 2060–2064, Dec. 2007.

[28] H. Cha and J. H. Patel, "A logic-level model for α particle hits in CMOS circuits," in *Proc. Int. Conf. Circuits Design*, Aug. 1993, pp. 538–542.

[29] Y. Tosaka, H. Hanata, T. Itakura, and S. Satoh, "Simulation technologies for cosmic ray neutron-induced soft errors: Models and simulation systems," *IEEE Trans. Nucl. Sci.*, vol. 46, no. 3, pp. 774–780, Jun. 1999.

[30] M. Omana, G. Papasso, D. Rossi, and C. Metra, "A model for transient fault propagation in combinatorial logic," in *Proc. Int. On-Line Test. Symp.*, Jul. 2003, pp. 111–115.

[31] K. Mohanram, "Closed-form simulation and robustness models for SEU-tolerant design," in *Proc. VLSI Test Symp.*, May 2005, pp. 327–333.

[32] D. T. Franco, M. C. Vasconcelos, L. Naviner, and J.-F. Naviner, "Signal probability for reliability evaluation of logic circuits," in *Proc. Eur. Symp. Rel. Electron. Devices, Failure Phys. Anal.*, vol. 48, nos. 8–9, pp. 1586–1591, Aug.–Sep. 2008.

[33] S. Mukherjee, M. Kontz, and S. Reihardt, "Detailed design and evaluation of redundant multi-threading alternatives," in *Proc. Int. Symp. Comput. Arch.*, May 2002, pp. 99–110.

[34] W. Bartlett and L. Spainhower, "Commercial fault tolerance: A tale of two systems," *IEEE Trans. Depend. Secure Comput.*, vol. 1, no. 1, pp. 87–96, Jan.–Mar. 2004.

[35] S. Mitra, N. Seifert, M. Zhang, Q. Shi, and K. S. Kim, "Robust system design with built-in soft error resilience," *IEEE Trans. Comput.*, vol. 38, no. 2, pp. 43–52, Feb. 2005.

[36] M. Zhang, T. M. Mak, J. Tschanz, K. S. Kim, N. Seifert, and D. Lu, "Design for resilience to soft errors and variations," in *Proc. Int. On-Line Test Symp.*, Jul. 2007, pp. 23–28.

[37] K. A. Bowman, S. G. Duvall, and J. D. Meindl, "Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration," *IEEE J. Solid-State Circuits*, vol. 37, no. 2, pp. 183–190, Feb. 2002.

[38] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De, "Parameter variations and impact on circuits and microarchitecture," in *Proc. Design Autom. Conf.*, Jul. 2003, pp. 338–342.

[39] S. Natarajan, M. A. Breuer, and S. K. Gupta, "Process variations and their impact on circuit operation," in *Proc. Int. Symp. Defect Fault Toler. VLSI Syst.*, Nov. 1998, pp. 73–81.

[40] H. Edamatsu, K. Homma, M. Kakimoto, Y. Koike, and K. Tabuchi, "Pre-layout delay calculation specification for CMOS ASIC libraries," in *Proc. Asian South Pacific Design Autom. Conf.*, Jan. 1998, pp. 241–248.

[41] F. Brglez and H. Fujiwara, "A neural netlist of ten combinational benchmark circuits and translator in Fortran," in *Proc. Int. Symp. Circuits Syst.*, 1985, pp. 1–8.

[42] P. Hazucha and C. Svensson, "Impact of CMOS technology scaling on the atmospheric neutron soft error rate," *IEEE Trans. Nucl. Sci.*, vol. 47, no. 6, pp. 2586–2594, Dec. 2000.

Austin C.-C. Chang received the M.S. degree in communication engineering from National Chiao Tung University, Hsinchu, Taiwan, in 2011.

He is currently an Engineer with MediaTek. His current research interests include design for reliability, Iddq testing, and scan-chain reordering for 3DIC.

Ryan H.-M. Huang received the M.S. degree in communication engineering from National Chiao Tung University, Hsinchu, Taiwan, in 2011, where he is currently pursuing the Ph.D. degree in electrical and computer engineering.

His current research interests include design for reliability, and automatic test pattern generation in computer aided design of electronic circuits and systems.

Charles H.-P. Wen (M'07) received the Ph.D. degree in VLSI verification and test from the University of California, Santa Barbara, 2007.

He is an Assistant Professor with National Chiao Tung University, Hsinchu, Taiwan, and is a specialist in computer engineering. Over the past few years, his work has been focused on applying data mining and machine learning techniques to SoC design (especially on statistical soft error rates and circuit diagnosability in nanometer technologies) and cloud computing (especially on performance analysis and architecture design of large-scale data centers).