國 立 交 通 大 學

電信工程研究所

碩 士 論 文

分析在全電量之粒子影響下之統計性軟性電子錯誤率

Fast Statistical Soft Error Rate (SSER) Analysis Considering Full-Spectrum Charge Collection

研究生：黃宣銘

指導教授：溫宏斌

中 華 民 國 100 年 7 月 4 日

分析在全電量之粒子影響下之統計性軟性電子錯誤率
Fast Statistical Soft Error Rate (SSER) Analysis Considering Full-Spectrum Charge Collection

研究生：黃宣銘　　Student : Hsuan-Ming Huang

指導教授：溫宏斌　　Advisor：Hung-Pin Wen

國 立 交 通 大 學
電 信 工 程 系
碩 士 論 文

A Thesis

Submitted to Department of Communication Enginerring

College of Electrical and Computer Engineering National

Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Communication Engineering

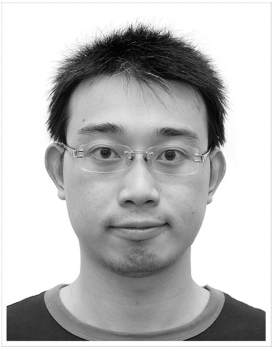
June 2011

Hsinchu, Taiwan, Republic of China

中華民國一百年六月

Hsuan-Ming Huang

EDUCATION



M.S.: Communication Engineering, National Chiao Tung University
- Advisor: Charles H.-P. Wen
- Laboratory: Computational Intelligence on Automation Lab

B.S.: Business Administration, National Central University

RESEARCH EXPERIENCE

Learning-based Statistical Soft-error Analysis *(2010-Pres.)*
- Applied machine learning (SVM) techniques to build an accurate soft-error analysis framework.

Design Optimization for TFT-LCD Panel *(2007-2009)*
- Implemented an integrated and automatic simulation platform for designing TFT-LCD panel.
- Designed a statistical-and-evolutionary-based approach for electrical-and-optical design optimization of TFT-LCD panel.

Random Dopant Induced Fluctuation in Nanowire FETs (2006-2007)
- Analyzed discrete-dopant-induced electrical and physical characteristics fluctuation in Nanowire FETs by the large-scale statistical approach.

PUBLICATIONS

Journal Papers

[1] Huan-Kai Peng, **Hsuan-Ming Huang**, Yu-Hsin Kuo, and Charles H.-P. Wen, “Statistical Soft Error Rate (SSER) Analysis for Scaled CMOS Designs,” ACM Transactions on Design Automation of Electronic Systems.

[2] Yiming Li, Chih-Hong Hwang, and **Hsuan-Ming Huang**, “Large-Scale Atomistic Approach to Discrete-Dopant-Induced Characteristic Fluctuations in Silicon Nanowire Transistors,” Physica Status Solidi (a), Vol. 205, No. 6, June 2008, pp. 1505-1510.

Conference Papers

[1] **Hsuan-Ming Huang**, Hui-Wen Cheng, Yiming Li, Tseng-Chien Tsai, Hung-Yu Chen, Kuen-Yu Huang and Tsau-Hua Hsieh, “An Optimal Design for Power Consumption of 2.2”~2.6” Display System of Mobile Phone,” International Meeting on Information Display (IMID 2009) Seoul, Korea, October 12-16, 2009.

[2] **Hsuan-Ming Huang**, and Yiming Li, “A Unified Parameterization Technique for

TFT-LCD Panel Design Optimization," Asia Symposium on Quality Electronic Design (ASQED 2009) Kuala Lumpur, Malaysia, July 15-16, 2009.

[3] **Hsuan-Ming Huang**, and Yiming Li, "Parameterized Display Performances Behavioral Modeling and Optimization for TFT-LCD Panel," NSTI Nanotechnology Conference and Trade Show (NSTI Nanotech 2009), Houston, Texas, U.S.A., May 3-7, 2009.

[4] Yiming Li, and **Hsuan-Ming Huang**, "Computational Statistics Approach to Capacitance Sensitivity Analysis and Gate Delay Time Minimization of TFT-LCDs," Accepted by The 7th International Conference on Scientific Computing in Electrical Engineering (SCEE), Finland, September 28 - October 3, 2008.

[5] **Husan-Ming Huang**, Chih-Hong Hwang, and Yiming Li, "Large-Scale "Atomistic" Approach to Discrete-Dopant Fluctuated Si Nanowire FETs," Accepted by Trends in Nanotechnology Conference (TNT), Kursaal congress facilities, San Sebastian, Spain, September 03-07, 2007.

[6] Chih-Hong Hwang, Ta-ChingYeh, **Hsuan-Ming Huang**, and Yiming Li, "High Frequency Characteristics of Discrete-Dopant Fluctuations in Nanoscale MOSFET Circuits," Accepted by The 12th IEEE International Workshop on Computational Electronics (IEEE IWCE), University of Massachusetts Amherst, USA, October 8-10, 2007.

[7] Yiming Li, Chih-Hong Hwang, **Hsuan-Ming Huang**, and Ta-Ching Yeh, "Discrete Dopant Induced Characteristic Fluctuations in 16nm Multiple-Gate SOI Devices," Accepted by The 2007 IEEE International SOI Conference (IEEE SOI), Miramonte Resort and Spa, Indian Wells, California, USA, October 1-4, 2007.

[8] Yiming Li, Chih-Hong Hwang, Shao-Ming Yu, **Hsuan-Ming Huang**, Ta-ChingYeh, and Hui-Wen Cheng, "Effect of Discrete Dopant on Characteristic Fluctuations in 16nm SOI-FinFETs," Accepted by The 12th IEEE International Conference on Simulation of Semiconductor Devices and Processes (IEEE SISPAD), TU Wien, Vienna, Austria, Sept. 25-27, 2007.

[9] Yiming Li, Chih-Hong Hwang, **Hsuan-Ming Huang**, and Ta-ChingYeh, "Discrete-Dopant-Fluctuated Threshold Voltage Roll-Off in Sub-16nm Bulk FinFETs," Accepted by The 2007 International Conference on Solid State Devices and Materials (SSDM), Tsukuba International Congress Center (EPOCHAL TSUKUBA), Ibaraki, Japan, September 18-21, 2007.

[10] Yiming Li, Chih-Hong Hwang, Shao-Ming Yu, and **Hsuan-Ming Huang**, "Three-Dimensional Simulation of Random-Dopant-Induced Threshold Voltage Fluctuation in Nanoscale Fin-typed Field Effect Transistors," Accepted by The IEEE 19th International Conference on Noise and Fluctuations (IEEE ICNF), Tokyo, Japan, Sept. 9-14, 2007.

[11] Chih-Hong Hwang, Yiming Li, **Hsuan-Ming Huang**, and Hsiang-Yu Lo, “Impact of Gate-Coverage on Immunity against Fluctuation of Silicon Naowire Transistor,” Accepted by Conference on Computational Physics (APS CCP), Brussels, Belgium, Sept. 5-8, 2007.

[12] Yiming Li, Chih-Hong Hwang, Shao-Ming Yu, and **Hsuan-Ming Huang**, “Random Dopant Induced Thermal Fluctuation in Nanoscale SOI FinFET,” Accepted by The 7th IEEE International Conference on Nanotechnology (IEEE NANO), Hong Kong Convention & Exhibition Centre, Hong Kong, China, August 2-5, 2007.

[13] Yiming Li, Chih-Hong Hwang, Shao-Ming Yu, and **Hsuan-Ming Huang**, “Electrical Characteristic Fluctuations in 16nm Bulk-FinFET Devices,” Presented in The IEEE 15th Biannual Conference Insulating Films on Semiconductors (IEEE INFOS), Athens, Greece, June 20-23, 2007.

[14] Yiming Li, Chih-Hong Hwang, Shao-Ming Yu, and **Hsuan-Ming Huang**, “Characteristic Fluctuation Dependence on Gate Oxide Thickness Scaling in Nano-MOSFETs,” Workshop Abstracts of The 2007 IEEE Silicon Nanoelectronics Workshop (IEEE SNW) Kyoto, Japan, June 10-11, 2007, pp. 79-80.

[15] Yiming Li, Chih-Hong Hwang, Ta-Ching Yeh, and **Hsuan-Ming Huang**, “Effects of Aspect- and Coverage-Ratio on Radio-Frequency Characteristics of Silicon Nanowire Transistors,” Workshop Abstracts of The 2007 IEEE Silicon Nanoelectronics Workshop (IEEE SNW) Kyoto, Japan, June 10-11, 2007, pp. 153-154.

[16] Yiming Li, Chih-Hong Hwang, Shao-Ming Yu, **Hsuan-Ming Huang**, and Hung-Ming Chen, “Random Discrete Dopant Fluctuated Sub-32 nm FinFET Devices,” Technical Proceedings of The 2007 NSTI Nanotechnology Conference and Trade Show (NSTI Nanotech), Santa Clara, California, U.S.A., May 20-24, 2007, vol. 1, pp. 189-192.

WORK EXPERIENCE

- Teaching Assistant, Object-Oriented Programming in C++, NCTU (2011-Pres.)
- Teaching Assistant, Computer-Aided Circuit Design and Analysis, NCTU (2009)
- Teaching Assistant, Computer-Aided Circuit Design and Analysis, NCTU (2008)
- Teaching Assistant, Numerical Semiconductor Device Modeling, NCTU (2007)
- Teaching Assistant, Numerical Analysis, NCTU (2007)

EXTRACURRICULAR ACTIVITIES

- Leader of NCUBA basketball team *(2004-2005)*

- Player in NCU soccer school team *(2002-2003)*
- Host of 2 camps, 3 ball games, and other activities *(2002-2006)*

SKILLS

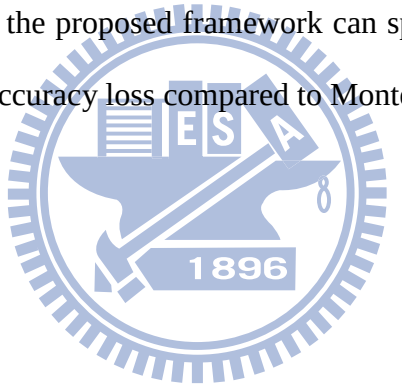
- Languages: C/C++, Qt, R, Matlab, Perl, and other script languages.
- Design Tools: ISE-TCADTM, RaphaelTM, EldoTM, HSpiceTM, NanosimTM, LakerTM,
- Publishing Software: Latex, Visio, Microsoft Office, Linux and Windows.

摘　要

近年來，隨著深次微米時代的來臨，製程變異對於系統的穩健帶來了極大的挑戰。其中，軟性電子錯誤率在先進電路的設計上被發現的機率也愈來愈高，對電路之可靠度而言又變成一個重要的研究題目。然而，在前人的研究中，並無一個可有效地估計在製程變異下之軟性電子錯誤率。因此，在本論文中建立出一個準確且快速的方法來有效地估計在製程變異下，軟性電子錯誤率對電路可靠度之影響，其中主要包涵有以下二個部分(1) 資料重建及改良式機器學習方法 (2) 粒子電量邊界選擇自動化。透過改良式機器學習配合資料重建，我們可快速建構出精確的軟性電子錯誤率模型。在建構精確模型後，此方法會自動選擇所需計算之粒子電量，並排除掉其它不需計算電量，以達加速計算軟性電子錯誤率之目的。實驗結果證明，此方法在 ISCAS 電路中與蒙地卡羅電路模擬相比可加速約 10^7 倍，且只有 0.8%的平均誤差。

Abstract

This thesis re-examines the soft error effect caused by radiation-induced particles beyond the deep sub-micron regime. Soft error has become one of critical reliability concerns due to the continuous technology scaling. Hence, it is necessary to develop an approach to accurately estimate soft error rate (SER) integrated with the process-variation impact. Due to inaccuracy of previously published approaches, an accurate-and-efficient framework is proposed in this thesis to perform statistical soft error rate (SSER) analysis considering full-spectrum charge collection. This framework mainly consists of two components (1) intensified learning with data reconstruction and (2) automatic bounding-charge selection. Experimental results show that the proposed framework can speed up SER estimation at the order of 10^7X with only 0.8% accuracy loss compared to Monte-Carlo SPICE simulation.



誌　　謝

本論文得以順利完成，首先要感謝的是我的指導教授溫宏斌老師。老師在我最徬惶無助的時候，願意擔任我的指導教授。在其指導期間中，老師在專業領域上的帶領，使我不斷地在未知領域中成長茁壯，也使我重新找回學習的感動與熱情，另外，不只是在專業上地引導，老師更教會了我許多待人處事的道理，受益良多。

另外，要感謝王俊堯及江蕙如教授願意撥冗擔任我的口試委員，並提供我寶貴的意見，使我的論文能夠更完整。也要感謝 CIA 實驗室的成員佳伶、千慧、家慶、欣恬、鈞堯、竣惟、玨璇、昱澤、凱華、洧駐、鉉崴。謝謝你們總是為實驗室帶來歡樂，使我總是能在開心的氣氛中學習，並常常在我無助地時候，給予我支持與鼓勵。

接著要感謝正凱學長、紹銘學長、至鴻學長、益廷學長、大慶、典燁，你們是我的碩士生涯中不可或缺的良師益友。

最後僅以此文獻給我摯愛的父母及姐姐。

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Soft errors have emerged to be one of the dominant failure mechanisms for reliability in modern CMOS technologies. Soft errors result from radiation-induced transient faults latched by memory elements and used to be of concern only for memory units but now becomes commonplace for logic units beyond deep sub-micron technologies. As predicted in [28] [8] [1], the soft error rate (SER) in combinational logic will be comparable to that of unprotected memory cells in 2011. Therefore, numerous studies have been proposed to modeling of transient faults [6] [30] [23] [11], propagation and simulation/estimation of soft error rates [35] [33] [25] [27] and circuit hardening techniques including detection and protection [20] [2] [18] [34]. Numerous previous works such as [23] [19] transient faults are propagated through one gate according to the logic function and in the meantime use analytical models to evaluate the electrical change of their pulse widths. A refined model presented in [11] to incorporate non-linear transistor current is further applied to different gates with different charges deposited. A static analysis is also proposed in [14] for timing masking by computing backwards the propagation of the error-latching windows efficiently. Moreover, in recent years, circuit reliability in terms of soft error rate (SER) has been extensively investigated. SERA [35] computes SER by means of a waveform model to consider the electrical attenuation effect and error-latching probability while ignoring logical masking. Whereas FASER [33] and MARS-C [16] apply symbolic techniques to logical and electrical maskings and scale the error probability according to the specified clock period, AnSER [14] applies signature observability and latching-window computation for logical and timing maskings to approximate SER for circuit hardening. SEAT-LA [25] and the algorithm in [27] simultaneously characterize cells, flip-flops, describe transient-fault propagation by waveform models and result in good SER estimate when comparing to SPICE simulation. However, all of these techniques are deterministic and may not be capable of explaining more sophisticated circuit behaviors due to the growing process variations beyond deep sub-micron era.

Process variations including numerous manufacturing defects have grown to be one of the major challenges to scaled CMOS designs [5] [4]. From [22] [4], 25%-30% variation on chip frequency are observed. For design reliability, 15%-40% SER variations are reported in [26] under the 70nm technology. Also, authors in [17] propose an symbolic approach to propagate transient faults considering process variations.

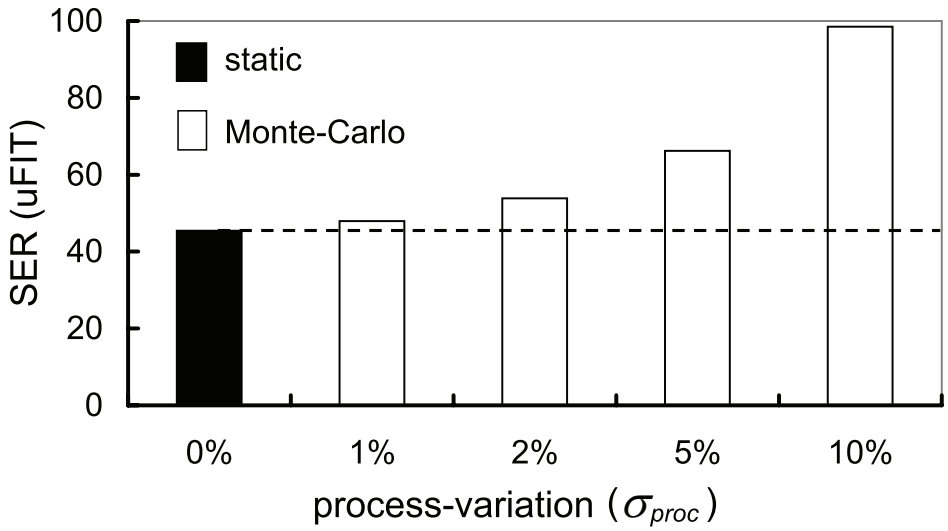


Figure 1.1: SER discrepancies between static and Monte-Carlo SPICE simulation w.r.t. different process variations

Using the 45nm Predictive Technology Model (PTM) [21], the impact of process variations on circuit reliability is illustrated in Figure 1.1, where SERs are computed by SPICE simulation on a sample circuit $c17$ from ISCAS'85 under different rates for process-variation (σ_{proc}'s) applied to perturbing the W/L ratio of each transistor in each cell's geometry. The X-axis and Y-axis denote σ_{proc} and SER, respectively, where FIT (Failure-In-Time) is defined by the number of failures per 10^9 hours. Nominal settings without variation are used in static SPICE simulation, whereas Monte-Carlo SPICE simulations are used to approximate process-variation impacts under different σ_{proc}'s.

As a result, SER from static SPICE simulation is **underestimated**. Considering different σ_{proc}'s in Monte-Carlo SPICE simulation, all SERs are higher than that from static SPICE simulation. As process variations deteriorate, the discrepancy between Monte-Carlo and static SERs further enlarges. In Figure 1.1, $(SER_{monte} - SER_{static})/SER_{static}$ under $\sigma_{proc} = 1\%, 2\%, 5\%$ and 10% are 6%, 19%, 46% and 117%, respectively. Such result suggests that the impact of process variations to SER analysis may no longer be ignored in scaled CMOS designs.

For considering process variation, authors in [24] propose an accurate statistical soft error rate (SSER) framework based on learning-based statistical models for transient-fault

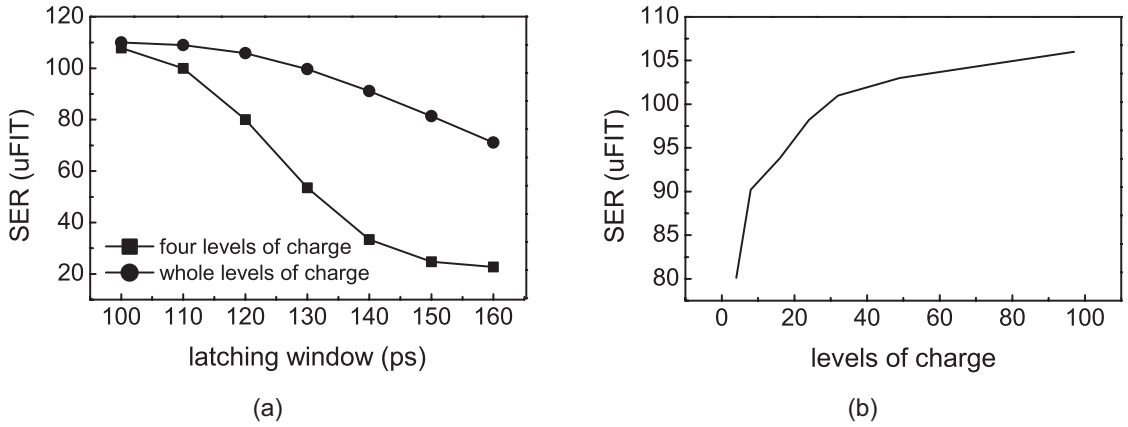


Figure 1.2: (a) Soft error rate comparison between two different analyses with different latching window(b) Soft error rate under different levels of charge collection

distributions. Using statistical tables for cell models in Monte-Carlo simulation, another SSER approach is also investigated in [15], which is more accurate but runs slower than the previous work. However, both works [24] [15] simplify their SER analysis by only injecting only four levels of electrical charges, and therefore motivate us to pose a fundamental but important question: *Are four levels of electrical charges enough to converge SER computation and completely explain the process-variation impact?*

Figure 1.2(a) illustrates the comparison of SERs from two SPICE simulations using different level of charges onto a sample circuit $c17$. The line with square symbol and the line with circle symbol represent the soft error rates under a 5% process variation by injecting only four levels of charges and full-spectrum charges, respectively. Actually, effective charge collection for SER analysis ranges from 35fC to 132fC, and SER difference from statistical analysis (with only four levels of charges) can go up to 69% (latching window = 150 ps). *Therefore, one more question comes up: If four levels of charges are not sufficient to explain the behavior of SER, how many levels of charges is sufficient?*

Figure 1.2(b) suggests the answer. Since the SER is increasing significantly until full-spectrum charges are considered, all of charges should be included. The cause of the SER difference can be further explained by an example shown in Figure 1.3 where the upper and lower parts of Figure 1.3 indicate the distribution of SSER induced by only four levels of charges and SSER induced by full levels of charges, respectively. X-axis and Y-axis

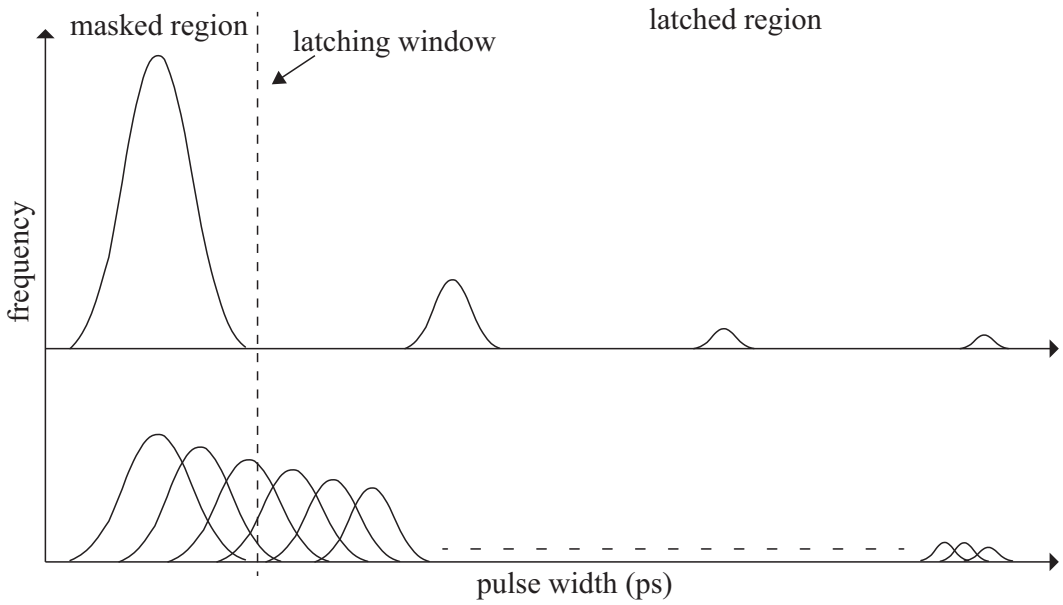


Figure 1.3: SER distributions induced by four levels of charges and by full-spectrum charges

denote the pulse width of the transient faults and effective frequency for a particle hit of charges. For SSER induced by four-level charges, only four transient-fault (TF) distributions are generated and contribute the final soft error rate. In other words, the soft errors will be concentrated in four blocks and may result in a misleading SER. For example, as the latching-window boundary of one flip-flop is far away from the first TF distribution, all soft errors induced by this TF distribution are masked due to timing masking as illustrated in Figure 1.3. But, in fact, only a part of them should be masked. Accordingly, SER analysis is no longer valid with only four levels of collection charges and instead should be comprehensively assessed by full-spectrum charge collection.

In this thesis, we present an efficient-and-accurate framework which integrates the impact of process variation and considers full levels of charges during SER analysis for combinational circuits. In addition, a bounding technique is proposed for accelerating SER computation and determines the necessary set of charges to apply statistical analysis. Advanced learning technique (i.e. support vector machine (SVM)) is also used to derive quality cell models for fast and accurate SER computation. The rest of the thesis is organized as follows: Chapter 2 describes the background of the generation and propagation of transient

faults and two related phenomena. Chapter 3 describes the formulation of the statistical soft error rate (SSER) problem. Chapter 4 presents the an intensified-learning framework including the automatic bounding-charge selection. Chapter 5 shows experimental results while Chapter 6 draws the conclusion.

Chapter 2

Background

In this chapter, the background of soft errors will be reviewed. First, radiation-induced transient faults and three masking mechanisms will be described in section 2.1 and 2.2, respectively. Then, two particular natures beyond deep sub-micron technologies will be illustrated in section 2.3 and 2.4, respectively. One makes the faults more unpredictable whereas the other causes the discrepancy in Figure 1.1. In these two sections, discussions are associated with the three masking mechanisms.

2.1 Radiation-induced Current Pulses

When a neutron particle strikes the silicon bulk of a device, it leads to the generation of electron-hole pairs. These freeing electron-hole pairs will result in a transient fault and may cause the operational failures. However, not each energy levels of particle strikes can result in a soft error. The transient fault induced by a low energy-level particle strike does not cause the soft error due to its output voltage changes less than V_{dd}/2. High energy-level particle strikes can also be ignored because the lower flux of neutrons (10X less than low energy-level particle strikes) [12].

Additionally, the transient fault can be modeled as a current source generated by the charge of particle. In [10], the author proposed a single exponential current source model to represent the transient current induced by a neutron particle as follows:

$$I(t) = \frac{Q}{\tau}\sqrt{\frac{t}{\tau}}e^{-t/\tau} \tag{2.1}$$

,where Q is the amount of injected charge deposition, τ is charge collection time constant.

Based on the (2.1), the deposited charge of neutron particle range from **35fC** to **132fC** is found by our extensive SPICE simulation. Note that, the energy levels of neutron particle were mapped into deposited charge using JEDEC89 Standard [12].

2.2 Three Masking Mechanisms

The transient fault propagating through a path to the flip-flop is affected by three masking mechanisms which collectively prevent the circuit to have a failure due to such transient glitches.

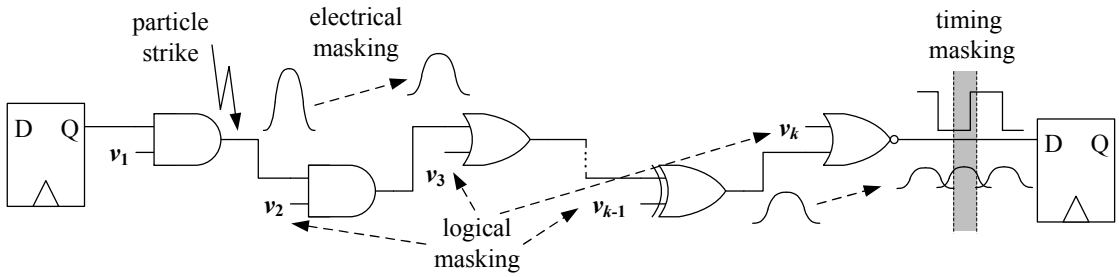


Figure 2.1: Three masking mechanisms for soft errors

Three masking mechanisms shown in Figure 2.1 from [28] are indicated as the key factors to determine if one transient fault can be latched by the memory elements to become a soft error. **Logical masking** occurs when the input value of one gate blocks the propagation of the transient fault under a specific input pattern. One transient fault attenuated by **electrical masking** may disappear due to the electrical properties of the gates. **Timing masking** represents the situation that the transient fault propagates to the input of one memory element outside the window of its clock transition.

2.3 To Be Electrically Better Or Worse?

The first observation is conducted by running static SPICE simulation on a path consisting of various gates (including 2 AND, 2 OR and 4 NOT gates) in the 45nm PTM technology. As shown in Figure 2.2, the radiation-induced particle first strikes on the output of the first NOT gate with a charge of $32fC$, and then propagates the transient fault along other gates with all side-inputs set properly. The pulse widths (pw_i's) in voltage of the transient fault starting at the struck node and after passing gates along the path in order are 171ps, 183ps, 182ps, 177ps, 178ps, 169ps, 166ps and 173ps, respectively. Each pw_i and pw_{i+1} can be compared to show the changes of voltage pulse widths during propagation in Figure 2.2.

As we can see, the voltage pulse widths of such transient fault go larger through gate #1, #4, and #7 while gate #2, #3, #5 and #6 attenuate such transient fault. Furthermore, gates of the same type behave differently when receiving different voltage pulses. To take AND-type gates for example, the output pw_1 is larger than the input pw_0 on gate

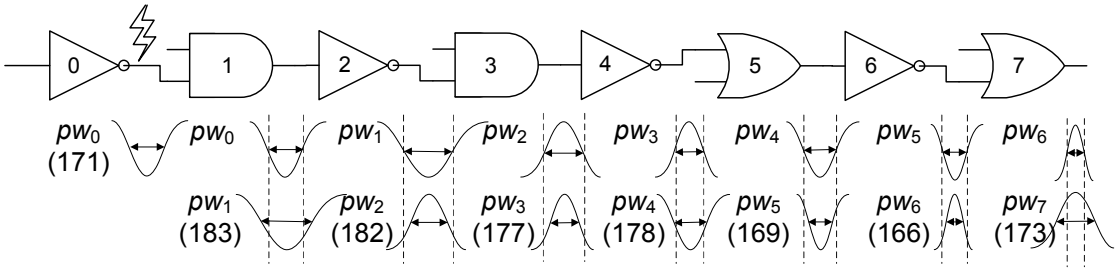


Figure 2.2: Static SPICE simulation of a path in the 45nm technology

$\#1$ while the contrary situation ($pw_3 < pw_2$) occurs on gate $\#3$. This result suggests that the voltage pulse width of a transient fault is not always diminishing, which contradicts to some assumptions made in traditional static analysis. A similar phenomenon called Propagation Induced Pulse Broadening (PIPB) was discovered in [9] and states that the voltage pulse width of a transient fault widens as it propagates along the long inverter chain.

2.4 When Error-Latching Probability Meets Process Variations

The second observation is dedicated to the timing-masking effect under process variations. In [16] [33], the error-latching probability (PL) for one flip-flop is defined as

$$PL = \frac{pw - w}{t_{clk}} \tag{2.2}$$

where pw, w and t_{clk} denote the pulse width of the arrival transient fault, the latching window of the flip-flop, and the clock period, respectively. However, process variations make pw and w become *random variables*. Therefore, we need to redefine Equation (2.2) as the following.

Definition ($\mathrm{P}_{err-latch}$, **error-latching probability**)
Assume that the pulse width of one arrival transient fault and the latching window ($t_{setup} + t_{hold}$) of the flip-flop are random variables and denoted as pw and w, respectively. Let $x = pw - w$ be another random variable and μ_x and σ_x be its mean and variance. The

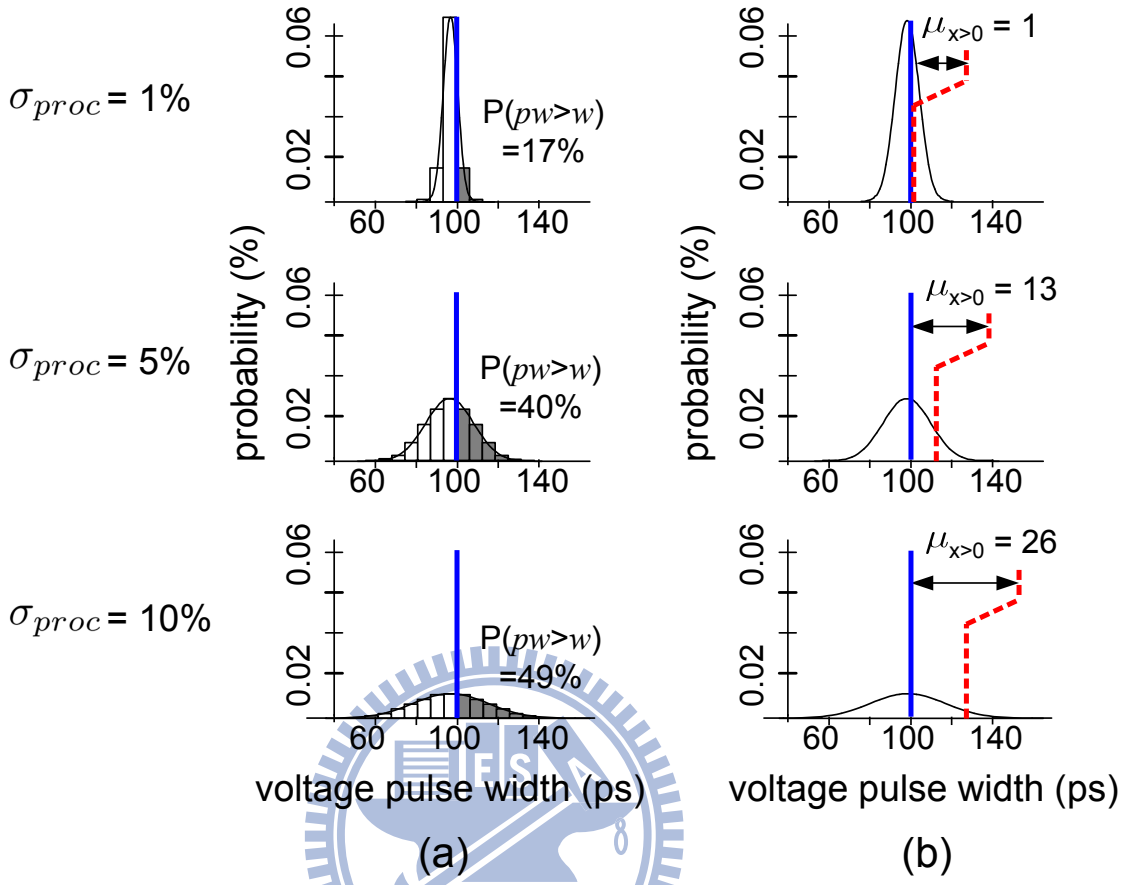


Figure 2.3: Process variations vs. error-latching probabilities

latch probability is defined as:

$$\mathrm{P}_{err-latch}(pw, w) = \frac{1}{t_{clk}} \int_{0}^{\mu_x + 3\sigma_x} x \times \mathrm{P}(x > 0) \times dx \tag{2.3}$$

With the above definition, we further illustrate the impact of process variations on SER analysis. Figure 2.3(a) shows three transient-fault distributions with the same pulse-width mean (95ps) under different σ_{proc}'s: 1%, 5% and 10%. A fixed latching window $w = 100$ps is assumed as indicated by the solid lines. According to Equation (2.2), static analysis result in zero SER under all σ_{proc}'s because $95 - 100 < 0$.

From a statistical perspective, however, these transient faults all yield positive and different SERs. It is illustrated using two terms: $\mathrm{P}(x > 0)$ and x in Equation (2.3). First, in Figure 2.3(a), the *cumulative probabilities* for $pw > w$ under three different σ_{proc}'s are 17%, 40%, and 49%, respectively. The largest σ_{proc} corresponds to the largest $\mathrm{P}(x > 0)$

term. Second, in Figure 2.3(b), we compute the pulse-width averages for the portion $x = pw - w > 0$ and they are 1, 13 and 26, respectively. Again, the largest σ_{proc} corresponds to the largest x term.

These two effects jointly suggest that a larger σ_{proc} lead to a larger $P_{err-latch}$, which has been neglected in traditional static analysis, and also explained the increasing discrepancy shown in Figure 1.1. In summary, process variations make traditional static analysis no longer effective and should be considered in accurate SER estimation for scaled CMOS designs.

Chapter 3

Problem Formulation of Statistical Soft Error Rate (SSER)

In this chapter, we formulate the statistical soft error rate (SSER) problem for general cell-based circuit designs. Figure 3.1 illustrates a sample circuit subject to process variations, where the geometries of each cell vary [22]. Once a high-energy particle strikes the diffusion region of these varying-size cells, according to Figure 1.1, 2.2 and 2.3, the electrical performances of the resulted transient faults also vary a lot. Accordingly, to accurately analyze the soft error rate (SER) of a circuit, we need to integrate both process-variation impacts and three masking effects simultaneously, which bring about the statistical soft error rate (SSER) problem.

The SSER problem is composed of three blocks: (1) electrical probability computation, (2) propagation probability computation and (3) overall SER estimation. A bottom-up mathematical explanation of the SSER problem will start reversely from overall SER estimation to electrical probability computation.

3.1 Overall SER Estimation

The overall SER for the circuit under test (CUT) can be computed by summing up the SERs originated from each individual struck node in the circuit. That is,

$$\text{SER}_{CUT} = \sum_{i=0}^{N_{node}} \text{SER}_i \tag{3.1}$$

where N_{node} denotes the total number of possible nodes to be struck by radiation-induced particles in the CUT and SER_i denotes the SERs from node i, respectively.

Each SER_i can be further formulated by integrating over the range $q = 0$ to Q_{max} (the maximum collection charge from the environment) the products of *particle-hit rate* and the total number of soft errors that q can be induced from node i. Therefore,

$$\text{SER}_i = \int_{q=0}^{Q_{max}} (\text{R}_i(q) \times \text{F}_{soft-err}(i,q))\text{d}q \tag{3.2}$$

In a circuit, $\text{F}_{soft-err}(i,q)$ represents the total number of *expected* soft errors from each flip-flop that a transient fault from node i can propagate to. $\text{R}_i(q)$ represents the effective frequency for a particle hit of charge q at node i in unit time according to [28] [35]. That

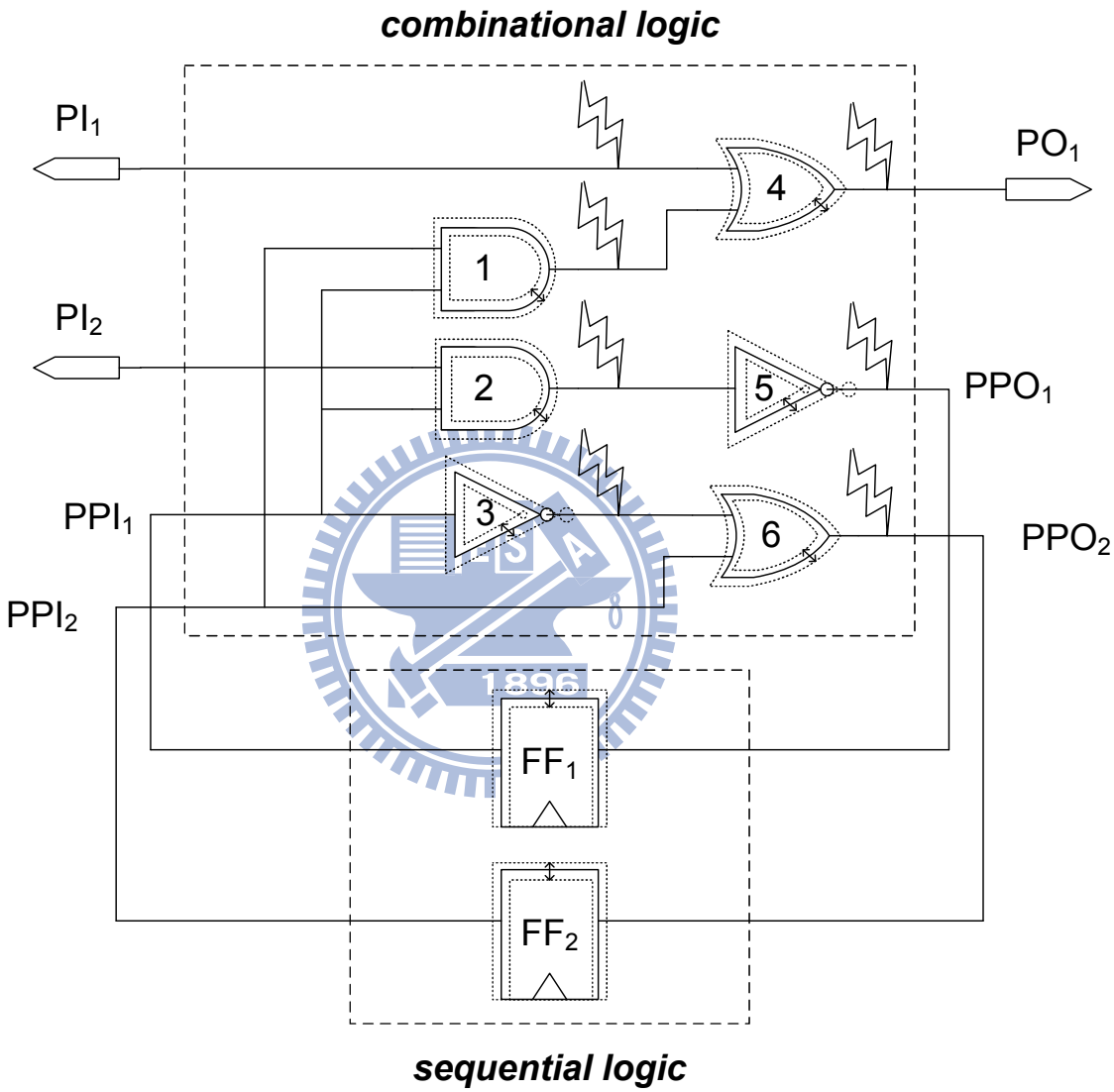


Figure 3.1: An illustrative example for the SSER problem

is,

$$\mathrm{R}_i(q) = F \times K \times A_i \times \frac{1}{Q_s} e^{\frac{-q}{Q_s}} \tag{3.3}$$

where F, K, A_i and Q_s denote the neutron flux ($>$ 10MeV), a technology-independent fitting parameter, the susceptible area of node i in cm^2, and the charge collection slope, respectively.

3.2 Logical Probability Computation

$\mathrm{F}_{soft-err}(i, q)$ depends on all three masking effects and can be decomposed into

$$\mathrm{F}_{soft-err}(i, q) = \sum_{j=0}^{N_{ff}} \mathrm{P}_{logic}(i, j) \times \mathrm{P}_{elec}(i, j, q) \tag{3.4}$$

where N_{ff} denotes the total number of flip-flops in the circuit under test. $\mathrm{P}_{logic}(i, j)$ denotes the overall logical probability of successfully generating a transient fault and propagating it through all gates along one path from node i to flip-flop j. It can be computed by multiplying the signal probabilities for specific values on target gates as follows.

$$\mathrm{P}_{logic}(i, j) = \mathrm{P}_{sig}(i = 0) \times \prod_{k \in i \rightsquigarrow j} \mathrm{P}_{side}(k) \tag{3.5}$$

where k denotes one gate along the target path ($i \rightsquigarrow j$) starting from node i and ending at flip-flop j, P_{sig} denotes the signal probability for the designated logic value, and P_{side} denotes the signal probability for the non-controlling values (i.e. 1 for AND gates and 0 for OR gates) on all side inputs along such target path.

Figure 3.2 illustrates an example where a particle striking net a results in a transient fault that propagates through net c and net e. Suppose that the signal probability of being 1 and 0 on one arbitrary net i is P_i and (1-P_i), respectively. In order to propagate the transient fault from a towards e successfully, net a needs to be 0 while net b, the side input of a, and net d, the side input of c, need to be non-controlling, simultaneously. Therefore, according

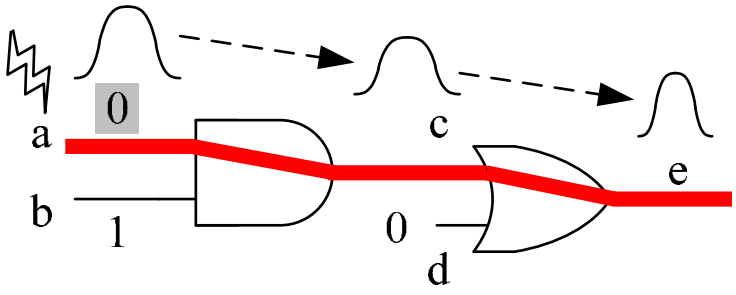


Figure 3.2: Logical probability computation for one sample path

to Equation (3.5),

$$\begin{aligned} \mathrm{P}_{logic}(a, e) &= \mathrm{P}_{sig}(a = 0) \times \mathrm{P}_{side}(a) \times \mathrm{P}_{side}(c) \\ &= \mathrm{P}_{sig}(a = 0) \times \mathrm{P}_{sig}(b = 1) \times \mathrm{P}_{sig}(d = 0) \\ &= (1 - \mathrm{P}_a) \times \mathrm{P}_b \times (1 - \mathrm{P}_d) \end{aligned}$$

3.3 Electrical Probability Computation

Electrical probability $\mathrm{P}_{elec}(i, j, q)$ comprises the electrical- and timing-masking effects and can be further defined as

$$\begin{aligned} \mathrm{P}_{elec}(i, j, q) &= \mathrm{P}_{err-latch}(pw_j, w_j) \\ &= \mathrm{P}_{err-latch}(\lambda_{elec-mask}(i, j, q), w_j) \end{aligned} \tag{3.6}$$

While $\mathrm{P}_{err-latch}$ accounts for the timing-making effect as defined in Equation (2.3), $\lambda_{elec-mask}$ accounts for the electrical-masking effect with the following definition.

Definition ($\lambda_{elec-mask}$, **electrical masking function**)
Given the node i where the particle strikes to cause a transient fault and flip-flop j is the destination that the transient fault finally ends at, assume that the transient fault propagates along one path ($i \rightsquigarrow j$) through $v_0, v_1, ..., v_m, v_{m+1}$ where v_0 and v_{m+1} denote node i and

flip-flop j, respectively. Then the electrical-masking function

$$\lambda_{elec-mask}(i,j,q) = \underbrace{\delta_{prop}(\cdots(\delta_{prop}(\delta_{prop}}_{m \text{ times}}(pw_0,1),2),\cdots),m) \tag{3.7}$$

where $pw_0 = \delta_{strike}(q,i)$ *and* $pw_k = \delta_{prop}(pw_{k-1},k)\ \forall k \in [1,m]$

In the above definition, two undefined functions, δ_{strike} and δ_{prop}, respectively, represent the *first-strike* function and the electrical *propagation* function of transient-fault distributions. $\delta_{strike}(q,i)$ is invoked one time and maps the deposited charge q at node i into a voltage pulse width pw_0. $\delta_{prop}(pw_{k-1},k)$ is invoked m times and *iteratively* computes the pulse width pw_k after the input pulse width pw_{k-1} propagates through the k-th cell from node i. These two types of functions are also the most critical components to the success of a statistical SER analysis framework due to the difficulty from integrating process-variation impacts.

The theoretical SSER in Equation (3.5) and Equation (3.7) is expressed from a path perspective. However, in reality, since both the signal probabilities and transient-pulse changes through a cell are independent to each other, SSER is processed stage by stage and can be implemented in a block-based fashion. Next, Chapter 4 follows such notions and presents the proposed intensified-learning framework for computing SSERs.

Chapter 4

A Statistical SER Analysis Framework

The table-lookup Monte-Carlo framework is inherently limited in execution efficiency because it computes δ_{strike} and δ_{prop} ***indirectly*** using extensive samplings of Monte-Carlo runs. In this chapter, we propose an alternative **learning-based** framework ***directly*** on the basis of support vector regression (SVR) and is found more efficient and accurate than the previous method.

4.1 A Baseline Learning Framework

Figure 4.1 shows our SVR-learning framework with respective learning models (δ_{strike} and δ_{prop}).

By definition, δ_{strike} and δ_{prop} are functions of pw that is a *random variable*. From Figure 2.2 and Figure 2.3, we assume pw follows the normal distribution, which can be written as:

$$pw \sim N(\mu_{pw}, \sigma_{pw}) \tag{4.1}$$

Therefore, we can decompose δ_{strike} and δ_{prop} into four models: δ^{μ}_{strike}, δ^{σ}_{strike}, δ^{μ}_{prop}, and δ^{σ}_{prop} where each can be defined as:

$$\delta : \vec{x} \mapsto y \tag{4.2}$$

where $\vec{x}$ denotes a vector of *input variables* and y is called the model's *label* or *target value*.

Integrating the impact of process variations, four models are traditionally built using look-up tables. However, look-up tables have two limitations on applicability: (1) inaccurate interpolation and (2) coarse model-size control. First, look-up tables can take only finite table indices and must use interpolation. However, interpolation functions are often not accurate enough or difficult to obtain, especially as the table dimensionality grows. Second, a look-up table stores data samples in a grid-like fashion, where the table will grow prohibitively large for a fine resolution. Meanwhile, the *information richness* often differs across different parts of a table. For example, we observe that pulse widths generated by strong charges behave much simpler than weaker charges do. Naturally, simple behaviors can be encoded with fewer data points in the model, whereas complicated behaviors need to be encoded with more.

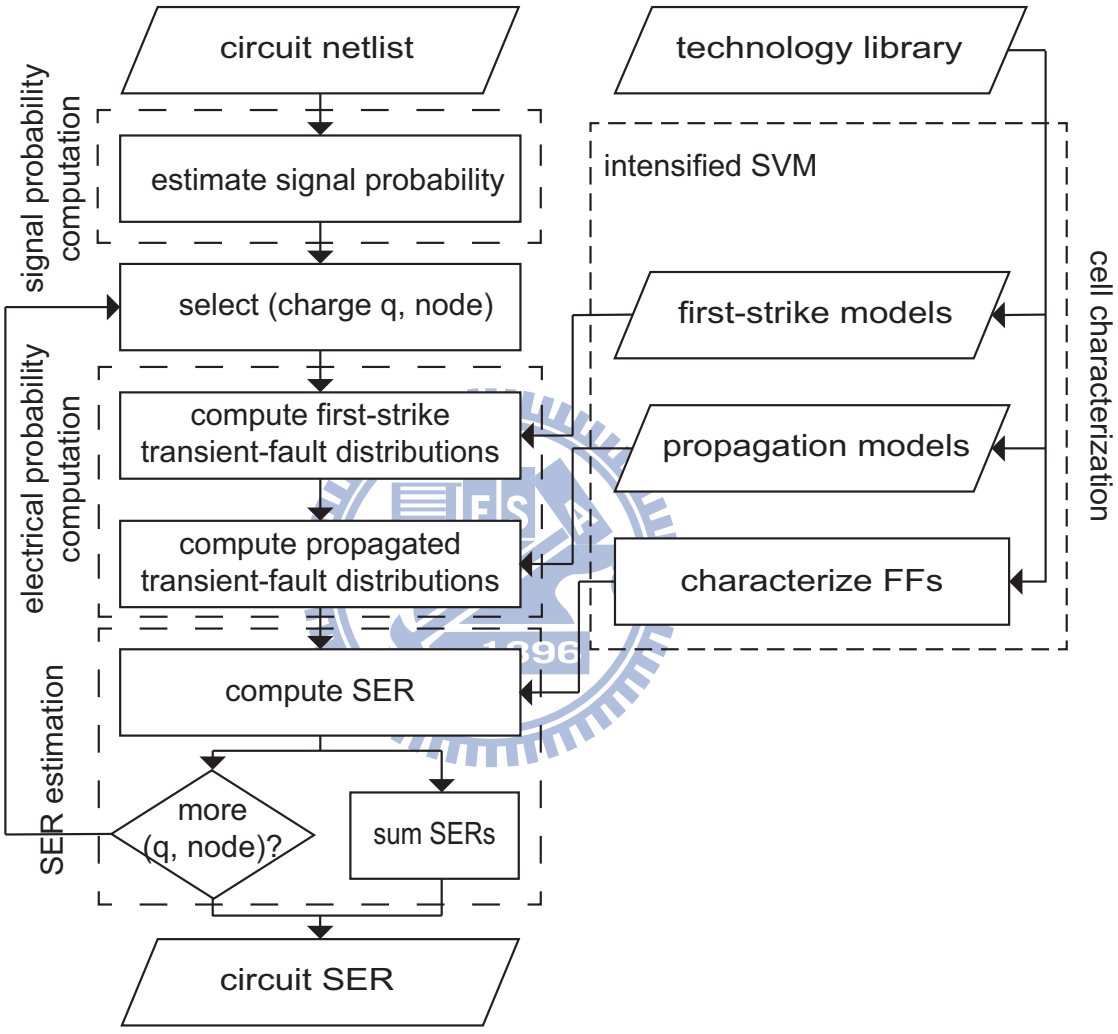


Figure 4.1: Proposed statistical SER framework

In statistical learning theory, such models can be built using regression, which can be roughly divided into linear [32] and non-linear [3] methods. Among them, Support Vector Regression (SVR) [31] [29] combines linear methods' efficiency and non-linear methods' descriptive power. It has two advantages over a look-up table: (1) It gives an explicit function and does not need interpolation. (2) It filters out unnecessary sample points and yields compact models.

In the following, we propose a methodology which comprises training sample preparation, SVR model training, parameter selection, and data reconstruction.

4.1.1 Training Sample Preparation

SVR models differ from lookup tables on the way we prepare training samples for them. For look-up tables, one starts from selecting a finite set of points along each table dimension. On one hand, they should be chosen economically; on the other hand, it is difficult to cover all corner cases with only limited numbers of points. For SVR models, we do not need to select these points. Instead, we provide a large set of *training samples*, and let the SVR algorithm do the selection task.

A training sample set S of m samples is defined as:

$$S \subset (\vec{X} \times Y)^m = \{(\vec{x}_1, y_1), \ldots, (\vec{x}_m, y_m)\} \tag{4.3}$$

where m pairs of input variables $\vec{x}_i$'s and target values y_i's are obtained from massive Monte-Carlo SPICE simulation. For δ^{μ}_{strike}, δ^{σ}_{strike}, we use input variables including charge strength, driving gate, input pattern, and output loading; for δ^{μ}_{prop}, δ^{σ}_{prop}, we use input variables including input pattern, pin index, driving gate, input pulse-width distribution (μ^{i-1}_{pw} and σ^{i-1}_{pw}), propagation depth, and output loading.

In our training samples, we denote output loading using combinations of input pins of arbitrary cells. Doing so preserves additional information of the output loading status and saves the labor (and risk) of characterizing the capacity of each cell's input pin. Although the number of such combinations can easily explode, there are usually only a limited number of representatives, which are automatically identified by SVR. Furthermore, from a learning perspective, since both peak voltage and pulse width are the responses of current

formulated in Equation (2.1), they are highly correlated. Empirically, using pulse-width information solely is sufficient to yield satisfactory SSERs and thus in our framework, we do not build models for peak voltages.

4.1.2 Support Vector Machine and Its Extension to Regression

Support vector machine (SVM) is one of the most widely used algorithms for learning problems [31] and can be summarized with the following characteristics:

- SVM is an efficient algorithm and finds a global minimum (or maximum) for a convex optimization problem formulated from the learning problem.
- SVM avoids the curse of dimensionality by capacity control and works well with high-dimensional data.
- SVM automatically finds the decision boundary for a collection of samples using a small subset where each sample is called a *support vector*.

The basic idea behind SVM is to find a function as the decision boundary with minimal errors and a maximal margin to separate data in multi-dimensional space. Given a training set **S**, with $\vec{x}_i \in \mathbb{R}^n$, $y_i \in \mathbb{R}$, the SVM learning problem is to find a function f (first assume $y = f(\vec{x}) = \langle \vec{w} \cdot \vec{x} \rangle + b$) that models **S** properly. Accordingly, the learning task is formulated into a constrained optimization problem as follows,

$$\begin{aligned} &\text{minimize} \quad \|\vec{w}\|^2 + C(\textstyle\sum_{i=1}^{m} \xi_i)^k \\ &\text{subject to} \quad \begin{cases} y_i(\langle \vec{w} \cdot \vec{x}_i \rangle + b) \geq 1 - \xi_i, \; i = 1, \ldots, m, \\ \xi_i \geq 0, \; i = 1, \ldots, m, \end{cases} \end{aligned} \tag{4.4}$$

ξ_i is a slack variable providing an estimate of the error induced by the current decision boundary; C and k are user-specified parameters indicating the penalty of function errors in control. Later, the *Lagrange multiplier* method can efficiently solve such a constrained optimization problem [31] and finds $\vec{w}$ and b for $f(\vec{x}) = \langle \vec{w} \cdot \vec{x} \rangle + b$ with a maximal margin $2/|\vec{w}|$ between $\langle \vec{w} \cdot \vec{x} \rangle + b = +1$ and $\langle \vec{w} \cdot \vec{x} \rangle + b = -1$. Figure 4.2 and 4.3 shows an

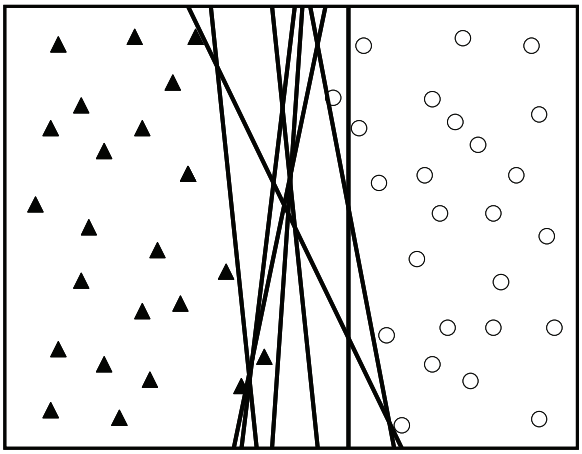


Figure 4.2: Many possible decision boundaries for a two-class data

example for a two-dimensional data set containing samples of two different classes. Figure 4.2 illustrates many possible decision boundaries to separate the data set whereas Figure 4.3 shows the one with the maximal margin and the minimal errors that the user can tolerate among all boundaries.

One SVM algorithm can be applied to regression problems with three steps: (1) *primal form* optimization, (2) *dual form* expansion, and (3) *kernel function* substitution. The primal form presents the nature of the regression whereas the dual form provides the key to the later non-linear extension using kernel functions. In our framework, ϵ-*SVR* [31] is implemented to realize a family of highly non-linear regression models $f(\vec{x}) : \vec{x} \mapsto y$ for δ^{μ}_{strike}, δ^{σ}_{strike}, δ^{μ}_{prop}, and δ^{σ}_{prop} for pulse-width mean and sigma of first-strike functions and pulse-width mean and sigma of propagation functions, respectively.

Primal Form Optimization

The regression's goal is to derive a function that minimizes slacks and meanwhile to make f as smooth as possible. The corresponding constrained optimization problem for

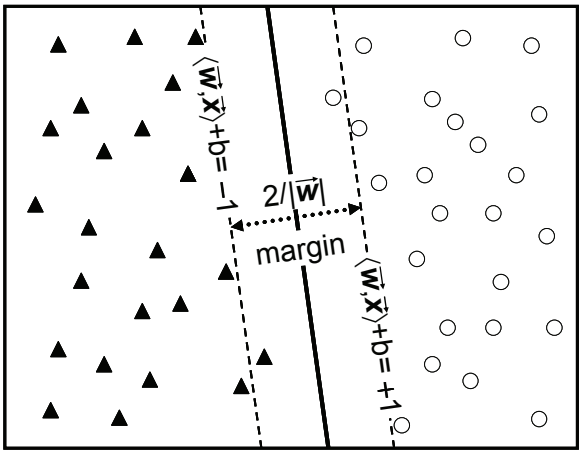


Figure 4.3: One with the maximal margin and the minimal errors that the user can tolerate among all boundaries

ϵ-*SVR* is modified as follows,

$$\begin{aligned} \text{minimize} \quad & \|\vec{w}\|^2 + C\sum_{i=1}^{m}(\xi_i^2 + \hat{\xi}_i^2) \\ \text{subject to} \quad & \begin{cases} (\langle \vec{w}\cdot\vec{x}_i\rangle + b) - y_i \le \epsilon + \xi_i, \; i = 1,\ldots,m, \\ y_i - (\langle \vec{w}\cdot\vec{x}_i\rangle + b) \le \epsilon + \hat{\xi}_i, \; i = 1,\ldots,m, \\ \xi_i, \hat{\xi}_i \ge 0, \; i = 1,\ldots,m, \end{cases} \end{aligned} \tag{4.5}$$

where the two slack variables ξ_i and $\hat{\xi}_i$ represent variations of the error exceeding and below the target value by more than ϵ, respectively. The parameter C determines the trade-off between the smoothness of $f(\vec{x}_i)$ and the variation amount of errors (ξ_i and $\hat{\xi}_i$) to be tolerated. Equation (4.5) is termed the regression's **primal form**.

Dual Form Expansion

Instead of finding $\vec{w}$ directly, the *Lagrange multiplier* method transforms the optimization problem from the primal form to its dual form and derives f as,

$$f(\vec{x}) = \sum_{i=1}^{m}(\alpha_i - \alpha_i^*)\langle \vec{x}\cdot\vec{x}_i\rangle + b \tag{4.6}$$

where α_i, α_i^* are Lagrange multipliers and b is a function of ϵ, C, α's and α^*'s [29].

Several findings can be inferred from Equation (4.6). First, the only inner product $\langle \vec{x} \cdot \vec{x}_i \rangle$ implies that only an *unseen* sample $\vec{x}$ and a *training* sample $\vec{x}_i$, are sufficient to predict a new unseen target value y. Second, only training samples $\vec{x}_i$'s that correspond to nonzero $(\alpha_i - \alpha_i^*)$'s contribute to the prediction outcome. All other samples are unnecessary for the model and are filtered out during the training process. Third, the inner product operation is a form of linear combination. As a result, the predicted target values of such a model are all linear combinations of training samples and thus f is a *linear* model. In practice, SVR often keeps only few samples (i.e., $\vec{x}_i$'s with nonzero coefficients) in its models and thus benefits from both smaller model size and faster prediction efficiency.

Kernel Function Substitution

According to the statistical learning theory [31], SVM remains valid if the inner product operation $\langle \vec{u} \cdot \vec{v} \rangle$ in Equation (4.6) is substituted by a *kernel* function $K(\vec{u}, \vec{v})$ [7]. That is,

$$f(\vec{x}) = \sum_{i=1}^{m} (\alpha_i - \alpha_i^*) K(\vec{x}, \vec{x}_i) + b \tag{4.7}$$

Radial Basis Function (RBF) is one kernel function used in our framework and can be formulated as $K(\vec{u}, \vec{v}) = e^{-\gamma \cdot \|\vec{u} - \vec{v}\|^2}$ where γ is a controlling parameter. Unlike the inner product operation, the RBF kernel is highly non-linear. This enables the SVM algorithm to produce families of non-linear models that are suitable to capture complicated behaviors, like that of generation and propagation of pulse-width distributions of transient faults.

4.2 Intensified Learning with Data Reconstruction

Although the Support-vector-regression learning provides accurate and efficient models to estimate the SER, there still remains two problems: (1) the training time for preparation data and (2) parameter search for high quality models. To resolve these two problems, traditional brute-force search cannot be effective and thus in our framework, a new meta-heuristic, particle swarm optimization (PSO), is employed to find the optimized training parameters.

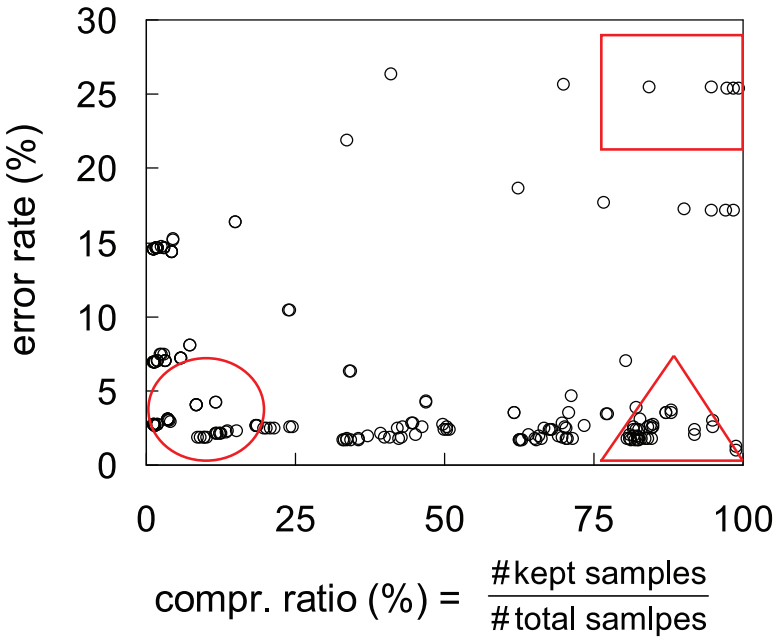


Figure 4.4: Quality comparison of 200 models using different parameter combinations

4.2.1 Parameter Search

Now we face the issue of selecting parameters (ϵ, C, γ) that have an unbounded number of combinations and is critical to achieving fine model quality. Figure 4.4 illustrate 200 models built from the same training sample set; each point represents one model using a distinct parameter combination. Their quality is measured along two coordinates: Y-axis denotes the error rate for prediction; X-axis denotes the sample compression ratio, the ratio between the number of samples kept by the model and the original size of S. Figure 4.4 shows that while it is possible to obtain an ideal model that is small and accurate (indicated by the circle), it is also possible to obtain a large and inaccurate model (indicated by the square). The differences are 20X in both axes, and there is so far no deterministic method to find the best combination.

In our framework, a meta-heuristic, particle swarm optimization (PSO), is employed to find the optimized training parameters. PSO is one of evolutionary computation techniques developed by James Kennedy and Russell Eberhart in 1995 [13]. PSO adopts a strategy to search the potential solution based on the behavior of particle swarm which is inspired by swarm intelligent of insects, such as bird flock, fish school, etc. Initially, PSO generates a

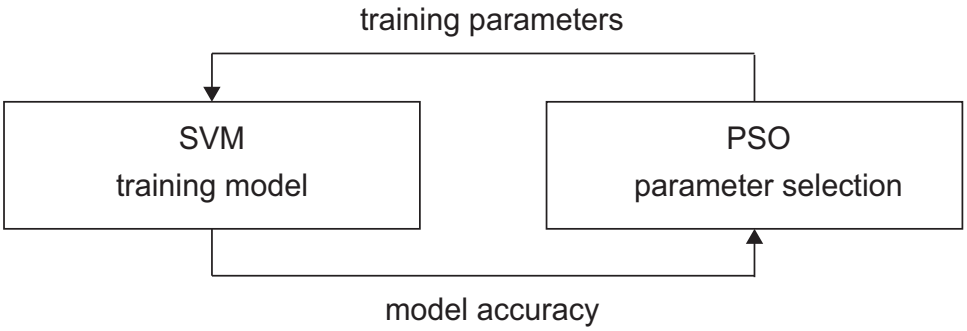


Figure 4.5: Intensified SVM learning with PSO

set of random particles in the multidimensional search space. Each particle is represented by its position and velocity, where the position indicates a possible solution of the optimization problem and the velocity is used to determine the searching direction. During each iteration, particles update their positions by tracking the best positions of all particles ($Gbest$) and their own best positions ($Pbest$). The velocity and position of particle i is manipulated by following equation:

$$\begin{aligned} V_i^{k+1} &= wV_i^k + c_1 r_1(Pbest - X_i^k) + c_2 r_2(Gbest - X_i^k) \\ X_i^{k+1} &= X_i^k + V_i^{k+1} \end{aligned} \tag{4.8}$$

,where k is the iteration index, w is the inertia weight, c_1 and c_2 are the learning factor, and r_1 and r_2 are random numbers among range [0,1].

The advantages of PSO are that it is easy to implement, requires only a few setting parameters to be adjusted and is capable of avoiding trapping at a local optimum solution more easily when compared with other evolutionary algorithms, like genetic algorithm (GA).

Figure 4.5 illustrates our intensified SVM-learning framework with the incorporating of PSO. First, PSO generates a set of training parameters for SVM to train the behavior models. After training models, the SVM reports the model accuracy to PSO as its fitness function. Based on the model accuracy, PSO will breed a new generation and generate better parameters for training again. This process iterates for a designated number of generations or until achieving stopping criteria.

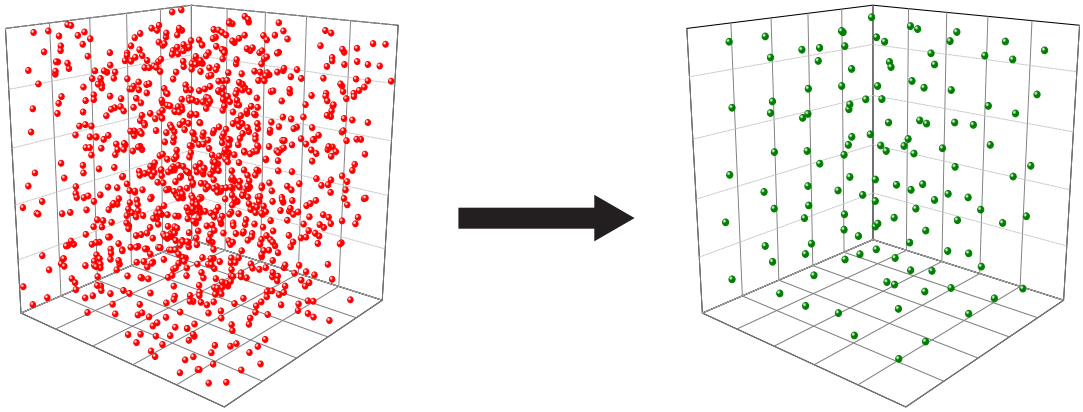


Figure 4.6: An example for data construction

4.2.2 Data Reconstruction

Although the support-vector-regression (SVR) learning provides accurate and efficient models to estimate the SER, there still remains one more problem: the training time for data preparation. A technique for data reconstruction is proposed to reduce the size of training data and then enhances the training time and the compression ratio of models. Our data reconstruction calculates the average value of training data in a user-specified range as illustrated in Figure 4.6. The red points represent the raw data generated by SPICE simulation and the average values of each block are calculated as green points as shown in Figure 4.6. After reconstructing training data, the size of training data is reduced to the number of blocks. Combining the intensified learning with data reconstruction, our framework can systematically finds a set of high quality parameter to build accuracy models. Furthermore, the training time is also reduced from the order of months to the order of hours.

4.3 Automatic Bounding-Charge Selection

However, Computing SER with full-spectrum charge collection is still challenging even after applying SVM. Hence, a technique of automatic bounding-charge selection is proposed to determine the charges that can be simply applied the traditional static analysis to

save time from statistical SER computation. In this section, a phenomenon on the variation of pulse width induced by different electrical charges is first reported. Then, the algorithm for automatic bounding-charge selection is described.

4.3.1 Variation of Pulse Width Induced by Different Charges

Figure 4.7 shows the mean, sigma, lower bound (mean-3*sigma) and upper bound (mean+3*sigma) of pulse widths which are induced by different electrical charges. Such results show that the mean of pulse widths increases rapidly as the deposited charge becomes larger. Meanwhile, a larger deposited charge also leads to a smaller sigma of its pulse width. Hence, bigger lower and upper bounds of the pulse widths can be observed along with the increasing deposited charge.

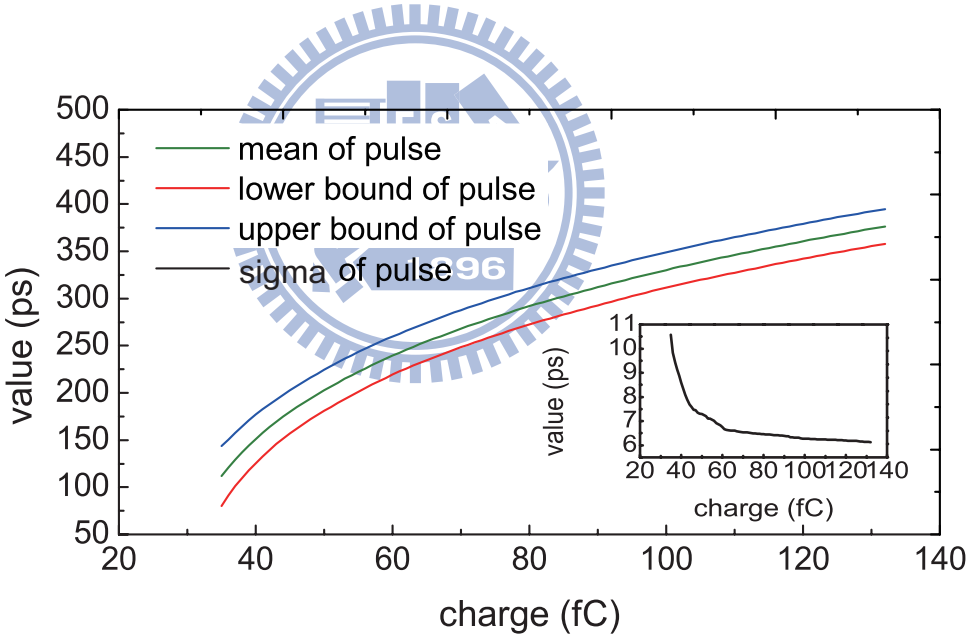


Figure 4.7: The mean, sigma, lower bound (mean-3*sigma) and upper bound (mean+3*sigma) of pulse widths which are induced by different electrical charges.

According to this phenomenon, an approach of bounding-charge selection is proposed to accelerate the overall SSER estimation. For computing overall SSER, we only need to consider the distribution of a pulse width which overlaps the latching window as illustrated

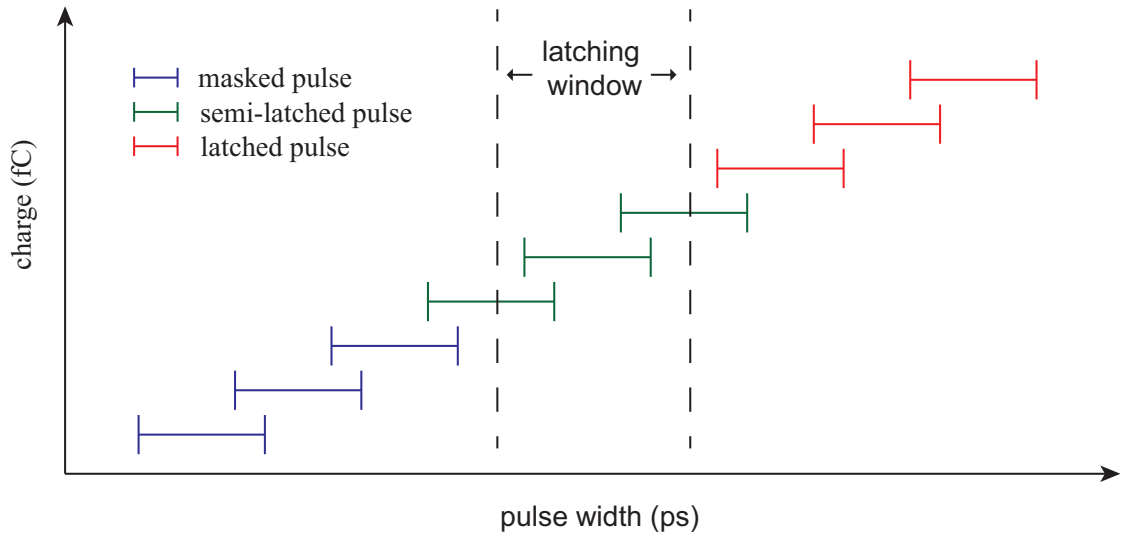


Figure 4.8: Different pulse-width distribution versus varying latching-window size

in Figure 4.8. The pulse-width distributions in blue and in red will be masked and result in SER, respectively. In other words, as the lower bound of a pulse width is bigger than the latching-window size, SER estimation for such distribution can be replaced by the corresponding static results. On the contrary, SERs for the distribution in green induced by a smaller charge will be masked completely and can be ignored when its upper bound is smaller than the latching-window size.

4.3.2 Automatic Bounding-charge Selection Algorithm

Figure 4.9 shows our algorithm for bounding-charge selection. First, a deposited charge q is picked to strike a gate in the circuit and then the algorithm computes the upper and lower bound of each latched pulses. If the upper bound of the pulse is smaller than the latching-window size, then the minimum charge (Q_{min}) is obtained. On the other hand, the maximum charge (Q_{max}) can be decided when the lower bound of the pulse width is bigger than the latching-window size. As a result, we only need to calculate the range of charges (Q_{min}, Q_{max}) to derive SERs for a gate in the circuit.

```
While Q_min and Q_max are decided
        1)  Pick a charge q
        2)  Obtain the pulses which are latched in FFs
        3)  MaxUpperBound = max(latched pulses)
        4)  MinLowerBound = max(latched pulses)
        5)  If MaxUpperBound < latching window then
                Q_min = q;
        6)  If MinLowerBound > latching window then
                Q_max = q;
End
```

Figure 4.9: Algorithm for bounding-charge selection

Chapter 5

Experimental Results

In this chapter, the accuracy of statistical cell models from the intensified learning with data reconstruction is first verified. Then, the SSER obtained by Monte-Carlo SPICE simulation is compared on four sample circuits with SSER obtained by our proposed framework. Finally, SSERs for other benchmark circuits are evaluated based on our approach. The proposed framework is implemented in C++ and run on a Linux machine with a Pentium Core Duo (2.4GHz) processor and 4GB RAM. The technology used is 45nm, Predictive Technology Model (PTM) [21] and the neuron flux rate at sea-level is assumed 56.5m^{-2}s^{-1}. In addition, the size of latching window is set to be 120ps.

Table 5.1: Model quality w.r.t. model type

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

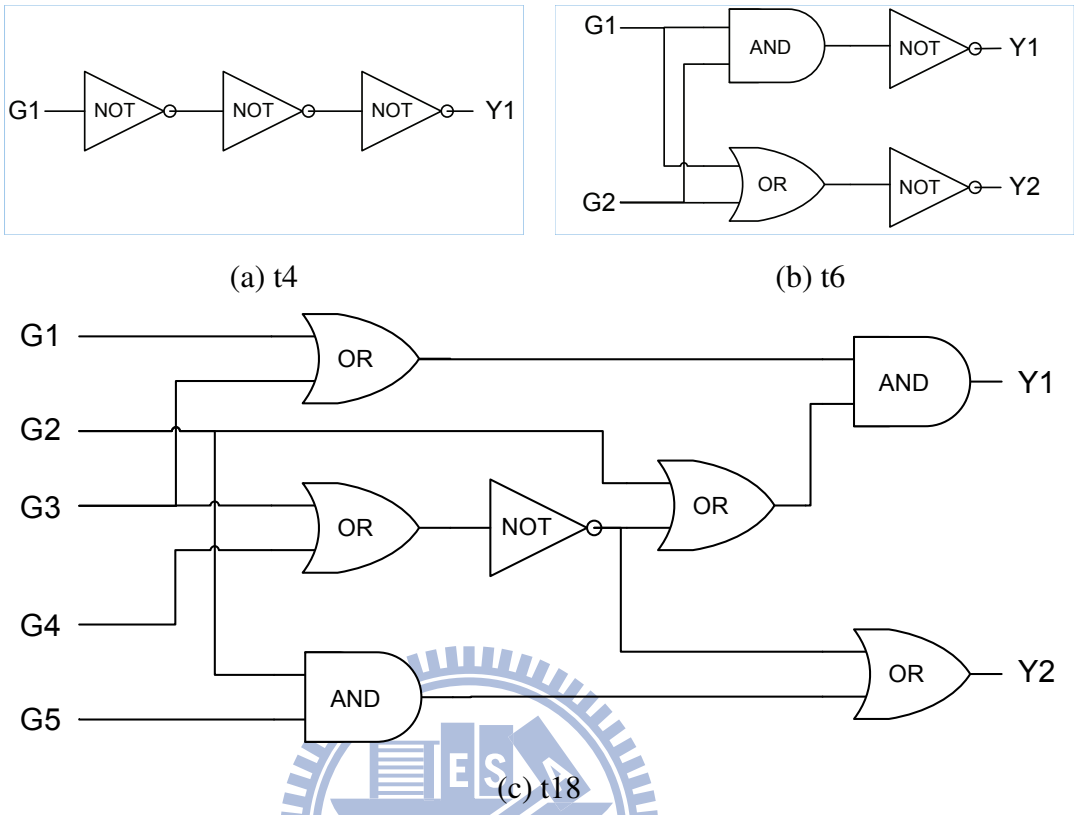


Figure 5.1: Three small circuits in our experiment

5.1 Model Accuracy

Table 5.1 shows the accuracy of built models including three types of cells and full levels of charge collection. From Table 5.1 we can find that the error rates of all models are less than 4.5%. Especially, error rates of mean values for the generated models are less than 0.45%. Such results demonstrate the effectiveness of the intensified learning and data reconstruction which collectively provide quality models for later SSER estimation.

Second, SSERs for c17 from ISCAS'85 and the other three sample circuits as shown in Figure 5.1 are derived from Monte-Carlo SPICE simulation to validate the accuracy and efficacy of our method. Considering the extremely long runtime for Monte-Carlo SPICE simulation, these four small-size circuits can only be affordable to compare SSERs on our machine. For example, the runtime of Monte-Carlo SPICE simulation for c17 with only 7

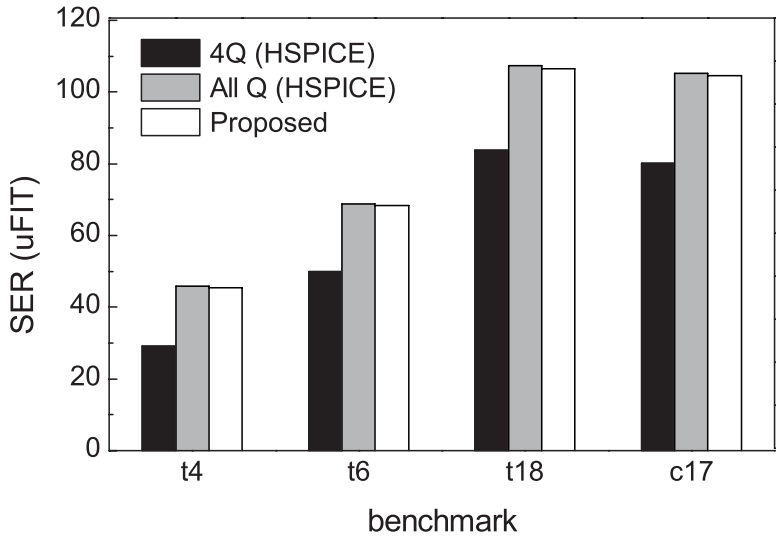


Figure 5.2: Soft error rate comparison between SPICE simulation and the proposed approach.

gates, 12 vulnerable nodes and 5 inputs takes more than three days.

Figure 5.2 compares the SSER results between SPICE simulation and our approach on four benchmark circuits. Besides, the SSER induced by only four levels of charges is also shown. Based on the results, two observations can be concluded: (1) The difference between SSER induced by four levels of charges and SSER induced by full-spectrum charges on t4, t6, t18 and c17 are 36.8%, 27.5%, 22%, and 23.9%, respectively. That manifests that SSER evaluated by four levels of charges is underestimated and not accurate enough. (2) The error rates between SPICE simulation and our proposed approach on t4, t6, t18 and c17 are 1.0%, 0.7%, 0.9%, and 0.5%, respectively. Such result represents that the our approach can yield accurate SSER with the maximum error rate as 1.0% and the average error rate as 0.8% when compared to Monte-Carlo SPICE simulation.

5.2 SER Measurement

Finally, we also apply the proposed framework to all ISCAS'85 circuits and a series of multipliers. The corresponding SSERs are shown in Table 5.2. Table 5.2 also includes information about the number of nodes (Column N_{node}), number of primary outputs (Col-

umn N_{po}), the selected charges range (Column Q_{range}), runtime using only Q_{range} (Column T_{range}), runtime using full-spectrum charges (Column T_{all}), and runtime speedup between T_{range} and T_{all} (Column Speedup). Compared with the Monte-Carlo SPICE simulation, runtime of t4, t6, t18 and c17 are less than 0.1 seconds in our framework, where the speedup is at the order of 10^7. Moreover, from Table 5.2, the levels of used charges are reduced from 98 to 10 and thus SSER estimation is accelerated at least $17.3X$ due to the automatic bounding-charge selection.

Table 5.2: Experimental results of each benchmarks

Circuit	N_{node}	N_{po}	SER (FIT)	Q_{range} (fC)	T_{range} (s)	T_{all} (s)	Speed (X)
t4	4	1	4.54E-05	(35,39)	0.002	0.06	30
t6	6	2	6.84E-05	(35,39)	0.004	0.1	25
t18	12	3	1.06E-04	(35,39)	0.012	0.3	25
c17	12	3	1.05E-04	(35,40)	0.016	0.4	25
c432	233	7	1.72E-03	(35,41)	5.6	113.9	20.3
c499	638	32	1.77E-03	(35,42)	30.1	692.3	23.0
c880	443	26	1.93E-03	(35,42)	4.3	138.0	32.1
c1355	629	32	2.24E-03	(35,42)	29.7	779.1	26.2
c1908	425	25	1.78E-03	(35,41)	15.6	307.2	19.7
c2670	841	157	3.95E-03	(35,42)	8.4	193.2	23.0
c3540	901	22	4.10E-03	(35,41)	35.5	753.7	21.2
c5315	1806	123	1.23E-02	(35,41)	30.6	628.0	20.5
c6288	2788	32	5.18E-03	(35,42)	628.3	11778.0	18.7
c7552	2114	126	5.92E-03	(35,44)	53.1	1041.1	19.6
m4	158	8	1.48E-03	(35,40)	2.0	39.2	19.6
m8	728	16	3.80E-03	(35,40)	33.3	699.2	21.0
m16	3156	32	7.92E-03	(35,42)	572.0	11656.4	20.4
m24	7234	48	1.21E-02	(35,42)	3599.3	66330.9	18.4
m32	13017	64	1.64E-02	(35,42)	9049.1	156606.9	17.3

Chapter 6

Conclusion

For accurate statistical soft error rate (SSER) analysis, full-spectrum charge collection should be considered instead of using four levels of charges only. In this thesis, we propose an accurate and efficient learning-based framework while considering full levels of charges to estimate SSER. High quality models (only 0.8% error) are built by the intensified learning and data reconstruction technique. Furthermore, an automatic bounding-charge selection is integrated into our framework and filters out charges which do not need statistical analysis to enable an average speedup of 17.3X on experimental circuits.

Statistical soft error rate (SSER) is an emerging topic. As the IC technology keeps evolving beyond deep sub-micron, we envision SSER analysis to become more critical for reliable scaled designs. A few future directions of SSER research include: (1) deriving more accurate learning models for σ_{pw}, and (2) applying SSER results to statistical circuit optimization.

Bibliography

[1] 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 Tran. Nuclear Science*, 54(6):2060–2064, Dec 2007.

[2] W. Bartlett and L. Spainhower. Commercial fault tolerance: a tale of two systems. *IEEE Tran. Dependable and Secure Computing*, 1(1):87–96, Jan-Mar 2004.

[3] D. M. Bates and D. G. Watts. *Nonlinear Regression Analysis and Its Applications.* John Wiley and Sons, 1988.

[4] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De. Parameter variations and impact on circuits and microarchitecture. In *Proc. Design Automation Conf.*, pages 338–342, Jul 2003.

[5] 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 Jour. Solid-State Circuits*, 37(2):183–190, Feb 2002.

[6] H. Cha and J. H. Patel. A logic-level model for α particle hits in cmos circuits. In *Proc. Int'l Conf. Circuit Design*, pages 538–542, Aug 1993.

[7] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2002.

[8] Paul E. Dodd and Lloyd W. Massengill. Basic mechanisms and modeling of single-event upset in digital microelectronics. *IEEE Tran. Nuclear Science*, 50(3):583–602, Jun 2003.

[9] V. Ferlet-Cavrois, P. Paillet, D. McMorrow, N. Fel, J. Baggio, S. Girard, O. Duhamel, J. S. Melinger, M. Gaillardin, J. R. Schwank, P. E. Dodd, M. R. Shaneyfelt, and J. A. Felix. New insights into single event transient propagation in chains of inverters-evidence for propagation-induced pulse broadening. *IEEE Tran. Nuclear Science*, 54(6):2338–2346, Dec 2007.

[10] L. Freeman. Critical charge calculations for a bipolar sram array. *IBM Journal of Research and Development*, 40(1), Jan-Mar 1996.

[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 Automation Conf.*, pages 918–923, Jul 2008.

[12] Joint Electron Device Engineering Council, Soild State Technology Association. *Measurement and Reporting of Alpha Particles an Terrestrial Cosmic Ray-Induced Soft Errors in Semiconductor Devices*, Aug 2001.

[13] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proc. Int'l Neural Network*, pages 1942–1948, Dec 1995.

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

[15] V. 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'l Smyp. Quality Electronic Design*, pages 831–838, 2010.

[16] N. Miskov-Zivanov and D. Marculescu. Mars-c: Modeling and reduction of soft errors in combinational circuits. In *Proc. Design Automation Conf.*, pages 767–772, Jul 2006.

[17] Natasa Miskov-Zivanov, Kai-Chiang Wu, and Diana Marculescu. Process variability-aware transient fault modeling and analysis. In *Proc. Int'l Conf. Computer Aided Design*, pages 685–690, Nov 2008.

[18] S. Mitra, N. Seifert, M. Zhang, Q. Shi, and K. S. Kim. Robust system design with built-in soft error resilience. *IEEE Tran. Computer*, 38(2):43–52, Feb 2005.

[19] K. Mohanram. Closed-form simulation and robustness models for seu-tolerant design. In *Proc. VLSI Test Symp.*, pages 327–333, May 2005.

[20] S. Mukherjee, M. Kontz, and S. Reihardt. Detailed design and evaluation of redundant multi-threading alternatives. In *Proc. Int'l Symp. Computer Architecture*, pages 99–110, May 2002.

[21] Nanoscale Integration and Modeling Group. *Predictive Technology Model*, 2008. Available at http://www.eas.asu.edu/ ptm/.

[22] S. Natarajan, M.A. Breuer, and S.K. Gupta. Process variations and their impact on circuit operation. In *Proc. Int'l Symp. Defect and Fault Tolerance in VLSI Systems*, pages 73–81, Nov 1998.

[23] M. Omana, G. Papasso, D. Rossi, and C. Metra. A model for transient fault propagation in combinational logic. In *Proc. Int'l On-Line Testing Symp.*, pages 111–115, 2003.

[24] 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'l Conf. Computer Aided Design*, pages 157–163, 2009.

[25] 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'l Conf. VLSI Design*, pages 499–502, Jan 2006.

[26] K. Ramakrishnan, R. Rajaraman, S. Suresh, N. Vijaykrishnan, Y. Xie, and M. J. Irwin. Variation impact on ser of combinational circuits. In *Proc. Int'l Smyp. Quality Electronic Design*, pages 911–916, 2007.

[27] 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 Automation and Test in Europe Conf.*, pages 164–169, March 2006.

[28] 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. Int'l Conf. Dependable Systems and Networks*, pages 389–398, 2002.

[29] A. J. Smola, B. Scholkopf, and B. S. Olkopf. A tutorial on support vector regression. Technical report, Statistics and Computing, 2003.

[30] Y. Tosaka, H. Hanata, T. Itakura, and S. Satoh. Simulation technologies for cosmic ray neutron-induced soft errors: Models and simulation systems. *IEEE Tran. Nuclear Science*, 46(3):774–780, Jun 1999.

[31] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., New York, NY, USA, 1995.

[32] S. Weisberg. *Applied Linear Regression, 3rd Edition*. John Wiley and Sons, 2005.

[33] B. Zhang, W.-S. Wang, and M. Orshansky. Faser: Fast analysis of soft error susceptibility for cell-based designs. In *Proc. Int'l Smyp. Quality Electronic Design*, pages 755–760, Mar 2006.

[34] 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'l On-Line Test Symp.*, pages 23–28, Jul 2007.

[35] M. Zhang and N. Shanbhag. A soft error rate analysis (sera) methodology. In *Proc. Int'l Conf. Computer Aided Design*, pages 111–118, Nov 2004.