An Efficient Static Algorithm for Computing the Soft Error Rates of Combinational Circuits

Rajeev R. Rao, Kaviraj Chopra, David Blaauw, Dennis Sylvester
Department of EECS, University of Michigan, Ann Arbor, MI 48109
{rrrao, kaviraj, blaauw, dennis}@eecs.umich.edu

Abstract

Soft errors have emerged as an important reliability challenge for nanoscale VLSI designs. In this paper, we present a fast and efficient soft error rate (SER) computation algorithm for combinational circuits. We first present a novel parametric waveform model based on the Weibull function to represent particle strikes at individual nodes in the circuit. We then describe the construction of the SET descriptor that efficiently captures the correlation between the transient waveforms and their associated rate distribution functions. The proposed algorithm consists of operations to inject, propagate and merge SET descriptors while traversing forward along the gates in a circuit. The parameterized waveforms enable an efficient static approach to calculate the SER of a circuit. We exercise the proposed approach on a wide variety of combinational circuits and observe that our algorithm has linear runtime with the size of the circuit. The runtimes for soft error estimation were observed to be in the order of about one second, compared to several minutes or even hours for previously proposed methods.

1 Introduction

A radiation particle passing through a strong electric field region in a semiconductor device generates a large number of free electron-hole pairs. If such an event occurs near the depletion region of a reverse biased p-n junction, the free mobile carriers are efficiently collected by the high electric field present across the p-n junction. Subsequently, a transient noise pulse is generated due to the current flowing through this junction. This single event transient (SET), if registered by a latch, can cause a functional/data error resulting in a single event upset (SEU). Faults occurring due to such transient upsets are referred to as soft errors, as no permanent damage is done to the device, and the rate at which they occur is referred to as the soft error rate. A quantitative metric called failures-in-time (FIT) denotes the number of errors that can occur in one billion device hours is used to provide a calculable estimate for the error rates in industrial logic blocks. In a typical IC, memory arrays, latch elements and combinational logic are all susceptible to soft errors.

The continued trend in technology scaling has resulted in soft errors becoming an increasing concern for digital circuits in the nanometer regime. Reduced feature sizes, higher logic densities, shrinking node capacitances, lower operating voltages and shorter pipeline depths have resulted in a significant increase in the sensitivity of integrated circuits to radiation induced single event upsets. A number of studies have been presented examining the impact of technology scaling on the soft error rates of CMOS circuits [1][2][3]. Although memory arrays represent a large portion of the chip area that is vulnerable to soft error strikes, a continued reduction in both the critical charge and collection efficiency has resulted in SRAM SER staying constant over several technology generations. In addition, the usage of ECC enables a high level of soft error protection for memory structures. Similarly, industrial estimates show that the nominal soft error rate of latches is nearly constant for the 130nm to 65nm technologies [4]. The development of radiation hardened latches [5] with minimal overheads has further lessened the possibility of soft errors occurring in latches. Consequently, for current and future technologies, the impact of soft errors on combinational elements is receiving significant attention. It has been predicted that at the 45nm technology node, a majority of the observed soft failures will be related to SET events that occur in logic blocks [2][6]. Hence, it is critically important to develop effective techniques to analyze and quantify the impact of soft errors on combinational logic circuits.

In this work, we develop an efficient analysis methodology to compute the soft error rate of combinational logic blocks that are susceptible to single event upsets due to high energy neutron strikes. We first utilize a transient current model to describe the gate level effects of a single particle strike on a diffusion region in the circuit [7]. The resultant voltage glitches are modeled using the Weibull probability density function that provides for accurate waveform representations. The amount of charge collected due to neutron strikes varies over a wide range of values; the rate distribution corresponding to the set of strikes in this range is modeled using the analytical expressions presented in [8][9]. We then describe the construction of the novel SET descriptor that integrates the transient waveform shapes with the corresponding SET rate distribution into a single object. The effect of particle strikes on a node is represented in an individual SET descriptor consisting of two simple functions: a waveform shape function described by a pair of linear parameters and a rate function described by a discrete set of error rate numbers. The proposed algorithm proceeds in a bottom-up fashion by injecting SET descriptors at each node and then propagating them along sensitizable paths in the circuit. We employ a merging operation to identify independent strike events with the same waveform shape function and combine the rate functions of the set of these SET descriptors into a single consolidated SET descriptor. The number of waveform shapes corresponding to injected strike events is enormous; however, after propagating through at most 3-4 gates, they converge into a small subset of waveform shapes. The merging operation efficiently recognizes such instances, thus minimizing the number of distinct waveforms to be propagated along the circuit. Similar to standard static timing analysis (STA), our algorithm requires a single pass through the circuit graph in topological order and hence the complexity is linear in the size of the circuit. Our algorithm shows that such an approach based on parameterized descriptors provides accurate and scalable soft error analysis for a variety of combinational circuits.

The remainder of this paper is organized as follows. In Section 2 we provide an overview of previous work in this area. In Section 3 we present the analytical function for modeling transient pulses and rate distributions and also describe the construction of the SET descriptor. We then detail the methods by which we propagate and merge the different SET descriptors in Section 4. In Section 5 we provide algorithm runtimes and present comparison of our method with SPICE simulations. Finally, we conclude in Section 6.

2 Prior Work

Previous approaches to soft error estimation can be broadly classified into two types: (i) System level approaches and (ii) Circuit level methods. System level estimation methods seek to compute the probability that a soft error at the gate level is manifested at the system level to the end user. The primary objective of these methods is to identify possible cases of undetected errors that could cause the so-called silent data corruption (SDC). A detailed overview of an industrial system level SER estimation method is presented in [10].

3-9810801-0-6/DATE06 © 2006 EDAA

A number of methods have also been proposed in the literature to estimate circuit level SER. The authors in [11] present a Monte Carlo based modeling program SEMM. A methodology based on the single event effect state transition model was developed in [12] to quantify the effect of SEUs on complex digital devices. Several methods proposed in literature are based on models for transient fault injection and propagation [13][14][15]. In these methods, the authors rely on simple device level equations to predict the appearance of the transient pulse at the primary output. Other approaches such as [16][17] analyze the impact of different masking mechanisms on the SER of combinational circuits.

Recently, two new approaches to logic level SER estimation have been proposed. The SERA methodology presented in [9] combines various aspects of probability theory, circuit simulation and fault simulation. The authors develop a path-based approach to introduce pulses at individual nodes in the circuit and propagate them to the primary outputs or the latches using probability models. The algorithms presented in [18][19] encode particle strikes and fault events at nodes using decision diagrams (BDDs/ADDs). The authors then use standard algorithmic methods to propagate these decision diagrams through the circuit. While these proposed methods are attractive, they are intractable for large combinational circuits. The presence of reconvergence increases the number of paths in a circuit exponentially, thereby limiting the applicability of a path based algorithm. Similarly, despite the prevalence of circuit partitioning techniques, BDD-based algorithms are inherently limited due to the memory blowup problems associated with them. Furthermore, these methods use overly simplified SPICE models (such as equivalent inverter chains and square pulses) while characterizing the cell library. As we describe in Section 3, it is important that the transient waveforms are characterized systematically to accurately capture the effects of various types of masking during the fault propagation operation.

In this paper, we present a static, block-based, linear-time algorithm to estimate the soft error rates of arbitrarily large combinational logic circuits. In contrast to previous approaches that use single parameters (such as pulse width or height) to describe the transient waveform, we present a Weibull function based model that provides for several degrees of freedom and allows a highly accurate fit for various types of transient pulses. Secondly, we present a unified model called the SET descriptor (explained in Section 3.2) that efficiently represents all SET strikes and their rate of occurrence at a victim node. The third key contribution of this paper is the observation that the shape of transient pulses originating at different victim nodes converges after a small number of propagations. This effect enables us to perform an efficient merging operation that allows for linear runtime analysis.

3 SER Analysis Model

In this section, we present an outline of the analytical models that are used in our algorithm. We first present the current model for a single particle strike. Using this model, we then present a novel parametric function that captures the effects of the entire range of particle strikes for one instance a library cell. Finally, we describe the methodology for cell library characterization for SER analysis.

3.1 Unitary SET Model

A particle hit at a sensitive region in a circuit injects a small amount of charge at the victim node. For a given injected charge Q_0, the subsequent transient current through the reverse biased p-n junction is modeled using the current pulse model presented in [7].

$$I(t) = \pm\frac{2Q_0}{\tau\sqrt{\pi}}\sqrt{\frac{t}{\tau}}\exp\left(\frac{-t}{\tau}\right) \qquad \text{(EQ 1)}$$

Here τ is a technology-dependent pulse-shaping parameter. The polarity of the current source is determined by the type of drain node (N/P-type) collecting the charge. This single parameter, time-dependent current waveform provides a simple yet accurate representation

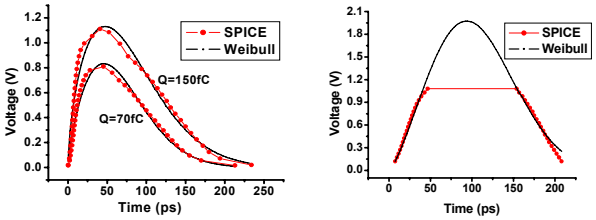

Figure 1. Comparison of SPICE and Weibull waveforms for (a) pulses with height < V_{dd} and (b) trapezoidal pulses.

of the current waveform due to an SET. By using this current pulse in conjunction with a library gate, we obtain the output voltage response of the cell to a given amount of injected charge.

Previously, soft error transients have been characterized using pulse width as a single parameter [18] or with simple trapezoidal shapes [14]. Such methods are inherently problematic since they do not accurately capture the range of non-linear waveform shapes that can be generated due to a particle strike. The characteristics of the first strike waveform play an important role in determining the impact of electrical masking for propagation through the initial stages of logic. It is indeed accurate to characterize a transient pulse after a few propagations as a trapezoidal waveform. However, at the drain node at which the cosmic particle strikes, the generated voltage waveform is, in general, vastly different from a standard trapezoidal shape.

To capture the large variety of transient waveforms that are possible in a circuit, we propose an empirical model based on the Weibull probability density function [20]. The three parameter Weibull function that we use in our analysis is as follows:

$$V(t) = c\left(\frac{b(a-1)}{a}\right)^{\left(\frac{1-a}{a}\right)}\exp\left(\frac{a-1}{a}\right)t^{(a-1)}\exp\left(\frac{-t^a}{b}\right) \qquad \text{(EQ 2)}$$

Here a is the shape parameter, b is the time-scale parameter and c is the normalization parameter. In contrast to the general form of the Weibull function, EQ2 has been normalized such that in this modified equation, the parameter c exactly corresponds to the height of the waveform. Further, by examining EQ2 in the context of voltage pulses, we observe that the shape parameter a indicates the general nature of the waveform (such as fast rise/slow decay) while the time-scale parameter b is representative of the width of the waveform. As an example, in Figure 1 (a), we plot the comparison between a transient waveform and the corresponding Weibull empirical model for an FO4 inverter injected with 70fC and 150fC of charge and observe that the Weibull provides a good fit.

As a pulse propagates through a circuit, assuming it is not attenuated due to various masking mechanisms, it will attain the shape of a trapezoidal waveform. We observe that the Weibull function can be modified slightly to model these pulse shapes as well. This modification is illustrated in Figure 1 (b). We empirically limit the Weibull function to only match the rising and falling edges of the voltage waveform in the range of voltage amplitude between 0 and V_{dd}. The values of the Weibull function that are greater than V_{dd} are irrelevant to our analysis. Although it is theoretically possible that a cosmic particle with sufficiently high energy will override the inherent capacitive clamping in a device and produce pulses with large overshoots (or undershoots), we observed that, according to the given current pulse model, such cases are non-existent.

3.2 SET Distribution Model

In the previous sub-section, we described the voltage waveform model for a single particle strike. For accurate SER analysis it is necessary to consider the cumulative effect of the entire spectrum of neutron strikes. The range for the injected charge due to neutron strikes can be determined to be [10fC, 150fC] for the 0.13µm tech-

nology [21]. SET strike events causing smaller charge collection occur much more frequently compared to strikes causing large charge collection. Using the empirical model presented in [8], the authors in [9] developed an analytical expression to describe the associated SET rate distribution.

$$R = F \times K \times A \times \left(\frac{1}{Q_s}\right) \times \exp\left(\frac{-Q_0}{Q_s}\right) \quad \text{(EQ 3)}$$

Here R = rate of SET strikes, F = neutron flux with energy>10MeV, A = area of the circuit susceptible to neutron strikes (in cm^2), K = a technology-independent fitting parameter, Q_0 = charge generated by the particle strike and Q_s = charge collection slope. We adopt this simple charge based model to correlate electric charge injected by a particle strike with the rate of SET occurrence. The collection slope Q_s is a measure of the magnitude of charge generated due to a neutron strike. In [8] the authors observe that since Q_s(NMOS) > Q_s(PMOS), the soft error rate due to strikes on NMOS drains is significantly greater (by about two orders of magnitude) than strikes on PMOS drains. Note that the proposed algorithm is independent of the empirical model used for the rate function. Since we utilize discrete vectors to describe the rate values, the analytical function used to generate the rate numbers does not influence the performance of the algorithm.

We use the aforementioned three parameter Weibull function to represent the individual voltage waveforms. For a given cell, as we sweep over the range [10fC, 150fC] of charge values, we observe that a wide set of voltage waveforms are generated. Each waveform in this set can be accorded a unique 3-tuple of Weibull parameters. In order to efficiently identify the entire set of waveforms corresponding to a range of energy levels, we seek to develop a functional relationship among the 3 parameters. In other words, capturing the relationship among the Weibull parameters using a simple polynomial equation is more efficient compared to identifying each waveform separately in a discrete manner.

If we assume that the three parameters are entirely unconstrained, then it is possible to determine the values of these constants using standard curve fitting techniques. However, such an unconstrained fitting mechanism will result in a one-to-many (aliasing) relationship between the transient waveforms and the Weibull 3-tuples such that one waveform can be represented with nearly equal error by two very different parameter 3-tuples. The existence of a unique, one-to-one representation between the Weibull parameters and voltage waveforms is an important requirement for the merging operation (described later in Section 4.2.2) in order to ensure the optimality of the algorithm. Hence, we place constraints on the values of these parameters such that we obtain unique representations.

First, we categorize the waveforms into three types: (Cat1) First strikes - These correspond to a direct strike at any particular node in the circuit. (Cat2) First propagation - These correspond to the case when a pulse due to a first strike has propagated through exactly one gate. (Cat3) Subsequent propagations - These correspond to the cases when the pulse has propagated through two or more gates. We also distinguish between rising and falling transitions in each case, resulting in six total categories. Cat2 is necessary to capture the crossover phase when the non-linear Cat1 transient waveform is transformed to the standard Cat3 trapezoidal shape. Using circuit simulations we have determined that these three categories capture the entire family of waveforms possible in the circuit.

Next, we determine the value of the shape parameter a in EQ2 for the three categories separately and fix it as a constant for the entire library. Intuitively, we see that this formulation is valid since the shape of different transient waveforms in a single category is constant and the waveforms vary only in their height and width. As mentioned previously, the parameter c is exactly equal to the height of the waveform. From this discussion it is clear that a and c can be uniquely determined from the characteristics of the waveform (distance from the strike and waveform height). Once these values are available, we use simple empirical fitting on EQ2 to calculate the value of b. Thus, in our analysis, we choose parameter b as the free variable.

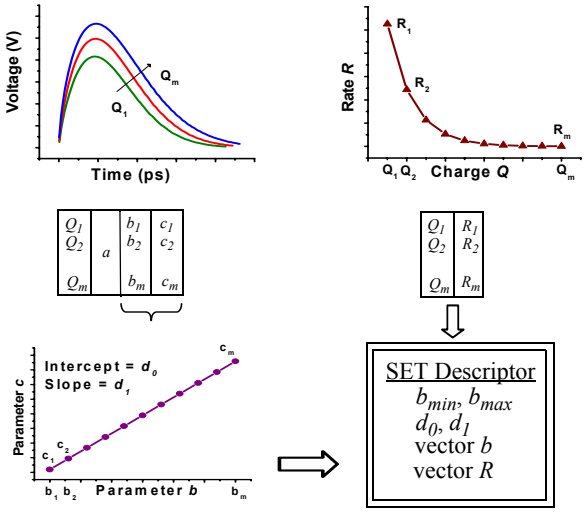

Figure 2. Construction of an

For the range of charge values, an entire family of voltage waveforms is generated. Using the procedure described here, we determine the values of the three Weibull parameters for each waveform. With b as the free variable, we observe that this set of (b,c) 2-tuples can be described in parametric form by considering a straight line in the bc plane.

$$c = d_0 + d_1 b \quad \text{(EQ 4)}$$

Here (d_0,d_1) are the intercept and slope parameters for this linear equation. This straight line represents the entire set of charge values that can produce a transient pulse at a particular node. Additionally, we also identify the minimum (b_{min}) and maximum (b_{max}) values of b corresponding to that bc line. For $b<b_{min}$ no transient pulse is generated at the node (electrical masking) and $b>b_{max}$ can be neglected since the probabilities associated with charge values above 150fC are negligible.

The construction of an individual SET descriptor is illustrated in Figure 2. The charge values are first discretized so that $Q_i \in \{Q_1, Q_2, ..., Q_m\}$. We first determine the rate value R_i corresponding to each Q_i using EQ3. We then generate voltage pulses corresponding to each Q_i and empirically calculate the Weibull 3-tuples. While a is fixed, b_i and c_i vary across the range of Q_i. After the set of b_i,c_i values are generated, we fit a linear equation to determine the pair of parameters (d_0,d_1) that identify this particular set of transient waveforms. It is evident that a one-to-one relationship exists between Q_i and R_i as well as between Q_i and b_i. Since we choose b as the free variable in our analysis, we store the strike rate information discretely in a pair of vectors. Thus an individual SET descriptor consists of b_{min}/b_{max}, the d_0,d_1 parameters that denote the waveform shape function and the b,R vectors that describe the strike rate values corresponding to each transient pulse. Note that the SET descriptor efficiently captures the effects of an entire set of waveforms. This is in contrast to previously proposed SER methodologies [9][15] that analyze each strike event individually and hence incur larger computational overheads.

3.3 Cell Library Characterization

While characterizing a cell library for SER analysis, we quickly recognize that an enormous number of transient waveforms are possible. The factors that influence the character of the transient wave-

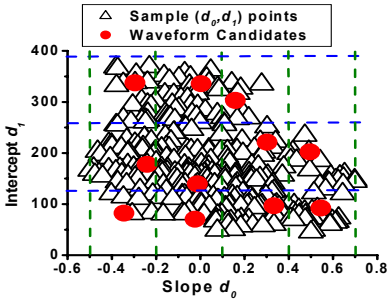

Figure 3. Selection of waveform candidates

forms are as follows - cell type, cell size, input state and output load. For every possible permutation among these factors, a pair of parameters d_0 and d_1 represents the resultant transient wave. When the set of all such (d_0,d_1) pairs is examined in the 2D plane (Figure 3) we see that a regular pattern exists such that clusters of points in this plane represent nearly identical transient waves. Consequently, to reduce the sample space of possible transient waveforms, we discretize this 2D plane by creating artificial grids. The grid sizes on the horizontal and vertical axes are carefully chosen such that all transients within a single grid exhibit near identical behavior when propagated through any cell in the library. We choose a representative waveform candidate for each grid that symbolizes the type of transient for that range of (d_0, d_1) values. In this manner, we see that the large sample space of all possible transients can be efficiently represented by a small set of representative candidates. In our analysis, we observed that typically the number of such candidates (equal to the number of grids) is around 8-12.

The candidate waveform set is generated in an incremental fashion for the three categories of waves described previously in Section 3.2. To characterize the Cat2 waves, we simulate the gates with input transients chosen only from the waveform candidates generated for the Cat1 waves. Since the Cat1 waveform candidates efficiently capture all possible types of first strike waves, it is sufficient to characterize for Cat2 (first propagation) waves using input transients from only this subset. Similarly, Cat3 waves are characterized using Cat2 waveform candidates as the input transients. In our analysis we observed that a total of about 40-45 waveforms (separate for rising/falling transitions) accurately encompass all possible SETs that can occur from the cells in the given gate library.

4 Our Algorithm

In this section we first mention the three different types of masking mechanisms that potentially eliminate any possible errors due to SEUs. We then describe the proposed algorithm including the procedures for the propagation and merging of waveforms. Finally, we examine the complexity of the proposed algorithm.

4.1 Masking Mechanisms

An SET generated at a victim node in a combinational circuit causes a soft error only if it propagates through the subsequent logic and is observable at an external output, or if it is latched into a memory element of the circuit. There exist three well-known masking mechanisms that prevent an SET in combinational logic from causing a soft error [6].

1. Temporal Masking – An SET pulse arriving at a memory element is temporally masked if the glitch on the data input node is outside the latching window of the memory register [17].

2. Logical Masking – The propagation of an SET is logically masked if there is no sensitizable path from the hazard to the primary output.

3. Electrical Masking – An SET can be electrically masked based on the characteristics of the driving cell and the input transient. For instance, given a weak driver with a large output load, it is possible that a majority of the SETs are attenuated when they propagate through such a gate.

Although these masking mechanisms serve as derating factors in reducing the probability of the occurrence of soft errors, it has been observed that their impact is lessening across technologies. Deeper processor pipelines have allowed higher clock rates which reduce temporal masking. As transistors are continually scaled to smaller feature sizes, the pulse attenuation effect is also decreased significantly so that the possibility of electrical masking is reduced [2].

For accurate SER estimation, it is crucial to account for all three masking mechanisms in the algorithmic framework. Both temporal and electrical masking mechanisms are strong functions of the SET pulse shape which, in turn, is a function of the neutron strike characteristics. A brute-force analysis would simulate a large number of neutron strikes for each gate in the circuit and propagate the subsequent SET pulse through all possible paths in the circuit. However, such a method is computationally intractable for even medium-sized circuits. Our algorithm inherently accounts for these masking mechanisms by utilizing the efficient representation of SETs using waveform shape and rate distribution functions.

4.2 Structure of the algorithm

The general structure of our algorithm is similar to standard STA. In STA, arrival times are propagated forward along the nodes using a single topological pass through the entire circuit. The propagation characteristics for this parameter are dependent on various factors such as cell type, output load and input slew. In our method we instead propagate SET descriptors along the nodes in the circuit. The algorithm traverses the circuit graph in topological order while performing the following two operations at each node n: (a) Propagation of each fan-in SET descriptor from input to output of n and (b) Merging of propagated waves and rate function at n to compute the total SET strike rate distribution at node n. This total SET strike rate at any node represents the contribution to the strike rate of all the nodes in its fan-in cone.

4.2.1 Propagation

The algorithm proceeds in a bottom-up fashion (using depth first search) by first injecting SETs at the input gates and proceeding along the nodes in a circuit towards the output. Depending on the input states of the gates, only a fraction of the entire set of gates is susceptible to soft error strikes. Initially, for a first strike Cat1 waveform, an SET descriptor is affixed at the injection node. We then propagate this SET descriptor forward through a sensitizable path in the circuit. Depending on the characteristics of the path (gate sizes, output load) we use a lookup table based transfer function to transform an input SET descriptor to an output SET descriptor of appropriate type. When a waveform is propagated through a gate, the waveform shape (ie, d_0,d_1 and vector b) parameters are transformed based on the pre-characterized table while the rate values (vector R) are propagated as is. Note that while performing the transfer function across a gate, we first determine all possible SET descriptors for its inputs and then transfer these SET descriptors using our pre-characterized library of waveform candidates. This is similar to the method used in STA where arrival times are first generated for all gate inputs and a subsequent sum/max operation determines the arrival time at the gate output.

As transient waveforms propagate through the nodes in a circuit, we observe that the resultant waves, after a few propagations, are nearly identical irrespective of the nature of the original first strike wave. This observation is based on the fact that most CMOS gates exhibit a unity transfer function after a few propagations. In other words, even if a large range of waveforms of type Cat1 are injected at various points in the fan-in cone of a particular node, after a small number of propagations along the paths that reach that node, the number of type Cat3 waves will be small. As a result, it becomes increasingly important to identify such cases of waveform equiva-

lence during the propagation operation. For instance, it is possible that for a gate with two inputs, although a large number of SET descriptors are generated for each input, the resultant set of SET descriptors at the output node will contain a large set of waves that can possibly be merged. This is in direct contrast to a path-based analysis method which treats each possible SET event as an independent instance and thus incurs a large penalty in cases where there are an exponential number of possible paths. The ability to identify such instances of waveform equivalence and then efficiently compact the large set of identical waves into a single output wave constitutes a key aspect of our proposed algorithm.

4.2.2 Merging

The usage of the (d_0,d_1) parameters for waveform identification enables us to quickly identify equivalent waves. At the end of the propagation operation at each gate output, we iterate through the resulting set of waves and compact SET descriptors with identical (d_0,d_1) parameters into a single output SET descriptor. Note that while the vector of b values in the SET descriptor will be identical, the vector of R values may be different because of the difference in the originating SET descriptor. We set the vector of R values for the output to be the vector sum of R values of the original pair of SET descriptors. In this manner, we see that for each node in the circuit, the set of all SET descriptors at that node is the combined effect of considering particle strikes at all possible nodes in the fan-in cone of that node.

As an example for the merging operation, consider a circuit that consists of a chain of ten identical minimum sized inverters [INV_1, INV_{10}]. The output of INV_{10} is connected to a capacitive load equivalent to a single inverter. Naturally, ten possible strike locations (at the drain nodes of all inverters) exist. A path-based algorithm would treat each possible strike independently and predict that ten possible waveforms can be generated at the output of INV_{10}. However, using our waveform compaction technique in conjunction with our gate library, we recognize that only four different types of waveforms are possible at the output of INV_{10}. Using the proposed algorithm we observe that transient waveforms injected at the outputs of INV_1-INV_7 manifest themselves as nearly identical waves at the output of INV_{10} since they converge to a single descriptor after four propagations. Combining this with three other waves generated at INV_8, INV_9 and INV_{10} we get only four different classes of waveforms at the output of the inverter chain.

4.2.3 Temporal Analysis

At the end of the bottom-up pass through the circuit, a final set of SET descriptors is generated at each output node. Each output node of the circuit is connected to a standard D-Flipflop. For a given input transient k, the temporal probability $z(k)$ of a flipflop is defined as the probability that this transient causes a faulty bit to be latched into the memory element. Using SPICE measurements we determined the value of $z(k)$ corresponding to each pulse in the set of waveform candidates belonging to the three categories.

In the final set of descriptors, each discrete (b,R) point corresponds to a single transient wave. For a given pulse in an individual descriptor, we then use the (d_0,d_1) parameters of the descriptor as indices in a pre-characterized lookup table to determine the exact value of $z(k)$ corresponding to that pulse. It is important to recognize that a one-to-one monotonic relationship exists between parameter b and the injected charge Q that generated this waveform so that $b_{min} \leftrightarrow Q_{min}$ and $b_{max} \leftrightarrow Q_{max}$. We denote the scaled strike probability $R_{sc}(b)$ value as $R_{sc}(b) = z(k)R(b)$. By repeating this computation for each pulse in the descriptor we convert the (b,R) vector into an equivalent (Q,R_{sc}) vector. The error rate value due to all pulses in a given descriptor d is then given by $SER(d) = \int_{Q_{min}}^{\infty} R_{sc}(Q)dQ$. Since we use discrete vectors to describe R_{sc}, we perform numerical integration to calculate $SER(d)$.The total circuit SER is the aggregate of the SER due to each individual descriptor at each output node in the circuit.

$$SER(total) = \sum_{\forall output} \sum_{\forall descriptor} SER(d) \qquad \text{(EQ 5)}$$

Note that since we disregard the effects of reconvergent paths in our analysis, this value of $SER(total)$ represents an effective upper bound on the actual SER value of the circuit. However, it has been observed that the presence of reconvergence does not influence the behavior of transient waveforms significantly [9].

4.3 Input Vector Dependence

The SER dependence on the circuit input vector state can be accounted for in two ways: (1) Compute the circuit SER over a large set of typical vectors. For each vector, the logic values are first propagated through the circuit and the SET descriptors corresponding to each input state is propagated only for the logically unmasked nodes. (2) Compute SER by first computing static state probabilities using a method such as [22]. For each gate, the rate vectors in the SET descriptors are then weighted by the state probabilities and conditional propagation probabilities during the propagate/merge operations. Although the second approach captures the entire input space, it is inherently difficult to accurately account for logic correlations due to reconvergence. Therefore, for the sake of simplicity, we implemented the first method and calculated average SER numbers.

4.4 Complexity Analysis

The algorithm proceeds as a single DFS topological pass through the circuit. Beginning at the inputs, the algorithm builds up the SET descriptors at all the nodes as it traverses up the circuit. The merging operation is essential in identifying equivalent waveforms thereby drastically reducing the number of propagated waves in the subsequent logic stages. For a given input vector, since a single pass through the circuit is sufficient to determine the SET descriptors at all possible outputs, the complexity of the algorithm is O(#Gates * #Waveform_Candidates). As mentioned previously, the number of waveform candidates is typically a small number (about 40-45). In Section 5 we present a plot to show that the average runtime of our algorithm over several input vectors is indeed linear in circuit size.

5 Results

We implemented the proposed algorithm using C++. We exercised our algorithm on a Pentium 4 machine with a 2.4GHz processor and 1GB RAM running Linux. We used a standard industrial 0.13µm cell library for circuit synthesis. In EQ3, we set the flux value F as 56.5 $m^{-2}s^{-1}$ corresponding to the rate of neutron flux at sea-level [8]. We characterized the gates and generated the candidate waveforms as described in Section 3.3. Note that this characterization process is a one-time effort that needs to be performed only once for a given library.

We first present the error rate and runtime results associated with our algorithm. In Table 1 we list the runtime and soft error rate values (in units of number of FIT) obtained by running our algorithm on various benchmarks such as the MCNC suite [23], the ISCAS-85 benchmarks [24] and standard multiplier circuits. For each benchmark, we applied 1000 random input vectors and extracted the average SER value from those runs. The runtimes reported here correspond to the SER computation for a single input vector. From this table, we see that for circuits with less than 250 gates the runtime is less than 0.01s and the largest circuit has runtime of 0.20s. In comparison, the authors in [9] report a runtime of about 40 minutes for the 16-bit multiplier and 10 hours for the 32-bit multiplier. In Figure 4 we plot the circuit size and the runtime of our algorithm and observe the linear complexity of the proposed approach.

Unlike circuit parameters such as area and power dissipation that are mainly a function of the number of gates, the number of outputs

Table 1. List of circuits, SER values and runtimes

Circuit	Gates	Outputs	SER (# FIT)	Rt (s)
i1	59	13	1.79E-05	<0.01
i2	222	1	1.40E-06	<0.01
i3	132	6	7.08E-06	<0.01
i4	236	6	2.06E-05	<0.01
i5	204	66	1.16E-04	<0.01
i6	735	67	3.13E-04	0.01
i7	937	67	3.32E-04	0.01
i8	1609	81	3.69E-04	0.02
i9	1018	63	2.69E-04	0.01
i10	3379	224	4.53E-04	0.04
c432	246	7	1.75E-05	<0.01
c499	750	32	6.26E-05	0.01
c880	591	26	6.07E-05	0.01
c1355	748	32	7.98E-05	0.01
c1908	760	25	7.50E-05	0.01
c3540	1951	22	9.03E-05	0.03
c6288	4836	32	4.06E-04	0.06
mul4x4	241	8	2.71E-05	<0.01
mul8x8	1320	16	7.29E-05	0.02
mul16x16	6175	32	1.54E-04	0.07
mul24x24	15678	48	1.79E-04	0.12
mul32x32	25618	64	4.60E-04	0.20

plays a significant role in determining the error rate of a circuit. Naturally, a larger number of outputs will increase the observability of possible transient pulses. We found that a small circuit with a large number of outputs can have a higher SER compared to a large circuit with a small number of outputs. This can be seen by comparing circuit *i6* with circuit *c3540* in Table 1. Although the circuits differ by 2.6X in circuit size, since the smaller circuit has 3X more outputs, the circuit SER of *i6* is about 3.5X larger than the SER of *c3540*.

To verify the accuracy of the proposed method, we compared the SER results from our algorithm to those obtained from SPICE simulations. To perform a full comparison with SPICE for a given circuit, we needed to first pick a sample input vector and simulate strikes node-by-node for that circuit. For about 30 discrete values for the injected charge in the range [10fC,150fC] this would involve about 30*#Gates number of simulation per circuit per input vector. Since the time required for such a simulation was prohibitively large, we used a smaller subset of the benchmark suite for SPICE comparison. In Table 2 we list the SER values of our algorithm against SPICE for a fixed input vector. We see that the error is usually within 20% with average error of 16.1%. Note that computation error on the order of 20% is relatively insignificant since the SER value typically varies by several orders of magnitude across different circuits on the chip.

6 Conclusions

In this paper we presented a static soft error rate computation algorithm for logic circuits. We first developed parametric representations for the transient pulses that affect the susceptible nodes in a circuit. We then developed the SET descriptor object to correlate waveform shapes with error rate distributions. The proposed algorithm employs an efficient merging mechanism to limit the number of waveforms thereby ensuring linear runtime. Experimental evaluations show that our algorithm is linear in the number of nodes in the circuit and that our approach is within 16.1% of SPICE simulations on average. In addition to predicting the presence of a transient pulse at the output, we also produce a soft error rate number (in terms of the number of FIT) for any given circuit. Such a rate number is useful for the system level designer to budget for extra resources for radiation hardening.

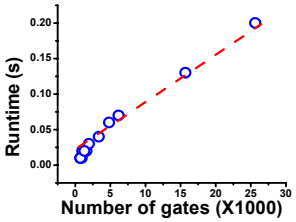

Figure 4. Runtime vs. Circuit size

Table 2. Comparison of proposed algorithm with SPICE

Circuit	Algorithm SER (# FIT)	SPICE SER (# FIT)	% Error
i1	1.99E-05	2.31E-05	16.1
i2	1.13E-06	9.71E-07	14.1
i5	1.03E-04	1.22E-04	18.1
i8	3.45E-04	3.98E-04	15.5
c432	2.01E-05	2.42E-05	20.5
c880	4.73E-05	4.01E-05	15.3
c6288	3.68E-04	4.40E-04	19.6
mul4x4	2.19E-05	2.49E-05	13.6
mul8x8	7.33E-05	8.23E-05	12.3
		Average	16.1

7 Acknowledgments

This work was supported in part by NSF, SRC and GSRC/ DARPA

References

[1] R. Baumann, "Soft errors in advanced computer systems," *IEEE D&T*, 22 (3), May 2005.

[2] P. Shivakumar, et al., "Modeling the effect of technology trends on the soft error rate of combinational logic," *DSN* 2002.

[3] T. Karnik, et al., "Scaling trends of cosmic ray induced soft errors in static latches beyond 0.18u," *VLSI Symposium*, 2001.

[4] S. Mitra, et al., "Robust system design with built-in soft-error resilience," *IEEE Computer*, 38(2), Feb 2005.

[5] W. Wang, "High performance radiation hardened register cell design on a standard CMOS process," *EDSSC*, 2003.

[6] S. Mitra, et al., "Logic soft errors in sub-65nm technologies design and CAD challenges," *DAC* 2005.

[7] L. Freeman, "Critical charge calculations for a bipolar SRAM array," *IBM Journal of R & D*, 40(1), 1996.

[8] P. Hazucha, C. Svensson, "Impact of CMOS technology scaling on atmospheric neutron soft error rate," *IEEE TNS,* 47(6), Dec 2000.

[9] M. Zhang, N. Shanbhag, "A soft error rate analysis (SERA) methodology," *ICCAD* 2004.

[10] H. Nguyen, Y. Yagil, "A systematic approach to SER estimation and solutions", *IRPS* 2003.

[11] P. Murley, G. Srinivasan. "Soft-error Monte Carlo modeling program, SEMM," *IBM Journal of R & D*, 40(1), 1996.

[12] K. Clark, et al., "Modeling single-event effects in a complex digital device," *IEEE TNS,* 50(6), December 2003.

[13] N. Kaul, et al., "Simulation of SEU transients in CMOS IC," *IEEE TNS*, 38(6), 1991.

[14] M. Omana, et al., "A model for transient fault propagation in combinational logic", *IOTS* 2003.

[15] A. Maheshwari, et al., "Accurate estimation of soft error rate in VLSI circuits," *DFT* 2004.

[16] M. Baze, S. Buchner, "Attenuation of single event induced pulses in CMOS combinational logic," *IEEE TNS*, 44(6), Dec 1997.

[17] P. Liden, et al., "On latching probability of particle induced transients in combinational networks," *FTCS* 1994.

[18] B. Zhang, M. Orshansky, "Symbolic simulation of the propagation and filtering of transient faulty pulses," *SELSE* 2005.

[19] N. Zivanova, D. Marculescu, "Circuit reliability techniques using symbolic techniques," *IWLS* 2005.

[20] A. Kasnavi, et al., "Analytical modeling of crosstalk noise waveforms using Weibull function," *ICCAD* 2004.

[21] Q. Zhou, K. Mohanram, "Cost effective radiation hardening technique for combinational logic," *ICCAD* 2004.

[22] S. Ercolani, et al., "Estimate of signal probability in combinational logic networks," *European Test Conference* 1989.

[23] S. Yang, Logic synthesis and optimization benchmarks user guide, Microelectronics Research Center of North Carolina, 1991.

[24] F. Brglez, H. Fujiwara, "A neural netlist of ten combinational benchmark circuits and translator in Fortran," *ISCAS* 1985.