Summer 2015

Reliability and security in low power circuits and systems

Hui Geng

RELIABILITY AND SECURITY IN LOW POWER CIRCUITS AND SYSTEMS

by

HUI GENG

A DISSERTATION

Presented to the Faculty of the Graduate School of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

DOCTOR OF PHILOSOPHY

in

COMPUTER ENGINEERING

2015

Approved by

Dr. Yiyu Shi, Advisor
Dr. Daryl G. Beetner
Dr. Minsu Choi
Dr. Jun Fan
Dr. Boping Wu

ABSTRACT

With the massive deployment of mobile devices in sensitive areas such as healthcare and defense, hardware reliability and security have become hot research topics in recent years. These topics, although different in definition, are usually correlated. This dissertation offers an in-depth treatment on enhancing the reliability and security of low power circuits and systems. The first part of the dissertation deals with the reliability of sub-threshold designs, which use supply voltage lower than the threshold voltage (V_{th}) of transistors to reduce power. The exponential relationship between delay and V_{th} significantly jeopardizes their reliability due to process variation induced timing violations. In order to address this problem, this dissertation proposes a novel selective body biasing scheme. In the first work, the selective body biasing problem is formulated as a linearly constrained statistical optimization model, and the adaptive filtering concept is borrowed from the signal processing community to develop an efficient solution. However, since the adaptive filtering algorithm lacks theoretical justification and guaranteed convergence rate, in the second work, a new approach based on semi-infinite programming with incremental hypercubic sampling is proposed, which demonstrates better solution quality with shorter runtime. The second work deals with the security of low power crypto-processors, equipped with Random Dynamic Voltage Scaling (RDVS), in the presence of Correlation Power Analysis (CPA) attacks. This dissertation firstly demonstrates that the resistance of RDVS to CPA can be undermined by lowering power supply voltage. Then, an alarm circuit is proposed to resist this attack. However, the alarm circuit will lead to potential denial-of-service due to noise-triggered false alarms. A non-zero sum game model is then formulated and the Nash Equilibria is analyzed.

ACKNOWLEDGMENTS

I would like to thank Dr. Yiyu Shi, my advisor and the chairman of my doctoral committee. The present work would not have been completed without his great advice and insightful guidance. It has been a pleasure working in his research group.

I would like to express my sincere gratitude to the remainder of my doctoral committee. Dr. Daryl G. Beetner and I had very fruitful discussion about the reliability of sub-threshold design, and Dr. Minsu Choi gave a lot useful suggestions to the security work in RDVS system. Dr. Jun Fan and Dr. Boping Wu have taken time to review my work and provided useful outside perspectives.

I would like to thank my parents, Jiansheng Geng and Guizhen Yu for their continuous encouragement and support. I thank my husband, Jianming Liu, for his great love, and our sweet little girl, Anya Geng Liu, who gives me a lot fun after the hard work.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

Figure Page

LIST OF TABLES

NOMENCLATURE

Symbol	Description
η	Delay Factor
C_s	Switching Load Capacitance
φ_s	Surface Potential
V_{dd}	Supply Voltage
γ	Body Biasing Coefficient
V_{th}	Threshold Voltage
V_T	Thermal Voltage
ρ	Sub-threshold Swing Parameter
V_{SB}	Body Biasing Voltage
V_{th0}	Threshold Voltage for V_{SB}=0
t	Gate Delay without V_{th} Variation and body biasing
t'	Gate Delay with V_{th} Variation
t"	Gate Delay with V_{th} Variation and body biasing
$\mathbf{A}$	Matrix with connection information of circuit
T	Target Delay of Circuit
$\mathbf{x}$	Decision Vector
ζ	Small Positive Number
$\mathbf{b}$	Dealy Improvement Vector
p	Gate Power
P_b	Gate Power with body biasing
$\mathbf{I_N}$	Unit Vector
μ	Step Size of Adaptive Filtering
λ	Lagrange Multiplier
δ	Positive Regularization Parameter

1. INTRODUCTION

With the massive deployment of mobile devices in sensitive areas such as healthcare and defense, hardware reliability and security have become hot research topics in recent years. These two topics, although different in definition, are usually correlated. This dissertation offers an in-depth treatment on enhancing the reliability and security of low power circuits and systems.

1.1. RELIABILITY

Both fabrication-time and run-time effects impact the reliability of integrated circuits (ICs). With the shrinking of feature size in complementary metal-oxide semiconductor (CMOS) technology, the fabrication-induced process variation, which may cause timing failure and impact the performance of circuits significantly, poses a major challenge for reliability of modern IC [62][63]. There are two different types of process variation, die-to-die variation and within-die variation. The die-to-die variation, resulting from wafer-to-wafer, impacts all transistors and interconnects on a die equally, and just effects the chip performance like frequency, leakage current etc. [64]. Conversely, the within die variations induces different electrical characteristics to different devices and interconnects within the same die.

Different approaches have been proposed to improve the reliability modern ICs in super-threshold designs, like Statistical Static Timing Analysis (SSTA) [65] and body biasing compensation technology [66], etc. However, there are few works on the sub-threshold designs, and this dissertation focuses on the effect of within die variation on sub-threshold designs. In fact, the effect of process variation on timing is extremely serious for sub-threshold design due to the exponential relationship between delay and V_{th}. Therefore, the first part of this dissertation focuses on solving timing problem caused by process variation in sub-threshold designs.

1.2. SECURITY

Mobile phones and computers are already the most common and basic communication tools now, and attacks to phones and computers keep increasing recently.

Considering that the phones and computers are able to access sensitive user data like bank information and photos etc., people are having higher and higher security requirements to these devices. In order to improve the security of electrical devices, both the security of software and hardware should be considered, and this dissertation focuses on the improvement of hardware security.

Side-channel attack (SCA) is a very popular hardware attack method. SCA explores the secret information by measuring the emitted outputs from physical devices, which includes execution timing, power consumptions, electromagnetic leaks, and even thermal/acoustic emanations [60]. Published SCAs include simple power analysis (SPA), differential power analysis (DPA), correlation power analysis (CPA), collision attack [61] and leakage power analysis (LPA), etc. Among them, DPA and CPA are the most popular and effective ones, which have been reviewed by numerous researchers on various crypto-systems [60].

Both DPA and CPA are power analysis attack methods, and the attackers measure the dynamic power consumption, which is dissipated during the transistors switching. The same input plaintext always results in the same power trace with a given key, thus, attackers can aggregate the small deviations between the power traces to identify the correct key. To address this issue, random dynamic voltage scaling (RDVS) has been proposed in the literature, which is demonstrated to be effective against DPA and CPA [48][49]. However, the resistance of RDVS to CPA can be undermined through lowering off-chip power supply voltage, and the second part of this dissertation focuses on addressing this problem in RDVS cryptosystem.

1.3. WORK SUMMARY

This dissertation consists of three projects.

The selective body biasing problem for post-silicon tuning in sub-threshold designs is proposed and solved with adaptive filtering algorithm in Section 2, in which, the selective body biasing problem is formulated mathematically as an integer programming problem with a statistical linear inequality constraint. In order to address this problem, the concept of adaptive filtering was borrowed from the signal processing community. A novel algorithm with binary attractors and power penalty was proposed to efficiently solve it. The experiments were based on a set of industrial designs with a 65 nm sub-threshold library. The pass rate is defined as the percentage of Monte Carlo samples without timing violations out of the 10K samples. Results show that, compared with a seemingly more intuitive approach, this approach can improve the pass rate by 57% on average with similar standby power and the same number of body biasing gates. This approach can reduce the standby power on average by 84%, with a 20% pass rate loss as compared to the approach that biases all of the gates.

Section 3 focuses on developing a more efficient algorithm to solve the proposed selective body biasing problem, which is formulated alternatively as a linear semi-infinite programming (LSIP) problem. In addition, the structure of the problem, associated with the physical meaning of the design, can lead to a novel Incremental Hypercubic Sampling (IHCS) algorithm. The algorithm solves the LSIP problem through a number of finite mixed-integer linear programming, and many nice properties of the algorithm are obtained through rigid mathematical derivations.

Though RDVS technology is a very efficient method to improve the resistance of cryptosystem against SCA, it is demonstrated that the resistance can be undermined by providing lower off-chip power supply voltage in Section 4. In order to address this issue, the following method is proposed in this dissertation. The off-chip power supply voltage will be monitored, and an alarm could be triggered to protect valued information once the power supply voltage is lower than the expected voltage (threshold voltage). However, considering both maintenance cost and the environment noise on power supply voltage, this problem is formulated as a non-zero sum game model, and the attacker and the circuit supplier (defender) are the players of this game. The analysis of the Nash equilibrium in this game shed light on the choice of the optimal threshold voltage, which is based on parameters of cryptosystem including the value of information, denial-of-service cost, etc.

Finally, Section 5 discusses the contribution of this dissertation and suggestions of future works.

2. PART 1: SELECTIVE BODY BIASING FOR POST-SILICON TUNING OF SUB-THRESHOLD DESIGNS: AN ADAPTIVE FILTERING

2.1. LITERATURE REVIEW

Power consumption has become an important design consideration for modern circuits, which is particularly true for energy constrained applications (e.g., battery-powered systems and wide-area surveillance) [1-4]. Sub-threshold designs (in which transistors work in a weak-inversion region under lower-than-threshold supply voltages) have emerged as a compelling solution for low power circuits and systems. The power consumption of sub-threshold circuits is expected to be within a range of 1pJ/instruction [4].

Despite significant power reductions, sub-threshold designs are subject to several challenges. The most critical challenge faced is related to predictability in the sub-threshold regime, where the V_{th} variation is the dominant variation source. The threshold variation is typically induced by random dopant fluctuations (RDFs). The placement and number of dopant atoms in the device channel cause random shifts in V_{th} [3, 4]. The RDF may increase path delays dramatically given the exponential relationship between V_{th} and the drive current in the sub-threshold designs. Gate delay variations can be as high as 300% of the nominal values [4]. Meanwhile, the V_{th} variation created by the RDF for each gate is purely random. Thus, the critical paths in different chips of the same design are likely to be completely different after the fabrication is complete. Therefore, obtaining a timing closure at the design time is extremely difficult. The post-silicon or runtime tuning techniques are imperative.

Body biasing is one of the most effective tuning techniques currently available. This technique uses body effect to modulate the V_{th} of transistors, and thus changes the circuit’s power and performance. A total of three different types of body biasing technologies can be used: forward body biasing (FBB), reverse body biasing (RBB), and bidirectional adaptive body biasing (ABB). Forward body biasing is typically used to speed up the circuit during the active mode. RBB is an effective technique that can be used to decrease the leakage power in standby mode. Finally, adaptive body biasing can be adjusted into either FBB or RBB at the cost of control and measurement circuitry [7]. However, many existing works on body biasing assumed super-threshold designs either explicitly or implicitly [7-11] [27-29], and they cannot be extended directly to the sub-threshold regime, primarily as a result of the exponential dependency of delay on the purely random V_{th} variation. The details are discussed in Section II.

Bo Zhai [4] derived a statistical model of the sub-threshold circuit delay and proposed the reduction of energy consumption by optimizing the pipeline’s depth. However, it assumes that path delays are independent random variables with lognormal distributions, and that the maximum of two lognormal random variables is still lognormal. Although the gate delays are independent, the path delays are possibly dependent because different paths may share gates.

Bo Liu, etc. in [6] were the first to examine the possibility of body biasing in sub-threshold designs. They suggested that both gate resizing and a fuzzy logic controller could be used to reduce the effect of process variation. However, the additional fuzzy logic controller involves extra dynamic power. In addition, in that approach body biasing is applied to all of the gates. That introduces large power and area overhead.

Nearly all of the body biasing studies conducted [6-11] assumed the presence of multiple body biasing voltage domains. These domains increase not only the routing but also the control complexity, which makes these schemes hard to be employed in large-volume commercial production. Multiple body biasing schemes also suffer from less latch-up immunity, less threshold voltage controllability, more substrate noise vulnerability and less gate oxide reliability. Considering the relative importance of reliability and cost, it is more practical to explore a scheme that uses only one body biasing voltage domain, preferably with binary control decision, i.e., whether a single body biasing voltage should be applied or not.

Towards this end, this dissertation puts forward a novel body biasing scheme for sub-threshold designs which tries to find a number of selective gates for body biasing so that the timing violation caused by random V_{th} variation is eliminated statistically with minimum standby power overhead. This scheme applies the same FBB voltage to all of the selected transistors so that the design cost can be reduced. The proposed algorithm is an offline algorithm, which selects the set of gates for body biasing at design-time. During post-silicon tuning, the biasing voltage is simply applied to the gates selected.

Simply assuming the worst case for all the gates will not be an option here, as the timing benefit brought by body biasing for a single gate in the sub-threshold regime can barely cancel its own delay increment in the worst-case scenario. Accordingly, for worst-case scenario, all the gates have to be selected as body biasing gates. On the other hand, the probability that all the gates are in the worst-case corner at the same time is extremely low. It is thus meaningful to explore an alternative statistical solution.

The remainder of this section is organized as follows. Both the background and motivation behind this study are summarized in Section 2.2. The timing model and problem formulation are introduced in Section 2.3. Section 2.4 presents the proposed adaptive filtering algorithm. Experimental results are presented in Section 2.5 and concluding remarks are given in Section 2.6.

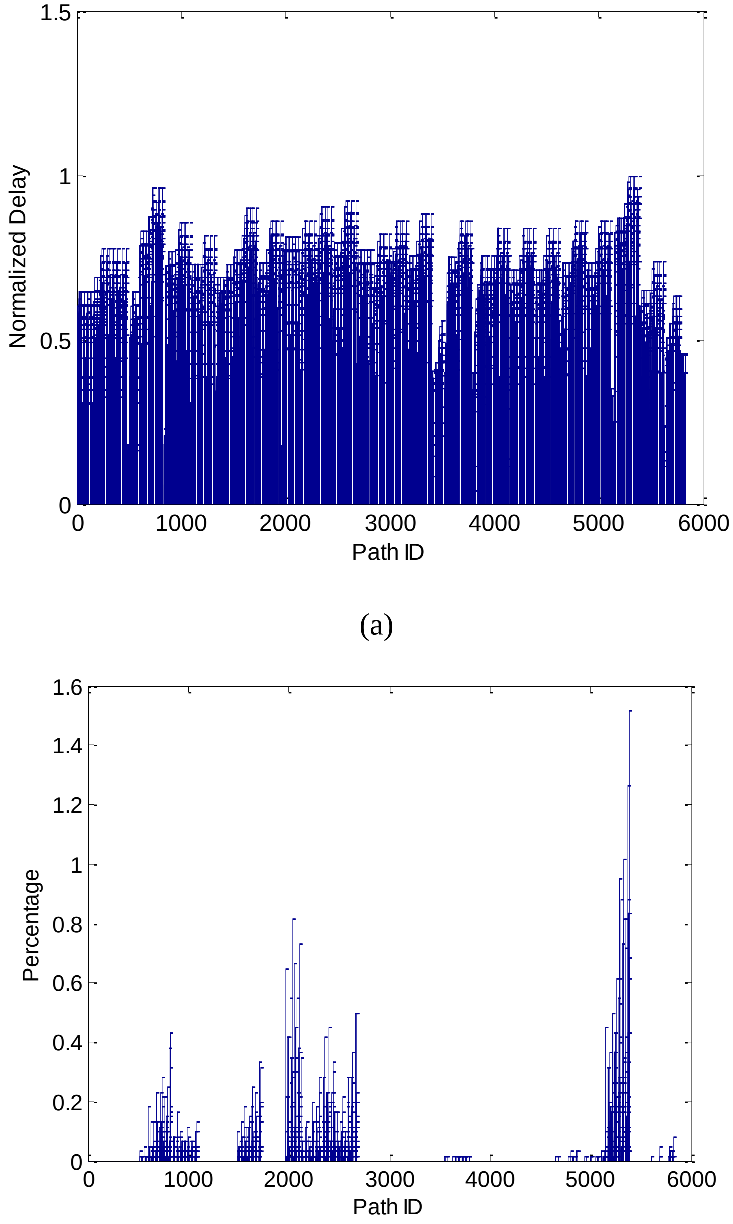
2.2. BACKGROUND AND MOTIVATION

The biggest challenge in sub-threshold designs is the unpredictability brought by the exponential dependence of delay variation on purely random threshold voltage variations.

A 65 nm industrial sub-threshold design was used as an example to illustrate this challenge. The normalized path delay, without variation, is given in Figure 2.1 (a). This design was processed through timing optimization, and the critical path ID was 5436. Meanwhile, 10K Monte Carlo simulations were run for $\sigma/\mu = 10\%$ V_{th} variation. The percentage that each path can become critical was calculated as in Figure 2.1 (b).

The highest possibility at which a path can become critical is approximately 1.46%, which means that many other paths may become critical. Accordingly, it is no longer possible to design for timing closure based on a selected number of critical paths, and post-silicon tuning is imperative to address the problem.

A number of researchers have proposed the use of body biasing for post-silicon tuning, but for super-threshold designs [7-11] [27-29]. Most used ABB technology, which can improve the performance of slow modules to fix the timing problem. It can also reduce the leakage power of fast modules. Additional circuitry is, however, needed for either runtime monitoring or voltage adjustments, increasing the area over-head.



(a)

(b)

Figure 2.1. Normalized path delay and critical path distribution: (a) normalized path delay without V_{th} variation, and (b) critical path distribution with V_{th} variation in a 65 nm sub-threshold design

A test chip with an on-chip body biasing generator was implemented to demonstrate the effectiveness of bidirectional ABB [7]. Critical paths were replicated as part of an ABB control circuit to generate one or multiply body biasing voltages. However, it assumes that a small set of critical paths can be identified. This assumption is incorrect for sub-threshold designs, as illustrated in Figure 2.1.

A lookup table based ABB scheme was proposed to control multiple functional blocks on a single chip independently. Different body biasing voltages were stored in a look-up table. Sensors were used to detect the temperature variation, and then suitable body biasing voltages were obtained for corresponding blocks [10]. One disadvantage associated with this method is the significant area overhead of additional circuitry. Another is the large number of delay and leakage measurements required to obtain the final solution. Though the authors tried to decrease the requirement of delay and leakage samples by using a polynomial timing model, it only worked for super-threshold designs and could not be extended to the sub-threshold region.

A joint co-optimization workflow was proposed to minimize the leakage power, under a given delay constraint, at a given yield in [27]. It included design-time gate-level sizing and post-silicon ABB adaptation. A robust adjustable linear program algorithm was given to solve the problem. Extending this solution to a sub-threshold design is difficult, because this optimization problem requires a linearized delay model (which is difficult to obtain in the sub-threshold region). In contrast, for multiple optimal body biasing voltages, additional body biasing voltage generators are required to implement the biasing voltages. This not only further increases the area overhead to the body biasing gate area and routing cost, but also suffers from less latch-up immunity, less threshold voltage controllability, more vulnerable to substrate noise and less gate oxide reliability.

One of the most common problems associated with ABB is the large area overhead when body biasing is provided to all of the gates. Considering multiple body biasing voltages are needed in ABB, almost all the ABB related papers use clustering methods to solve this problem. However, most cluster technologies are based on blocks that utilize the spatial correlation of process parameters. [9] [28] proposed the use of variability-aware technique to cluster gates using distributions and correlations of gate delays. This technique assumes that the relationship between the delay and the variation source is linear. It also assumes that the variations at different gates are spatially correlated. Neither of these, however, hold true in sub-threshold designs.

Several researchers have tried to provide body biasing to part of gates to decrease area overhead. [8] tried to minimize the standby power by selecting a subset of rows from a row-based standard cell layout. Body biasing was applied to those selected rows that contained the most timing critical gates. The timing constraints, however, were deterministic. This does not fit the strong log-normal random variations in sub-threshold designs.

To sum up, most of the previous papers on body biasing only work for super-threshold. They either assume a small set of critical paths, or utilize a linearized delay model and spatially variation information which do not hold in sub-threshold designs. Meanwhile, there exists huge routing area and standby power overhead for multiple body biasing voltages which prohibits them from being used in practical industrial application.

In order to address the above issues, this paper proposes to only apply one post-silicon body biasing voltage to a limited number of selected gates in order to optimize the standby power overhead. Both the reduced routing area cost and improved reliability are preferable in practice, because only one body biasing voltage is utilized.

2.3. MODELING AND PROBLEM FORMULATION

As mentioned in the introduction section, in order to reduce as much standby power overhead as possible for sub-threshold designs, a minimal set of gates needs to be identified. Thus, the timing violations can be statistically eliminated by providing body biasing to this set. The following is divided into three subsections. The timing model is defined in the first section, and the problem is formulated mathematically in the second. The third subsection contains an example that illustrates this formulation.

2.3.1. Models and Assumptions. As will be shown later, a large number of delay samples under the random V_{th} variation is needed to train the adaptive algorithm to be proposed. The time-consuming Monte Carlo simulation is not a viable option here. An analytic delay model that is less accurate but much more efficient is thus used.

The gate delay is assumed to follow the sub-threshold statistical model [4-5] [12-13]:

$$t_d = \frac{1}{2}\eta C_s V_{dd} \frac{1}{I_{s0}} e^{-\frac{V_{dd}}{\rho V_T}} e^{\frac{V_{th}}{\rho V_T}} = \tau e^{\frac{V_{th}}{\rho V_T}}, \qquad (2.1)$$

where η is the delay factor and C_s is the switching load capacitance. V_{dd} is the supply voltage and V_{th} is the threshold voltage. V_T is the thermal voltage and ρ is the sub¬-threshold swing parameter. The constant τ can be obtained by using HSPICE to measure the gate delay at typical V_{th}.

In addition, the threshold with body biasing can be calculated based on the equation

$$V_{th} = V_{th0} + \gamma(\sqrt{|-2\varphi_s + V_{SB}|} - \sqrt{|-2\varphi_s|}), \qquad (2.2)$$

where γ (the body biasing coefficient) and φ_s (the surface potential) are obtained from the technology library [12]. V_{SB} is the body biasing voltage, and V_{th0} (the threshold voltage for $V_{SB} = 0$) is a random variable due to the process variation.

The following assumptions are also made to hold for sub-threshold designs:

- The routing of body biasing nets has a minimal impact on signal nets, and thus, signal net's wire delay remains constant.
- Most of paths in the sub-threshold design can become critical paths. This is due to purely random V_{th} variation and the exponential relationship between delay and V_{th} variation. The example given in Section 2.2 (Figure 2.1 (b)) was used to demonstrate this assumption.
- Only one body-biasing domain will be used (i.e., all of the gates selected for body biasing will use the same body voltage, and only one user-specified biasing voltage is available). This assumption is made to significantly reduce the hardware cost associated with body biasing.

The last two assumptions make this work unique. The system to be identified is sparse (considering a very limited number of gates) and binary; there are only two controlling options for each gate: using body biasing or not. This permits the formulation of the problem in a very interesting way as given below.

2.3.2. Problem Formulation. Without loss of generality, only combinational logic is considered in the following discussion, because for sequential logic, the combinational logic can be checked between each pair of the registers separately.

For a design with a total of N gates and M paths, a matrix $\mathbf{A}$ with M rows and N columns is defined to represent the circuit connection in which each row stands for a path and each column stands for a gate. If a gate G_n appears in path P_m, the element at m-th row and n-th column of the matrix A (i.e. $\mathbf{A}_{m,n}$) will be set to 1, otherwise it is set to 0. Three $N{\times}1$ vectors are defined as $\mathbf{t}$, $\mathbf{t'}$, and $\mathbf{t''}$, in which each element stands for a gate's t , t' and t'' delay in different scenarios:

- t is the gate delay without V_{th} variation or body biasing,
- t' is the gate delay with only V_{th} variation,.
- t'' is the gate delay with both V_{th} variation and body biasing.

Thus, t is the ideal gate delay with no process variation and t' is the gate delay sample with random normal distributed V_{th} variation that may cause timing violations. In theory, since V_{th} could deviate from its nominal value in each direction, t' may be either longer or smaller than t. In order to address the possible timing violation, some gates will be selected for body biasing, and their gate delay after body biasing is t''.

With the definitions above, when there are no variations or body biasing, the maximum delay, T for this circuit could be obtained by

$$T = \max\left(\mathbf{At}\right). \tag{2.3}$$

The circuit is assumed to have passed the timing sign-off when there is no process variation. Accordingly, this delay T will be taken as the target of the post-silicon body biasing tuning. The delay matrix $\mathbf{A}_{\mathbf{t}'}$ ($M \times N$) when there is V_{th} variation can be calculated as

$$\mathbf{A}_{t'} = \mathbf{A}diag\left\{\mathbf{t}'\right\}, \tag{2.4}$$

where $diag\left\{\mathbf{t}'\right\}$ is a $N \times N$ diagonal matrix with diagonal vector $\mathbf{t'}$. Similarly, the delay matrix $\mathbf{A}_{t''}$ ($M \times N$) when there are both V_{th} variation and body biasing is

$$\mathbf{A}_{t''} = \mathbf{A}diag\left\{\mathbf{t}''\right\}. \tag{2.5}$$

Therefore, when body biasing is applied, the delay improvement matrix $\mathbf{A}_{tt'}$ ($M \times N$) is

$$\mathbf{A}_{tt'} = \mathbf{A}_{t'} - \mathbf{A}_{t''} = \mathbf{A}diag\left\{\mathbf{t}' - \mathbf{t}''\right\}. \tag{2.6}$$

The overall required delay improvement vector $\mathbf{b}$ ($M \times 1$) required to fulfill the target delay is

$$\mathbf{b} = \mathbf{At}' - T\mathbf{I}_M, \tag{2.7}$$

where $\mathbf{I}_M$ is a unit vector ($M \times 1$).

A decision vector $\mathbf{x}$ ($N \times 1$) is defined, in which each element represents the decision of body biasing control on that gate, i.e., 1 means applying body biasing to that gate and 0 means no body biasing. The proper $\mathbf{x}$ must be chosen to ensure the overall delay improvement for every path is larger than the required delay improvement, thereby fulfilling the timing constraint,

$$\text{prob}\left\{\mathbf{A}_{tt'}\mathbf{x} \geq \mathbf{b}\right\} > 1\text{-}\zeta, \tag{2.8}$$

where ζ is a small positive number and $\mathbf{A}_{tt'}\mathbf{x} \geq \mathbf{b}$ represents a component-wise inequality (i.e., $\mathbf{a}_{m,tt'}\mathbf{x} \geq b_m, m = 1...M$, where $\mathbf{a}_{m,tt'}$ is the m^{th} row of $\mathbf{A}_{tt'}$, and b_m is the m^{th} element of $\mathbf{b}$).

When the delay is modelled in a statistical way, this is a robust linear programming problem which could be considered in a statistical framework [15]. However, due to the fact that the delay is log-normal and the paths are not independent with each other, it is not tractable since the joint distribution is very difficult to obtain. Therefore, the approach in [15] cannot be used, and (2.8) can be further converted to a minimization problem using the logarithmic barrier function and solve it using adaptive filtering based algorithm later in Section 2.3.

Additional two $1 \times N$ vectors are defined as $\mathbf{p}$, and $\mathbf{p}_b$, in which each element stands for G_n gate's power p_n without body biasing, and power $p_{b,n}$ with body biasing. Therefore, for $\mathbf{x}$, the total power P will be

$$\begin{aligned} P &= \mathbf{p}_b\mathbf{x} + \mathbf{p}\left(\mathbf{I}_N - \mathbf{x}\right) \\ &= \mathbf{p}\mathbf{I}_N + \left(\mathbf{p}_b - \mathbf{p}\right)\mathbf{x}, \end{aligned} \tag{2.9}$$

where $\mathbf{I}_N$ is a unit vector ($N \times 1$).

In order to decrease the power, it is proposed to solve the following optimization problem

$$\begin{aligned} &\boldsymbol{min} && \left(\mathbf{p}_b - \mathbf{p}\right)\mathbf{x} \\ &\text{subject to} && \text{prob}\left\{\mathbf{A}_{tt'}\mathbf{x} \geq \mathbf{b}\right\} > 1\text{-}\zeta. \end{aligned} \tag{2.10}$$

where $\left(\mathbf{p}_b - \mathbf{p}\right)\mathbf{x}$ is used as the optimization target. Considering that $\mathbf{p}\mathbf{I}_N$ is constant, it is the same as $\mathbf{p}\mathbf{I}_N + \left(\mathbf{p}_b - \mathbf{p}\right)\mathbf{x}$. It should be noted that the integer set in (2.10) is non-convex and the coefficients of the constraint are statistical. Accordingly, the problem is NP hard. A novel adaptive filtering algorithm will be proposed to solve it below. Before that, an example to illustrate the above modeling process is first given in sub-section 2.2.3 as below.

2.3.3. An Example. Taking the circuit T0 as an example, there are a total of six 2-input NAND gates identified by number G1 - G6 respectively in network T0, shown in Figure 2.2. There are 11 paths in total. If the same worst-case gate delay is used for any path going through a gate (regardless of the input pin difference), the path number can be further reduced to 7, as shown in Table 2.1.

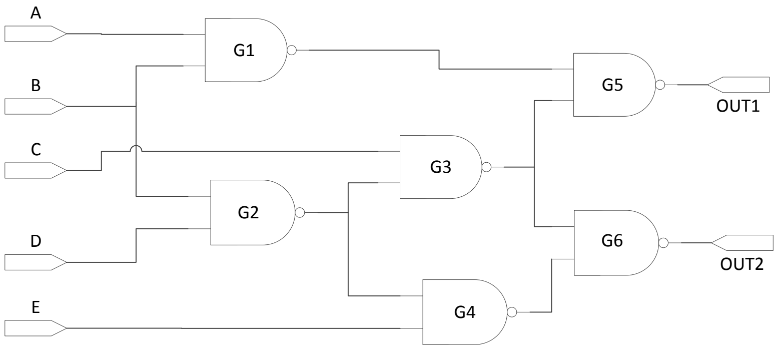


Figure 2.2. Network of T0.

Table 2.1 Paths of T0.

Path ID	**Gates In The Path**
P1	G1,G5
P2	G2,G3,G5
P3	G2,G3,G6
P4	G2,G4,G6
P5	G3,G5
P6	G3,G6
P7	G4,G6

According to TABLE I, the matrix $\mathbf{A}$ of T0 is

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}. \tag{2.11}$$

And the $\mathbf{t}$ vector (6×1) can be express as

$$\mathbf{t} = \begin{bmatrix} t_1 & t_2 & t_3 & t_4 & t_5 & t_6 \end{bmatrix}^T, \tag{2.12}$$

where t_n ($n=1,2,...,6$) is the gate delay without process variation or body biasing for gate number n. Considering the longest path includes 3 NAND gates, the target delay T is

$$\begin{aligned} T &= \max\left(\mathbf{At}\right) \\ &= \max\{t_2+t_3+t_5, t_2+t_3+t_6, t_2+t_4+t_6\}. \end{aligned} \tag{2.13}$$

The $\mathbf{t}'$ and $\mathbf{t}''$ vector could be obtained similarly, in which $t_n^{'}$ and $t_n^{"}$ means the t' delay with process variation only and $t"$ the delay with both process variation and body biasing for gate number n. The delay improvement matrix $\mathbf{A}_{tt'}$ (7×6) of T0 can then be expressed as

$$\begin{bmatrix} t'_1-t"_1 & 0 & 0 & 0 & t'_5-t"_5 & 0 \\ 0 & t'_2-t"_2 & t'_3-t"_3 & 0 & t'_5-t"_5 & 0 \\ 0 & t'_2-t"_2 & t'_3-t"_3 & 0 & 0 & t'_6-t"_6 \\ 0 & t'_2-t"_2 & 0 & t'_4-t"_4 & 0 & t'_6-t"_6 \\ 0 & 0 & t'_3-t"_3 & 0 & t'_5-t"_5 & 0 \\ 0 & 0 & t'_3-t"_3 & 0 & 0 & t'_7-t"_7 \\ 0 & 0 & 0 & t'_4-t"_4 & 0 & t'_7-t"_7 \end{bmatrix}. \tag{2.14}$$

Then the required delay improvement vector $\mathbf{b}$ (7×1) is

$$\begin{aligned} \mathbf{b} = [t'_1+t'_5-T \quad t'_2+t'_3+t'_5-T \quad t'_2+t'_3+t'_6-T \\ t'_2+t'_4+t'_6-T \quad t'_3+t'_5-T \quad t'_3+t'_6-T \quad t'_4+t'_6-T]^T. \end{aligned} \tag{2.15}$$

Note that both t' and $t"$ for each gate are independent lognormal random variables since the V$_{th}$ variations follow independent Gaussian distributions.

2.4. ALGORITHM

In this section, a novel adaptive filtering based framework is proposed to solve the problem formulated in (2.10), along with its complexity analysis. The framework overview is illustrated in Figure 2.3, which includes three steps: data pre-processing, representative path selection, and APA-BA-PP.

The pre-processing step is based on the gate level Verilog file. Both path and gate information are extracted first. The sub-threshold timing model is then built through HSPICE simulation. The matrices and vectors are prepared last as described in Section 2.2.

A representative path selection stage is employed because the path number can have an exponential relationship with the gate number. It selects a set of representative paths, the number of which does not exceed the number of gates. The details are described in Section 2.3.1.

Finally, the APA-BA-PP algorithm actually selects the gates for body biasing and is the most important part of the framework. The details are described in Section 2.3.2.

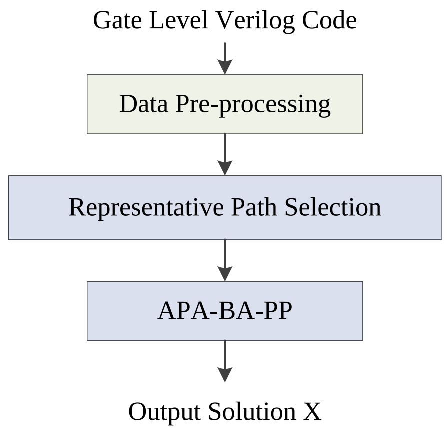


Figure 2.3. Adaptive filitering framework overview

2.4.1. Representative Path Selection. This step suppresses the potentially huge path number. A similar approach as described in [14] [26] is employed. To illustrate the idea, circuit T0 is used again as an example. Its 7 paths are expressed in the graph in Figure 2.4. Based on this graph, the delay of path P2: G2→G3→G5 and P4: G2→G4→G6 can be rewritten accurately as the linear combination of the remaining paths. For example, $P2 = P5 + P3 - P6$, therefore, P2 can be well represented by P5, P3, and P6.

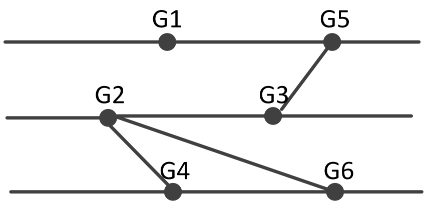


Figure 2.4. Network of T0.

In general, the representative path selection algorithm can find r paths, the linear combination of which can represent the delay of other paths. This will reduce the dimension of $\mathbf{A}$ to $r \times N$, in which $r = rank(\mathbf{A})$. Since $rank(\mathbf{A})$ will not exceed its column number which in turn is equal to the gate number, the number of representative paths selected will not exceed the gate number. The details are as follows.

To select these representative paths, singular value decomposition (SVD) is performed on matrix $\mathbf{A}$ firstly,

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Delta}\mathbf{V}^T, \tag{2.16}$$

and then apply QR decomposition using column pivoting on $\mathbf{U}_r$, which is a sub-matrix formed by the first r columns of $\mathbf{U}$

$$\mathbf{U}_r^T\mathbf{P}_r = \mathbf{QR}. \tag{2.17}$$

The matrices $\mathbf{Q}$ and $\mathbf{R}$ are found during the procedure and help identify the output permutation matrix $\mathbf{P}_r$. After obtaining $\mathbf{P}_r$, to identify the r representative paths, $\mathbf{P}_r^T\mathbf{A}$ is computed and the sub-matrix formed by the first r rows is taken as $\mathbf{A}_r$ which in turn corresponds to r paths.

It should be noted that considering the delay is modelled as log-normal distribution, it is not guaranteed that the other non-presentative paths can also satisfy the timing constraints. Even for the representative paths, it is not guaranteed that they can satisfy the delay constraints for every possible sample of the log-normal distributions. However, the use of representative paths is still helpful to conquer the complexity issue and to highlight the essential circuit structure information.

The SVD and QR decomposition in this algorithm are only a one-time cost for each design. Sophisticated algorithms are available for solving these procedures. Furthermore, because matrix $\mathbf{A}$ is very sparse, the runtime of SVD could be further reduced through exploiting this sparsity. The analysis of the computation complexity will be detailed in Section 2.3.3.

2.4.2. APA-BA-PP Algorithm. After reducing $\mathbf{A}$ to $\mathbf{A}_r$, the formulation in (2.10) can be simplified as:

$$\boldsymbol{min} \qquad (\mathbf{p}_b - \mathbf{p})\mathbf{x} \tag{2.18}$$

$$\text{subject to} \qquad \text{prob}\{\mathbf{A}_{tt'}\mathbf{x} \geq \mathbf{b}\} > 1\text{-}\zeta, \tag{2.19}$$

$$\begin{aligned}&\mathbf{x} \in \{0,1\},\\ &m = 1...r.\end{aligned} \tag{2.20}$$

The formulated problem in (2.18)-(2.20) contains three parts: the inequality timing requirement (2.19), the integer set constraint (2.20) and the minimization target of power cost (2.18). In this subsection, a novel algorithm is proposed to solve this problem: the inequality timing constraint (2.19) is first transformed to a logarithmic barrier optimization target; the power minimization target (2.18) is then added in the cost function as power penalty, and the integer set constraint (2.20) converted as the binary attractor in the cost function. The details will be elaborated below.

At first, in order to handle the inequality constraint, the logarithmic barrier function [14] is utilized to convert the inequality constraint to the following unconstrained optimization

$$\boldsymbol{min} \qquad \text{-log}\sum_{m=1}^{r}\left(\mathbf{a}_{\text{m},tt'}^{T}\mathbf{x} - b_m\right). \tag{2.21}$$

This problem is still challenging as the coefficients are all statistical. Towards this, the adaptive filtering concept could be borrowed from the signal processing community to aid in developing a novel algorithm to solve it.

Considering that the delay improvement matrix $\mathbf{A}_{tt'}$ and the required delay improvement vector $\mathbf{b}$ are all lognormal distributions, following the way [15] handles the impulse noise, a constraint on the update of the adaptive filter coefficients is added:

$$\begin{aligned}&\boldsymbol{min} \qquad \text{-log}\sum_{m=1}^{r}\left(\mathbf{a}_{\text{m},tt'}^{T}(k)\mathbf{x}(k) - b_m(k)\right)\\ &\text{subject to} \qquad \left\|\mathbf{x}(k) - \mathbf{x}(k-1)\right\|_2^2 \le \mu^2,\end{aligned} \tag{2.22}$$

where k is the sampling time index for the different random lognormal distributions, and μ is a parameter ensuring that $\mathbf{x}(k)$ and $\mathbf{x}(k\text{-}1)$ does not change dramatically. This can also be viewed as the minimum disturbance constraint. The variable μ controls the convergence level of the algorithm and it should be as small as possible. The a-posterior error is

$$\varepsilon_m(k) = b_m(k) - \mathbf{a}_{m,tt'}^{T}(k)\mathbf{x}(k), m = 1...r. \tag{2.23}$$

Considering that the optimization target in (2.22) is convex, the method of Lagrange multipliers could be used, and the unconstrained cost function of (2.22) can be obtained as

$$\begin{aligned} \mathrm{J}\left(\mathbf{x}(k)\right) = &\text{-}\log\sum_{m=1}^{r}\left(-\varepsilon_m(k)\right) \\ &+\lambda\left[\left\|\mathbf{x}(k)-\mathbf{x}(k-1)\right\|_2^2-\mu^2\right], \end{aligned} \tag{2.24}$$

where λ is the Lagrange multiplier. The derivative of the cost function with respect to the vector **x(k)** is

$$\begin{aligned} \frac{\partial J\left(\mathbf{x}(k)\right)}{\partial\mathbf{x}(k)} = &\text{-}\frac{\mathbf{A}_{tt'}^{T}(k)\mathbf{I}}{\sum_{m=1}^{r}\left(-\varepsilon_m(k)\right)} \\ &+2\lambda\left[\mathbf{x}(k)-\mathbf{x}(k-1)\right]. \end{aligned} \tag{2.25}$$

Setting the derivative to zero, (2.26) is gotten as

$$\mathbf{x}(k)=\mathbf{x}(k-1)+\frac{1}{2\lambda}\frac{\mathbf{A}_{tt'}^{T}(k)\mathbf{I}}{\sum_{m=1}^{r}\left(-\varepsilon_m(k)\right)}. \tag{2.26}$$

Substituting (2.26) into the constraint (2.22), (2.27) is obtained

$$\frac{1}{2\lambda}=\frac{\mu\sum_{m=1}^{r}\left(-\varepsilon_m(k)\right)}{\sqrt{\mathbf{I}^T\mathbf{A}_{tt'}(k)\mathbf{A}_{tt'}^{T}(k)\mathbf{I}}}. \tag{2.27}$$

Substituting (2.27) into (2.26), the constant $\sum_{m=1}^{r}\left(-\varepsilon_m(k)\right)$ can be canceled, and the update equation for the $\mathbf{x}$ vector is:

$$\mathbf{x}(k)=\mathbf{x}(k-1)+\frac{\mu\mathbf{A}_{tt'}^{T}(k)\mathbf{I}}{\sqrt{\mathbf{I}^T\mathbf{A}_{tt'}(k)\mathbf{A}_{tt'}^{T}(k)\mathbf{I}}}. \tag{2.28}$$

In practice, a small positive regularization parameter δ is added in the denominator of the second term to prevent the divide-by-zero problem. The update equation is then

$$\mathbf{x}(k)=\mathbf{x}(k-1)+\mu\frac{\mathbf{A}_{tt'}^{T}(k)\mathbf{I}}{\sqrt{\mathbf{I}^{T}\mathbf{A}_{tt'}(k)\mathbf{A}_{tt'}^{T}(k)\mathbf{I}+\delta}}. \tag{2.29}$$

Next, consider the integer constraint, which is non-convex and NP hard. In order to conquer the NP-hard integer set problem, the idea of the zero-point attractor in [16] is extended to a binary attractor. First, the concept of the l_0 norm, that counts the number of non-zero entries, is introduced.

Considering that l_0 norm is a Non-Polynomial (NP) hard problem, it is generally approximated by a continuous function. A popular approximation [17] is

$$\left\|\mathbf{x}(k)\right\|_0 \approx \sum_{n=0}^{N}\left(1-e^{-\beta\left|x_n(k)\right|}\right), \tag{2.30}$$

where the two sides of (2.30) are strictly equal when the parameter β approaches infinity. Here, it is extended to involve both 0 and 1 as below

$$\left\|\mathbf{x}(k)\right\|_{0,1} \approx \sum_{n=0}^{N}\left(1-e^{-\beta\left|x_n(k)\right|}-e^{-\beta\left|x_n(k)-1\right|}\right). \tag{2.31}$$

To illustrate how (2.31) works, the function $1-e^{-\beta\left|x_n(k)\right|}-e^{-\beta\left|x_n(k)-1\right|}$ is plotted in Figure 2.5. It is clear that it outputs 1 if the input is neither 0 nor 1, which is also the number of non-zero and none-one entries. The motivation behind the binary attractor is that the following target

$$\boldsymbol{min} \qquad \left\|\mathbf{x}(k)\right\|_{0,1} \tag{2.32}$$

minimizes the total number of non-zero and non-one entries, which requires the optimal solution to belong to the $\{0,1\}$ integer space. Considering the power minimization target too, the proposed cost function in (2.21) can be rewritten as

$$\begin{aligned}&\text{-log}\sum_{m=1}^{r}\left(\mathbf{a}_{\mathrm{m},tt'}^{T}(k)\mathbf{x}(k)-b_m(k)\right)\\&\quad+\varphi\left(\mathbf{p}_b-\mathbf{p}\right)\mathbf{x}(k-1)+\kappa\left\|\mathbf{x}(k)\right\|_{0,1},\end{aligned} \tag{2.33}$$

where $\varphi\left(\mathbf{p}_b-\mathbf{p}\right)\mathbf{x}(k-1)$ is the power penalty term, $\varphi>0$ is a step-size factor of the power penalty, and $\kappa>0$ is the strength for binary attractor.

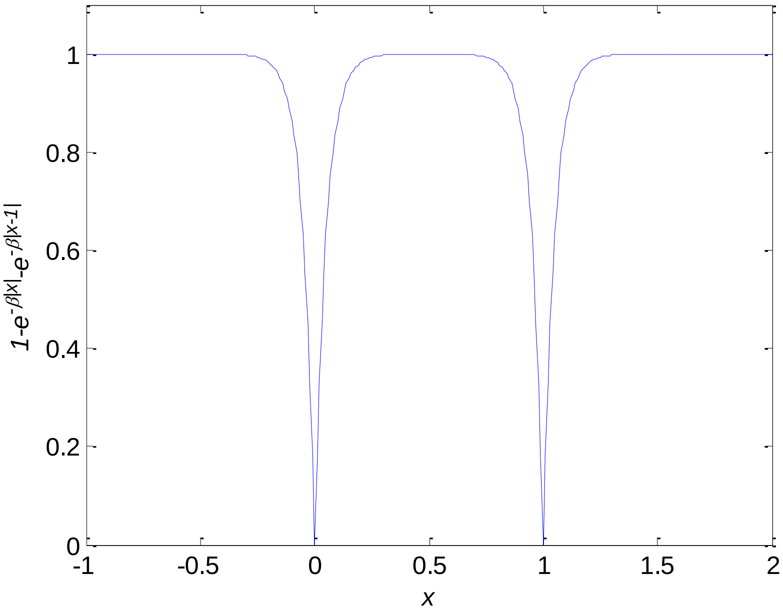


Figure 2.5 The continuous approximated function of the new l0,1 norm counting both non-zero and non-one entries, which outputs 1 when the input is neither 0 nor 1.

Therefore, by minimizing (2.33), the update of the adaptive filtering should be modified to:

$$\begin{aligned}\mathbf{x}(k) = \mathbf{x}(k-1) + \mu \frac{\mathbf{A}_{tt'}^{T}(k)\mathbf{I}}{\sqrt{\mathbf{I}^{T}\mathbf{A}_{tt'}(k)\mathbf{A}_{tt'}^{T}(k)\mathbf{I}+\delta}} \\ -\varphi(\mathbf{p}_b-\mathbf{p}) - \kappa\nabla\left\|\mathbf{x}(k)\right\|_{0,1},\end{aligned} \tag{2.34}$$

where $\nabla\left\|\mathbf{x}(k)\right\|_{0,1}$ is the gradient with respect to $\mathbf{x}(k)$. Note that the current form of (2.34) cannot be implemented in practice, since the term depends on $\mathbf{x}(k)$, the exact a priori value of which is not known. Since the additional constraint $\left\|\mathbf{x}(k)-\mathbf{x}(k-1)\right\|_2^2 \le \mu^2$ is applied in (2.22), it is reasonable to assume $\left\|\mathbf{x}(k)\right\|_{0,1} \approx \left\|\mathbf{x}(k-1)\right\|_{0,1}$, $\nabla\left\|\mathbf{x}(k)\right\|_{0,1} \approx \nabla\left\|\mathbf{x}(k-1)\right\|_{0,1}$. Therefore, the update of the adaptive filtering should be modified to:

$$\begin{aligned}\mathbf{x}(k) = \mathbf{x}(k-1) + \mu \frac{\mathbf{A}_{tt'}^T(k)\mathbf{I}}{\sqrt{\mathbf{I}^T\mathbf{A}_{tt'}(k)\mathbf{A}_{tt'}^T(k)\mathbf{I}+\delta}} \\ -\varphi(\mathbf{p}_b - \mathbf{p}) - \kappa\nabla\left\|\mathbf{x}(k-1)\right\|_{0,1}.\end{aligned} \tag{2.35}$$

Substituting (2.31) into (2.35), the final update of the adaptive filtering is:

$$\begin{aligned}\mathbf{x}(k) = \mathbf{x}(k-1) + \mu \frac{\mathbf{A}_{tt'}^T(k)\mathbf{I}}{\sqrt{\mathbf{I}^T\mathbf{A}_{tt'}(k)\mathbf{A}_{tt'}^T(k)\mathbf{I}+\delta}} \\ -\varphi(\mathbf{p}_b - \mathbf{p}) - \kappa\beta\,\mathrm{sgn}\left(\mathbf{x}(k-1)\right)\otimes e^{-\beta\left|\mathbf{x}(k-1)\right|} \\ -\kappa\beta\,\mathrm{sgn}\left(\mathbf{x}(k-1)-\mathbf{I}\right)\otimes e^{-\beta\left|\mathbf{x}(k-1)-\mathbf{I}\right|},\end{aligned} \tag{2.36}$$

where $\otimes$ is the component-wise multiplication, $\mathbf{I}$ is a unit vector, and $\mathrm{sgn}(\cdot)$ is component-wise sign function defined as

$$\mathrm{sgn}(x) = \begin{cases} \dfrac{x}{|x|}, & x \neq 0; \\ 0, & \mathit{elsewhere.} \end{cases} \tag{2.37}$$

The algorithm described by (2.36) is denoted as affine projection algorithm with binary attractor and power penalty (APA-BA-PP). The final algorithm is shown in Algorithm 1.

The interpretation of the proposed adaptive algorithm is presented as follows. The adaptive filtering algorithm sequentially updates its filter coefficients along the negative gradient direction for each input $\mathbf{A}_{tt'}(k)$ with log-normal distribution, and can be expressed as

$$\begin{aligned}\mathbf{x}_{new} = \mathbf{x}_{prev} + \text{gradient correction} + \text{power penalty} \\ + \text{binary attraction}\end{aligned} \tag{2.38}$$

Algorithm 1: APA-BA-PP

```
a. Initialization:
   $\delta, \mu_0, \varphi_0, \kappa_0, \eta$
   $k = 0, \mathbf{x}_0 = [0,0,...,0]^T$
b. Load $\mathbf{A}, \mathbf{p}, \mathbf{p}_b$ and $r = rank(\mathbf{A})$
c. Perform SVD decomposition on $\mathbf{A} = \mathbf{U\Delta V}^T$
d. Perform QR with column pivoting on matrix $\mathbf{U}_r$ composed
   by the first $r$ columns of $\mathbf{U}$. $\mathbf{U}_r^T\mathbf{P}_r = \mathbf{QR}$, where $\mathbf{P}_r$ is
   permutation matrix.
e. Take $\mathbf{A}_r$ to be the sub-matrix formed by the first $r$ row of
   $\mathbf{P}_r^T\mathbf{A}$.
f. Calculate delay $\mathbf{t}$ and $T = \max(\mathbf{At})$
g. Loop: while $k < K$
   h. Calculate random delay $\mathbf{t}'$ and $\mathbf{t}''$, and
         $\mathbf{A}_{t'} = \mathbf{A}diag\{\mathbf{t}'\}, \mathbf{A}_{t''} = \mathbf{A}_r diag\{\mathbf{t}''\}$
         $\mathbf{A}_{tt'} = \mathbf{A}_{t'} - \mathbf{A}_{tt'} = \mathbf{A}_r diag\{\mathbf{t}' - \mathbf{t}''\}$
         $\mathbf{b} = \mathbf{A}_r\mathbf{t}' - T\mathbf{I}$
   i. Calculate step-size
         $\mu_k = \eta\mu_{k-1}, \varphi_k = \eta\varphi_{k-1}, \kappa_k = \eta\kappa_{k-1}$
   j. Update the filter
         $\mathbf{x}_k = \mathbf{x}_{k-1} + \mu_k \frac{\mathbf{A}_{tt'}^T\mathbf{I}}{\sqrt{\mathbf{I}^T\mathbf{A}_{tt'}\mathbf{A}_{tt'}^T\mathbf{I} + \delta}}$
         $-\varphi_k(\mathbf{p}_b - \mathbf{p}) - \kappa_k\beta\,\text{sgn}(\mathbf{x}_{k-1}) \otimes e^{-\beta|\mathbf{x}_{k-1}|}$
         $-\kappa_k\beta\,\text{sgn}(\mathbf{x}_{k-1} - \mathbf{I}) \otimes e^{-\beta|\mathbf{x}_{k-1} - \mathbf{I}|}$
   k. $k = k + 1$
```

For the first term of gradient correction, according to the definition of matrix $\mathbf{A}_{tt'}$, $\mathbf{A}_{tt'}^T(k)\mathbf{I}$ is the summation of delay improvement of each gate over all the representative paths at timing index k, and after normalization, this information is used to select the gate. At each iteration, the algorithm adopts the body biasing assignment according to a single case of variation. Different samples in subsequent iterations may cause the assignment solution to move in different directions, and a move in one iteration may deteriorate the timing yield of previous samples. However the assignment solution will move towards the direction that moderates most of the variations, in the long run. It is reasonable that the gates with accumulated higher overall delay improvement over different samples of the random distribution tend to be critical and this works in a similar way of sorting the gates based on their statistical summations of delay improvement contribution.

The second term, power penalty, is a constant negative vector based on the extra power cost for each gate if it is selected to apply body bias. This power penalty is used to suppress the gates with higher power cost to be selected.

Finally, the third term, binary attraction, imposes an attraction to zero and one on the coefficients near zero and one. To further illustrate the impact of the binary attractor, it is plotted in Figure 2.6. After each iteration, a filter weight will increase a little when it is less than zero, or decrease a little when it is larger than zero. Meanwhile, the filter weight less than one will increase a little and decrease a little if larger than one. Therefore, the binary attractor will attract the non-zero and non-one weights around zero and one. The range of attraction depends on the parameter β. In the adaptation, the gate with lower delay improvement contribution (near zero) in the past indicates a higher possibility of remaining not selected, and the one with higher contribution (near one) tends to remain selected. According to Figure 2.6., the closer it is to zero and one, the greater the attraction intensity is. In conclusion, the binary attractor could smooth and remove the effect of abrupt delay variation due to exponential delay relationship.

According to the above analysis, it is clear that the step-size φ and κ determine the performance of the proposed algorithm. In order to further improve the performance, a simple variable step-size strategy is used in which the step-sizes are reduced by a constant factor [18]. This is because when more and more samples are accumulated in the sorting, the resulting rank should become more stable, therefore the power penalty and binary attractor can gradually be removed. More complicated variable step-size algorithm could be studied in the future [18]. Assume the algorithm runs for K iterations in which K

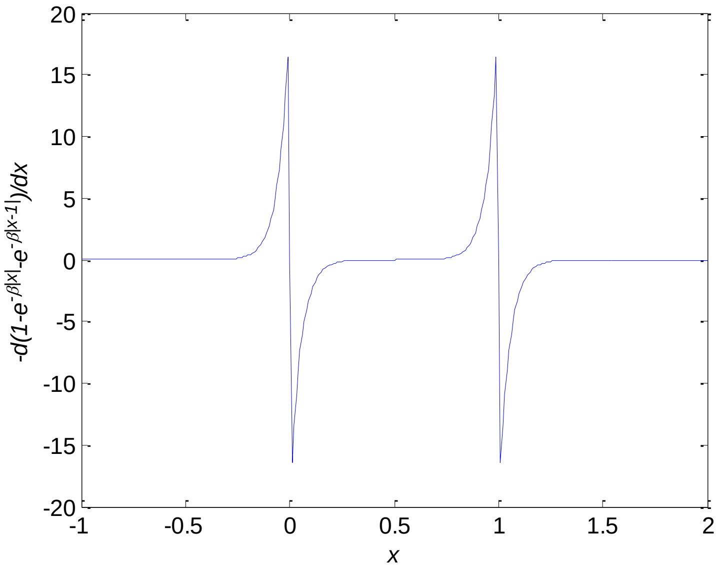


Figure 2.6 The plot of the binary attractor, which is the negative gradient direction of new l0,1 norm, imposes an attraction to zero and one on the coefficients near zero and one.

is a user-specified number. The larger K is, the more gates will be selected, the impact of K on the pass rate and gate selection will be shown later.

To sum up, the proposed APA-BA-PP algorithm can sort the gates based on their statistical contributions to overall delay improvement considering both the standby power cost and the binary decision rule. In practice, in order to further regulate the right-hand side of (2.36), a sigmoidal function could be applied [25], and the final gate set should be selected according to the final ranking of the solution based on the balance between the pass rate requirement and the tolerance of standby power. However, it should be noted that the rigid convergence analysis of the proposed algorithm is still open for future work. In practice, the algorithm could be enforced to stop after a maximum number of iterations K are achieved (in experiments of this dissertation $K = 5000$). Experimental results show that such a heuristic can provide solutions with reasonably good quality.

2.4.3. Complexity Analysis. This section is concluded with the complexity analysis of this APA-BA-PP algorithm. SVD decomposition is done to path matrix **A** ($M \times N$), where M is the number of paths and N is the number of gates. This APA-BA-PP algorithm deals with the combinational circuits between registers, similarly to those in the timing analysis. For modern designs with millions of gates, the number of gates would still be in the range of thousands.

For most designs, **A** is usually very sparse. Therefore, this sparse property can be utilized to significantly decrease the runtime of SVD. For example, the randomized SVD [30] [33] runs very fast on large scale sparse matrix. According to reference [30], the complexity of randomized schemes SVD could achieve $O\left(MNlog(r)\right)$, where r is number of representative paths, which ideally should be the rank of the matrix **A** . For extremely large number of paths, the runtime could be further decreased through using smaller r than the real rank of A at the cost of slight accuracy loss, as will be shown in the experimental results.

The complexity of QR with column pivoting on matrix $\mathbf{U}_r$ with size $M \times r$ is $O\left((M + rlog_f(r))r^2\right), f > 1$ [31], which is also linear w.r.t. the path number. Yet compared with SVD decomposition, it is well known that QR has a much smaller affine. Therefore, the runtime of QR is also acceptable.

Finally, for the proposed APA-BA-PP algorithm, matrix operations are involved with complexity $O\left(Nr\right)$ in each iteration. Accordingly, the total complexity is $O\left(KNr\right)$, where K is the total iteration number.

2.5. EXPERIMENT RESULTS

The path extraction was implemented in Python and the remaining parts were implemented in C++. Eigen 3.0 library was used for the QR decomposition [18], and redsvd package was used for SVD [19]. Simulations are performed based on five 65 nm industrial sub-threshold designs as listed in Table 2.2, which includes the path number, gate number along with the rank of matrix **A** (the number of representative paths) for each design. The supply voltage is 200 mV, and the body biasing voltage V_{BS} is set to -400 mV for PMOS and 400 mV for NMOS. The body biasing voltage is selected so that when all the gates are biased, 100% pass rate can be achieved for V_{th} variance less than 8%. All pass rates reported in this section were obtained through 10K Monte Carlo simulations. In each simulation, it works only if all the paths satisfy the target delay, otherwise it fails. The pass rate is the ratio of the number of working simulations to the total number of simulations. The experiments were conducted on a workstation equipped with quad-core 2.4 GHz CPU and 96 GB RAM.

The proposed adaptive algorithm is compared with two intuitive methods. The first one “all body biasing” applies body biasing to all the gates in the design. The second one “weighted delay” selects body biasing gates based on the descending order of the weighted delay of each gate. The weighted delay of a gate is defined as the product of the number of paths going through a gate and its corresponding delay improvement when body biasing is applied. This metric evaluates the overall importance of a gate in timing. For a fair comparison, the same number of gates are selected for the weighted delay method and the proposed adaptive filtering method.

Table 2.2 Summary of five 65nm sub-threshold designs

Design	***Gate number***	***Path number***	***Rank***
T1	168	75,383	122
T2	496	5830	248
T3	546	916,719	311
T4	892	729,045	460
T5	1,269	462,140	379
T6	3,513	685,908	-

The pass rate and standby power are compared among the three methods in the six designs for different V_{th} variations (σ/μ), and the results are shown in Table 2.3. Different V_{th} variations are used to report meaningful pass rates as the designs are of different sizes. For the adaptive filtering method and the weighted delay method, the percentage of gates selected for body biasing is also reported. The number of iterations used in the adaptive filtering method is between 5,000 and 6,000. According to the results in this table, compared with the all body biasing method which always achieves 100% pass rate , the adaptive filtering method reduces the standby power by 86% at the cost of 20% pass rate loss on average. On the other hand, compared with the weighted delay method, the adaptive filtering method increases the pass rate by up to 57% with the similar standby power. Also, the number of gates selected for body biasing only constitutes a small percentage of the total gates, which verifies the sparsity assumption in Section 2.4.

Design T2 is then used to study the relationship between pass rate and V_{th} variation for the three methods shown in Figure 2.7. The adaptive filtering approach is stopped after 6,000 iterations. 97 out of 496 gates are selected by the adaptive filtering approach and the weighted delay approach. As expected, the proposed adaptive filtering method outperforms the weighted delay method significantly. But it is worse compared with the all body biasing method, which is the cost paid to achieve the significant standby power reduction. Note that the performance difference between APA-BA-PP and "All body bias" can be further reduced by choosing more selected gates as studied later in Figure 2.8.

Table 2.3 Comparison of pass rate among different approach

Design	**Pass rate**				**Standby Power (µW)**			**Selected Gate Percentage**
	σ/μ	***All body biasing***	***Weighted delay***	***APA-BA-PP***	***All body biasing***	***Weighted delay***	***APA-BA-PP***	***APA-BA-PP/ Weighted delay***
T1	0.04	100%	0%	80%	0.35	0.09	0.10	25.60%
T2	0.03	100%	8%	70%	0.86	0.17	0.15	18.71%
T3	0.06	100%	11%	78%	1.08	0.24	0.23	19.60%
T4	0.04	100%	10%	96%	1.70	0.31	0.30	17.26%
T5	0.07	100%	56%	84%	2.25	0.27	0.27	10.87%
T6	0.04	100%	12%	65%	6.35	1.64	1.61	20.00%

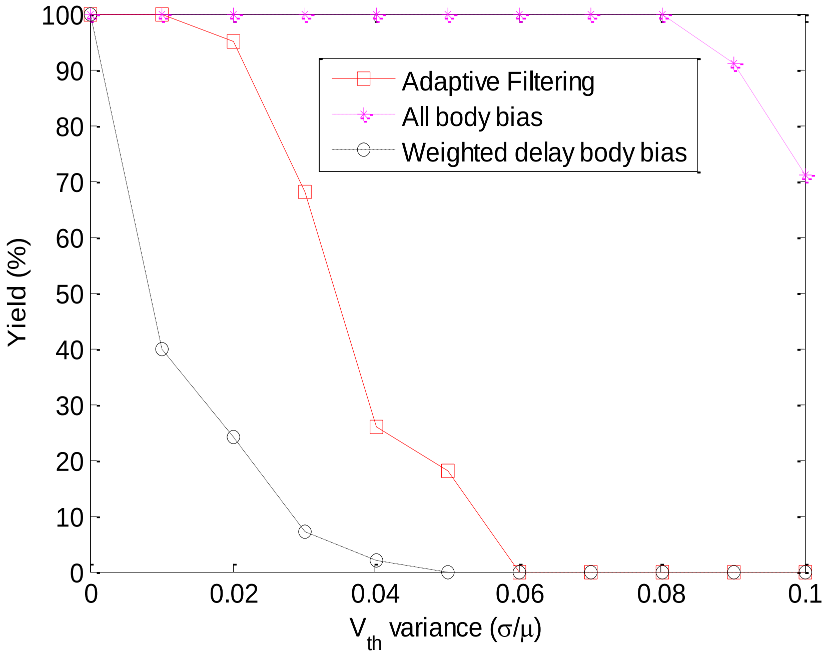


Figure 2.7 Pass rate v.s. V_{th} variance comparison among different methods for T2.

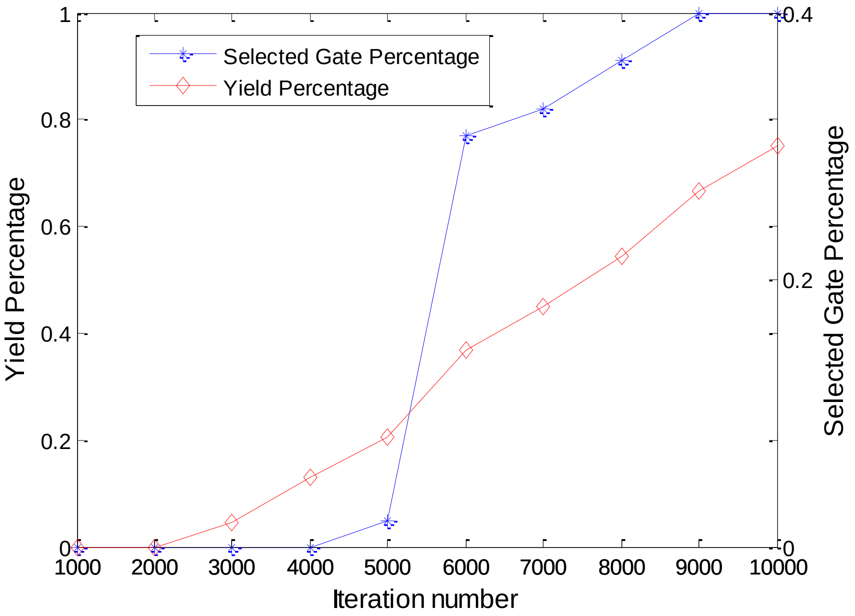


Figure 2.8 Pass rate and the percentage of gates selected v.s. iteration number for design T2 with 10% V_{th} variance

The same design is used again to demonstrate the relationship between the pass rate/percentage of the total gates selected and the iteration number for the adaptive filtering method. The results are depicted in Figure 2.8 for 10% V_{th} variance. As mentioned earlier, the selected gate number will increase in each iteration to achieve higher pass rate. This can be clearly observed in the figure. Interestingly, when the iteration number increases over 6,000, a sharp increase in pass rate can be observed. Therefore, it is necessary to try different numbers of iterations, and choose the most cost-efficient size for the target gate set.

Still using the same design, the impact from the representative path selection step is studied. The pass rates from the adaptive filtering method with and without this step are compared for different V_{th} variance, and the results are shown in Figure 2.9. From the figure it can be seen that using representative paths can result in slightly lower pass rate for the same V_{th} variation. But it results in drastic runtime reduction by reducing the exponential path number to linear w.r.t. the gate number.

As mentioned in Section 2.3.3, the proposed algorithm deals with the combinational circuits between registers, and for modern designs with millions of gates, the number of gates will be in the range of thousands. In order to demonstrate the scalability of the proposed scheme, the runtime of SVD, QR and APA-BA-PP are separately reported for different cases in Table 2.4. It verifies that even for the largest design T6 with 3,513 gates, the runtime is still acceptable.

Meanwhile, the impact of the number of representative paths r on the runtime and the quality of the solution is very interesting. T5 is used here as an example, and the runtimes of SVD and QR w.r.t. different r are shown in Figure 2.10. It is clear that the runtime could be significantly decreased with smaller r. On the other hand, the impact of r on pass rate is presented in Figure 2.11, which verifies that the pass rate loss could be slight for r reduced from 379 (full rank) to 127 (10% of the gate number).

Finally, using Cadence encounter, the core layout of T2 is finished to study the area overhead of the proposed algorithm. The layout without body biasing is given in Figure 2.12, and layout with selected body biasing is compared in Figure 2.13. Based on the proposed APA-BA-PP, 97 out of 496 gates are selected in T2, and the total area overhead is 0.8% compared with the original design without body biasing, which is really small (the size is labeled in the layout). Considering that the area overhead is from both biasing gates and routing, the wire length overhead is also given to show the contribution of different parts. The total wire length is 3,387 um for the design without body biasing, and 3,447 um for the design with body biasing, a 2% overhead.

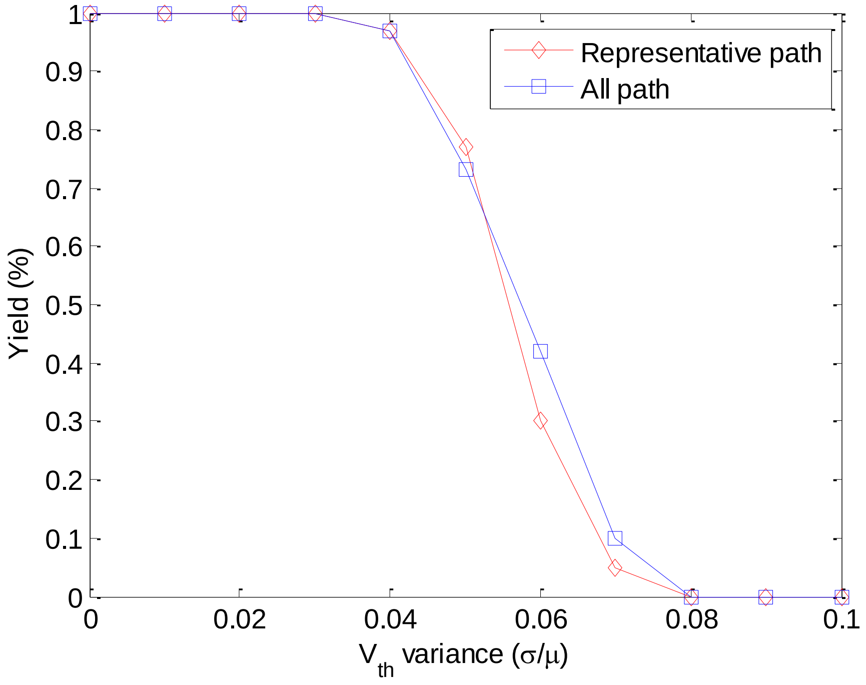


Figure 2.9 The impact of representative path selection on pass rate at different V_{th} variance for design T2.

Table 2.4 The runtime of SVD, QR and APA-BA-PP v.s. gate number for design T1-T6

Design	***r***	***Runtime (mins)***		
		SVD	***QR***	***APA-BA-PP***
T1	122	0.91	0.58	7.23
T2	248	0.34	0.15	17.72
T3	311	65.37	40.28	34.99
T4	460	111.72	63.35	47.01
T5	254	21.18	13.19	75.04
T6	702	228.05	129.52	140.01

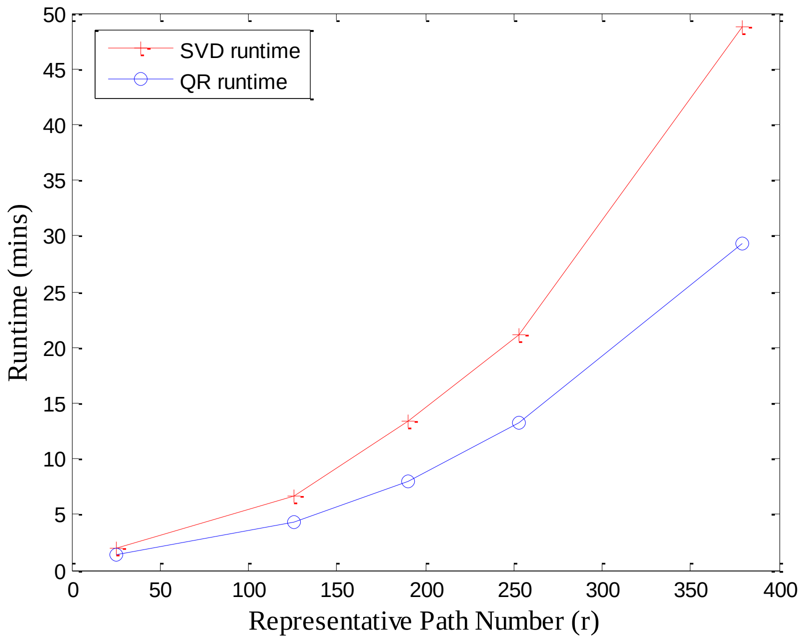


Figure 2.10 The impact of r on runtime for design T5.

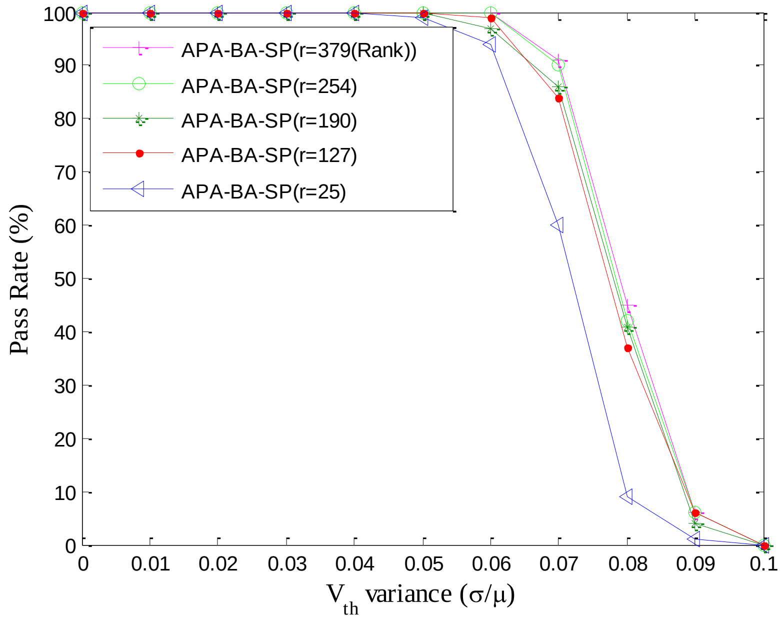


Figure 2.11 The impact of r on pass rate for design T5.

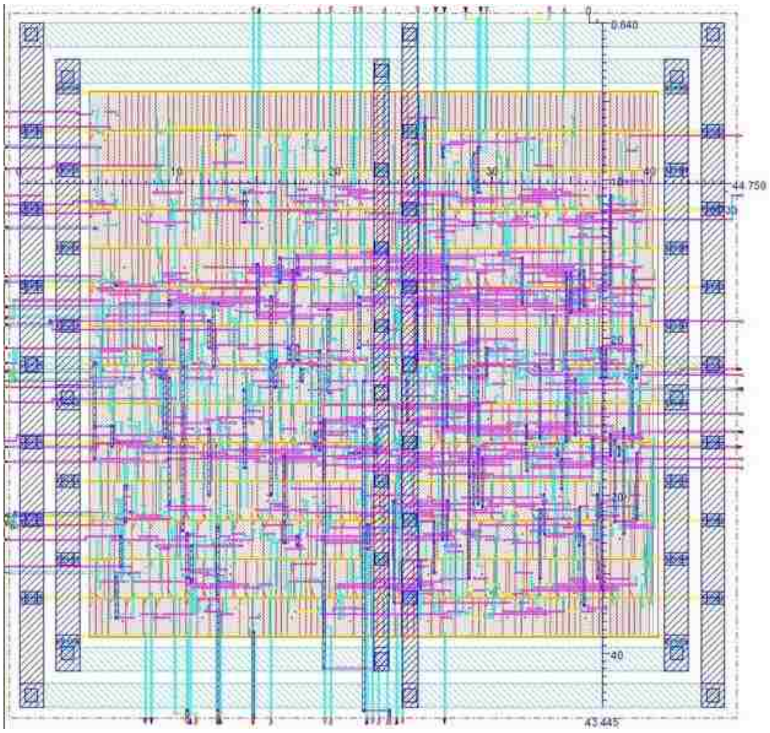


Figure 2.12 Layout of T2 without body biasing.

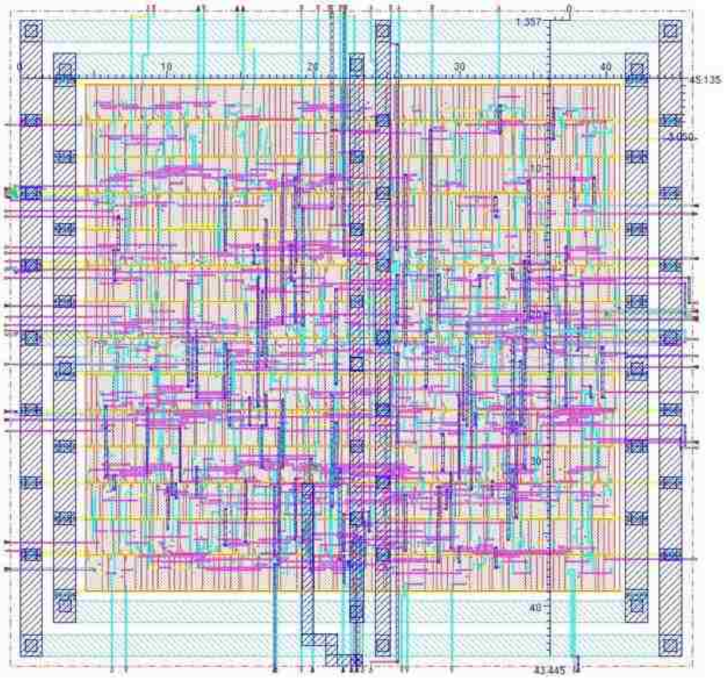


Figure 2.13 Layout of T2 with body biasing gates selected by APA-BA-PP

2.6. CONCLUSION

In this project, the problem of body biasing for sub-threshold designs with one body biasing voltage domain and one body biasing voltage level is formulated firstly. It is demonstrated that the problem can be modeled as a linearly constrained statistical sparse optimization problem, and the adaptive filtering concept from the signal processing community is borrowed to develop an efficient algorithm to solve it. Experimental results on industrial designs using 65nm sub-threshold library suggest that compared with the weighted delay approach, the proposed approach can improve the pass rate by 57% on average with similar standby power and the same number of body biasing gates. Compared with the approach to bias all the gates, the proposed approach can reduce the standby power by 84% on average with 20% pass rate loss.

3. PART I: SELECTIVE BODY BIASING FOR POST-SILICON TUNING OF SUB-THRESHOLD DESIGNS: A SEMI-INFINITE PROGRAMMING APPROACH WITH INCREMENTAL HYPERCUBIC SAMPLING

3.1. BACKGROUND AND MOTIVATION

A practical selective body biasing scheme, with only one body biasing voltage domain to fix the timing violation, is proposed in Section 2. However, though the formulated problem used uncertain constraints with probability distribution, the proposed adaptive algorithm does not explicitly prioritize the constraints with larger probability (i.e., constraints that are more likely to occur). In addition, the power optimization was achieved with power penalty heuristic, and the integer constraints in the formulation were conquered by the binary attractor heuristic. These heuristics lack theoretical justification and have no guaranteed convergence rate. All these issues will significantly degrade the performance of the algorithm.

The interesting fact is noticed that the problem of selective body biasing with single biasing voltage domain can be formulated alternatively as a linear semi-infinite programming (LSIP) problem. In addition, the structure of the problem, associated with the physical meaning of the design, can lead to a novel Incremental Hypercubic Sampling (IHCS) algorithm. The algorithm solves the LSIP problem through a number of finite mixed-integer linear programming. Finally, many nice properties of the algorithm is demonstrated through rigid mathematical derivations. Experimental results on industrial designs using 65nm sub-threshold library demonstrate that, compared with the state-of-art adaptive filtering approach in Section 2, the IHCS approach can improve the pass rate by up to 7.3x with a speed up to 4.1x, using the same number of body biasing gates with about the same power consumption.

The remainder of this section is organized as follows: Section 3.2 gives the new LSIP problem formulation. The proposed IHCS algorithm is presented in Section 3.3. The experimental results are demonstrated in Section 3.4 and concluding remarks are given in Section 3.5.

3.2. PROBLEM FORMULATION

In this section, an alternative formulation for the problem of selective body biasi-

-ng with only one biasing voltage domain is shown, which will lead to a novel and efficient algorithm.

This section is started by defining the notations necessary for the discussion. Following the delay model in Section 2, all gates are simplified as a simple pull-up or pull-down transistor or appropriate driving strength, and only the worst case falling and rising time is considered. The gate delay is given here again in the following (3.1) for convenience, which is exponential with a single Gaussian distributed random variable V$_{th}$.

$$t_d = \frac{1}{2}\eta C_s V_{dd} \frac{1}{I_{s0}} e^{-\frac{V_{dd}}{\rho V_T}} e^{\frac{V_{th}}{\rho V_T}} = \tau e^{\frac{V_{th}}{\rho V_T}}, \tag{3.1}$$

where η is the delay factor and C_s is the switching load capacitance. V_{dd} is the supply voltage and V$_{th}$ is the threshold voltage. V_T is the thermal voltage and ρ is the sub-threshold swing parameter. The constant τ can be obtained by using HSPICE to measure the gate delay at typical V$_{th}$.

The threshold with body biasing can be calculated as

$$V_{th} = V_{th0} + \gamma(\sqrt{|-2\varphi_s + V_{SB}|} - \sqrt{|-2\varphi_s|}), \tag{3.2}$$

where both γ (the body biasing coefficient) and φ_s (the surface potential) are obtained from the technology library [12]. V_{SB} is the body biasing voltage, and V_{th0} (the threshold voltage for $V_{SB} = 0$) is a random variable due to the process variation.

The gate threshold voltage vector V$_{th}$ is defined as $[V_{th1}, V_{th2}, \cdots, V_{thN}]$, which is multivariate Gaussian random vector. The threshold voltage mean value is defined as V$_{th0}$**1**$_N$, where **1**$_N$ is $N \times 1$ vector of all ones.

The delay improvement matrix $\mathbf{A}_{tt'}$ is formulated as

$$\mathbf{A}_{tt'} = \mathbf{A} diag\{\mathbf{t}' - \mathbf{t}''\}. \tag{3.3}$$

where matrix **A** with M rows and N columns is used to represent the connection information of design. Each row represents a path, and each column represents a gate. If n-th gate appears in m-th path, the element at the m-th row and n-th column of the matrix

A (i.e. $\mathbf{A}_{m,n}$) is set to 1, otherwise it is set to 0. $\mathbf{t'}$, and $\mathbf{t''}$ are two $N \times 1$vectors, and each element represents the gate's delay t' and t'' in different scenarios:

- t' is the gate delay with only V_{th} variation
- t'' is the gate delay with both V_{th} variation and body biasing

Both the delay vectors $\mathbf{t'}$, and $\mathbf{t''}$ are lognormal distribution [4].

Furthermore, considering the path number can have an exponential relationship with the gate number, a representative path selection stage was employed in Section 2, in which the number of representative paths does not exceed the number of gates. Therefore, the matrix $\mathbf{A}$ is replaced with the must smaller representative path matrix $\mathbf{A}_r$. The same technique in this section is adopted.

Meanwhile, a vector $\mathbf{b}$ $(M \times 1)$ is defined as

$$\mathbf{b} = \mathbf{At}' - T\mathbf{I}_M, \tag{3.4}$$

where T is the target delay of the circuit. Considering $\mathbf{t'}$ is the delay vector with process variation, vector $\mathbf{b}$ is the required delay improvement vector under process variation.

Based on the above definitions, constraint $\mathbf{A}_{tt'}\mathbf{x} \geq \mathbf{b}$ is gotten, which represents component-wise inequality, $\mathbf{a}_{m,tt'}\mathbf{x} \geq b_m, m = 1...M$, where $\mathbf{a}_{m,tt'}$ is the m^{th} row of $\mathbf{A}_{tt'}$, and $\mathbf{b}_m$ is the m^{th} element of $\mathbf{b}$). The constraint means the overall delay improvement for paths must be larger than the required delay improvement vector $\mathbf{b}$. This constraint cannot be satisfied for the entire variation space of V_{th}, and this problem is solved by adding a penalty term in the objective function in Section 2.

Here, an alternative formulation of (2.10) is explored, by specifying the variation space of V_{th} that the constraints need to be satisfied, i.e.,

$$\begin{array}{ll} \boldsymbol{min} & (\mathbf{p}_b - \mathbf{p})\mathbf{x} \\ \text{subject to} & \mathbf{A}_{tt'}\mathbf{x} \geq \mathbf{b}, \boldsymbol{V_{th}} \in \Omega. \end{array} \tag{3.5}$$

in of decision variables is finite (i.e. N), while the constraints are infinite, which the number due to the fact that they include a linear combination of N lognormal distributed random variables. The variation space Ω has a direct impact on the number of gates selected as well as the pass rate and needs to be provided by the designer. In this dissertation, $\mu \pm 3\sigma$ is used here to define the boundary Ω. Since the number of constraints is infinite (V$_{th}$ can be any value in Ω), the above formulation is in essence a linear semi-infinite programming (LSIP).

Typical methods to solve LSIP problem include stochastic approach, fuzzy approach, interval approach, robust approach and parametric approach, etc. [33]. The stochastic approach is difficult to apply here because the dimension of random variables in this problem is very high, which makes the empirical distribution is impossible to get. The robust approach cannot be used due to the fact that it is based on the uncertain set as mentioned in [33]. As such, it is imperative to devise a new method to solve the problem, which will be detailed in the following Section 3.3.

3.3. PROPOSED IHCS ALGORITHM

In this section, an efficient algorithm will be proposed to solve the LSIP problem (3.5) using a novel concept of Incremental Hypercubic Sampling (IHCS) approach specially tailored to the problem structure.

The algorithm is inspired by the observation that in the V$_{th}$ variation space Ω, those scenarios (constraints) with smaller variation should be satisfied with higher priority, as they have higher probability to occur. Accordingly, it is very interesting to devise an iterative algorithm that handles constraints with smaller V$_{th}$ variations first. The linear programming (LP) problem is considered first

$$\begin{aligned} \boldsymbol{min} \quad & \left(\mathbf{p}_b - \mathbf{p}\right)\mathbf{x} \\ \text{subject to} \quad & \hat{\mathbf{A}}_{tt'}\mathbf{x} \text{ - } \hat{\mathbf{b}} \geq \varepsilon \boldsymbol{1}_N. \end{aligned} \tag{3.6}$$

where $\hat{\mathbf{A}}_{tt'}$ and $\hat{\mathbf{b}}$ in the constraint are a sample of the LSIP problem (3.5) with threshold vector V$_{th0}$, and ε is the timing margin which is positive and less than the timing constraint T. The following assumptions for the selective body biasing problem are gotten:

Assumption 1. There exists a small positive number ε that the linear programming problem in (3.6) is solvable for any V$_{th}$ within hypercube centered at mean threshold vector with side length $2 \times 3\sigma$.

It has been shown in section 2 that, the timing benefit brought by body biasing for a single gate in the sub-threshold regime can cancel its own delay increment in the worst-case scenario under variation with 3σ/μ = 10%. Here, it is demonstrated that this ***assumption 1*** is reasonable using a commercial 65nm library as an example. The parameters of 65nm library are listed in Table 3.1.

Consider the threshold with body biasing in (3.3), and the body biasing voltage is 400 mV, substitute the parameters in Table 3.1 and the 400 mV (0.4 V) body biasing voltage into (3.3), and the change in the threshold voltage caused by body biasing is:

$$\begin{aligned}\Delta V_{th,body} &= \gamma(\sqrt{|-2\varphi_s + V_{SB}|} - \sqrt{|-2\varphi_s|}) \\ &= 0.3757 \times (\sqrt{|2 \times 0.8016 + 0.4|} - \sqrt{2 \times 0.8016}) \\ &= 0.0560\end{aligned} \tag{3.7}$$

Meanwhile, the maximum threshold voltage variation is

$$\Delta V_{th,\max} = \pm 3\sigma = \pm \mathit{0.1}\mu = \pm 0.1 V_{th0,\mu} = \pm 0.05610. \tag{3.8}$$

Therefore, there is $\Delta V_{th,body} - |\Delta V_{th,\max}| \approx 0$, which means the threshold voltage changes can cancel the maximum threshold voltage variation with margin. Therefore, even for maximum threshold voltage variation ($\mu \pm 3\sigma$), the solution with all ones can make sure $\mathbf{A}_{tt'}\mathbf{x} \geq \mathbf{b}$. Therefore, Assumption 1 will hold for this 65nm library. Assumption 1 will be workable for other libraries by setting suitable body biasing voltage V_{SB} .

Table 3.1 Parameters of 65nm library

parameters	$V_{th0,\mu}$	γ	φ_s
NMOS	0.5610	0.3757	-0.8016
PMOS	-0.5330	-0.4046	0.8016

Based on ***Assumption 1***, the following ***Theorem 1***is gotten:

Theorem 1. There exists a small V_{th} hypercube centered at V_{th0} with side $2\Delta V_{th}$ that the solution x_0 of the tighter problem (3.6) can satisfy the constraints $\mathbf{A}_{tt'}\mathbf{x} \ge \mathbf{b}$ for any V_{th} within the small hypercube.

Proof:

Considering the definitions of $\mathbf{t'}$ and $\mathbf{t''}$,

$$\mathbf{t'} = \left[\tau_1 \mathrm{e}^{\frac{V_{th1}}{\rho V_T}}, \cdots, \tau_N \mathrm{e}^{\frac{V_{thN}}{\rho V_T}} \right], \tag{3.9}$$

$$\mathbf{t''} = \left[\tau_1 \mathrm{e}^{\frac{V'_{th1}}{\rho V_T}}, \cdots, \tau_N \mathrm{e}^{\frac{V'_{thN}}{\rho V_T}} \right], \tag{3.10}$$

where V_{thi} is threshold voltage with process variation for i-th gate, and V'_{thi} is threshold voltage with both process variation and body biasing for i-th gate. Therefore,

$$\begin{aligned} \mathbf{A}_{tt'} &= \mathbf{A}diag\left\{\mathbf{t}' - \mathbf{t}''\right\} \\ &= \mathbf{A}diag\left\{\left[\left[\tau_1 \mathrm{e}^{\frac{V_{th1}}{\rho V_T}}, \cdots, \tau_N \mathrm{e}^{\frac{V_{thN}}{\rho V_T}} \right] - \left[\tau_1 \mathrm{e}^{\frac{V'_{th1}}{\rho V_T}}, \cdots, \tau_N \mathrm{e}^{\frac{V'_{thN}}{\rho V_T}} \right]\right]\right\} \end{aligned} \tag{3.11}$$

and

$$\mathbf{b} = \mathbf{t}' - T\boldsymbol{1}_N = \left[\tau_1 \mathrm{e}^{\frac{V_{th1}}{\rho V_T}} - T, \cdots, \tau_N \mathrm{e}^{\frac{V_{thN}}{\rho V_T}} - T \right] \tag{3.12}$$

where T is the target timing constraint of the design.

Solution $\boldsymbol{x}_0$ is obtained by solving LP problem (3.6), which means

$$\mathbf{A}_{tt'}\boldsymbol{x}_0 - \mathbf{b} \ge \varepsilon \boldsymbol{1}_N \tag{3.13}$$

and substitute (3.11) and (3.12) into (3.13),(3.14) is gotten

$$\mathbf{A}diag\left\{\left[\left[\tau_1\,\mathrm{e}^{\frac{V_{th1}}{\rho V_T}},\cdots,\tau_N\,\mathrm{e}^{\frac{V_{thN}}{\rho V_T}}\right]-\left[\tau_1\,\mathrm{e}^{\frac{V'_{th1}}{\rho V_T}},\cdots,\tau_N\,\mathrm{e}^{\frac{V'_{thN}}{\rho V_T}}\right]\right]\right\}\boldsymbol{x}_0 \\ -\left[\tau_1\,\mathrm{e}^{\frac{V_{th1}}{\rho V_T}}-T,\cdots,\tau_N\,\mathrm{e}^{\frac{V_{thN}}{\rho V_T}}-T\right]\geq\varepsilon\boldsymbol{1}_N. \tag{3.14}$$

Now considering a hypercube space centered at $V_{th}=\left[V_{th1},V_{th2},\cdots,V_{thN}\right]$ with size length $2\Delta V_{th}$, and on its vertices, the solution $\boldsymbol{x}_0$ in (3.6) should satisfy the constraint

$$\mathbf{A'}_{tt'}\boldsymbol{x}_0\,\text{-}\,\mathbf{b'}\geq 0. \tag{3.15}$$

in which

$$\mathbf{A'}_{tt'}=\mathbf{A}diag\left\{\left[\left[\tau_1\,\mathrm{e}^{\frac{V_{th1}\pm\Delta V_{th}}{\rho V_T}},\cdots,\tau_N\,\mathrm{e}^{\frac{V_{thN}\pm\Delta V_{th}}{\rho V_T}}\right]-\left[\tau_1\,\mathrm{e}^{\frac{V'_{th1}\pm\Delta V_{th}}{\rho V_T}},\cdots,\tau_N\,\mathrm{e}^{\frac{V'_{thN}\pm\Delta V_{th}}{\rho V_T}}\right]\right]\right\} \tag{3.16}$$

and

$$\mathbf{b'}=\left[\tau_1\,\mathrm{e}^{\frac{V_{th1}\pm\Delta V_{th}}{\rho V_T}}-T,\cdots,\tau_N\,\mathrm{e}^{\frac{V_{thN}\pm\Delta V_{th}}{\rho V_T}}-T\right] \tag{3.17}$$

Therefore, substituting (3.16) and (3.18) into (3.15), and after several straightforward mathematical operations, (3.18) is gotten

$$\mathrm{e}^{\frac{\pm\Delta V_{th}}{\rho V_T}}\,\mathbf{A}diag\left\{\left[\left[\tau_1\,\mathrm{e}^{\frac{V_{th1}}{\rho V_T}},\cdots,\tau_N\,\mathrm{e}^{\frac{V_{thN}}{\rho V_T}}\right]-\left[\tau_1\,\mathrm{e}^{\frac{V'_{th1}}{\rho V_T}},\cdots,\tau_N\,\mathrm{e}^{\frac{V'_{thN}}{\rho V_T}}\right]\right]\right\}\boldsymbol{x}_0 \\ -\mathrm{e}^{\frac{\pm\Delta V_{th}}{\rho V_T}}\left[\tau_1\,\mathrm{e}^{\frac{V_{th1}}{\rho V_T}}-T,\cdots,\tau_N\,\mathrm{e}^{\frac{V_{thN}}{\rho V_T}}-T\right]-\mathrm{e}^{\frac{\pm\Delta V_{th}}{\rho V_T}}\,T\boldsymbol{1}_N+T\boldsymbol{1}_N\geq 0 \tag{3.18}$$

Considering (3.11)-(3.12), (3.18) can be rewritten as

$$\mathrm{e}^{\frac{\pm\Delta V_{th}}{\rho V_T}}\left(\mathbf{A}_{tt'}\boldsymbol{x}_0\,\text{-}\,\mathbf{b}\right)-\mathrm{e}^{\frac{\pm\Delta V_{th}}{\rho V_T}}\,T\boldsymbol{1}_N+T\boldsymbol{1}_N\geq 0. \tag{3.19}$$

$$\begin{aligned} &\mathrm{e}^{\frac{\pm\Delta V_{th}}{\rho V_T}}\left(\mathbf{A}_{tt'}\boldsymbol{x}_0 - \mathbf{b}\right) \geq \mathrm{e}^{\frac{\pm\Delta V_{th}}{\rho V_T}}\, T\boldsymbol{1}_N - T\boldsymbol{1}_N \\ &\mathbf{A}_{tt'}\boldsymbol{x}_0 - \mathbf{b} \geq T\,\mathrm{e}^{\frac{\mp\Delta V_{th}}{\rho V_T}}\left(\mathrm{e}^{\frac{\pm\Delta V_{th}}{\rho V_T}} - 1\right)\boldsymbol{1}_N \end{aligned} \tag{3.20}$$

It is already known that $\mathbf{A}_{tt'}\boldsymbol{x}_0 - \mathbf{b} \geq \varepsilon\boldsymbol{1}_N$, and in order to satisfy (3.20), ΔV_{th} should satisfies

$$\varepsilon \geq T\,\mathrm{e}^{\frac{\mp\Delta V_{th}}{\rho V_T}}\left(\mathrm{e}^{\frac{\pm\Delta V_{th}}{\rho V_T}} - 1\right) \tag{3.21}$$

It should be noted that (3.21) stands for the following four equations:

$$\varepsilon \geq \mathrm{e}^{\frac{\Delta V_{th}}{\rho V_T}}\left(\mathrm{e}^{\frac{\Delta V_{th}}{\rho V_T}} - 1\right)T \tag{3.22}$$

$$\varepsilon \geq \mathrm{e}^{\frac{-\Delta V_{th}}{\rho V_T}}\left(\mathrm{e}^{\frac{-\Delta V_{th}}{\rho V_T}} - 1\right)T \tag{3.23}$$

$$\varepsilon \geq \mathrm{e}^{\frac{\Delta V_{th}}{\rho V_T}}\left(\mathrm{e}^{\frac{-\Delta V_{th}}{\rho V_T}} - 1\right)T \tag{3.24}$$

$$\varepsilon \geq \mathrm{e}^{\frac{-\Delta V_{th}}{\rho V_T}}\left(\mathrm{e}^{\frac{\Delta V_{th}}{\rho V_T}} - 1\right)T \tag{3.25}$$

Solving (3.22)-(3.23), the required range of ΔV_{th} can be gotten as

$$\left[0, \rho V_T \ln\left(\frac{1}{2} + \sqrt{\frac{1}{4} + \frac{\varepsilon}{T}}\right)\right] \tag{3.26}$$

Solving (3.24)-(3.25), another range requirement of ΔV_{th} can be gotten as

$$\left[0, \rho V_T \ln\left(\frac{T}{T - \varepsilon}\right)\right] \tag{3.27}$$

Considering both (3.26) and (3.27), the allowed maximum value of ΔV_{th} is

$$\Delta V_{th} = \rho V_T \ln\left(\frac{1}{2} + \sqrt{\frac{1}{4} + \frac{\varepsilon}{T}}\right) \tag{3.28}$$

This ends the proof of ***Theorem 1***.

A lot of interesting facts can be observed from ***Theorem 1***. The maximum side length $2\Delta V_{th}$ of the hypercube is related with both the target timing constraint T, and the timing margin ε according to (29). Therefore, the normalized timing margin is defined as ε/T. Meanwhile, the normalized side length with threshold voltage is defined as step size, i.e. $2\Delta V_T/V_{th0,\mu}$. For a commercial 65nm technology, the relation between the step-size $2\Delta V_T/V_{th0,\mu}$ and the normalized time margin ε/T is plotted in Figure 3.1. It can be observed that the hypercube is monotonically increasing with the normalized timing margin, which is intuitive. Meanwhile, according to ***Assumption 1***, ΔV_{th} should also be limited by 3σ too (i.e. $2\Delta V_T/V_{th0,\mu} < 0.1$).

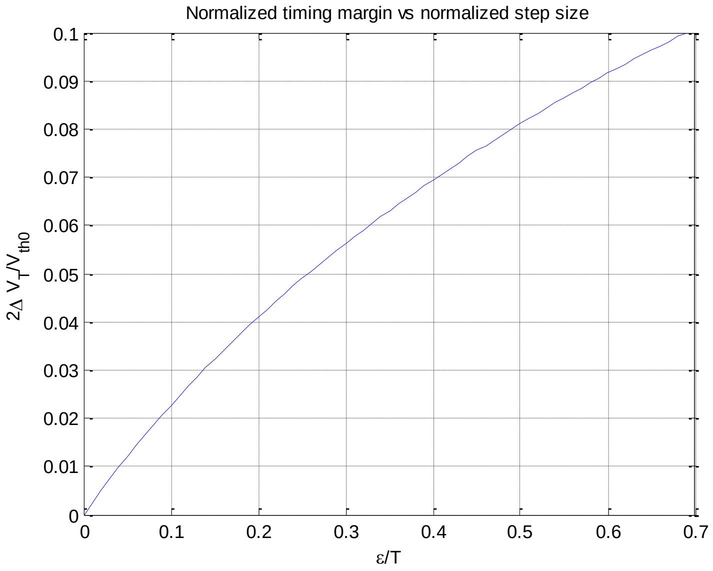


Figure 3.1 Relation between the step size $2\Delta V_T/V_{th0,\mu}$ and the normalized time margin ε/T.

Finally, based on ***Theorem 1***, the following corollaries are derived easily:

Corollary 1. There exist a minimum spacing r between the samples to cover the entire $\mu \pm 3\sigma$ variation space.

Corollary 2. There exist a finite number of samples to cover the entire $\mu \pm 3\sigma$ variation space.

The corollaries pave the road for the IHCS algorithm, which is summarized in ***Algorithm 1***. The algorithm samples V_{th} on an incrementally increased hypercubic surface, and for each sample $\hat{\boldsymbol{V}}_{th}$, and calculate the corresponding $\hat{\mathbf{A}}_{tt'}$ and $\hat{\mathbf{b}}$ according to (3.3) and (3.4). The following ILP problem will be solved

$$\begin{aligned} \boldsymbol{min} \quad & (\mathbf{p}_b - \mathbf{p})\mathbf{x} \\ \text{subject to} \quad & \hat{\mathbf{A}}_{tt'}\boldsymbol{x} - \hat{\mathbf{b}} \geq \varepsilon \boldsymbol{1}_N \end{aligned} \tag{3.29}$$

Assume the solution of (3.29) is $\boldsymbol{x}_0$, and the selected gates for each sample are combined together. In the next iteration, the elements in *x* corresponding to the gates selected in all previous iterations will be set to 1. Since the surface area of the hypercube increases in each iteration, an increased number of samples are used in each iteration. In addition, to reduce computation complexity, only a small number of samples will be used which may not be enough to cover the entire surface. In the experiments, it is found that setting step size as $\Delta\sigma = 0.01\mu$ and the number of samples as $N \times 2^i$ (where *i* is the iteration number) offers a good balance between runtime and solution quality. Moreover, from the corollaries as long as in each iteration the surface with a step size $\Delta\sigma$ is expanded, the entire variation space can be covered. The iterations will be stopped when the threshold specified by the designer is reached (if it is smaller than $\mu \pm 3\sigma$) or more gates than predefined parameter K have been selected as body biasing.

Apparently, the algorithm has the benefit that scenarios with smaller variations are covered with higher priority (in earlier iterations). This should lead to significantly improved pass rate, as will be validated in experimental results. Meanwhile, it is straightforward to have the following ***Theorem 2***:

Theorem 2. The pass rate increases monotonically with the number of iterations in IHCS.

Algorithm 1: Incremental Hypercubic Sampling (IHCS)

```
Initial: Set all elements of $\boldsymbol{x}_s$ to be zeros; $i=1,\sigma_0=\mu$;
Loop: $for\,(\sigma=\sigma_0;\sigma<\sigma_0+0.1\mu;\sigma=\sigma+\Delta\sigma)$
    $a.1$ Loop: $for\,(j=1;\ j\le n_i;\ j=j+1)$
            Sample $V_{th,j}$ on the surface of hypercube
            centered at $V_{th0}$ with side length $2(\sigma_0+(i-1)\Delta\sigma)$
            Get the optimal solution $\boldsymbol{x}_j$ by solving (3.29)
            End Loop
    $a.2$   Merge all ones in solutions $\boldsymbol{x}_1,\boldsymbol{x}_2,...,\boldsymbol{x}_{ni}$ to
            form a new vector $\boldsymbol{x}_{s_i}$
    $a.3$   $i=i+1$
    $a.4$   Update the elements of $\boldsymbol{x}_s$ to be ones
            according to the solution $\boldsymbol{x}_{s_i}$.
    $a.5$   $if\ (\|\boldsymbol{x}_s\|_0>K)\ \ exit;\ else\ \ goto\ a.1$
Output: Final solution $\boldsymbol{x}_s$
```

The proof is trivial considering the fact that each iteration keeps the gates already selected and adds more gates for body biasing.

Finally, a single path with two gates as a simple two-dimensional example is used to illustrate the IHCS algorithm. In this example there are only two random variables V_{th1} and V_{th2}. The variation space of V_{th1} and V_{th2} is shown as Figure 3.2 (a), in which the origin of coordinate has been relocated at (V_{th1},V_{th2}). Then a few V_{th} on the square with side $2\Delta\sigma$ is randomly sampled. By solving the related ILP problems and merging the solutions, the area surrounded by the square will be covered, as shown in Figure 3.2 (b). Then the side of the square is expanded to $4\Delta\sigma$, and sampled on the new surface. After solving the related ILP problems and merging the solution, the area surrounded by the expanded square is covered as shown in Figure 3.2 (d). This process continues until the required variance specified by the designer is covered or too many gates have been selected as body biasing.

It should be noted that, even though the power overhead is still used as the minimization target as Section 2. It is straightforward to extend the proposed IHCS approach to take into account other optimization targets, such as the layout and routing overhead, etc. It is only necessary to modify the optimization targets of the ILP.

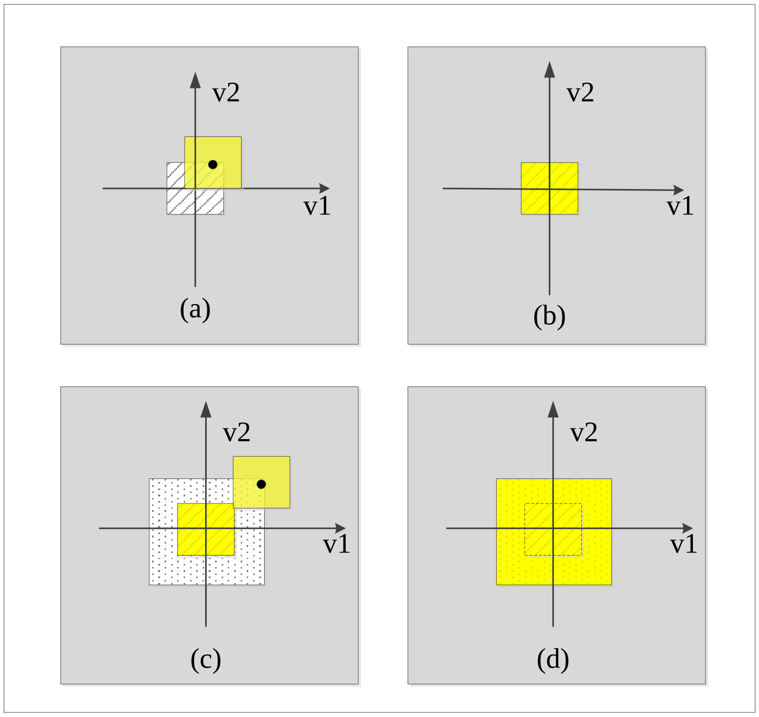


Figure 3.2 A two-dimension example of proposed IHCS algorithm workflow

3.4. EXPERIMENTAL RESULTS

The proposed IHCS algorithm is implemented and the APA-BA-PP algorithm in MATLAB. The simulations were performed based on six 65 nm industrial sub-threshold designs as listed in Table 3.2, which includes the path number, gate number along with the rank of matrix **A** (the number of representative paths) for each design. The supply voltage is 200 mV, and the body biasing voltage is set to -400 mV for PMOS and 400 mV for NMOS. The body biasing voltage is selected so that when all the gates are biased, 93% pass rate can be achieved for V_{th} variance less than 5%. All pass rates reported in this section were obtained through 10K Monte Carlo simulations. In each simulation, it works only if all the paths satisfy the target delay, otherwise it fails. The pass rate is defined as the percentage of Monte Carlo samples without timing violations out of the 10K samples. The experiments were conducted on a workstation equipped with quad-core 2.4 GHz CPU and 96 GB RAM.

Table 3.2 Summary of six 65 nm sub-threshold designs

Design	***Gate number***	***Path number***	***Rank***
T1	168	75,383	122
T2	496	5830	248
T3	546	916,719	311
T4	892	729,045	460
T5	1,269	462,140	379
T6	3,513	685,908	-

Firstly, the pass rate is compared and standby power of the proposed IHCS algorithm with all body biasing and APA-BA-PP algorithm in section 2 for all six test cases. For fair comparison, the pass rate is compared under the same selected gate percentage, and the results are shown in Table 3.3. Different V_{th} variations are used to report meaningful pass rates as the designs are of different sizes. The percentage of gates selected for body biasing is also reported for both APA-BA-PP and IHCS algorithm. According to the data in Table 3.3, all body biasing always achieve 100% pass rate, but with much higher power consumption compared with APA-BA-PP and IHCS algorithm. Meanwhile, compared with APA-BA-PP, the proposed IHCS algorithm has up to 82% pass rate improvement, which means that IHCS achieves better performance (pass rate).

The runtime comparison between the proposed IHCS algorithm and APA-BA-PP is shown in Table 3.4. According to the table, IHCS algorithm can improve the runtime up to 2x on average compared with APA-BA-PP. The only case where IHCS requires longer runtime is Design T1. This is probably because the proposed algorithm has not been optimized for small designs, where the number of samples can be reduced.

Table 3.3 Comparison of pass rate among different approach

design	**Pass rate**				**Standby Power (μW)**			**Selected Gate Percentage**
	σ/μ	***All body biasing***	***APA-BA-PP***	***SIP-IHCS***	***All body biasing***	***APA-BA-PP***	***SIP-IHCS***	***SIP-IHCS /APA-BA-PP***
T1	0.04	100%	45%	79%	0.35	0.09	0.09	19.64%
T2	0.05	100%	30%	93%	0.86	0.18	0.18	21.17%
T3	0.03	100%	96%	96%	1.08	0.29	0.29	26.73%
T4	0.04	100%	39%	55%	1.70	0.23	0.23	13.34%
T5	0.06	100%	86%	86%	2.25	0.22	0.22	8.06%
T6	0.03	100%	13%	95%	6.35	1.06	1.06	16.43%

Table 3.4 The runtime of APA-BA-PP and IHCS for designs T1-T6

Design	Runtime (mins)	
	APA-BA-PP	IHCS
T1	7.23 (1)	13.90 1/(0.5X)
T2	17.72 (1)	17.64 (1/1.0X)
T3	34.99 (1)	13.67 (1/2.6X)
T4	47.01 (1)	17.03(1/2.8X)
T5	75.04 (1)	17.99(1/4.1X)
T6	140.01 (1)	139.00(1/1.0X)

According to ***Theorem 1***, the step size of ΔV_{th} will impact the final solution drastically, and design T2 is used to verify the effect of step size on pass rate. The result is shown in Figure 3.3. From the figure, it is clear that the pass rate reduces with the increase of step size. This is intuitively in accordance with ***Corollaries 1*** and ***2***: with too big of a step size, the solutions from the samples on the expanded hybercubic surface are not enough to cover the gap. On the other hand, the runtime also increases rapidly as the step size decreases. It is found that setting the step size to 0.01, which has been used throughout the experiments, offers a good balance between runtime and pass rate.

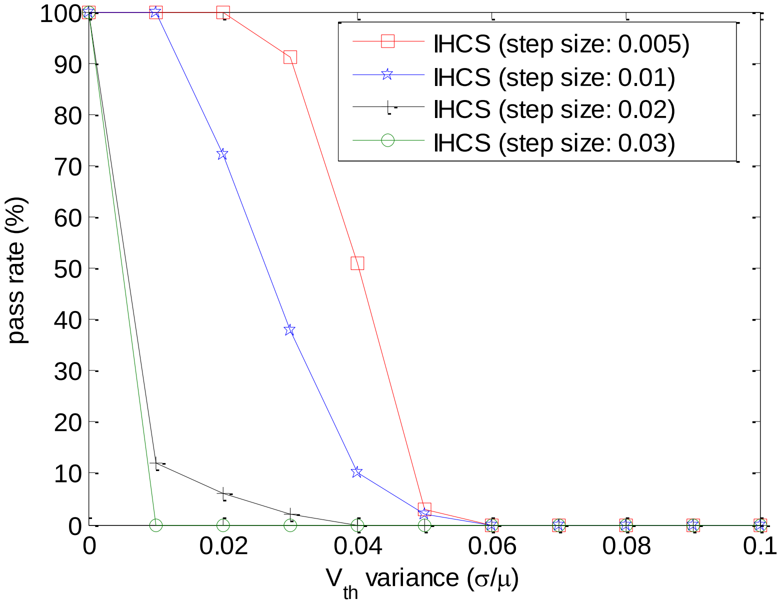


Figure 3.3 The impact of step size on pass rate at different V_{th} variance for design T2.

Then, the same design T2 is used to demonstrate the relationship between the pass rate and the number of samples used in each iteration, which is shown in Figure. 3.4. From the figure it is seen that the pass rate improves as more samples are taken in each iteration, which is as expected. The increase of the sample number on the other hand has a negative impact on runtime, and in the experiments it is figured out that setting the number of samples to be $N \times 2^i$ offers a good balance between runtime and pass rate.

Finally, using Cadence encounter, the core layout of T2 with selected body biasing is given in Figure 3.5 to study the area overhead of the proposed algorithm. Based on the proposed IHSC, 105 out of 496 gates are selected in T2, and the total area overhead is 0.9% compared with the original design without body biasing, which is really small (the size is labeled in the layout). Considering that the area overhead is from both biasing gates and routing, the wire length overhead is also given to show the contribution of different parts. The total wire length is 3,387 um for the design without body biasing, and 3,703 um for the design with body biasing, a 2% overhead.

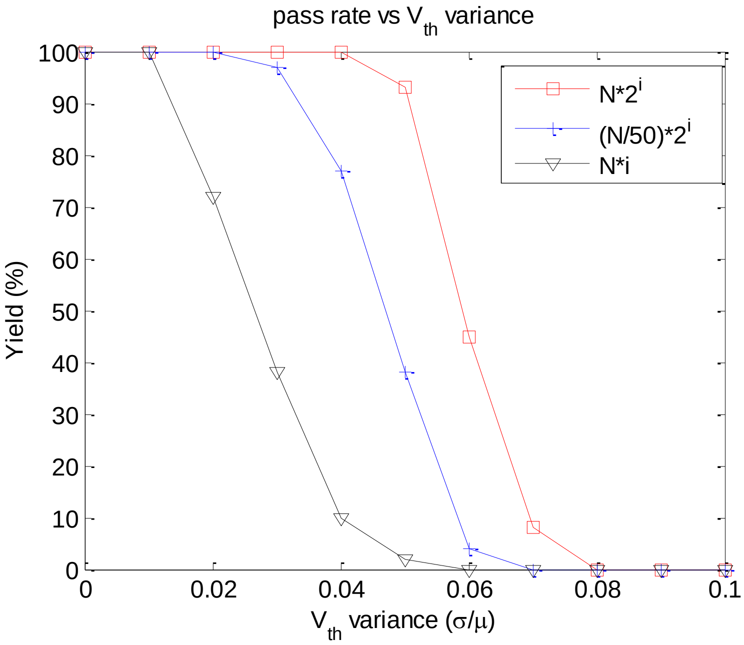


Figure 3.4 The impact of sample numbers in each iteration on pass rate at different V_{th} variance for design T2.

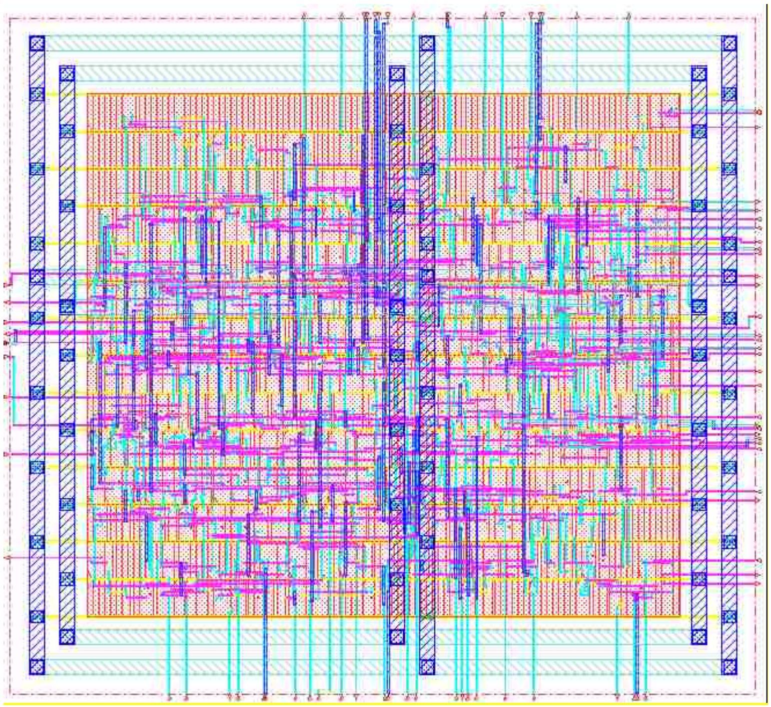


Figure 3.5 Layout of design T2 with body biasing.

3.5. CONCLUSIONS

In this Section, the selective body biasing problem for sub-threshold designs is reformulated as an SIP problem. Then an efficient algorithm based on the novel concept of Incremental Hypercubic Sampling (IHCS), specially tailored to the problem structure, is proposed along with the convergence analysis. Compared with the state-of-the-art approach based on adaptive filtering, experimental results on industrial designs using 65nm sub-threshold library demonstrate that the proposed approach can improve the pass rate by up to 7.3x with a speedup to 4.1x, using the same number of body biasing gates with about the same power consumption.

4. PART II: ON RANDOM DYNAMIC VOLTAGE SCALING IN CORRELATION POWER ANALYSIS: A GAME-THEORETIC APPROACH

With the massive deployment of mobile devices and sensor networks, the demand of low powered devices with high security becomes more and more urgent. RDVS technology is a very efficient method to improve the resistance of cryptosystem against side-channel attacks. In this section, it is demonstrated that the resistance can be undermined by providing lower off-chip power supply voltage. In order to address this issue, the following method is proposed in this dissertation. The off-chip power supply voltage will be monitored, and an alarm could be triggered to protect valued information once the power supply voltage is lower than the expected voltage (threshold voltage). However, considering the maintenance cost and the environment noise on power supply voltage, this problem is further formulated as a non-zero sum game model, and the attacker and the circuit supplier (defender) are the players of this game. The analysis of the Nash equilibrium in this game shed light on the choice of the optimal threshold voltage based on parameters of cryptosystem including the value of information, denial-of-service cost, etc.

The remainder of this section is organized as follows: Section 4.1 introduces the basic background and motivation, and Section 4.2 reviews cryptography and CPA, together with the motivation of the proposed game theoretic approach. Section 4.3 details the proposed game theoretic approach and Section 4.4 presents the experimental results. Concluding remarks are given in Section 4.5.

4.1. INTRODUCTION

With the massive development of mobile devices like phones and tablets as well as sensor networks for civil and military applications, securing cryptographic device against SCA has become a very hot research topic in recent years. One of the most popular cryptographic devices used in these security sensitive devices is the smart card, which provides the security identification and authentication for those applications. How to secure these cryptographic devices from various attacks has grown to become an active research topic recently.

Conventional security measures employ various cryptographic algorithms, including Advanced Encryption Standard (AES), Data Encryption Standard (DES) [34] [35], RSA (named after its inventors Rivest, Shamir and Adleman) [36]-[38] and Elliptic Curve Cryptography (ECC) [39] [40]. Theoretically, those algorithms shall provide strong resistance against cyber attacks. However, when they are implemented in hardware, side-channel attackers can still get the correct key easily.

One early example of side-channel attack is the DPA [41], which computes a differential trace based on a hypothesis of the secret key to manifest the points where the key bits were manipulated. It can directly reveal the private key or significantly narrow down the key search space. DPA later evolved and the CPA was developed. With details reviewed in Section 4.2.1, CPA is more efficient and effective compared with DPA.

The only way to resist side-channel attack is to enhance hardware implementation and reduce power peaks and correlations. Various works exist in this regard [42]-[45]. However, most of them are deterministic in nature. In other words, the same plaintext always results in the same power trace under a given key. Accordingly, attackers can still repeat the same pattern multiple times to accumulate small deviations in power traces to assist the attack.

An early attempt to address this issue through random precharging was discussed in [46], which is still not completely secure against DPA. Random Dynamic Voltage and Frequency Scaling (RDVFS) as a countermeasure was proposed in [47], which did enhance the resistance of the cryptosystem to power attacks. It is later demonstrated in [48] that RDVS alone provides better resistance against DPA, and the evaluation about the effectiveness of RDVS against CPA was given in [49].

However, according to the analysis given below, the attacker can improve the performance of power analysis by providing lower power supply voltage to RDVS cryptosystem.

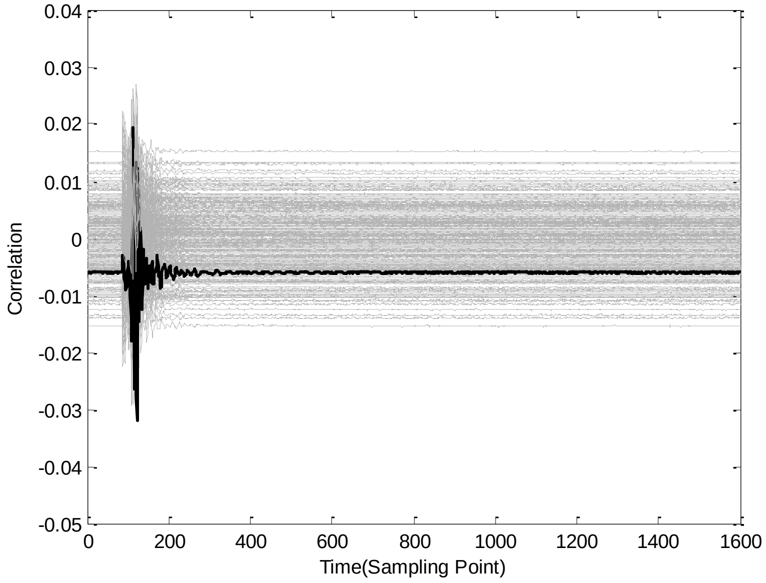
The following Figure 4.1 (a) and Figure 4.1 (b) demonstrate the impact of scaling range on CPA resistance. In Figure 4.1, the X-axis represents the indices of the power trace samples, and the Y-axis represents the correlation coefficients of each hypothetical key. The correct key is plotted in black, and all other keys are plotted in grey. For CPA, the instantaneous power consumption during the DATA transitions is the most important information, which means the highest correlation peaks happen at the data transition. Therefore, if the correlation coefficients of the correct key (black) are higher than those of other keys (grey) at this time, CPA is successful. It can be observed that that larger scaling range in RDVS can improve the CPA resistance. In contrast, lower voltage scaling range decreases the resistance of system to power attack. Therefore, this allows the possibility that the attacker can decrease the voltage scaling range by manipulating the DC-DC converters in order to facilitate the success of CPA.

For many DC-DC converters including the switched-capacitor DC-DC buck converters, and inductive buck converter , there exist a linear relationship between output and input, and it will be demonstrated that lower supply voltage to DC-DC converters will give smaller output voltage range (i.e. smaller scaling voltage range in RDVS) in Section 4.2.2. Therefore, the attackers can retrieve the secret key of the system more quickly by using lower attack power supply. In order to improve the resistance to CPA of RDVS cryptosystem under low supply voltage, the following method is proposed in this dissertation. Firstly, the off-chip power supply voltage will be monitored, and an alarm could be triggered to protect valued information once the power supply voltage is lower than the expect

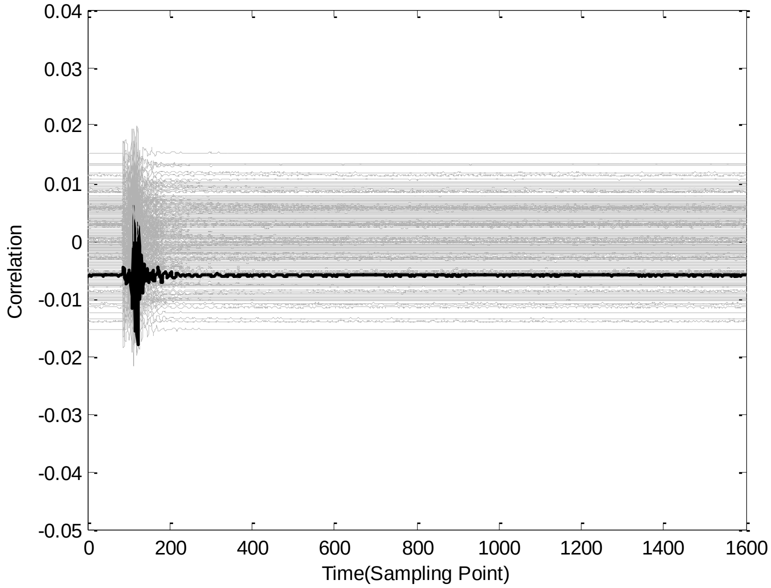
In theory, larger threshold voltage could help filtering the power attacks, however, normal changes on supply voltage caused by noise may be also taken as power attacks, which will cause loss of defenders. In fact, either for the defender or the attacker, both the threshold voltage and the attack power supply voltage impact their gain or loss. The strategy of the defender is to take different threshold voltages and corresponding strategy of the attacker is to use different attack power supply voltages. The problem between the attacker and the defender can be formulated as a game, and it is very interesting to figure out the optimal strategy for both the defender and the attacker. Therefore, in this paper, a game theoretic approach is proposed to analyze the problem between the attacker and the defender in RDVS power attack resistant cryptosystem.

4.2. PRELIMINARIES

With the massive development of mobile devices like phones and tablets as well as sensor networks for civil and military applications, securing cryptographic device



(a)



(b)

Figure 4.1 CPA result with Temporal DVS (a) 1.8/1.5V Temporal DVS. (b) 1.8/1.2V Temporal DVS

against SCA has become a very hot research topic in recent years. One of the most popular cryptographic devices used in these security sensitive devices is the smart card, which provides the security identification and authentication for those applications. How to secure these cryptographic devices from various attacks has grown to become an active research topic recently.

4.2.1. Cryptographic System and CPA. Symmetric key ciphers have simpler and faster encryption and decryption processes. However, because of their shared key policy, they are more prone to attacks. Hence, generally keys are distributed between two channels using more secured and computationally intensive asymmetric key ciphers, and information is shared using symmetric key ciphers. AES and DES are some of the well-known symmetric key ciphers. The best known asymmetric ciphers include RSA and ECC.

CPA is a more effective attacking method compared with DPA [50]. It analyzes the correlative relationship between the plaintext/cipher-text and instantaneous power consumption of the cryptographic device. In order to do CPA attack, a power model (e.g. Hamming Weight (HW) model) [54] needs to be built to predict the power consumption in terms of hypothetical keys and input/output data; then, the predicted power is compared with the measured power using a Person correlation coefficient algorithm [51]. If CPA is successful, the correlation coefficients between the predicted power and the measured power will be significantly higher when the hypothetical key is the secret key [52].

4.2.2. Design Description. In the previous work [49] [52], an AES S-Box in VHDL has been designed to demonstrate the effectiveness of RDVS. Here, this S-Box is used again as the vehicle to verify the proposed game theoretic approach in this paper. The designed S-Box is synthesized in Xilinx ISE EDA tool, and is downloaded on an evaluation FPGA board (SASEBO-GII), which is designed and developed for the purpose of side-channel attack experiments by the Research Center for Information Security (RCIS) in Tokyo, Japan [53].

By measuring the voltage across a shunt resistor of the FPGA board, the current of the FPGA can be gotten. By multiplying it with the core voltage of the FPGA, the power consumption can be obtained. Such an approach improves the efficiency to obtain realistic data compared with software simulation.

The collected power data, measured under constant supply voltage, is then processed through MATLAB to include the RDVS based on the common knowledge that power scales quadratically with the supply voltage [54]. This mimics the real S-Box design with RDVS. The supply voltage is chosen randomly from a number of pre-set voltages, which can either be provided externally or obtained through on-chip DC-DC converters, based on a random number generator (RNG).

4.2.2.1 RDVS System. The RDVS Cryptosystem diagram is shown in Figure 4.2, where the power gird of the system is chosen randomly from four different power grids according to the RNG. The source of power grids are outputs of DC-DC converters and off-chip power supply, and the simplest switched-capacitor DC-DC converter is the best choice in this application considering the balance between cost and performance.

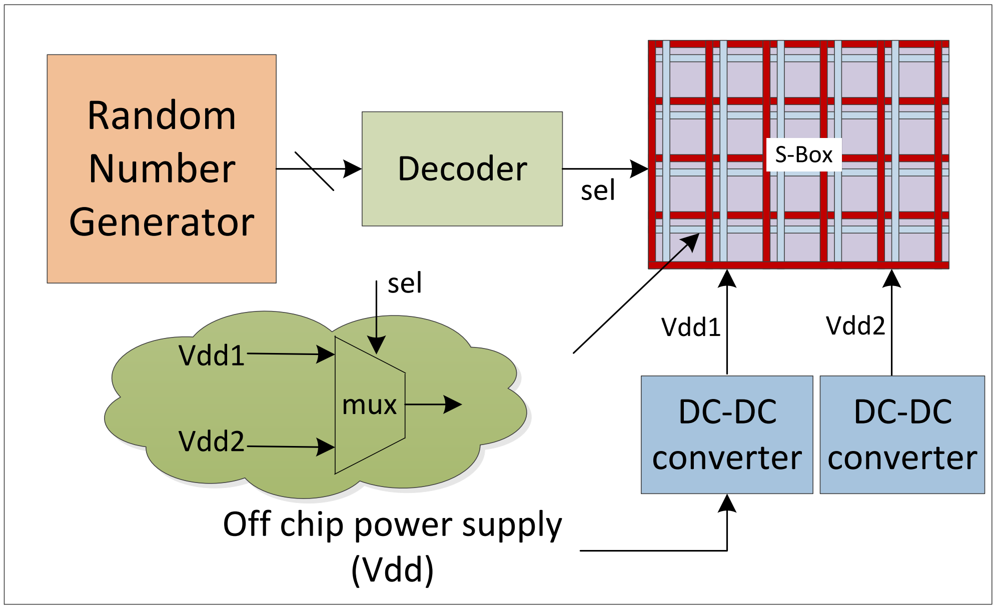


Figure 4.2 System diagram of RDVS S-Box.

4.2.2.2 Random number generator. To implement RNG, a simple Linear Feedback Shift Register (LFSR) counter [55] is explored here, which is easy to implement with standard cell libraries in real circuits. The simulation result is shown in

Figure 4.3, where the period of pseudo random number generated is 256. More shift registers can provide stronger randomness, but at the cost of area overhead. On the other side, for some applications such as smart cards, RNGs with stronger randomness may be readily available and can be directly utilized. In previous experiments, it has been demonstrated that stronger randomness can significantly improve CPA resistance [48].

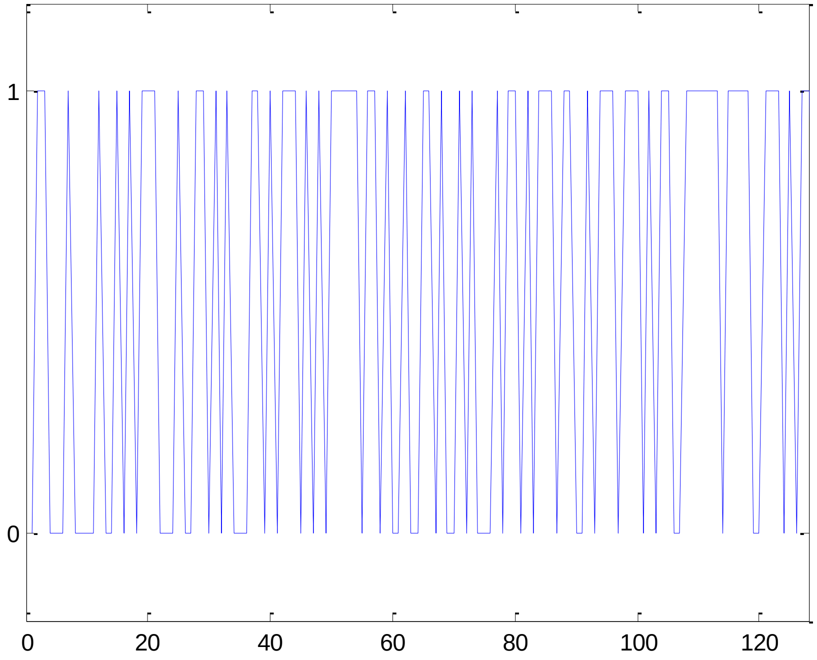


Figure 4.3 Part of the random number generator cycles of LSFR.

4.2.2.3 DC-DC converter. There are a lot of linear DC-DC converters. and here the DC-DC converter from [57] is chosen, which is shown in Figure 4.4. There are two DC-DC converters in the system, and they share the same structure with different resistors and inductors to obtain different output voltages. The relationship between the output voltage and input voltage is linear for this DC-DC converter [57], therefore, it is proved that the voltage range decreases by providing lower power supply voltage (input voltage of the DC-DC converters). Therefore, the following (4.1) can be gotten as:

$$\begin{aligned}
&\frac{V_{dd1}}{V_{dd}} = \frac{V_{dd1}'}{V_{dd}'} = a \\
&\frac{V_{dd2}}{V_{dd}} = \frac{V_{dd2}'}{V_{dd}'} = b \\
&\begin{cases} a < b < 1 \\ V_{dd} > V_{dd}', \end{cases}
\end{aligned} \tag{4.1}$$

where, V_{dd} is the normal power supply voltage and the input voltage of DC-DC converter, and V_{dd1} and V_{dd2} are the outputs of DC-DC converter under V_{dd}. V_{dd}' is the lower power supply voltage used by the attacker, and V_{dd1}' and V_{dd2}' are the corresponding output voltage of DC-DC converter. a and b are the coefficients of DC-DC converter, which are smaller than 1. Then the voltage range of RDVS S-box can be gotten, which shows that the lower power supply voltage V_{dd}' leads to smaller voltage range.

$$\begin{aligned}
&V_{dd2} - V_{dd1} = bV_{dd} - aV_{dd} = V_{dd}(b-a) \\
&V_{dd2}' - V_{dd1}' = bV_{dd}' - aV_{dd}' = V_{dd}'(b-a) \\
&\because \begin{cases} V_{dd} > V_{dd}' \\ 0 < b - a \end{cases} \\
&\Rightarrow V_{dd}(b-a) > V_{dd}'(b-a) \\
&V_{dd2} - V_{dd1} > V_{dd2}' - V_{dd1}'
\end{aligned} \tag{4.2}$$

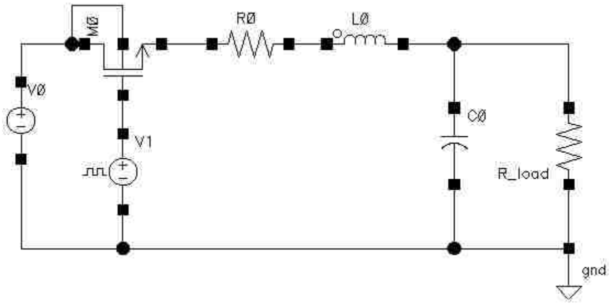


Figure 4.4 DC-DC converter

4.2.2.4 Power supply voltage monitor. Considering the fact that the change of power supply voltage will impact the gate delay, the power supply voltage can be monitored by measuring the gate delay. The idea of critical-path monitor (CPM) is borrowed here to monitor the change of gate delay, which was proposed to measures critical path delay and the effects of noise and localized Vdd droops on delay in [59]. Due to the fact that many timing paths may be critical, the CPM in [59] uses a small number of delay paths with different delay versus process, voltage and temperature curves to synthesize the critical paths. However, in this dissertation, only changes of power supply voltage need to be monitored, and do not have to synthesize the critical path.

Therefore, a simplified power supply voltage monitor (PVM) is designed here based on the structure of CPM, which has less power and area overhead. The block diagram of VPM is shown in Figure 4.5, which is mainly composed of three parts: AND path delay unit, edge detector and comparator. The minimum resolution of the edge detector is one FO2 delay. The threshold voltage is coded to 6-bit, and will be the reference input of the comparator. For the edge detector, taking the 1-to-0 transition of B as an example, the outputs of inverters is (0)101010, where the first 0 in brackets is value of B. When the rising edge of B arrives and just propagates the first two inverters, the output of inverters is (1)011010. Therefore, the edge of B could be captured on the rising edge of the clock and the code based on the position of edge of B is gotten after the XOR/XNOR array, which indicates the AND path delay.

4.3. GAME MODEL AND ANALYSIS

Considering the fact that lower power supply voltage results in smaller scaling range, which decreases the resistance of system to power attack [49], the attacker can improve the efficiencies of attack by using a lower power supply voltage. In order to verify this point, some experiments will be firstly presented. Figure 4.6 shows the CPA analysis of the same RDVS S-Box system for 1.7v power supply and a lower 1.5v power supply voltage. The X-axis represents the number of traces and Y-axis represents the correlation value at the time of data transients. The correct key is plotted in black, and other keys are plotted in grey. For the attack using 1.7v power supply, the correlation coefficient of the correct key is buried by the correlation coefficient of other keys even

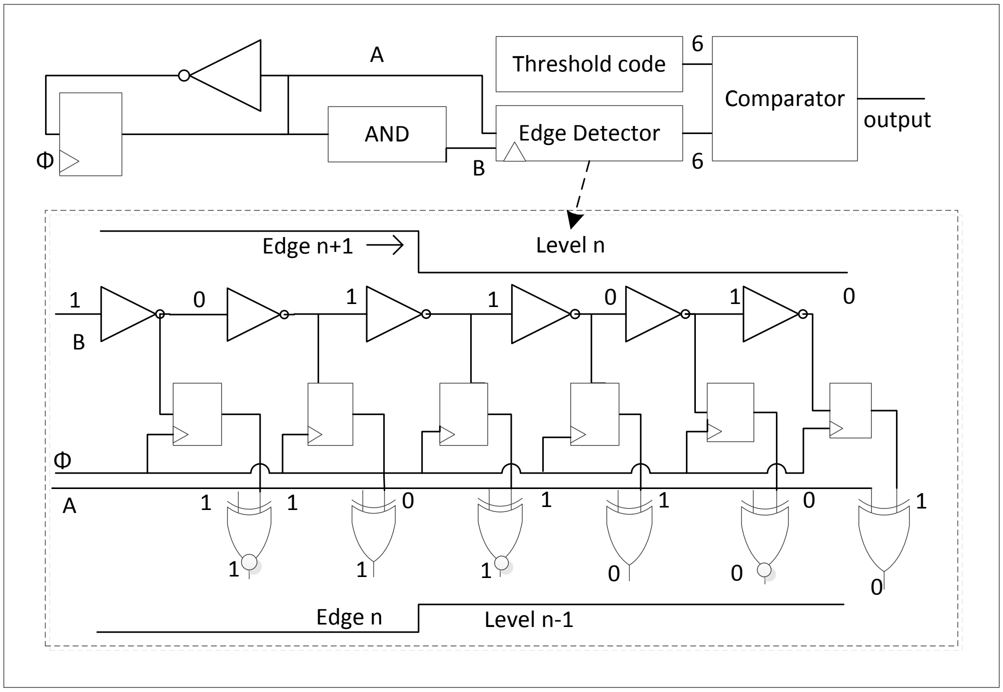
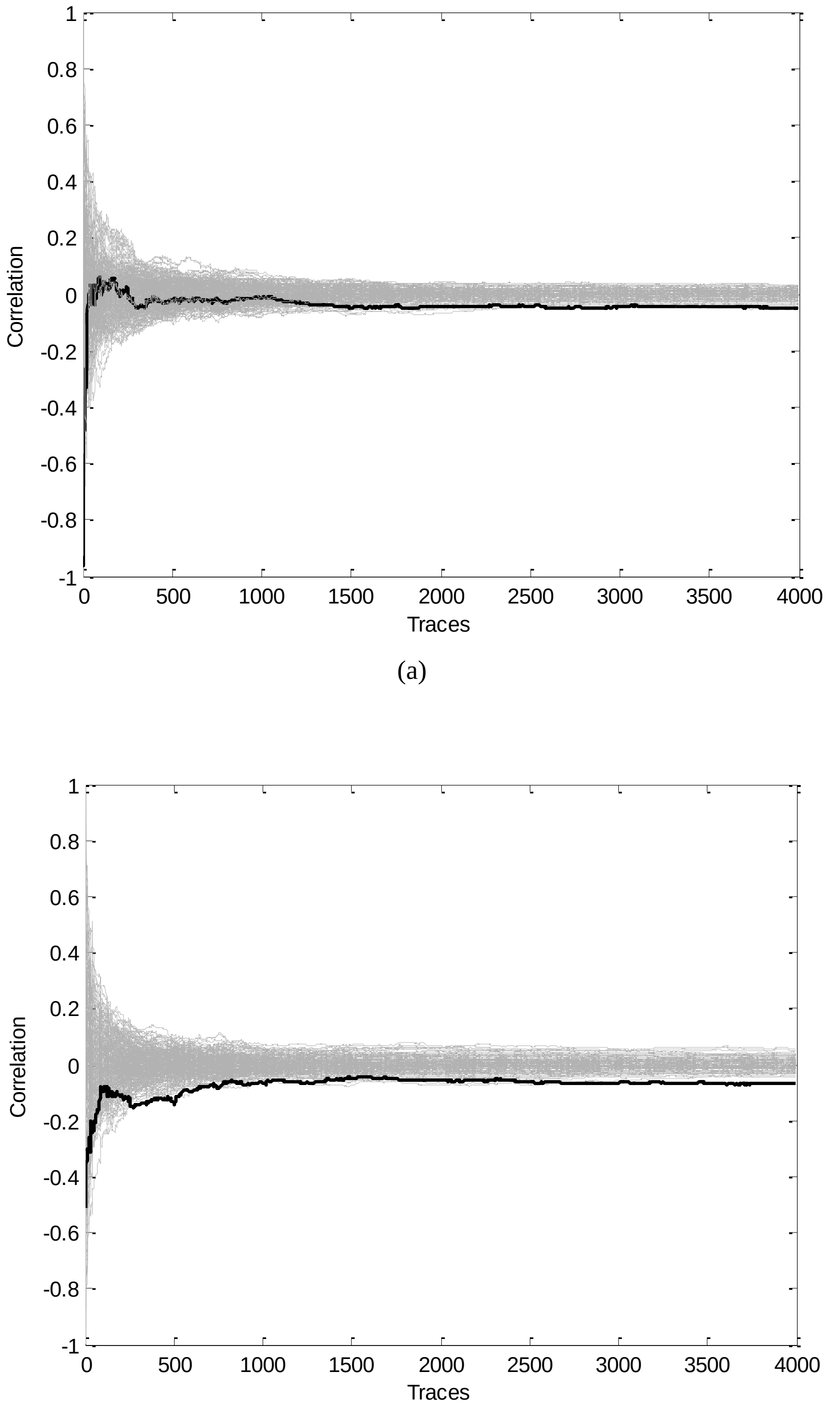


Figure 4.5 Block diagram of the VPM and edge detector

the number of power traces achieves 4000, which means that CPA would fail. However, for the attack using 1.5v power supply, the correct key stands out from other keys when the number of traces is around 3500. Therefore, it demonstrates that lower power supply voltage makes the attack easier, and it is very necessary for the defender to take this issue into account when using DVS in cryptosystem.

In all, the problem can be summarized as below: the designer/user of cryptosystem is the defender, and their purpose is to avoid information leakage in power trace at lower attack voltage by monitoring the supply voltage. On the other side, the attacker will try to get the secret key using CPA approach by decreasing the voltage scaling range without triggering the alarm, and the gain of the attacker definitely is the loss of the defender. Both the defender and the attacker are trying their best to maximize their own gain. Therefore, the problem could be formulated as a non-cooperative game between the attacker and the defender, and the following assumptions are given before introducing the strategies of the attacker and the defender.



(a)

(b)

Figure 4.6 Number of traces needed for successful CPA attack (a) for 1.8v power supply attack. (b) for 1.7v power supply attack

Firstly, integrated DC-DC converters are widely employed to convert a single off-chip voltage V0 to multiple on-chip voltages V1, V2, V3, etc. Meanwhile, as mentioned in Section 4.2, the output of DC-DC converter is linear with the input. This assumption is demonstrated both by simulation result in Section 4.2.2 and reference [34].

Furthermore, attackers are rare among all users, which is very reasonable in practice. The attacker's target is to retrieve the secret key of the S-Box, from which he/she can gain access to valuable information of value F. Otherwise, if he fails to gain the information, the gain of the attacker will be 0. It's of the attacker's interest to obtain the secret key as quickly as possible, and the attacker will not give up until they get the secret key. Here, the following behavior of the attacker is assumed, and it will not affect the problem formulation and the game model. Firstly, the attacker will gather K power traces, and then the attacker will try keys one by one following the order of correlation coefficient since the top one key. Accordingly, the cost of attack is composed of two parts, which is the cost of K power traces E and the cost of $R\beta$ from trying keys. Here, E is the cost of gathering K power traces, R is the rank of correct key, and β is the trying cost of per key.

Finally, it is assumed that the noise of power grid is additively Gaussian distributed, with zero mean and known standard deviation, and the noise is long-time noise. Therefore, the noise will not impact on attacker during the short time attacking. However, it does impact on the defender, who needs to maintain the cryptosystem during its lifetime.

Now, the description of the system could be summarized here. The defender built a cryptosystem with RDVS to improve the resistance of cryptosystem to power analysis attack (CPA in this paper). The cryptosystem is composed of RDVS S-Box, random number generator and DC-DC converters, and the diagram of the system is shown in Figure 4.2. On the other hand, the attacker prefers to attack the system using a lower off-chip supply voltage, which reduces the voltage range of RDVS cryptosystem, and makes the attack easier.

In summary, the defender's strategy is: monitor the voltage coming from off-chip and set the threshold to Si, Si is from a threshold sets S= {S1, S2, ..SN}. When the voltage drops below Si, an alarm will be triggered and the secrete key and all information stored in the memory will be erased. However, the alarm can either be triggered by the attacker, or due to environmental noise. When this happens, the chip needs to be returned to the defender, who will then pick up another chip and reprogram a different secret key. To avoid attackers from detecting the threshold Si through multiple trials, the threshold needs to be randomly distributed among chips.

The attacker's strategy is to use a lower off-chip voltage set V(V_1, V_2, …, V_M), and try different keys until the information is obtained. Assume that both the attacker and the defender are rational, and the attacker will not stop until the secret key is gotten. Because, the gain will be definitely negative if the attacker stops attacking, and it is very possible that the secret key goes out in next trying.

The gain for attack and defender is defined respectively as below.

For the attacker, the following situations are defined:

1) Gain is G .

If the alarm of cryptosystem is triggered, it will cause maintenance cost of the defender, and it is defined as the denial-of-service cost G, which is positive value. Under this situation, the gain of attacker is G.

2) Gain is $F-(E+R\beta)$.

If the alarm is not triggered, the attacker will not stop until the secret key is gotten, and the cost of attack includes power trace measurement cost E and key trying cost $R\beta$. Therefore, the gain of attacker is $F-(E+R\beta)$ under this case.

NOTE 1: The probability of failed attack is zero. Failed attack is defined as, the attacker does not get secret key and the effort has already be spent. That is because, if the probability is not zero, which means that cryptosystem is strong enough to resist power attack analysis.

NOTE 2: F is larger than G, otherwise it makes no sense to detect the attacker.

On the other hand, the gain for the defender is defined as:

1) DS1: Gain is $-G$.

When alarm is triggered because of the attacker, it will result in the denial-of-service cost to the defender, in the form of $-G$.

2) DS2: Gain is $-PC$.

PC is the cost when the alarm is triggered due to the noise, where P is the probability of the alarm is triggered due to the noise, and C is the denial-of-service cost if the system is down due to noise. Note, for defender, whatever the alarm is triggered by attacker or the noise, the denial-of-service cost should be the same. Considering our model is built between one attacker and the defender, the denial-of-service cost caused by attacker just happen to one single device. However, the denial-of-service cost due to noise could happen to all devices is the cryptosystem is network with multiple devices. Therefore, C may be either G if the cryptosystem is a single device system or $K \times G$ when the system is network, where is the number of components of the network. NOTE 3: Just as mentioned above, the noise of power grid is long-time noise, and it only impacts the defender but not the attacker.

Both the attacker and the defender are rational. The attacker runs CPA analysis to maximize the gain of attack, while the defender looks for optimal threshold voltage to minimize their loss. Assume that the defined costs above are common knowledge between the attacker and the defender. More precisely, both the attacker and the defender share the following knowledge: Denial-of-service cost G, Value of information F and Parameters of attacker's cost E, β.

Now, a simple example is used to show how the game theoretic approach works in this section. Without loss of generality, assume the defender has two (N=2) different threshold voltages, and

$$S = \{S_1, S_2\} = \{\text{low}, \text{high}\} \tag{4.3}$$

The attacker also has 2 (M=2) different attack voltages

$$T = \{T_1, T_2\} = \{\text{low}, \text{high}\}. \tag{4.4}$$

According to the gain definition, the normal form of the game between the attacker and the defender can be gotten in Table 4.1. For $\{T = \text{low}, S = \text{low}\}$, the attacker voltage T1 equals to the threshold voltage S1, and the low supply voltage alarm will be triggered. Therefore, the gain of attacker is G. On the other hand, the gain of defender is $-G - P_{\text{low}}C$, where P_{low} the probability of the alarm is triggered due to the noise under low threshold voltage. For $\{T = \text{low}, S = \text{high}\}$, the attacker voltage T1 is smaller than the threshold voltage S1, which will trigger the low supply voltage alarm, too. The gain of attacker is G, and the gain of defender is $-G-P_{high}C$, where P_{high} is the probability of the alarm is triggered due to the noise under high threshold voltage. For $\{T=high, S=low\}$, the attacker voltage T2 is larger than the threshold voltage S1, and the secret key could be gotten after K power traces are gathered. Under this situation, the gain of attacker is $F-(E+R\beta)$, and the corresponding gain of the defender is $-F-P_{low}C$. Just like $\{T=low, S=high\}$, same gain for attacker and defender can be gotten under $\{T=high, S=high\}$.

Table 4.1 Normal form of the game

Defender Attacker	Low	High
Low	G, $-G-P_{low}C$	G, $-G-P_{high}C$
High	$F-(E+R\beta)$, $-F-P_{low}C$	G, $-G-P_{high}C$

Note, this model still works when considering any other different sets S and T, and M do not have to equal to N.

The Nash equilibria are gotten by analyzing the Table 4.1, and there are total three different possible cases in this example.

Note, there is $P_{low} < P_{high}$ in Table II.

1). Case 1:

If $G \geq F-(E+R\beta)$, it means that the denial-of-service cost is high, or the cost attack is high. Under this case, the attacker will try to create denial of service, and the gain of attacker G is maximized. Therefore, (low, low), (low, high) and (high, high) may be Nash equilibria. On the other hand, the defender tries to avoid the denial of service due to noise, and considering that $P_{low} < P_{high}$, (low, low) is the Nash equilibria 1 (NE1).

2.) Case 2:

If $G < F - (E + R\beta)$ & $-F - P_{low}C \geq -G - P_{high}C$, it may happen when the denial-of-service cost is low or the cost per trial is low, and the information value is high. The attacker tries to use a high supply voltage to guarantee the success of attack, and considering $-F - P_{low}C \geq -G - P_{high}C$, the defender also tries to use a low threshold voltage to achieve the gain $-F - P_{low}C$. Therefore, in this case, (high, low) is the Nash equilibria 2 (NE2).

Note, for this case, the following conditions should be satisfied:

$$\because \begin{cases} G < F - (E + R\beta) \\ -F - P_{low}C \geq -G - P_{high}C' \end{cases} \tag{4.5}$$

$$\Rightarrow (P_{high} - P_{low})C > E + R\beta, \tag{4.6}$$

which means only when parameters satisfy relationship of (6), it is possible to achieve NE2.

3.) Case 3:

If $-F - P_{low}C < -G - P_{high}C$, the defender will try to use high threshold voltage to maximum the gain, then either (low, high) named as NE3 or (high, high) named as NE4 could be Nash equilibria.

Note, the condition of case 1 and case 3 may be true at the same time, which means both NE1 and NE3/NE4 may be Nash equilibria for some cases. Therefore the game model will be a mixed strategy game, and this will be proved in Section 4.4 using the given example.

4.4. NUMERICAL RESULTS

Different applications and cryptosystems mean different parameters (G, F, E, β, C). In order to observe the effect of different parameters on Nash equilibria, CPA experiments are run based on analysis in Section 4.3.

In the CPA experiments, 1.8/1.5V RDVS S-box is used as the cryptosystem, and simulations are run following the attacker behavior defined in Section 4.3. In the experiments, 5000 power traces are gathered firstly following the way mentioned in Section 4.2. Then the rank of secret key is gotten by sorting result of CPA analysis in Matlab. The experimental results are summarized in Table 4.2, where T is the power supply voltage using by the attacker. 1.8v is included here also as reference, and the rank of 1.8v is 110, which is much larger than the rank using lower power supply voltage and demonstrates the high efficiency of CPA using lower power supply. All following experiments are using the values of R in Table 4.2.

Table 4.2 Rank of secret key in CPA analysis to DVS S-box

T(v)	R (Rank of secret key)
1.8	110
1.7	49
1.5	23

In order to observe the effect of each parameter on Nash equilibrium, the value of parameters are chosen as shown in Table 4.3, which is obtained based on the S-box system, and use 1.5v/1.7v as the low/high voltage. Each time, only one parameter is changed and the rest of parameters are fixed to value in Table 4.3. Therefore, five different experiments are done in the following Sub-section 4.4.1.

Table 4.3 Value of parameters

Parameter	Meaning of Parameter	Value
G	Denial-of-service cost	3
F	Value of information	5
C	Denial-of-service cost caused by noise	60
β	trying cost of per key	0.005
E	cost of 5000 power traces	0.5

Furthermore, considering that the noise of power grid is additive with zero mean Gaussian distribution, the value of $P_{1.5}$ (P_{low}) and $P_{1.7}$ (P_{high}) can be gotten by setting the standard deviation $\sigma = 0.05$, which is reasonable considering most of the noise will not exceed $[-3*0.05v, +3*0.05v]$, i.e. $[-0.15v, +0.15v]$.

$$P_{1.5} \approx 0.00 \tag{4.7}$$

$$P_{1.7} \approx 0.02 \tag{4.8}$$

4.4.1. Mixed Nash Equilibrium Analysis. Substitute $P_{1.5}, P_{1.7}, R$ in Table 4.2 and Table 4.3 into Table 4.1, it is verified that the game does not admit a pure strategy Nash equilibrium. If the defender chooses 1.5v as the threshold voltage to minimize the loss under (1.5v, 1.5v), however the attacker will use 1.7v to achieve the maximum gain under 1.5v threshold voltage, which makes the defender suffer more loss (-5). Therefore, the defender then goes to 1.7v as the threshold voltage, and the Nash equilibria will be either NE3 or NE4.

Two mixed strategy Nash equilibria are calculated by the game solver from [58] that they are denoted as (5) and (6). The mixed-strategy Nash solution for the attacker is shown in Figure 4.7, on the other hand, the defender just has one strategy (1.7v) as shown in (5) and (6). In all the two possible Nash equilibria, the gain of the attacker is 3.0 while the gain of the defender is -4.2.

$$\{0.4(1.5) + 0.6(1.7); 1.0(1.7)\} \tag{4.9}$$

$$\{1.0(1.7); 1.0(1.7)\} \tag{4.10}$$

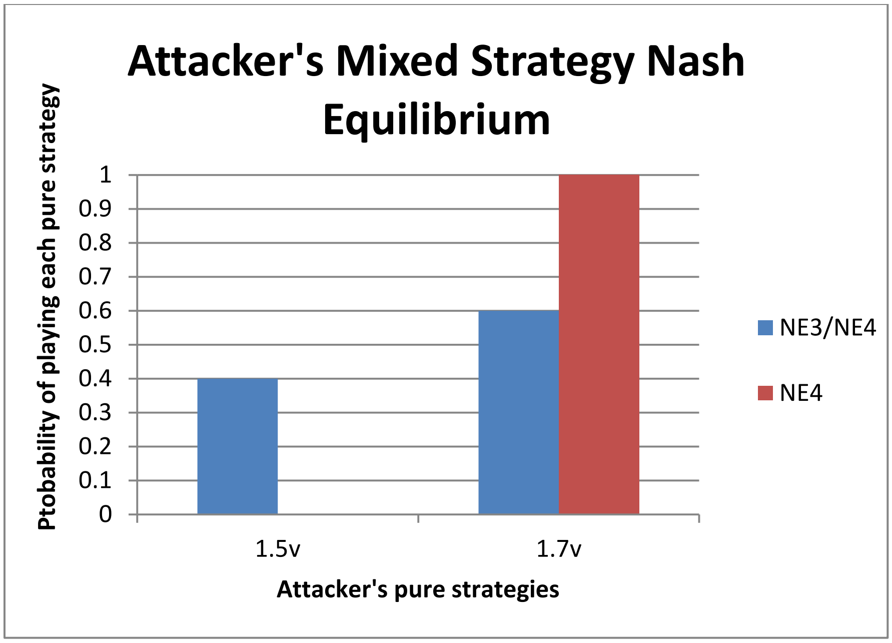


Figure 4.7 Attacker’s mixed strategy Nash equilibrium.

4.4.2. Effect of Parameters on Nash Equilibrium. Still using the same parameters in Table 4.3, and above $P_{1.5}, P_{1.7}, R$ in Table 4.1, experiments are done to observe the effect of parameters on Nash equilibrium. All following results are from game solver in [58].

Taking G as the variable, and the other parameters as constant with values in Table 4.3, Figure 4.8 is presented to show the Nash equilibria and gain of attacker/defender v.s. G. According to Figure 4.8, the Nash equilibria will be NE1 when increasing G, and NE1 will also be a possible Nash equilibria when β and E are increased as shown in Figure 4.9 and Figure 4.10. This experimental result matches the analysis about case 1 in Section 4.3, which means when denial-of-service cost or the cost attack (β and E) is high, and Nash equilibria is NE1. On the other hand, the gain of attacker increases with G, in contrast the gain (loss) of the defender decreases (increases) with G, which means the defender should try to decrease the denial-of-service cost G in order to minimize the loss. Please note, when β / E is larger than 0.03/1.755, there are three possible Nash equilibria NE3, NE4 and NE1, therefore, the gain of attacker is always 3.0, and the gain of the defender could be either -3.0 or -4.2.

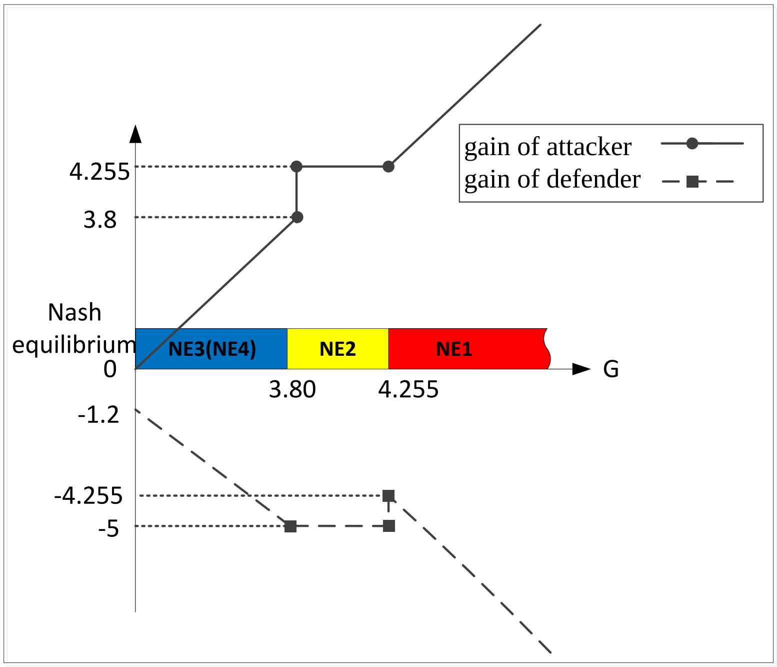


Figure 4.8 Nash equilibria v.s. G.

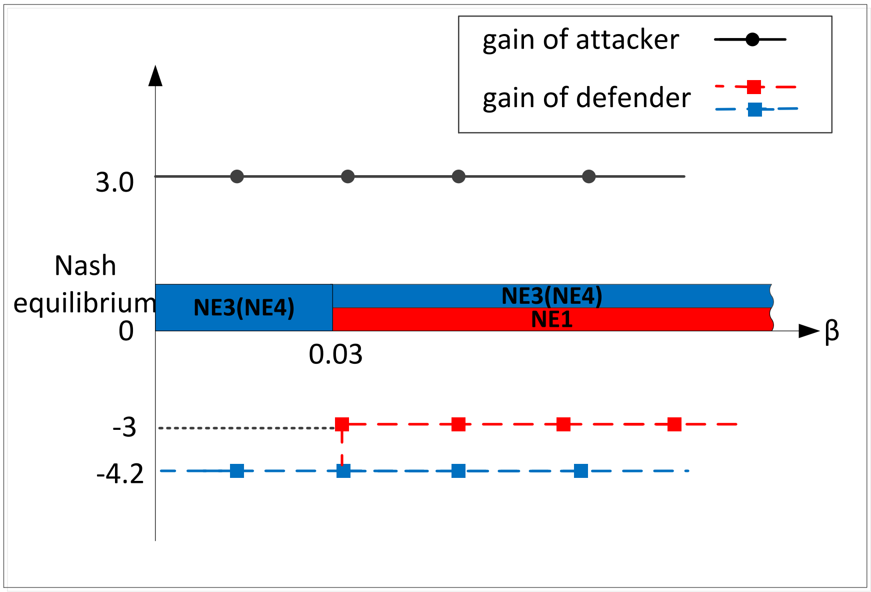


Figure 4.9 Nash equilibria v.s. β.

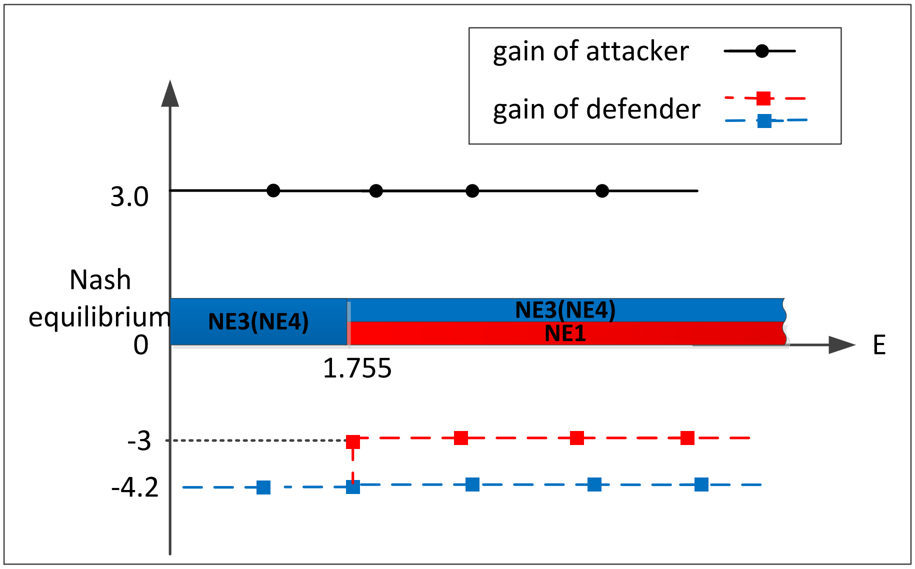


Figure 4.10 Nash equilibria v.s. E.

When information value F is very high, the gain of defender $-F-P_{1.5}C$ under (1.7v, 1.5v) will be far smaller than other gains, therefore, defender will choose 1.7v as the threshold voltage to avoid heavy loss, which means NE3(NE4) will be Nash equilibria as in Figure 4.11. When F is large enough, the gain of attacker and the loss of defender keep constant, i.e., 3.0 as the gain of attacker and 4.2 as the loss of defender, which are shown in Figure 4.11.

According to Table 4.3, when the cost caused by noise C is large, $-G-P_{1.7}C$ decreases, the defender will use low threshold voltage under this situation, and the Nash equilibria may be either NE1 or NE2. Then the attacker will decide the equilibrium based on their gain, in this example, gain of attacker under NE2 is larger than under NE1. Therefore, with the increase of C, NE2 will be the Nash equilibria as shown in Figure 4.12.

Based on above experimental results, the gain of attacker and defender will keep constant with increasing F, E, β, C except for G. Therefore, it is very important for the defender to minimize the denial-of-serve cost G.

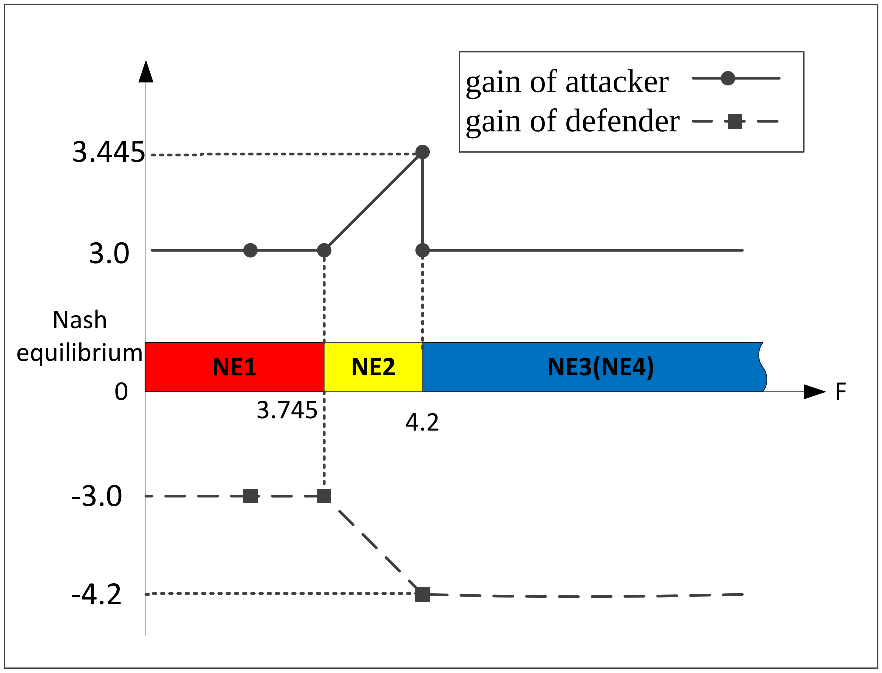


Figure 4.11 Nash equilibria v.s. F.

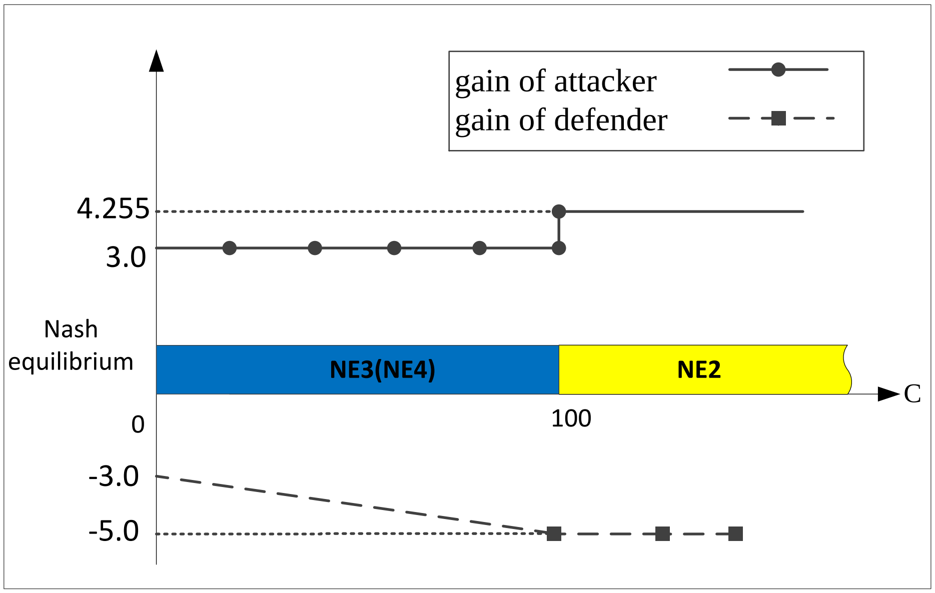


Figure 4.12 Nash equilibria v.s. C.

4.5. CONCLUSION

Based on the observation in [49], it is demonstrated that the attacker could improve the effectiveness of CPA to RDVS S-box cryptosystem by using low power supply voltage. On the other hand, the defender could protect the cryptosystem by setting threshold voltage to trigger an alarm when the power supply voltage is low. Furthermore, the problem between the attacker and the defender is formulated as a game, and a game-theoretic methodology is proposed to analyze both the optimal attacking power supply voltage for the attacker and the detection threshold voltage for the defender.

Theoretical analysis is done to calculate the Nash equilibria, which is also verified using a game solver in this dissertation. The following important properties are gotten based on the analysis:

1) When denial-of-service cost or the cost of attack is high, Nash equilibrium prefers to locate at low threshold voltage and low attack power supply voltage (NE1 in the example).
2) When information value F is very high, high threshold voltage and high attack power supply voltage goes to Nash equilibria (NE4 in the example).
3) Only when the cost of alarm triggered due to noise is high, NE2 may be the Nash equilibria, which represents the attack behavior.

Based on these properties, only when the condition of (3) is true, the attacker will attack the system, therefore, the defender can avoid attack under most of cases by choosing right threshold voltage.

Another important result of the proposed methodology is that the loss of the defender is decided by the value of secret information, and the payoff when the alarm is triggered, which is already known and can be decided by the defender. Furthermore, the probability of alarm trigged by noise for different threshold voltage can be also derived, and then the defender will know the exact range of the loss under different threshold voltage. Therefore, the proposed approach could help the defender find the conditions that limit their loss to expected value for different cryptosystems with different information value.

5. SUMMARY AND FUTURE WORK

5.1. SUMMARY OF CONTRIBUTIONS

The first part of this dissertation studies and solves the timing violation problem caused by process variation in sub-threshold designs. Sub-threshold designs are a popular option in many energy constrained applications. However, a major bottleneck for these designs is the challenge in attaining timing closure. Most of the paths in sub-threshold designs can become critical paths due to the purely random process variation on threshold voltage, which exponentially impacts the gate delay. In order to address timing violations caused by process variation, post-silicon tuning is widely used through body biasing technology at the cost of heavy power and area overhead. Therefore, it is imperative to select only a small group of the gates with body biasing for post-silicon-tuning.

Existed body biasing schemes with multiple body biasing voltage domains and multiple body biasing voltage levels involve not just routing overhead but also design control complexity. Therefore, the first contribution of this dissertation is to explore the possibility of using only one body bias voltage domain with a single body bias voltage in sub-threshold design. This selective body biasing scheme only involves modest area overhead based on the experiment result, and no extra control circuit is needed. Therefore, the proposed scheme is very practical for low cost and low power design.

In the first work, the selective body biasing problem is formulated as an optimization model with statistical inequality constraints. To the best knowledge of authors, this is the first time that this selective body biasing problem is formulated as a optimization problem, and it is further solved through an adaptive algorithm, which is the second contribution of this dissertation.

In order to achieve a better solution for the selective body biasing problem, the problem is reformulated alternatively as a LSIP problem, and propose an efficient IHCS algorithm to solve the LSIP problem with a finite number of mixed-integer linear programming. Compared with the adaptive filtering approach, experimental results on industrial designs using 65nm sub-threshold library demonstrate that the new IHCS approach can improve the pass rate by up to 7.3x with a speed up to 4.1x, using the same number of body biasing gates and about the same power consumption.

The second part of this dissertation studies the attack and defense problem in power analysis attack to RDVS system. It is demonstrated that the resistance of RDVS system to CPA can be undermined by providing lower off-chip power supply voltage. In order to improve the resistance under lower attack voltage, the defender can insert power voltage monitor into the cryptosystems and trigger an alarm to protect valued information once the power supply voltage is lower than the expected voltage (threshold voltage). The problem is further formulated as a non-zero sum game model, and the attacker and the circuit supplier (defender) are the players of this game. The S-box is used in this dissertation as the vehicle to show how the game model works, but it can be extend to other cryptosystem easily. The analysis of the Nash equilibrium in this game shows valued guidelines to defender for a success protection.

5.2. FUTURE WORK

For the selective body biasing scheme, the future works include but are not limited to the following topics:

1) The current optimization target is power, besides that, the area overhead should also be a very important factor in practice. When considering area overhead, choice of body biased gates will be a strong function of layout due to well design rules, routing complexity, etc. Therefore, in future work, area could be taken into account as the objective function or constraint to achieve the target of low area overhead.
2) Monte Carlo simulations are used to verify the performance of design with selective body biasing, which is very timing consuming. The theoretical analysis of pass rate will be helpful to predict and evaluate the performance of the algorithm. Furthermore, the pass rate in closed form expression can also be involved in the constraint or optimization target.
3) The formulation of selective body biasing is path based and in order to solve the scaling problem, representative path method is used. However, the extraction of the representative path still consumes considerable time for large design. Therefore, block-based formulation could be studied in future work.

For the second part job about the game theoretic approach on RDVS system, the following improvement could be done in future:

4) Only random dynamic voltage scaling is considered in this dissertation, and it is worthwhile to consider random dynamic frequency. The game model can be extended to find out the optimal scaling frequency.
5) There are various SCAs including SPA, DPA, CPA and LPA, etc. Only CPA is considered in this dissertation. It will be meaningful to study the resistance of proposed game theoretic approach to other popular SCAs. Furthermore, all experiments in this dissertation are based on FPGA, and it would be interesting to evaluate the performance based on ASICs.

BIBLIOGRAPHY

[1] Wang, A.; Chandrakasan, A., "A 180mV FFT processor using subthreshold circuit techniques," Solid-State Circuits Conference, 2004. Digest of Technical Papers, IEEE International, vol., no., pp.292,529 Vol.1, 15-19 Feb. 2004.

[2] Bo Zhai; Pant, S.; Nazhandali, L.; Hanson, S.; Olson, J.; Reeves, A.; Minuth, M.; Helfand, R.; Austin, T.; Sylvester, D; Blaauw, D, "Energy-Efficient Subthreshold Processor Design," Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, vol.17, no.8, pp.1127,1137, Aug. 2009.

[3] Kwong, J.; Chandrakasan, A.P., "Advances in Ultra-Low-Voltage Design," Solid-State Circuits Society Newsletter, IEEE , vol.13, no.4, pp.20,27, Fall 2008.

[4] Bo Zhai; Hanson, S.; Blaauw, D; Sylvester, D, "Analysis and mitigation of variability in subthreshold design," Low Power Electronics and Design, 2005. ISLPED '05. Proceedings of the 2005 International Symposium on, vol., no., pp.20,25, 8-10 Aug. 2005.

[5] L. A. P. Melek, M. C. Schneider, C. Galup-Montoro; "Body Bias Compensation Technique for Subthreshold CMOS Static Logic Gates," Symp. Integrated Circuits Systems Design, pp. 7-11, Sep. 2004.

[6] Bo Liu; Pourshaghaghi, H.R.; Londono, S.M.; de Gyvez, J.P., "Process Variation Reduction for CMOS Logic Operating at Sub-threshold Supply Voltage," Digital System Design (DSD), 2011 14th Euromicro Conference on , vol., no., pp.135,139, Aug. 31 2011-Sept. 2 2011.

[7] Tschanz, J.; Kao, J.; Narendra, S.; Nair, R.; Antoniadis, D.; Chandrakasan, A.; Vivek De, "Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage," Solid-State Circuits Conference, 2002. Digest of Technical Papers. ISSCC. 2002 IEEE International , vol.1, no., pp.422,478 vol.1, 7-7 Feb. 2002.

[8] Sathanur, A.; Pullini, A.; Benini, L.; De Micheli, G.; Macii, E., "Physically clustered forward body biasing for variability compensation in nanometer CMOS design," Design, Automation & Test in Europe Conference & Exhibition, 2009. DATE '09. , vol., no., pp.154,159, 20-24 April 2009.

[9] Kulkarni, S.H.; Sylvester, D.M.; Blaauw, D.T., "Design-Time Optimization of Post-Silicon Tuned Circuits Using Adaptive Body Bias," Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on , vol.27, no.3, pp.481,494, March 2008.

[10] Kumar, S.V.; Kim, C.H.; Sapatnekar, S.S., "Body Bias Voltage Computations for Process and Temperature Compensation," Very Large Scale Integration (VLSI) Systems, IEEE Transactions on , vol.16, no.3, pp.249,262, March 2008.

[11] Mostafa, H.; Anis, M.; Elmasry, M., "A Novel Low Area Overhead Direct Adaptive Body Bias (D-ABB) Circuit for Die-to-Die and Within-Die Variations Compensation," Very Large Scale Integration (VLSI) Systems, IEEE Transactions on , vol.19, no.10, pp.1848,1860, Oct. 2011.

[12] http://www-device.eecs.berkeley.edu/bsim/?page=BSIM3.

[13] http://www.eecs.berkeley.edu/~alanmi/abc/abc.htm.

[14] Lin Xie; Davoodi, A., "Representative path selection for post-silicon timing prediction under variability," Design Automation Conference (DAC), 2010 47th ACM/IEEE , vol., no., pp.386,391, 13-18 June 2010.

[15] S. Boyd and L. Vandenberghe, Convex Optimization, 2004: Cambridge Univ. Press.

[16] T. Shao, Y. R. Zheng, and J. Benesty, “An affine projection sign algorithm robust against impulsive interferences,” IEEE Signal Proces.s Lett., vol 17, no. 4, pp. 327-330, Apr. 2010.

[17] Jianming Liu and Steven L. Grant, "An Improved Variable Step-size Affine Projection Sign Algorithm for Echo Cancellation," Signal Processing Conference (EUSIPCO), 2013 Proceedings of the 21st European, pp. 1-5, 2013.

[18] Yuantao Gu, Jian Jin, and Shunliang Mei, “l0 norm constraint LMS algorithm for sparse system identification,” IEEE Signal Proces.s Lett., vol 16, no. 9, pp. 774-777, Sep. 2009.

[19] Jianming Liu and Steven L. Grant, “A new variable step-size zero-point attracting projection algorithm,” in Proc. Signals, Systems and Computers, 2013 Asilomar Conference, pp. 1524-1528, 2013.

[20] P. S. Bradley, and O. L. Mangasarian, “Feature selection via concave minimization and support vector machines,” in Proc. 13th ICML, 1998, pp. 82-90.

[21] D. L. Duttweiler, “Proportionate normalized least-mean-squares adaption in echo cancellers,” IEEE Trans. on Speech Audio Process., vol.8, no. 5, pp.508-518, 2000.

[22] Jianming Liu and Steven L. Grant, “A generalized proportionate adaptive algorithm based on convex optimization,” in Proc. Signals and Information Processing (ChinaSIP), 2014 IEEE China Summit & International Conference on, pp. 748-752, 2014.

[23] http://eigen.tuxfamily.org/

[24] http://code.google.com/p/redsvd/

[25] Han, Jun; Morag, Claudio, "The influence of the sigmoid function parameters on the speed of backpropagation learning". In Mira, José; Sandoval, Francisco. From Natural to Artificial Neural Computation. pp. 195–201, 1995.

[26] Lin Xie; Davoodi, A., "Post-Silicon Failing-Path Isolation Incorporating the Effects of Process Variations," Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on , vol.31, no.7, pp.1008,1018, July 2012.

[27] Mani, M.; Singh, AK.; Orshansky, M., "Joint Design-Time and Post-Silicon Minimization of Parametric Yield Loss using Adjustable Robust Optimization," Computer-Aided Design, 2006. ICCAD '06. IEEE/ACM International Conference on , vol., no., pp.19,26, 5-9 Nov. 2006.

[28] Kulkarni, S.H.; Sylvester, D; Blaauw, D., "A Statistical Framework for Post-Silicon Tuning through Body Bias Clustering," Computer-Aided Design, 2006. ICCAD '06. IEEE/ACM International Conference on , vol., no., pp.39,46, 5-9 Nov. 2006.

[29] Ananthan, H.; Kim, C.H.; Roy, K., "Larger-than-Vdd forward body bias in sub-0.5V nanoscale CMOS," Low Power Electronics and Design, 2004. ISLPED '04. Proceedings of the 2004 International Symposium on, vol., no., pp.8,13, 9-11 Aug. 2004.

[30] N. Halko, P. Martinsson and J. Tropp, &ldquo, “Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions,” &rdquo, SIAM Rev. vol. 53, no. 2, pp. 217-288, 2011.

[31] M. Gu and S.C. Eisenstat. “Efficient algorithms for computing a strong rank-revealing QR factorization,” ssSIAM J Sci Comp, 17:848–869, 1996.

[32] Daisuke Okanohara, Experiments of RedSVD, https://code.google.com/p/redsvd/wiki/English

[33] M. A. Goberna and M. A. Lopez, "Post-Optimal Analysis in Linear Semi-Infinite Optimization, " ser. Briefs in Optimization. Springer, 2014.

[34] Davis, R., "The data encryption standard in perspective," IEEE Communications Magazine, vol. 16, no. 6, pp. 5-9, Nov 1978.

[35] Smid, M.E., and Branstad, D.K., "Data Encryption Standard: past and future," in Proc. of the IEEE , vol. 76, no. 5, pp. 550-559, May 1988.

[36] Rivest, R., Shamir, A., and Adleman, L., "A Method for Obtaining Digital Signatures and Public Key Cryptosystems," Communications of the ACM, Vol. 21, No. 2, pp. 120 - 126, 1978.

[37] Taek-Won, K., "Two implementation methods of a 1024-bit RSA crypto-processor based on modified Montgomery algorithm," in Proc. of the IEEE International Symposium on Circuits and Systems, pp.650-653, 2001.

[38] Cao, Y. Y., and Fu, C., "An Efficient Implementation of RSA Digital Signature Algorithm," in Proc. of the Internatoinal Conference on Intelligent Computation Technology and Automation, pp. 100-103, 2008.

[39] Koblitz, N., "Elliptic Curve Cryptosystems," Mathematics of Computation, Vol. 48, pp. 203-209, 1987.

[40] Wang, Y. B., Dong X. J., and Tian, Z. G., "FPGA Based Design of Elliptic Curve Cryptography Coprocessor," in Proc. of the International Conference on Natural Computation, no., pp. 185-189, 2007.

[41] Kocher, P., Jaffe, J., and Jun, B., "Differential Power Analysis," in Proc. of the Annual International Cryptology Conference on Advances in Cryptology, pp. 388-397, 1999.

[42] Muresan, R., and Gregori, S., "Protection Circuit against Differential Power Analysis Attacks for Smart Cards," IEEE Transactions on Computers, vol. 57, no. 11, pp. 1540-1549, Nov. 2008.

[43] Chari, S., Jutla, C., Rao, J. R., and Rohatgi, P., "Towards Sound Approaches to Counteract Power-Analysis Attacks," in Proc. of the Annual International Cryptology Conference on Advances in Cryptology, pp. 389-412, 1999.

[44] Sundaresan, V., Rammohan, S., and Vemuri, R., "Defense against Side-Channel Power Analysis Attacks on Microelectronic Systems," in Proc. of the National Aerospace and Electronics Conference, pp. 144-150, 2008.

[45] Ahn, M., and Lee, H. J., "Experiments and Hardware Countermeasures on Power Analysis Attacks," in Proc. of the International Conference on Computational Science and Applications, pp. 48-53, 2006.

[46] M. Bucci, M. Guglielmo, R. Luzzi and A. Trifiletti, “A power consumption randomization countermeasure for dpa-resistant cryptographic processors,” Integrated Circuit and System Design, Power and Timing Modeling, Optimization and Simulation. 14th International Workshop, PATMOS 2994, 2004.

[47] Yang, S., Wolf, W., Vijaykrishnan, N., Serpanos, D. N., and Xie, Y., “Power attack resistant cryptosystem design: A dynamic voltage and frequency switching approach,” in Proc. of the conference on Design, Automation and Test in Europe, pp. 64–69, 2005.

[48] Baddam, K., and Zwolinski, M., “Evaluation of Dynamic Voltage and Frequency Scaling as a Different Power Analysis Countermeasure,” Proceedings of the International Conference on VLSI Design, pp. 854-862, 2007.

[49] Hui Geng; Jun Wu; Jianming Liu; Minsu Choi; Yiyu Shi, "Utilizing random noise in cryptography: Where is the Tofu?," Computer-Aided Design (ICCAD), 2012 IEEE/ACM International Conference on , vol., no., pp.163,167, 5-8 Nov. 2012.

[50] S. Guilley, L. Sauvage, F. Flament, V.N. Vong, P.Hoogvorst, and R. Pacalet, "Evaluation of power constant dual-rail logics countermeasures against DPA with design time security metrics," IEEE Transactions on Computers, vol. 59, pp. 1250-1263, Sept 2010.

[51] S. Mangard, E. Oswald, and T. Popp, Power Analysis Attacks-Revealing the Secrets of Smart Cards. Springer, March 12, 2007.

[52] J. Wu, Y. Shi, and M. Choi, "Measurement and evaluation of power analysis attacks on asynchronous S-Box," in IEEE Transactions on Instrumentation and Measurement, Dec. 2012.

[53] http://www.risec.aist.go.jp/project/sasebo/download/SASEBO-GII_Spec _Ver1.01_English.pdf

[54] CMOS Power Consumption and Cpd Calculation; http://www.ti.com/ lit/an/scaa035b/scaa035b.pdf

[55] P. Alfke, “Efficient shift register, lfsr counter, and long pseudo-random sequence generators,” 1996.

[56] Kamhoua, Charles A., Manuel Rodriguez, and Kevin A. Kwiat. "Testing for Hardware Trojans: A Game-Theoretic Approach." Decision and Game Theory for Security. Springer International Publishing, 2014. 360-369.

[57] Wibben, J.; Harjani, R., "A High-Efficiency DC–DC Converter Using 2 nH Integrated Inductors," Solid-State Circuits, IEEE Journal of , vol.43, no.4, pp.844,854, April 2008.

[58] Savani, R.: Solve a Bimatrix Game,http://banach.lse.ac.uk, June 07, 2015.

[59] Drake, A.; Senger, R.; Deogun, H.; Carpenter, G.; Ghiasi, S.; Nguyen, T.; James, N.; Floyd, M.; Pokala, V., "A Distributed Critical-Path Timing Monitor for a 65nm High-Performance Microprocessor," Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International , vol., no., pp.398,399, 11-15 Feb. 2007.

[60] J. Kocher, P. Jaffe and B. Jun, “Introduction to differential power analysis and related attacks,” in Technical Report, Cryptography Research Inc., San Francisco, California, 1998.

[61] A. Bogdanov, “Multiple-differential side-channel collision attacks on AES,” in Cryptographic Hardware and Embedded Systems (CHES), Washington, D.C., USA, August, pp. 30–44, 2008.

[62] S. Borkar, T. Karnik, and V. De, “Design and reliability challenges in nanometer technologies,” in Design Automation Conference, 2004, p.75.

[63] Y. Lu, L. Shang, H. Zhou, H. Zhu, F. Yang, and X. Zeng, “Statistical reliability analysis under process variation and aging effects,” in Proc.Design Automation Conf. (DAC), pp. 514–519, 2009.

[64] Bowman, K.A.; Alameldeen, A.R.; Srinivasan, S.T.; Wilkerson, C.B., "Impact of die-to-die and within-die parameter variations on the throughput distribution of multi-core processors," Low Power Electronics and Design (ISLPED), 2007 ACM/IEEE International Symposium on , vol., no., pp.50,55, 27-29 Aug. 2007.

[65] Mehr, S.K.; Mehr, A.R.A.; Mozaffari, S.N.; Afzali-Kusha, A., "A new block-based SSTA method considering within-die variation," Quality Electronic Design (ASQED), 2010 2nd Asia Symposium on , vol., no., pp.260,263, 3-4 Aug. 2010.

[66] Azizi, N.; Najm, F.N., "Compensation for within-die variations in dynamic logic by using body-bias," IEEE-NEWCAS Conference, 2005. The 3rd International , vol., no., pp.167,170, 19-22 June 2005.

VITA

Hui Geng was born in Anyang, Henan, China. She received her B.S. degree in Electronic Information Technology from Beijing Information Technology Institute, Beijing, China, in 2006, and M.S. degree in Electrical and Communication Engineering from Tsinghua University, Beijing, China in 2009. She joined Beijing Voiceon Technology Co. Ltd, Beijing in March 2010 as an ASIC Design Engineer, where she implemented two co-processor ASICs for speech recognition. She started to pursue Ph.D. degree in Electrical and Computer Engineering at Missouri University of Science and Technology, Rolla, Missouri in August 2011. Her research interests focus on ultra-low-voltage design, IC design, and CAD. In August 2015, she received her PhD in Computer Engineering from Missouri University of Science and Technology. Upon graduation, she will be an ASIC design engineer at Broadcom Corporation, Irvine, CA. in August, 2015