ABSTRACT

YELTEN, MUSTAFA BERKE. Variability and Reliability in Nanoscale Circuits: Simulation, Design, Monitoring, and Characterization. (Under the direction of Michael B. Steer and Paul D. Franzon.)

At the nanoscale, variability of transistor process parameters and time-based reliability degradation (also called device aging) significantly impact circuit performance metrics. It is necessary to individually model the variability of each transistor in a circuit to accurately represent die-to-die and within-die variations. Traditionally, one of several sets of process parameters is used for all transistors in a circuit in what is called process corner analysis. Reduced-order models are developed to rapidly analyze and simulate circuit variability and reliability simultaneously. The main advantage of reduced-order models is integrating multiple parameter sweeps by greatly reducing the dimensionality of a transistor process model to just those parameters subject to significant variability. The surrogate models also include models of aging that are not part of a transistor model. Other effects could also be incorporated.

The drain-source currents of 65 nm n- and p- channel transistors are modeled using surrogate models, neural networks, and support vector machines. The models are in terms of eleven parameters, including a few process parameters that essentially cover the process variation space, terminal voltages, temperature and device age. The reduced-order models are compared on the basis of the model generation and evaluation times, model size, and accuracy. Kriging-based surrogate models and neural network-based models are shown to have sufficient accuracy to perform circuit variability and reliability analysis with Kriging-based models having a shorter evaluation time for smaller circuits and neural network-based models having shorter evaluation time for larger circuits.

Variability in operating conditions of differential amplifiers is accurately captured using the reduced-order models. The effects of die-to-die and within-die variations on voltage transfer characteristics of an XOR gate are presented. Individual contributions of crucial process parameters to drain current mismatch of two transistors are quantified with the largest contributions coming from intrinsic threshold voltages and effective length variations of devices. Reliability degradation in differential amplifiers in the presence of process variations are investigated. Design for reliability guidelines are formulated suggesting that the biasing of analog circuits should be modulated by changing nodal voltages within a circuit to increase device aging resilience.

An analog circuit-based negative bias temperature monitor is designed and implemented in 65 nm device technology yielding a sensitivity of 3.15 V^{-1}. Finally, experimental aging characterization examples have been conducted on major analog and digital building blocks.

Variability and Reliability in Nanoscale Circuits: Simulation, Design, Monitoring, and Characterization

by
Mustafa Berke Yelten

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Electrical Engineering

Raleigh, North Carolina

2011

APPROVED BY:

Mo-Yuen Chow	Joseph B. Tracy Minor Program Representative
Michael B. Steer Chair of Advisory Committee	Paul D. Franzon Co-chair of Advisory Committee

DEDICATION

To my mother Ufuk, my father Muhammet, and my sister Azade,

for their continuous support, patience and love.

BIOGRAPHY

Mustafa Berke Yelten was born on the 10^{th} August 1982, in Istanbul, Turkey. In 2001, he finished his high school education in Istanbul (Erkek) Lisesi. Afterwards, he attended Bogazici University Electrical & Electronics Engineering Department, Istanbul and graduated from there in June 2006, receiving his Bachelor of Science (BS) degree with High Honors. Then he decided to continue his academic endeavors in North Carolina State University Electrical and Computer Engineering Department (NCSU ECE) and came for this purpose to Raleigh, NC in August 2006. He specialized in analog/radio frequency (RF) integrated circuit design and solid state electronics and received his MS degree under the direction of Dr. Kevin G. Gard with his thesis "Theoretical Analysis and Design Methodologies for Low Noise Amplifiers based on Tunable Matching Networks" in July 2008. In August 2008 he started his PhD studies in NCSU ECE department and in February 2009 he joined the project team "Self-Healing Advanced Multifunction Receiver-On-Chip—SHAMROC" under the guidance of Dr. Michael Steer and Dr. Paul Franzon. This project was part of the program "Self-HEALing Mixed Signal Integrated Circuits—HEALICS" co-sponsored by Defense Advanced Research Projects Agency (DARPA) and United States Air Force Research Laboratory (US-AFRL). During his PhD studies, he developed response surface modeling-based and machine learning-based techniques for analog and digital circuit variability and reliability analysis. He also contributed to the algorithmic development for self-healing analog/RF building blocks.

Mustafa Berke Yelten has been a member of the Institute of Electrical and Electronics Engineers (IEEE) since 2003 and he has been inducted into the highly selective academic excellence society Phi Kappa Phi in October 2007. Moreover, he received the Certificate in Teaching Accomplishment (CoAT) of NCSU in April 2008. This program

aims to flourish the skills of teaching in academic researchers. He also received the Mentored Teaching Assistant Fellowship from the NCSU College of Engineering in January 2009. Mustafa Berke Yelten was the President of ECE Graduate Student Association (ECEGSA) in 2009-2010 academic year and was twice the chairman of the ECEGSA seminars (Fall 2009 and Spring 2010).

Apart from the engineering career, Mustafa Berke Yelten is a published poet in Turkey and has been awarded several prizes for his poems and short stories. His first poetry book was published in 2006. Currently, he is working on a second poetry book with poems written during his MS and PhD studies.

ACKNOWLEDGEMENTS

First of all, I would like to thank to my advisors, Dr. Michael B. Steer and Dr. Paul D. Franzon. During my PhD, both Dr. Steer and Dr. Franzon always encouraged me to pursue my own ideas, develop the research goals, and achieve them. Dr. Steer's meticulous analysis approach to each piece of my research work gave me inspiration to be more creative and perfectionist. In addition, Dr. Franzon provided crucial support during all stages of my PhD studies, in particular, for the realization of the experimental work. Thus, I am deeply grateful to both of them.

I would like to express my best feelings to Dr. Mo-Yuen Chow and Dr. Joseph Tracy for serving in my committee.

The SHAMROC project, that I participated during my PhD, was part of the DARPA HEALICS program in cooperation with the US-AFRL and Raytheon Company (Contract number: FA8650-09-C-7925). I would like to courteously acknowledge DARPA and US-AFRL for their support to sustain my research. Also important was the guidance of Raytheon Company (Dr. Anthony Kopa and Dr. Charles Wang) during the tapeout, for which I am grateful. Finally, Dr. Steve Lipa of NC State University gave me great technical assistance for the realization of my experimental work. I am thankful to him for his help.

During my PhD, I had a wonderful academic and social environment in NC State University. I am glad to know many great people from the Electronic Circuits and Systems research group who accompanied me during coffee breaks or lunches. I am indebted to all of them for very enjoyable conversations.

Finally, my biggest gratefulness belongs to Ufuk, Muhammet and Azade Yelten, my dear family. By various means of communication, they shared with me the pains of being

far away. Their love and support gave me the biggest motivation to create this work.

The research work here has the distribution statement "A" (Approved for Public Release, Distribution Unlimited). The views expressed are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Variability and Reliability Concerns in Nanoscale Circuits

1.1 Sources of Variation and Reliability Degradation

For the last few decades integrated circuit (IC) technologies have evolved based on the famous Moore's Law which anticipated that chip functionality has to be twice as big every 1.5 to 2 years [1]. In order to achieve this target the number of transistors should be increased which necessitates that the sizes of the transistors to scale down. Scaling has to be applied not only to device dimensions but also to operating voltages to prevent possible breakdown of the devices. It should be noted that with every new generation of scaled device technology, novel fabrication technologies are developed which can be used to manufacture ICs built using the novel technology. Although the scaling process has been successfully realized for technologies of channel length $L_{\text{eff}} \geq 100$ nm, with the introduction of deep submicron ($L_{\text{eff}} < 100$ nm) devices, increasing variations in the device and circuit performance are noticed. Complex circuit designs composed of many

transistors were observed to fail right after fabrication [2]. This observation focused the attention on the variations in device fabrication technologies which can be referred to as process variations.

Process variations can happen at different levels of IC fabrication including the fab-to-fab (fab is an abbreviation for the semiconductor fabrication plants), lot-to-lot, wafer-to-wafer, within-wafer (die-to-die), and within-die (intra-die) variations [3]. In this hierarchy, the nature of process variations can be described as systematic or random. It has been observed that fab-to-fab and lot-to-lot process variations are generally random since different equipment and manufacturing methodologies can be used for different lots in various labs [3]. However, the wafer-to-wafer and within-wafer variations have many properties in common since they are subjected to the same fabrication processes so the variation changes in a systematic way that can be modeled once the source of the variation is correctly detected. For device technologies of $L_{\text{eff}} \geq 100$ nm, layout techniques such as common centroid are developed to deal with these systematic process variations as well as to achieve better transistor matching in analog circuit domain [4]. However, with the advancement to the deep submicron regime, within-die variations that are of random nature [5] become more significant with respect to the total device process variability. Furthermore, no spatial correlation of variations has experimentally been observed in within-die variations although it is commonly seen in systematic variations [6].

The process variations can be classified into two subcategories based on the fabrication steps. The front-end-of-line (FEOL) variability comprises the process variations in the device fabrication steps such as lithography, oxidation and implantation. Conversely, the back-end-of-line (BEOL) variability is related with the variations developed during process stages such as the wiring and building interconnects during which etching and chemical-mechanical polishing (CMP) take place [7]. FEOL variability will be more of

interest in this dissertation work since the aim is to capture the process variations of individual devices in nanoscale circuits both at the die-to-die and within-die levels.

The most dominant factor in FEOL variability is the variation in L_{eff}. L_{eff} scales down with every new device technology and the fluctuation of its values can result in large degradations of important device quantities such as the DC drain current I_{ds} or performance metrics like the speed or the noise figure (NF) of the circuit. Origins of L_{eff} variations have been attributed to several non-idealities including the reticle mask errors, temperature and etch rate non-uniformities, aberrations of lenses, and scanner/stepper illumination fluctuations [8]. Another reason stems from the wavelength of the light used in lithography. In order to maintain the lithographic precision, the wavelength of the light employed should scale with each technology of smaller critical dimension (CD). However, the planned switching from $\lambda = 193$ nm to $\lambda = 157$ nm of light in lithography process to be employed in nanoscale circuits has failed [7]. The resulting low lithographic precision has caused proximity effects (linewidth variation), corner rounding and line-end-shortening [7].

While some of these phenomena can be described as a result of both systematic and random processes, the truly random part of the variation in L_{eff} is caused by the line edge roughness (LER). LER is defined as the local variation of the polysilicon gate edge and its origins can be traced back to random fluctuations of the absorption rate, chemical reactivity and number of photons during the exposure stage in the lithography [9]. Although the spatial correlation of L_{eff} variation gives rise to layout techniques to accommodate variability to some extent LER is getting more dominant for device technologies below 90 nm [10]. This in turn increases the random component of L_{eff} variation which may weaken the benefits of these layout techniques.

Another important source of variation is the variation in the intrinsic threshold voltage

of devices, $V_{\mathrm{th},0}$. This is a direct consequence of the scaling to deep submicron operation regime where the atomistic effects largely prevail. The main reason leading to the variation in $V_{\mathrm{th},0}$ is the random dopant fluctuation (RDF). RDF refers to the variability in the number and location of dopants in the transistor channel during the ion implantation stage of device fabrication which changes the channel doping concentration (N_{ch}) profile. Experimental data has suggested that RDF does not exhibit any spatial correlation both in die-to-die and within-die variations and its impact to the device variability can be observed more as the driving voltages scale down [10]. This also indicates that upcoming technologies may severely be degraded due to RDF where both transistor sizes and the supply voltages are reduced.

The oxide thickness (t_{ox}) of a transistor is subject to variations that are generally random and spatially uncorrelated [11]. Due to scaling of sizes, t_{ox} approached to a few atomic layers where the thermal oxidation process is hard to control thereby inducing variability [7]. In addition, as t_{ox} gets smaller the gate tunneling current increases which contributes to the total leakage power. As a remedy, high-κ (relative permittivity) dielectrics such as HfO_2 can replace the traditional SiO_2 so that the capacitance of the oxide is increased. However, these new dielectric materials cause silicon-interface traps which lead to further reliability problems.

The mobility of charge carriers can present variability due to process steps. In particular, application of appropriate type of mechanical stress can yield higher charge carrier mobility by reducing the effective mass of the charge carriers. A tensile stress is needed for n-channel metal-oxide-semiconductor (MOS) transistors and a compressive stress should be applied to the p-channel MOS transistors for the desired increase in the mobility. Nonetheless, shallow trench isolation (STI) which is applied during the planarization steps of device fabrication imposes a compressive stress on n-channel devices due to the

thermal expansion coefficients mismatch [12]. This decreases the effective mobility of electrons but increases that of holes. It has also been shown that this effect becomes more pronounced with devices of smaller active area which will add up to the variability of nanoscale devices [7].

Finally, the drain-source series resistance (R_{ds}) of transistors should be mentioned as one of the sources of process variations in emerging technologies. R_{ds} is bias dependent and the effects of the process conditions should be considered when the bias is not present. In deep submicron devices, halo/pocket implantation, which is done to prevent punchthrough effect, and the lateral source/drain abruptness can dramatically change the magnitude of R_{ds} [13]. Since both of these factors are becoming more difficult to control in nanoscale transistors, R_{ds} is prone to be affected by excessive variability of process parameters.

Besides the process variations, two different factors pose also a concern of causing significant variability on design metrics of nanoscale circuit designs. These are the environmental variations and IC reliability degradations. Environmental variations are composed of factors that can affect the circuit operation any time during its lifetime and they are capable of severely deteriorating the design performance. Supply voltage-based, thermal and radiation-induced variations should be included within the scope of the environmental variations [11].

Supply voltage-based variations can have a dramatic impact on nanoscale circuits since in scaled device technologies larger currents will be delivered per power pin which can lead to bigger IR drops [1]. Even though process-invariant current source topologies have already been proposed [2], the complex structure of nanoscale IC designs include sophisticated interconnect networks within a chip which makes ICs more susceptible to even a small variation of the power supply potential.

Thermal variations were always a concern since each IC design should be functionally correct for a certain range of temperatures. High temperatures result in competing effects on the transistor parameters such as the reduction in the mobility and threshold voltages [11], which can be even more complicated when there is a temperature gradient in the circuit layout. In actuality, given that the impact of temperature on deep submicron circuits is much more significant than on long-channel devices and the fact that the biasing as well as the activity duty cycle of each transistor is different, every transistor in a circuit can be at a different temperature at any time point. This makes the thermal analysis very complex and the associated thermal variation difficult to predict.

The radiation-induced variation arises due to cosmic rays or alpha particles which induce significant charge accumulations at any part of the circuit when they strike [11]. These charge accumulations may lead to unexpected long delays or logic errors which are defined to be as soft errors. Depending on the context, soft errors may also be treated as a reliability problem.

Reliability degradations of the circuits have become a focus of interest as the deep submicron circuits are widely used in analog and mixed-signal ICs. In general, the reliability problems stem from the increased electric fields across and along the transistor channel due to the scaling in L_{eff} and t_{ox}. Over the lifetime of transistors, high levels of electric fields cause the generation of traps at different locations of the channel and these traps increase the magnitude of V_{th} and decrease the effective μ_0 (μ_{eff}). Since the device degradation is time dependent, it is also called aging. Hot carrier injection (HCI) and negative bias temperature instability (NBTI) are the main aging phenomena in NMOS and in p-channel MOS (PMOS) transistors, respectively, which have the consequences described above. Both aging phenomena have been very well studied in the literature, and compact semi-empirical equations were derived to express the time-dependent degra-

dation of different transistor parameters [14]. Both HCI and NBTI will be covered in depth in this dissertation.

There are few more aging mechanisms besides HCI and NBTI. As t_{ox} is scaled down further, the electric field across the gate oxide increases and results in trap creation within the oxide [15] over the lifetime of the transistor. This is the start of the time dependent dielectric soft-breakdown. The gate tunneling current increases that eventually degrades the gate oxide integrity. Many charge carriers will be able to tunnel through the gate oxide and create a conducting path [16]. These conducting paths turn into short circuits in the gate oxide over time that destroy the device oxide completely thereby resulting in the failure of the transistor. This is called the dielectric hard-breakdown which can be characterized as an abrupt increase in the gate leakage current. Time to dielectric soft- and hard-breakdown is estimated using a Weibull distribution [11]. At the circuit level the degradation due to dielectric soft-breakdown can be modeled by two resistances, one between the gate and source (R_{gs}) and the other one between the gate and drain (R_{gd}). Nonetheless, latest studies suggest that the device and circuit functionality will not be substantially impacted since the gate leakage current does not increase much during the dielectric soft-breakdown phase as well as V_{th} and the small signal transconductance g_m remain unaffected [15].

The other well-known reliability problem is electromigration. Electromigration is not a device aging phenomenon and it affects the interconnects. The basic reason for electromigration is the increased current densities due to scaling. Mostly atomic migration is observed at the regions of interconnects where the current density is maximized. This creates the conditions of an open circuit (very high resistance) at the locations where the atoms have migrated from and short circuit conditions (very low resistance) where the atoms are moving to [17]. The problem is generally alleviated by physical verification

tools in layout design by imposing upper bounds for the maximum current density [11].

1.2 Previous Research on IC Variability Analysis

Traditionally process corner analysis is used by a designer to characterize circuit variability. The process corners aim to show the general trends in the design quantities caused by the die-to-die process variations [11]. In the process corner analysis, sets of worst-case device models are used in full transient circuit simulation but this does not yield a sufficient representation of the actual distribution of important design quantities such as ($V_{\rm th}$) and transistor saturation drain current ($I_{d,\rm sat}$) [18]. This is especially true when the relative variability is large. An alternative approach to circuit variability analysis is the use of Monte Carlo analysis in which multiple circuit simulations are performed with each simulation using a different set of device model parameters representative of the process distributions [19]. In Monte Carlo analysis, the number of runs is particularly high with the significant process variations in a deep submicron process. Besides, Monte Carlo methods are computationally intensive and prohibitively so if large within-die variations are to be considered. This makes it difficult for designers to capture the effects of within-die variations at the circuit level.

In analog and mixed-signal circuit domains, variability modeling research has been focused on replacing the Monte Carlo analysis with nontraditional and computationally less intensive methods. Inspired by the increasing level of variability in nanoscale circuits different approaches involving mathematical and statistical modeling as well as device and circuit modeling techniques are suggested. In [20] a technology computer-aided design (TCAD) technique was proposed that takes 17 device fabrication parameters constituted by the device $V_{\rm th}$, lightly-doped drain (LDD) and source-drain implantation (S/D) doses,

energies and annealing temperatures. A sensitivity analysis is applied to screen out less important parameters. Using the remaining parameters, device models of 0.18 μm (BSIM3v3) are generated through TCAD device simulator. These device models are used in a typical circuit simulator (e.g. SPICE) to acquire responses for the oscillation frequency and the total power consumption of a 33-stage ring oscillator. The process parameters are optimized to minimize the variation in these both output quantities. Screening out parameters, however, significantly complicates capturing the interactions of process variations in deep submicron circuits [21]. Instead, input factors (similar to the group used in [20]) can be divided into smaller groups where a polynomial-based response surface model is built for each group using TCAD device simulations of 65 nm technology in order to derive compact models for $I_{d,\mathrm{sat}}$, V_{th} and the leakage current $I_{d,\mathrm{leak}}$. The constant terms of each response surface model is found using the parameters of other groups so that an interdependence of the models is provided. Nevertheless, no specific application of the constructed models is suggested [21]. In addition, these models are of little use to circuit designers since they do not have any control on device fabrication parameters once the design kits are fixed. Another approach similar to [21] has been implemented for the V_{th} of n-channel MOS (NMOS) transistors of 95 nm technology to optimize the process factors such that the variation in the modeled quantity is minimized [22].

Parametric yield estimation should also be mentioned as part of the variability modeling for devices and circuits. In [23] response surface models are constructed to calculate the performance metrics for a given circuit in the presence of process variations. Then, the statistical operator MAX is employed to estimate the yield of the circuit for the given process factor distribution. This operator is defined as a linear combination of two random variables x and y, $\mathrm{MAX}(x, y) = \alpha x + \beta y + \gamma$, where $\alpha = P(x \geq y)$ and

$\beta = P(y \geq x)$ are the tightness probabilities and γ is a constant [23]. Even though the method is more computationally efficient than the Monte Carlo method no practical insight about the design of the analyzed circuit can be extracted out of it to counter the negative impact of process variations.

A different approach involves modeling the current variation in a circuit as a noise source [24]. A regression model for the variation of the on-current I_{on} based on 65 nm bulk process in terms of the average I_{ds}, L_{eff}, effective width W_{eff}, the drain-source and gate-source potentials V_{gs} and V_{ds} are derived. This model is used to find the variability in the node voltages of a differential amplifier [24]. In [25] the radio frequency design quantities NF, input referred third order intercept point (IIP3), and the transconductance gain (G_m) of a low noise transconductance amplifier are modeled by polynomial-response surface modeling techniques in terms of a few device process parameters such as the CD or V_{th} variation to extract design insights. In [26] response surface modeling based on a quadratic model with Cornish-Fisher expansion is used to find an upper and lower bound for the gain of an operational transconductance amplifier and also the leakage current of an XOR gate instead of using Monte Carlo simulation methods. Finally, the transient response variability of common digital blocks such as the two-input NAND gate, simple latches are captured by means of models of device current-voltage (I-V) and capacitance-voltage (C-V) curves at 65 nm technology node [27]. Here critical points in the I-V and C-V curves, which indicate device operation region changes, are located and based on these points the I-V and C-V curves are then interpolated using fitted analytical functions.

Since most of the process variability arises from quantum mechanical phenomena, atomistic simulators can be used to analyze the origins of the process variations and aging. In [28] LER and RDF effects on metal gate high-κ dielectric devices are simulated

in the presence of aging (NBTI) using atomistic three dimensional drift-diffusion simulator. Results show that the degradation caused by aging increases the level of LER and RDF effects on the analyzed devices. This information underlines the fact that process variations and aging effects cannot be treated independent of each other.

1.3 Design of Process Variation Tolerant Circuits

The modeling and simulation for process variations and aging requires novel computer-aided design techniques to be developed. In addition to this, novel circuit design methods and topologies are needed with which process variation and aging tolerant circuits can be built. This approach is a continuation of design for fabrication (DFM) which aims to produce circuit designs that can be manufactured repeatedly by taking into account process, voltage, temperature, and time-based variations and degradations [29].

One easy way of building process variation tolerant circuits is to use transistors with thick oxide layer and capable of withstanding higher supply voltages [30]. Another suggested method is employing sense circuits such as a current mirror within a low noise amplifier to monitor the changes in the bias current and then appropriately adjust the biasing voltage in order to increase the yield [31]. However with nanoscale circuits, no sufficient headroom is left for a third transistor in a cascode structure. Even so, this yield increase benefit comes with a large overhead penalty. A different strategy is iteratively sizing a constant-g_m biasing circuit elements (transistors and a resistor) of a low noise amplifier (LNA) and a mixer so that the LNA and mixer can fulfill the specification requirements [32]. Novel biasing circuit topologies have also been proposed based on the implementation of a circuit that can significantly reduce the variations on the main biasing current [2]. Here an addition-based current source is introduced where the sum

of two currents that are negatively correlated with each other is held constant. A similar circuit topology can be employed to reduce the negative impact of HCI on an NMOS device as shown in [33].

Digital calibration of appropriately selected knobs to rectify the performance degradation of a circuit is another method for the design of process variation tolerant circuits. This approach has been validated in the case of an LNA with three external voltages being used as individual knobs [34]. The main concept here relies on the reduced cost and complexity compared to individual performance calibration of each device in a given circuit. Similarly, the calibration of a charge pump in a phase locked loop (PLL) has been realized in [35] such that the mismatch between the pull-up and pull-down currents is minimized. This will eventually reduce the jitter and the phase noise in the PLL when process variations are present. The benefits of digital calibration can also be extended to heal the aging-based degradation of circuits as shown in the case of a digitally-assisted current steering digital-to-analog converter [36].

1.4 Original Contributions

The review of the origins and modeling of process variations and aging as well as the review of the state-of-the art research in variation tolerant nanoscale circuit design indicate a major challenge to both IC designers and the computer-aided design specialists. Many design and performance quantities are subject to large deviations from their expected values due to variability effects. Also, the reliability degradations can result in significant degradation in these quantities with the time progress.

Clearly, the concept of DFM should evolve further to design for variability (DFV) and design for reliability (DFR) as robust design methods must be configured in such

a way that they will be able to specifically counteract process variations and aging. As noted recently by several authors there is a huge need and demand for an approach that is capable of unifying design for variability and design for reliability methods with the regular circuit design techniques [37]. The benefits of such an approach are two-fold: First, the interaction of process variations and reliability degradations are experimentally proven [14]. Thus a joined analysis of both effects in computer-aided design domain is necessary. Secondly, many process variation tolerant circuits discussed in this chapter have ideas that can be applied to alleviate aging-based performance drops as well. An integrated variability and reliability analysis approach will broaden the perspective of circuit designers to simultaneously contend with variations due to process parameters or aging.

The main contribution of this dissertation is the development of reduced order modeling techniques that are capable of analyzing the process variations and aging degradations both at die-to-die and within-die scales. The key differentiation between these two scales is made using the fact that variability and reliability analysis in deep submicron circuits requires each device in a circuit to be modeled with different sets of parameter vectors. Traditionally just a few vectors (corresponding to the process corners) are used for the variability analysis with the same vector being applied to every device in the circuit. This approach will not be sufficient to capture with accuracy all the interactions of process and aging parameters. Conversely, making an excessive number of individual Monte Carlo simulations result in a huge computational cost. Reduced-order models can be built to represent an output quantity with relatively high accuracy for a specified range of multiple inputs. In this work I_{ds} of 65 nm transistors are modeled in terms of six process parameters —that capture a significant proportion of the whole process space—, three terminal voltages, the operation temperature and the device age (see Chapter 2). A

circuit analysis tool is constructed that can use the developed I_{ds} reduced-order models to evaluate the DC operating conditions of a given circuit. Since all transistors in the circuit are expressed as a vector of eleven elements, the designer has the capability of evaluating all different combinations and interactions of process parameters, temperature, aging and voltage variations. Three types of reduced-order models, Kriging-based surrogate models, neural network-based models and support vector machine-based models are developed and analyzed for this purpose. Thus the first significant contribution of the dissertation is to show how various reduced-order models can be employed to build a variability-reliability analysis tool that can yield the circuit designers great flexibility to observe systematic and random process variations and aging trends in a nanoscale circuit.

The second contribution of the dissertation is regarding the in-depth comparison of different reduced-order modeling strategies and their development techniques. A thorough analysis has been conducted in Chapter 3 that compares model accuracy, size as well as generation and evaluation times. In addition, all these models employed sample vectors produced using a novel sampling method based on orthogonal Latin hypercube sampling (see Subsection 2.2.3). This method provides a solution to building sampling schemes with uniformly distributed samples in a multidimensional design space. The complete methodology for reduced-order modeling as well as the algorithmic techniques employed enables an integrated variability and reliability analysis of circuits during electrical simulation.

The third contribution of the dissertation is demonstrating the extent of variability and reliability degradations on circuit building blocks. Within-die variations are shown to have a significant impact on the distribution of I_{ds} in a circuit (see Section 4.3). The significant process parameters responsible for the mismatch of two transistors are iden-

tified as the intrinsic threshold voltage and effective length variation (see Section 4.5). Finally, in Subsection 5.3.2, biasing of analog circuits are shown to be decisive in determining the aging characteristics which can be used as a design guideline for building aging resilient circuits.

The final contribution is made by developing an analog circuit-based NBTI monitor that is explained in Chapter 6. Previous approaches of building NBTI monitors is based up on digital circuits and they were sensing changes in quantities that are predominantly important for digital applications such as the shift in the oscillation frequency of a ring oscillator. Here, the design of the monitor is done using a common gate amplifier whose biasing is modulated. The biasing is dependent on a PMOS transistor thus the modulation results in a gain drop linearly increasing with the degradation of this device. Such a design can easily be integrated into an analog circuit to monitor the time-based degradation of critical design quantities.

1.5 Preview of the Dissertation

The dissertation is organized as follows: The methodology of developing a reduced-order model is discussed in Chapter 2. In particular, the focus will be on building a surrogate model. Chapter 3 presents the comparison of the three reduced-order modeling strategies in terms of different modeling metrics. The variability and reliability analysis of circuits with examples using reduced-order models are discussed in Chapters 4 and 5, respectively. In Chapter 6, the NBTI monitor is presented with its principles of design and performance details. Also, characterization of reliability effects on major analog and digital building blocks is accomplished in this chapter. Finally, the dissertation will be concluded with the possible future work alternatives in Chapter 7.

1.6 Publications

The following papers are based on the work presented in the dissertation.

Journal Papers:

- M. B. Yelten, P. D. Franzon, M. B. Steer, "Surrogate Model Based Analysis of Analog Circuits—Part I: Variability Analysis", *IEEE Transactions on Device and Materials Reliability*, in press, DOI: 10.1109/TDMR.2011.2160062.

- M. B. Yelten, P. D. Franzon, M. B. Steer, "Surrogate Model Based Analysis of Analog Circuits—Part II: Reliability Analysis", *IEEE Transactions on Device and Materials Reliability*, in press, DOI: 10.1109/TDMR.2011.2160063.

- M. B. Yelten, P. D. Franzon, M. B. Steer, "Comparison of Modeling Techniques in Circuit Variability Analysis", submitted to *International Journal of Numerical Modelling: Electronic Networks, Devices, and Fields*.

- M. B. Yelten, P. D. Franzon, M. B. Steer, "Analog Circuit Based NBTI Monitor", submitted to *IEEE Transactions on Device and Materials Reliability*.

Conference Paper:

- T. Zhu, M. B. Yelten, M. B. Steer, P. D. Franzon, "Application of surrogate modeling in variation-aware macromodel and circuit design", *2011 Simulation-Driven Engineering Design Optimization and Modeling Conference, (SDDOM'11)*, Noordwijkerhout, the Netherlands.

Chapter 2

Reduced-Order Model-Based Device Modeling

2.1 Introduction

Scaling of the dimensions of integrated circuits to deep submicrons (65 nm, 45 nm, and below) achieves higher speeds and compactness but results in increased design complexity. In particular, the variability and reliability of circuits poses a significant concern in design due to significantly increased variations of V_{th}, μ_{eff}, t_{ox}, L_{eff}, and W_{eff}. These variations are exaggerated by aging and ambient temperature changes. With smaller device sizes, physical phenomena that were negligible with longer channel transistors become significant. Increased drain induced barrier lowering (DIBL), short channel effects, and gate tunneling are a few examples. Lithography-derived variations do not scale down with the sizes of transistors and so tolerances have become a significant percentage of nominal device parameters [7]. These factors lead to experimentally observed higher variability of devices both for die to die and, of significance, within die levels [5]. Mod-

eling within-die variations requires separate descriptions of the transistors in a circuit since the variability induced differs from device to device for each process parameter. Furthermore, aging of transistor-based circuits is dependent on the entire operational history, including temperature and biasing history, and reduces circuit reliability. The high power density of sub 100 nm transistor circuits leads to operating temperatures up to 100°C where circuit operation deviates significantly from nominal behavior.

The impact of increased device variability is observed in the complexity of the response surface of the drain-source current, I_{ds}, and transconductance, g_m, etc, of nanoscale devices. Increased variability of critical design quantities requires more advanced modeling approaches [18]. Adding semi-empirical, empirical or statistics-based parameters to analytical device model expressions, as is sometimes done, reduces intuitive understanding by circuit designers which increases the difficulty of designing variability-aware circuits.

In classical circuit design these problems are addressed through process corner simulations of a design developed using the nominal process. Alternatively, in Monte Carlo analysis circuit-level models are repeatedly evaluated with the parameters of the models changed to cover the process space. However with advanced processes the traditional approach of simulating circuits at the process corners alone, or with the same values of parameters for all transistors, is proving to be inadequate. It is necessary to consider the variability of individual transistors due to significant within-die process variations and transistors aging at different rates. The solution to this issue requires an approach capable of analyzing variability and aging at the individual transistor level. Reduced-order models which are only a function of those quantities that vary significantly such as certain process parameters, terminal voltages, temperature, and device age can be employed for that purpose thereby making it an alternative to the full modeling at the process corner and Monte Carlo-based circuit analysis. Reduced-order models establish

the connection between the input space and the output space of a strongly nonlinear function using a group of sample vectors and their outputs. This connection can be a number of basis functions with several coefficients to be optimized or it can be neural network with a variable number of neurons and associated weights to be found out for the best representation of the system. A reduced-order model can also be a machine-learning based model whose parameters to be trained with every sample vector. In this chapter the construction of the first type of reduced-order models, also called surrogate modeling, will be described in detail.

Surrogate modeling is a response surface modeling method that takes as input a group of sample vectors and their corresponding outputs (also called true model outputs), and develops a nonlinear reduced-order model by adjusting the coefficients of basis functions [38]. For circuits, this technique has been applied to modeling the input referred noise and admittance matrix elements of a low-noise amplifier [39] and in modeling using space mapping [40]. The space mapping technique was initially employed within the context of the design and optimization of microwave filters [41]. The approach utilizes a coarse model for the desired output quantity and develops appropriate (and usually low order) input and output mapping functions to yield an accurate input-output relation. Various mapping models and functions have been utilized to simplify the design process of microstrip transmission lines, microwave filters and waveguides [42].

Surrogate modeling has also been applied in different contexts in electrical engineering. In [43] surrogate models for the difference between the simulated and measured textile antenna input return ratios are established for the given ranges of permittivity and loss tangent of the textile material. In [44] the look-up tables of input/output buffer information specification (IBIS) characterizing the input/output (I/O) circuits are modeled. In particular, surrogate modeling of the I-V and the rising/falling transition

voltage-time relations are performed with the output voltage (V_{out}), supply voltage, V_{th} and temperature are taken as inputs. In [45] surrogate models for the design of integrated transformers operating at high frequencies are developed given the inner diameters and widths of the primary and secondary inductors as input. Finally a surrogate model for a design of substrate integrated waveguide interconnect with transitions to conductor backed coplanar waveguides using polynomial basis functions is presented in [46].

As can be inferred from the examples in the literature surrogate modeling aims to reduce the complexity of a system governed by many parameters. This is achieved by describing the system in terms of a handful of parameters whose variations have a physical meaning for the designer. This concept is applied here to the BSIM model which has more than 100 parameters [47], however only a small number of these change significantly due to process variations and aging. Thus the response of a device can be captured by a surrogate model that has only a few parameters in addition to the voltages at the device terminals, temperature, and the device's age. All these parameters establish the state of a transistor and will be called variables. The surrogate model is developed by fitting the response of, in this case, the BSIM model with nominal parameters (as in the designed circuit) and the handful of variables subject to process variability and aging. The resulting surrogate model is much simpler than the full transistor model and computationally efficient.

In this chapter the effect of variability on DC drain current is to be modeled by a surrogate model which will captures the DC drain current response of 65 nm transistors over the entire process space including all possible temperatures, terminal voltages, and aging effects. The process space is defined by the major sources of variation. The models generated will be employed to merge variability and reliability analysis with electrical circuit simulation in the next chapters of the dissertation. This enables simultaneous

design for performance, variability, and reliability.

The chapter presents the description of each stage in surrogate modeling in Section 2.2 and the results of the constructed surrogate models in Section 2.3.

2.2 Framework for the Surrogate Model-Based Device Modeling

A surrogate model is a representation of a system that captures the essential characteristics of mathematically complex functions of often many parameters [48]. Evaluating the outcome for each unique combination of the input variables will be exact, yet require lengthy simulation. The surrogate transistor model used here has only a few input variables of parameterized basis functions which are evaluated to yield the model output. Those input variables are a (relatively small) subset of the inputs to the full transistor model. The model fitting problem is then to determine the coefficients of the basis functions of the surrogate model for each variable in the input vector so that the response surfaces of the surrogate model and of the full transistor model match. A set of sample input vectors (sample vectors) is required which should be representative of a large proportion of the entire process variation space as well as nodal voltage space.

The surrogate modeling process described in [38] has been reformulated here yielding the model development flowchart shown in Figure 2.1. In the first stage parameter sets are selected using Latin Hypercube Sampling to create a set of sample vectors. Then the responses to these are evaluated using the transistor model (or true model). The third stage is to choose the basis functions of the surrogate model. Common choices include polynomials, radial basis functions (RBFs), Kriging basis functions, and support vectors

[38]. The coefficients of these are optimized using local or global search algorithms.

Surrogate model accuracy is determined using the root mean square error (RMSE) or root relative square error (RRSE) metrics. If these metrics reveal that the model accuracy is not sufficient, more sample vectors as well as more highly-parameterized basis functions should be used. A trade-off exists between the model complexity and model accuracy. Typically more sample vectors are selected from the regions of the surrogate model response surface that have the largest deviation from the response of the true model.

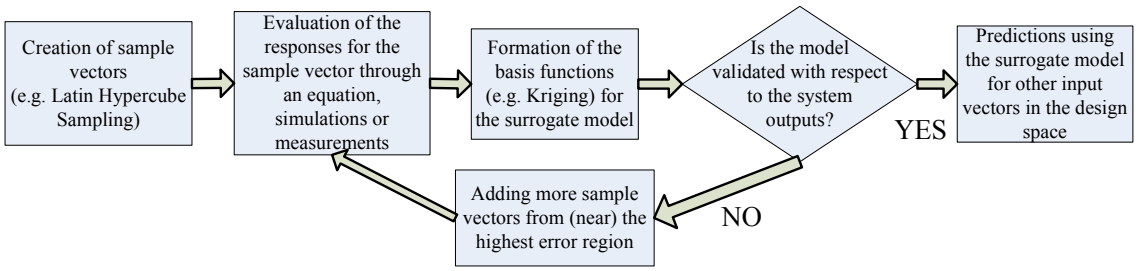


Figure 2.1: Basic flowchart describing the surrogate modeling process. The loop can be repeated for a prespecified times or until a desired accuracy level is reached.

2.2.1 Drain Current Modeling Problem

The quantity that has the biggest impact on the variability of analog, digital and mixed-signal CMOS circuits is the drain current. It is also a good indicator of the variability of other circuit quantities. The focus of the modeling in this paper is the DC current of 65 nm devices (in the IBM 10SF process). In the simplest form the model would be a multidimensional look-up table populated using the results of transistor simulations and nominal device parameters as well as unique sets of input variables with each set defining

what is called a sample vector. In this study each sample vector has eleven elements. The output corresponding to each sample vector can be found using the results of successive circuit simulations or an analytical expression. The latter approach has been adopted here, in particular, I_{ds} in the BSIM device model is used [47]:

$$I_{ds} = \frac{I_{ds0} N_{\mathrm{F}}}{1 + R_{ds} I_{ds0} / V_{ds,\mathrm{eff}}} \times \left[1 + \frac{\ln\left(V_{\mathrm{A}} / V_{\mathrm{A,sat}}\right)}{C_{\mathrm{clm}}}\right] \times \left(1 + \frac{V_{ds} - V_{ds,\mathrm{eff}}}{V_{\mathrm{A,dibl}}}\right) \times \left(1 + \frac{V_{ds} - V_{ds,\mathrm{eff}}}{V_{\mathrm{A,dits}}}\right) \times \left(1 + \frac{V_{ds} - V_{ds,\mathrm{eff}}}{V_{\mathrm{A,scbe}}}\right), \quad (2.1)$$

with terms described in Table 2.1. In the BSIM model each of these terms is the function of other parameters some of which are physical parameters, while the others are semi-empirical fitting parameters. The functional dependencies of all these parameters are given in the BSIM manual [47].

In (2.1) there are four main components which can alter the operation of a nanoscale transistor in saturation mode where most devices are biased in all analog and mixed-signal circuits. The first component is the channel length modulation (CLM). The physical reasoning of CLM stems from the increased V_{ds} beyond the saturation drain-source voltage $V_{d,\mathrm{sat}}$ (or equivalently $V_{ds,\mathrm{eff}}$). The excess voltage beyond $V_{ds,\mathrm{eff}}$ should drop across a certain section of the transistor channel [49]. Thus the L_{eff} becomes shorter than what it was before which makes the transistor even more susceptible to any variation in L_{eff}. This effect also causes I_{ds} to increase in the saturation region.

The second component is DIBL. DIBL effect is also an outcome of device scaling. Due to short channel length, the voltage applied at the drain side can lower the barrier between the drain end and the source end of the transistor thereby making it a function of V_{ds}. That forces the transistor to operate in a stronger inversion mode and reduces

Table 2.1: Explanation of the terms in (2.1).

Variable	Explanation	Variable	Explanation
C_{clm}	Channel length modulation parameter	I_{ds0}	Saturation Drain Current
N_{F}	Number of fingers	R_{ds}	Bias-dependent source-drain resistance
V_{ds}	Drain-source voltage	$V_{ds,\mathrm{eff}}$	Effective drain-source voltage
V_{A}	$V_{\mathrm{A,clm}} + V_{\mathrm{A,sat}}$	$V_{\mathrm{A,sat}}$	Early Voltage at $V_{ds} = V_{ds,\mathrm{eff}}$
$V_{\mathrm{A,clm}}$	Early Voltage contribution due to the channel length modulation	$V_{\mathrm{A,dibl}}$	Early Voltage contribution due to the drain-induced barrier lowering
$V_{\mathrm{A,dits}}$	Early Voltage contribution due to the drain-induced threshold shift	$V_{\mathrm{A,scbe}}$	Early Voltage contribution due to the substrate current-induced body effect

V_{th} [49]. Since V_{th} drops down, I_{ds} is increased. [50].

The third component is due to the drain induced threshold shift (DITS). DITS is a direct consequence of the halo (pocket) doping. Even for long channel devices V_{th} can be observed to decrease with the application of increasing V_{ds}, thus its consequences present great similarity with that of DIBL [51]. An increase in I_{ds} and a decrease in the small signal drain-source resistance r_{ds} (also called the output resistance r_o) can be expected due to the presence of DITS.

The substrate current-induced barrier lowering (SCBE) is a result of the increased substrate current in nanoscale devices. Especially, due to large electric fields along the

transistor channel, the charge carriers acquire a very large velocity thereby causing impact ionization [50]. Some of the charge carriers generated by impact ionization are swept to the substrate yielding an increased substrate current. This substrate current flows through the substrate resistance which increases the bulk voltage thereby changing the threshold voltage of the device.

$$V_{\text{th}} = V_{\text{th},0} + \gamma\Big(\sqrt{2\varphi_s - V_{bs}} - \sqrt{2\varphi_s}\Big). \tag{2.2}$$

In (2.2), γ is the body effect coefficient, and φ_s is the surface potential. Given the source voltage remains constant, as V_{bs} increases, V_{th} decreases. Thus I_{ds} rises with an associated drop in r_o.

The scales of all these effects change with slight variations in the process parameters thereby significantly weakening the performance robustness of nanoscale circuits.

2.2.2 Determination of the Input Parameters

In this work, a transistor-level model implemented in a circuit simulator is called the true model and the nonlinear reduced-order models try to estimate this model as accurately and efficiently as possible. The models are established using design space samples which include terminal voltages, temperature, device age and a few process parameters that are significant contributors to variability.

Sample vector selection is based on two perspectives: the total number and the distribution of sample vectors. Clearly as the number of sample vectors grows the accuracy of the surrogate models improves, but model complexity and development time increases. Also sample vectors should be distributed as uniformly as possible since initially no information on the true response is available.

Ideally the correlation of the elements of the sample vector is negligible. For the transistor drain current, I_{ds}, a careful investigation of the BSIM model was conducted indicating that most of the parameters of the semi-empirical BSIM model are highly correlated. One method to alleviate the correlations between the parameters is to apply principal component analysis (PCA) where the axes of each parameter are rotated so that the modified parameters become orthogonal to each other [7]. However the uncorrelated parameters resulting from PCA have little physical significance [52]. It has also been shown that the mathematical operations required to obtain uncorrelated parameters result in underestimation of the large process variations of nanoscale devices [53]. From the perspective of design, the PCA method obscures the dependency of critical design quantities on process variations. Alternatively, it has been suggested that performance-aware and application-specific model cards [52] be created where the variations in selected electrical test quantities, such as on-current (I_{on}), off-current (I_{off}), or V_{th}, are decomposed into associated process parameters and then the individual contributions modeled separately. This yields an abundance of process files for different design purposes and their interpretation requires considerable effort by the designer.

The approach adopted in the work presented here integrates variability analysis of crucial process parameters with circuit simulation. This is achieved by first determining the BSIM device parameters that span the process corner space and then assigning ranges that correspond to the expected variations. These parameters are strictly of physical origin, i.e. they are not fitting parameters and they define the process space, but they are not fully uncorrelated. Nonetheless, the aim here is not to acquire the complete statistical profile for a given process (e.g. device parameter extraction), rather it is to quantify the extent of variations arising from different fabrication sources which are weakly correlated.

In this work the parameters chosen as elements of the sample vector are the intrinsic mobility μ_0 (`U0`), intrinsic threshold voltage $V_{\text{th},0}$ (`VTH0`) at zero bulk-source potential ($V_{bs,0}$), the electrical gate equivalent oxide thickness t_{ox} (`toxe`), the variation due to mask/etch processes of effective gate length ΔL_{eff} (`XL`), and the zero-bias lightly-doped drain-source resistance $R_{ds,0}$ (`RDSW`). Among these, the correlation of μ_0 and $V_{\text{th},0}$ has been recently investigated for 90 nm CMOS technology indicating low correlation [54]. Even though the channel doping parameter N_{ch} (`NDEP`) is not typically included within the process space, small variations can affect the level of the random dopant fluctuation (RDF) effect, thus it is included as an element of the sample vector. In addition to process parameters, voltages characterizing the operation of the transistor (the gate-source voltage V_{gs}, the drain-source voltage V_{ds} and the bulk-source voltage V_{bs}), operating temperature T, and age, t_{age}, are part of the sample vector. A total of 11 variables comprise the sample vector of the surrogate model of the drain current. For the process parameters t_{ox}, L_{eff}, $V_{\text{th},0}$, μ_0 and $R_{ds,0}$, the ranges are determined from their values at the slow and fast process corners as a percentage of their nominal values and expressed as Δt_{ox}, ΔL_{eff}, $\Delta V_{\text{th},0}$, $\Delta \mu_0$ and $\Delta R_{ds,0}$, respectively. ΔN_{ch} accounts for random variations from its nominal value $N_{\text{ch},0}$. The voltage levels V_{gs}, V_{ds}, and V_{bs} cover all possible biasing conditions with $V_{dd} = 1.2$ V, the supply voltage of the design. The temperature T accounts for expected operating conditions. Finally, the range of the transistor age, t_{age}, is set based on the expected product life time. All parameters other than these are set to their nominal values. The ranges of the variables are provided in Table 2.2. The sample vector of the j^{th} NMOS device is

$$\mathbf{NM}^{(i)} = [T, t_{\text{ox}}, N_{\text{ch}}, \Delta L_{\text{eff}}, V_{bs}, V_{\text{th},0}, \mu_0, V_{gs}, V_{ds}, R_{ds,0}, t_{\text{age}}]\,, \tag{2.3}$$

and that of the j^{th} PMOS device is

$$\mathbf{PM}^{(j)} = [T, t_{\text{ox}}, N_{\text{ch}}, \Delta L_{\text{eff}}, V_{\text{sb}}, |V_{\text{th,0}}|, \mu_0, V_{sg}, V_{sd}, R_{sd,0}, t_{\text{age}}]\,. \tag{2.4}$$

Table 2.2: Surrogate model parameters and their ranges

Parameter	Range	Parameter	Range
$\Delta V_{\text{th,0}}$	$\pm 20\%\, V_{\text{th,0,nom}}$	V_{gs}	0 – 1.2 V
ΔL_{eff}	0 – $40\%\, L_{\text{min}}$	V_{ds}	0 – 1.2 V
$\Delta \mu_0$	$\pm 7\%\, \mu_{0,\text{nom}}$	V_{bs}	-0.4 V – 1.2 V
Δt_{ox}	$\pm 10\%\, t_{\text{ox,nom}}$	T	-100°C – 100°C
ΔN_{ch}	$\pm 5\%\, N_{\text{ch,nom}}$	t_{age}	0 – 10 years
$\Delta R_{ds,0}$	$\pm 10\%\, R_{\text{ds,0,nom}}$		

A concluding remark to this section can be made on the choice of input parameter ranges. Some authors claim that the variation range estimates based on design-time analysis (not including the measured process parameter distributions and related test data of actually fabricated wafers) are overly pessimistic. A wafer-specific variability characterization in conjunction with an appropriate machine learning scheme is proposed which shows a smaller variance of process distribution [55]. However, most circuit designers do not have a direct access to the foundries and individual wafer test results. Furthermore, training the active learning model in a certain foundry and at a given technology node can result in a large error if applied to the data extracted from a different foundry with a different version of the same process technology.

2.2.3 Sampling Scheme Development

In this section an orthogonal sampling scheme termed orthogonal Latin hypercube sampling (OLHS) that efficiently chooses samples in a highly dimensional design space is presented.

Appropriate sampling across the process and design space, concentrating on areas of high sensitivity, is crucial to ensuring model accuracy. To start with, the range of each parameter in Table 2.2 is divided into n equal intervals in each dimension and thus n determines the resolution of the sampling design. In conventional random sampling, samples would be chosen arbitrarily from k regions in k-dimensional space without regards to the uniformity of the sample distribution. Thus for a 2D problem there are n^2 possible samples. A more efficient 2D sampling design uses Latin hypercube sampling (LHS) [56]. With this method and for a 2D space, only one sample per row and column is selected. Thus only n samples are chosen with the samples represented by a list of n (x_i,y_i) pairs.

Although LHS is better than random sampling of the entire space in terms of the sample distribution, large regions of the sample space may be empty. A more uniform distribution of samples across the design space can be achieved by maximizing the Euclidean distance between individual samples and simultaneously minimizing the number of same distance sample-pairs. This essentially creates a maximin sampling plan [57], [58]. However, the maximin sampling design principle may not be sufficient to ensure a uniform sampling of the design space, called the space filling property. Here good space filling is achieved by introducing orthogonality to the sampling design. The concept of orthogonality aims to partition the design space into several subspaces such that there is a certain number of samples contained in each subspace [59]. An example of this or-

thogonal LHS (OLHS) scheme is seen in Figure 2.2. In this figure, a 2D (8×8) design space (i.e. $n = 8$ in each dimension) is divided into 4 subspaces of size (4×4). It is important to observe that each subspace has two samples and the distance between each sample in the design space is maximized. Thus the sampling plan in Figure 2.2 presents an orthogonal maximin sampling plan that can be used to represent the design space.

In 2D the number of divisions in the x and y directions are denoted x_{div} and y_{div}, respectively ($x_{\text{div}}, y_{\text{div}}$ are nonnegative integers, i.e. $x_{\text{div}}, y_{\text{div}} \in \mathbb{Z}^+$). Thus there are $x_{\text{div}} y_{\text{div}}$ subspaces where n is an integer multiple of both x_{div} and y_{div}. The orthogonality constraint for the entire design space is defined as the number of samples, ξ, to be contained in each of the $x_{\text{div}} y_{\text{div}}$ subspaces:

$$\xi = \frac{n}{x_{\text{div}} y_{\text{div}}} + \delta \qquad |\delta| \leq \delta_{\max}. \tag{2.5}$$

Here $\delta \in \mathbb{Z}^+$ is an (integer) flexibility factor limited by $\delta_{\max}$. In the final step, among the initially constructed LHS designs, one design which satisfies both the maximin and orthogonality constraints is chosen. In this work, $n = 64$ is chosen along with $x_{\text{div}} = y_{\text{div}} = 4$, and $\delta_{\max} = 1$.

The above text described a two-dimensional OLHS scheme. Directly extending this to a k-dimensional OLHS would result in a very large number of samples and thus would be computationally expensive. Instead $(k-1)$ 2D OLHS schemes are employed. In k-dimensional space the dimensions are $y^{(1)}$, $y^{(2)}$, ..., $y^{(k)}$. One dimension, the i^{th} dimension, is randomly chosen and this becomes the first dimension in a 2D OLHS scheme, i.e. $x^{(i)}$. The samples of $x^{(i)}$ are $x_1^{(i)}$, $x_2^{(i)}$, ..., $x_n^{(i)}$. Now $x^{(i)}$ is paired with one of the other dimensions $y^{(j)}$, $j \neq i$, and a 2D OLHS scheme is created by applying the sampling rules described above. Repeating the same procedure for the remaining $y^{(i)}$

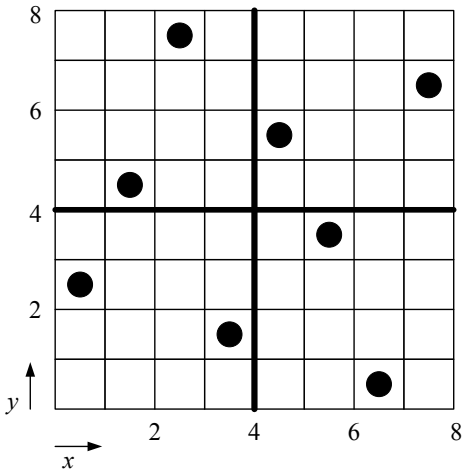


Figure 2.2: A 2D orthogonal Latin hypercube sampling example with $n = 8$ and $x_{\text{div}} = y_{\text{div}} = 2$.

dimensions yields $(k-1)$ 2D OLHS schemes:

	2D OLHS Schemes $(j \neq i)$					
	$x^{(i)}y^{(1)}$	$x^{(i)}y^{(2)}$	$\ldots$	$x^{(i)}y^{(j)}$	$\ldots$	$x^{(i)}y^{(k)}$
Samples	$x_1^{(i)}\ y_2^{(1)}$	$x_1^{(i)}\ y_4^{(2)}$	$\ldots$	$x_1^{(i)}\ y_3^{(j)}$	$\ldots$	$x_1^{(i)}\ y_2^{(k)}$
	$x_2^{(i)}\ y_n^{(1)}$	$x_2^{(i)}\ y_n^{(2)}$	$\ldots$	$x_2^{(i)}\ y_n^{(j)}$	$\ldots$	$x_2^{(i)}\ y_3^{(k)}$
	$x_3^{(i)}\ y_4^{(1)}$	$x_3^{(i)}\ y_n^{(2)}$	$\ldots$	$x_3^{(i)}\ y_2^{(j)}$	$\ldots$	$x_3^{(i)}\ y_1^{(k)}$
	$x_4^{(i)}\ y_3^{(1)}$	$x_4^{(i)}\ y_2^{(2)}$	$\ldots$	$x_4^{(i)}\ y_1^{(j)}$	$\ldots$	$x_4^{(i)}\ y_4^{(k)}$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
	$x_n^{(i)}\ y_1^{(1)}$	$x_n^{(i)}\ y_1^{(2)}$	$\ldots$	$x_n^{(i)}\ y_4^{(j)}$	$\ldots$	$x_n^{(i)}\ y_3^{(k)}$

. (2.6)

In the next step, one of the $(k-1)$ OLHS schemes, or equivalently a column of the multidimensional array in (2.6), (e.g. $x^{(i)}y^{(j)}$, $i \neq j$) is randomly chosen. For the first element of $x^{(i)}y^{(j)}$, which is $x_1^{(i)}y_q^{(j)}$, $(q \leq n)$, q is selected and then all of the y^m

$(m \leq k, m \neq j)$ elements of the q^{th} row are collected. There are $(k-2)$ of such elements. By attaching the initial pair $x_1^{(i)} y_q^{(j)}$ to these elements, a vector of k elements is composed. This procedure is repeated for all n elements of the chosen $x^{(i)} y^{(j)}$. A demonstration of the described technique to create the desired set of samples using the initial multidimensional array of (2.6) is provided below.

$$\begin{bmatrix} y_4^{(1)} & y_n^{(3)} & \dots & x_1^{(i)} & y_3^{(2)} & \dots & y_1^{(k)} \\ y_1^{(1)} & y_1^{(3)} & \dots & x_2^{(i)} & y_n^{(2)} & \dots & y_3^{(k)} \\ y_n^{(1)} & y_3^{(3)} & \dots & x_3^{(i)} & y_2^{(2)} & \dots & y_n^{(k)} \\ y_2^{(1)} & y_4^{(3)} & \dots & x_4^{(i)} & y_1^{(2)} & \dots & y_2^{(k)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ y_3^{(1)} & y_2^{(3)} & \dots & x_n^{(i)} & y_4^{(2)} & \dots & y_4^{(k)} \end{bmatrix}. \tag{2.7}$$

At the end of this process, an $(n \times k)$ matrix is generated where each row corresponds to a sample vector. If the number of samples N_s required is larger than n, the same process is repeated and the new $(n \times k)$ matrix is concatenated to the old one after being compared in order to expel the same sample vectors if any. Finally, the true model outputs are computed using (2.1) for each sample vector of the final $(N_s \times k)$ sample vector matrix.

2.2.4 Basis Function Selection

The next stage of the surrogate modeling process is the selection of the basis functions that establish the surrogate model. For each input variable a separate basis function is assigned with distinct coefficients to be fitted. Both polynomials and Kriging basis functions are commonly used with surrogate models [38]. As pointed out in the previ-

ous chapter, many variability modeling approaches employed polynomial-based response surface models in different contexts. However, as the number of variables increases in a response surface model the number of interaction terms in a polynomial rises sharply making the modeling process very complex. Instead, nonlinear functions should be chosen to capture the variations as accurately as possible while keeping the number of fitting parameters in the model at a reasonable level. Thus Kriging basis functions are more suited to fitting complex response surfaces and were previously used in design and optimization of analog integrated circuits [60], [61], [39].

In Kriging basis functions, the coefficients of basis functions are adjusted so that the combination of basis functions has the best fit to the true model. These functions form the complete surrogate model using the correlation of every two sample vectors that can be computed by

$$C\left[\mathbf{x}^{(i)}, \mathbf{x}^{(\ell)}\right] = \exp\left(-\sum_{j=1}^{k} \theta_j \left|\mathbf{x}_j^{(i)} - \mathbf{x}_j^{(\ell)}\right|^{p_j}\right), \tag{2.8}$$

where C is for the correlation of the sample vectors $x^{(i)}$ and $x^{(\ell)}$, θ_j and p_j are the coefficients of the Kriging basis function, and k is the dimension of the sample vector [38]. Collecting correlations forms the correlation matrix Ψ [38]:

$$\Psi = \begin{pmatrix} C\left[\mathbf{x}^{(1)}, \mathbf{x}^{(1)}\right] & \dots & C\left[\mathbf{x}^{(1)}, \mathbf{x}^{(N_s)}\right] \\ \vdots & \ddots & \vdots \\ C\left[\mathbf{x}^{(N_s)}, \mathbf{x}^{(1)}\right] & \dots & C\left[\mathbf{x}^{(N_s)}, \mathbf{x}^{(N_s)}\right] \end{pmatrix}. \tag{2.9}$$

The values of the coefficients, θ_j and p_j with $j \leq k$ are to be estimated for each dimension of the sample vector. Generally p_j is between 1 and 2 with p_j= 2 corresponding to the Gaussian basis function and this is used here. With the p_js fixed, the problem reduces to optimizing the θ_j values based on the sample vectors. A large θ_j reduces the

variance of the basis functions for the j^{th} variable thereby increasing its contribution on the model response.

Once the complete correlation matrix is calculated for all sample vectors, model response of an input vector $\tilde{\mathbf{x}}$ can be found by computing the correlation of $\tilde{\mathbf{x}}$ with all sample vectors. Each of these terms is individually a Kriging basis function with the i^{th} of them being given as [38]

$$\psi^{(i)} = \exp\left(-\sum_{j=1}^{k} \theta_j \left|\mathbf{x}_j^{(i)} - \tilde{\mathbf{x}}_j\right|^{p_j}\right), \tag{2.10}$$

where $\mathbf{x}_j^{(i)}$ is the j^{th} element of the i^{th} sample vector. The basis function is applied for each $\tilde{\mathbf{x}}$ to every sample vector. For each variable there are two fitting coefficients, θ_j and p_j.

The model response uses the conventional interpolation equation

$$\tilde{y} = \mu + w^{\mathrm{T}}\psi(\tilde{\mathbf{x}}), \tag{2.11}$$

where $\tilde{y}$ is the model's response for an input vector $\tilde{\mathbf{x}}$, μ is the offset term and w is the weighting vector [38]. Both μ and w are functions of θ_j, p_j and the sample vectors. The vector of basis function evaluations is

$$\psi(\tilde{\mathbf{x}}) = \begin{bmatrix}\psi^{(1)}(\tilde{\mathbf{x}}) & \psi^{(2)}(\tilde{\mathbf{x}}) & \cdots & \psi^{(N_s)}(\tilde{\mathbf{x}})\end{bmatrix}^{\mathrm{T}}, \tag{2.12}$$

and there are N_s sample vectors. In effect the Kriging basis function approach is analogous to a smart table lookup procedure. A full mathematical treatment of Kriging basis functions can be found in [38] and [62].

Optimum θ parameters can be determined via a local or global search algorithm. In this study, the MATLAB® toolbox, DACE (Design and Analysis of Computer Experiments) was used to determine the θ_js and hence establish the surrogate model [63]. DACE finds the θs through a pattern search procedure similar to the Hooke-Jeeves method [63]. Compared to a standard global search method, such as the genetic algorithm, this approach yields the desired set of coefficients much quicker with a considerable accuracy.

2.2.5 Testing the Surrogate Models for Accuracy

Accuracy of surrogate models must be characterized precisely since the model prediction error directly depends on that. In this work, two different metrics have been used to express the accuracy of a surrogate model. Mean square error (MSE) computes the average sum of differences squared between the model response and the true model output for a number of test vectors. Root mean square error (RMSE) is the square root of MSE. Conversely, root relative square error (RRSE) expresses the error by referencing MSE to the deviation of true model outputs from their mean value squared for all test vectors:

$$\text{RMSE} = \sqrt{\frac{1}{N_t}\sum_{n=1}^{N_t}\left(y_{\text{model},n} - y_{\text{true},n}\right)^2}, \tag{2.13}$$

$$\text{RRSE} = \sqrt{\frac{\sum_{n=1}^{N_t}\left[y_{\text{model},n} - y_{\text{true},n}\right]^2}{\sum_{n=1}^{N_t}\left[y_{\text{true},n} - \frac{1}{N_t}\sum_{n=1}^{N_t} y_{\text{true},n}\right]^2}}. \tag{2.14}$$

In (2.13) and (2.14) N_t denotes the number of test vectors, $y_{\text{true},n}$ and $y_{\text{model},n}$ are the outputs of the true model and the generated model, respectively, for the n^{th} test vector. Since RRSE expresses the accuracy as a fraction rather than in absolute terms —as RMSE does— it is more useful in comparing the quality of a model.

2.3 Surrogate Modeling Results

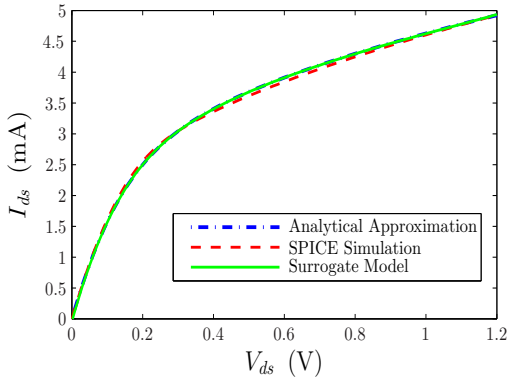
Surrogate models were developed for the IBM 10SF (65 nm) process using the analytic drain current expresion in (2.1) slightly modified to reconcile it with simulations. The transistor width was chosen to be 10 μm for both NMOS and PMOS devices. W_{eff} variations on I_{ds} are negligible above values of $W_{\text{eff}} \geq 1$ μm, thus W_{eff} has not been included within input vector parameters. This also means that it is sufficient to take W_{eff} of a certain transistor, divide it to 10 μm and then multiply the result with the I_{ds} surrogate model outcome to yield the I_{ds} of that transistor.

The generation of the surrogate models were realized using an INTEL I7 Quad Core 3.5 GHz processor. In developing the NMOS drain current surrogate model with $t_{\text{age}} = 0$, 3840 sample vectors and 3200 test vectors were used to evaluate an RMSE of 37.3 μA and an RRSE of 1.09%. For the PMOS drain current surrogate model ($t_{\text{age}} = 0$), 1920 sample vectors and 3200 test vectors were employed which yields an RMSE of 27.4 μA and an RRSE of 1.81%.

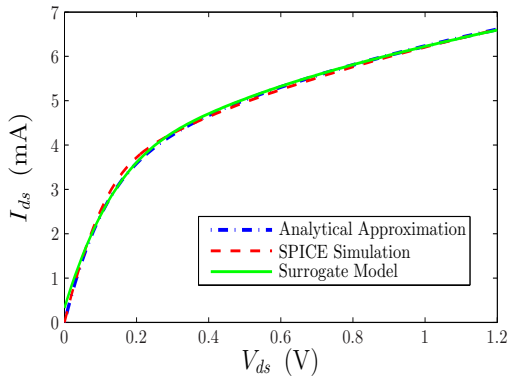
The generated surrogate models are MATLAB® objects which encompass two important quantities. These are the optimized θ values and the correlation matrix Ψ which is a $N_s \times N_s$ matrix where N_s is the number of sample vectors allocated to construct the surrogate model. The size of Ψ not only determines the memory space used for the surrogate model but also the model generation and evaluation times.

The surrogate model results were also compared to the analytical approximation, (2.1), and the SPICE circuit simulation results for a single transistor, see Figures 2.3 and 2.4. Various process corners are chosen along with maximum and minimum values for the operating temperature. The body effect has also been tested by assuming different V_{bs} levels. The graphs depict good agreement as shown over the full range of V_{gs} and V_{ds}

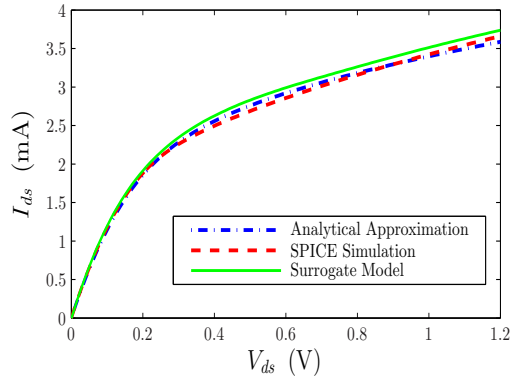
and at the process corners and temperature extremes. The importance of these graphs comes from the fact that surrogate models are able to capture even the most excessive variability in the output quantity with sufficient accuracy. Thus they are potentially a good candidate for circuit variability and reliability analysis.



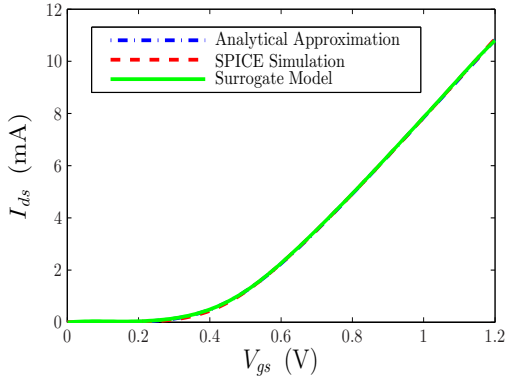
(a): NN, $T = 27°\text{C}, V_{bs} = 0$ V



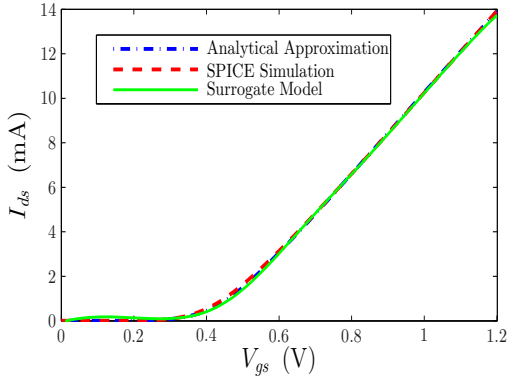
(b): FS, $T = -100°\text{C}, V_{bs} = 0.4$ V



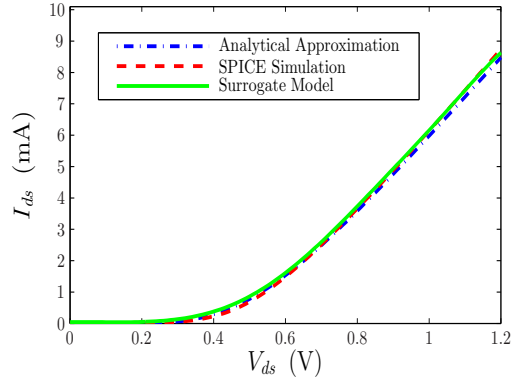
(c): SS, $T = 100°\text{C}, V_{bs} = -0.4$ V



(d): NN, $T = 27°\text{C}, V_{bs} = 0$ V

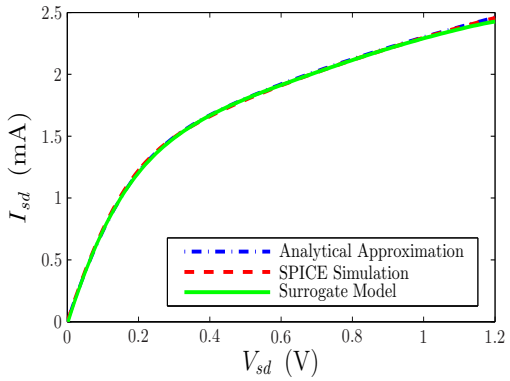


(e): FS, $T = -100°\text{C}, V_{bs} = 0.4$ V

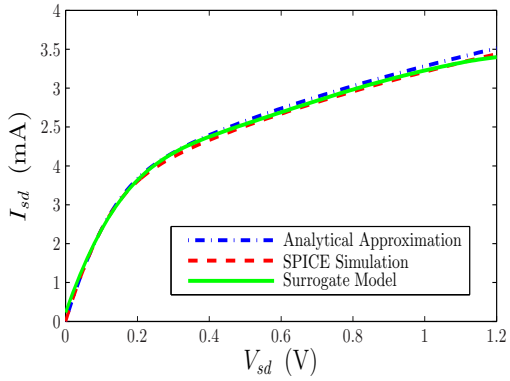


(f): SS, $T = 100°\text{C}, V_{bs} = -0.4$ V

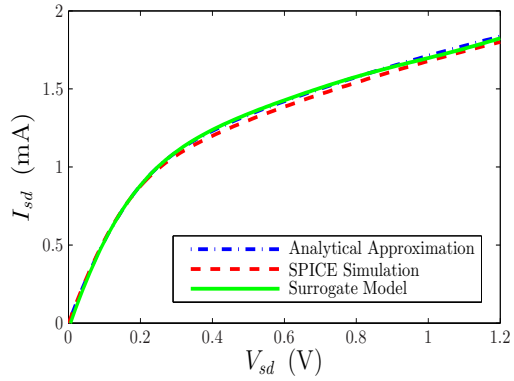
Figure 2.3: Comparison of extracted surrogate models, approximate analytical drain current model equations, and SPICE simulations. In the graphs where $|V_{ds}|$ is varied, $|V_{gs}|$ is held at 0.8 V. Conversely, when $|V_{gs}|$ is swept, $|V_{ds}|$ is fixed to 1.2 V. Good agreement (maximum error $\approx 5\%$) has been achieved.



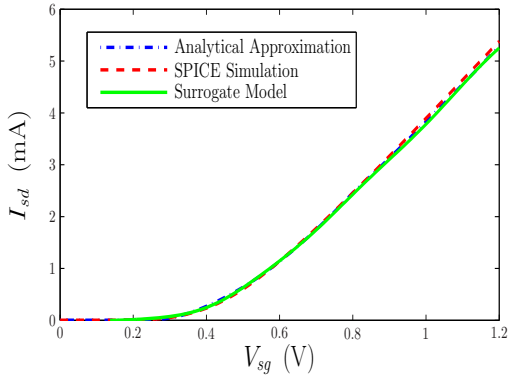
(a): NN, $T = 27°\text{C}, V_{sb} = 0$ V



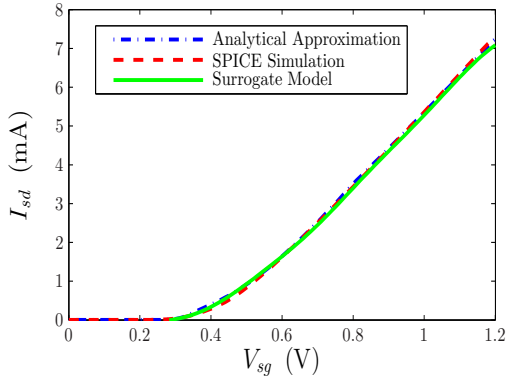
(b): FF, $T = -100°\text{C}, V_{sb} = 0.4$ V



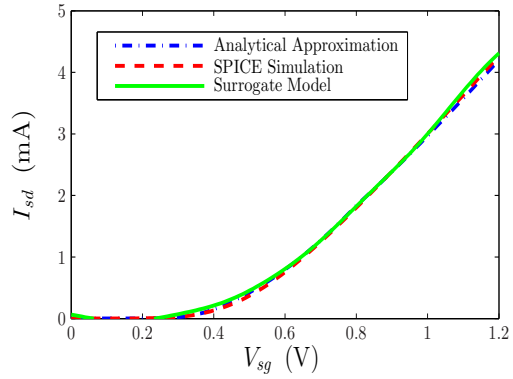
(c): SS, $T = 100°\text{C}, V_{sb} = -0.4$ V



(d): NN, $T = 27°\text{C}, V_{sb} = 0$ V



(e): FF, $T = -100°\text{C}, V_{sb} = 0.4$ V



(f): SS, $T = 100°\text{C}, V_{sb} = -0.4$ V

Figure 2.4: Comparison of extracted surrogate models, approximate analytical drain current model equations, and SPICE simulations. In the graphs where $|V_{ds}|$ is varied, $|V_{gs}|$ is held at 0.8 V. Conversely, when $|V_{gs}|$ is swept, $|V_{ds}|$ is fixed to 1.2 V. Good agreement (maximum error $\approx 5\%$) has been achieved.

2.4 Summary

In order to achieve a variability and reliability analysis of analog, mixed-signal, and digital circuits that takes into account both the die-to-die and within-die variations, entities for each transistor in a given topology should be defined. These entities must capture the process variations space, terminal voltages, temperature, and device age. Reduced-order models can be used for this purpose where models for design quantities of interest such as I_{ds} are developed in terms of a handful of parameters.

In this chapter the surrogate modeling concept was introduced. A surrogate model is a reduced order model which takes samples that are vectors of input parameters. The surrogate model for I_{ds} described in this chapter has eleven parameters. Among these parameters, six of them are BSIM process parameters all of which contribute to the variability of nanoscale transistors as explained in detail in Chapter 1. Additionally, three terminal voltage differences as well as the device temperature and device age were included to fully describe the each transistor. An orthogonal Latin hypercube sampling scheme was developed to create the appropriate sample vectors for efficient model generation. The true model for I_{ds} is chosen to be the analytic BSIM model equation which incorporates multiple physical effects that lead to further variability in nanoscale circuits. While circuit simulations could be preferred for the true model that results in a higher computational cost. Kriging basis functions are used to for which the coefficients are optimized using a MATLAB® toolbox called DACE. The accuracy of the models has been quantified using RMSE and RRSE.

The methodology has been applied and models for I_{ds} of unaged ($t_{\text{age}} = 0$) of 65 nm NMOS and PMOS transistors are constructed. The model predictions, the analytic expression on which the surrogate models are based, as well as the single transistor

circuit simulations were compared. The results indicate that the developed models can be used for variability and reliability analysis of circuits.

Chapter 3

Comparison of Different Reduced-Order Modeling Schemes

3.1 Introduction

In Chapter 2, the concept of reduced-order modeling was introduced and a special class, surrogate models, was described with all stages of development. In this chapter, two more types of reduced-order modeling are presented and compared with surrogate modeling on the basis of several important modeling metrics. These are the neural-network based models and least squares support vector machine (SVM)-based models. The models will be analyzed with respect to their accuracy, establishment time, size, and evaluation time. In particular, I_{ds} modeling of 65 nm transistors is considered as a case study. The aim here is to determine the trade-offs between alternative modeling schemes and make the most useful choices for the variability and reliability analyses described in the next chapters.

It should be noted that the same input parameters along with the same sampling

techniques are used to generate the data set for all modeling strategies. The same true model for I_{ds} of NMOS and PMOS transistors is used to evaluate the sample vectors. The surrogate model examples demonstrated in this chapter are developed based on the methodology that was presented in Chapter 2. An additional technique is presented with which the accuracy of constructed surrogate models can be increased by adding a few sample vectors to the list of existing sample vectors. The addition of new sample vectors follows an error-based criterion that is explained in detail in Subsection 3.3.1.

Similar to surrogate modeling, neural network modeling is used to capture complex response surfaces in Subsection 3.3.2. The inputs of the true model (process parameters, terminal voltages and temperature) are connected to the neurons in what is called the hidden layer through weighted connections. The value of the input variables affects the outcome of the activation function at the neuron which has the characteristics of a continuous threshold function [64]. Neural networks have a large application domain ranging from analog design synthesis [65], to behavioral modeling of RF circuit blocks [66], and to process control modeling for integrated circuit (IC) fabrication [67]. Hybrid modeling strategies involving neural networks and space mapping have been employed in the context of microwave filter design [68], and large-signal modeling of high-frequency devices [69].

The support vector machine (SVM), discussed in Subsection 3.3.3, is a machine learning technique that can be applied to the variability analysis problem both as a classification tool and as a regression tool. As a regression tool, SVM treats modeling as an optimization problem for which the Euclidean norm of the weight vector is minimized such that the modeled curve can be estimated within a specified confidence interval [70]. SVMs have been used in analog design synthesis and performance modeling of amplifiers [71], [72], oscillators [73], and resistive mixers [74]. There are different mathematical

formulations of SVMs for functional approximation including ε-SVM regression, ν-SVM regression, and least squares SVM [64], [70]. In this work, the least squares SVM is adopted as the representative of SVM-classes of modeling.

In the literature, the models produced using these modeling techniques have been compared with each other in modeling the admittance matrix elements and input referred noise of a low noise amplifier (with the number of dimensions being six) [75], and in the sensitivity analysis of a computational fluid dynamics model (with three dimensions) [76]. The number of model dimension in our study is ten ($t_{\text{age}} = 0$ is assumed) which is a much higher dimensional modeling problem than is normally considered. It will be shown that with the same number of samples, models of better accuracy can be established using an orthogonal approach to sample selection.

3.2 Model Comparison Criteria

Reduced-order models developed in this chapter have been compared on the basis of four important metrics that are model accuracy, model establishment time, model size, and model evaluation time.

The number of samples from which a model is constructed has an important impact on its accuracy. Using too few samples results in low accuracy, but too many samples leads to excessive model extraction and (usually) model evaluation times, as well as resulting in what is called generalization error which comes from overfitting data [64]. The optimal number of samples can be found empirically using a group of test vectors, differing from the sample vectors used in generating the model. In this study, 3200 test vectors are employed to evaluate the accuracy of the extracted models. Accuracy is characterized by RMSE and RRSE as defined in (2.13) and (2.14), respectively.

The model establishment time is a measure of the computational cost of developing the models. It is aimed to select the modeling strategy that minimizes the time elapsed until the model generation is fully completed. In that way, the complexity of the modeling approach can also be limited to some extent.

The model size is a measure of the memory required to store the model. Generally, many model quantities can be of interest at any point in a circuit electrical simulation, thus bulky models with large memory consumption can restrict the usage of surrogate models in memory-limited applications.

Finally, the model evaluation time is a crucial memory for the models to be established as a very large number of model evaluations is required to quantify the variability of circuits. Shortest evaluation time will be the most decisive characteristic in the comparison of modeling strategies.

3.3 Modeling Results

In this section the results of Kriging-based surrogate models, neural network-based models and SVM-based models are described and then compared.

3.3.1 Kriging-Based Surrogate Modeling

Surrogate modeling approach that has been introduced in Chapter 2 is utilized in this section. The sample vectors are generated using the OLHS methodology as described previously. Using these sample vectors, the two coefficients of Kriging basis functions for the j^{th} input parameter, p_j and θ_j are optimized where $j \leq k$, and k is the number of input parameters.

In order to reduce complexity, p_j is typically set to 2, which essentially brings the form

of the Kriging basis function close to the Gaussian basis function. As previously, optimum θ_js are found using the MATLAB® toolbox DACE (Design and Analysis of Computer Experiments) [63]. The metrics of the I_{ds} model developed using sample vector sets of different sizes are tabulated in Table 3.1. It is seen that more samples (higher N_s) results in lower RRSE. Correspondingly a larger correlation matrix Ψ must be retained resulting in longer model generation and model evaluation times.

Table 3.1: Kriging-based surrogate model results for I_{ds}.

N_s	Accuracy (RRSE)		Generation Time (s)		Model Size (MB)		Evaluation Time (ms)	
	NMOS	PMOS	NMOS	PMOS	NMOS	PMOS	NMOS	PMOS
640	7.74%	7.53%	16.3	18.1	1.64	1.64	0.36	0.36
1280	5.68%	4.93%	67	73	6.71	6.71	0.44	0.47
1920	4.38%	3.77%	170	177	15.55	15.55	0.53	0.56
2520	3.71%	2.40%	319	323	28.3	28.32	0.91	0.92
3200	3.16%	2.12%	516	520	44.7	44.74	0.89	0.90
3840	2.80%	1.90%	790	778	64.8	64.86	1	1

When a surrogate model is constructed, DACE produces a final set of θ elements. In the developed procedure, these are fed back to the model as the initial set of θ in the new iteration. Furthermore, a group of test vectors is prepared in each loop. Depending on the computational cost, test vectors can be partitioned in several groups corresponding to each iteration or they can be reproduced. Each test vector contained in the group is evaluated using both the generated surrogate model and the true model. The difference between the value that the surrogate model projects and the true model output for each

test vector is collected in a separate array. The test vector that leads to the highest error is added to the existing sample vectors list. Thus each iteration of model construction starts with different θ_js and one additional sample vector. This approach resembles to the error based sampling criteria often claimed to produce clustering which complicates the θ-search procedure [39]. However if the test vectors are created after a random sampling step, the highest error in each iteration does not have to stem from the same region of the design space. Thus clustering problem has not been observed in this study. The whole procedure is depicted in the flowchart of Figure 3.1.

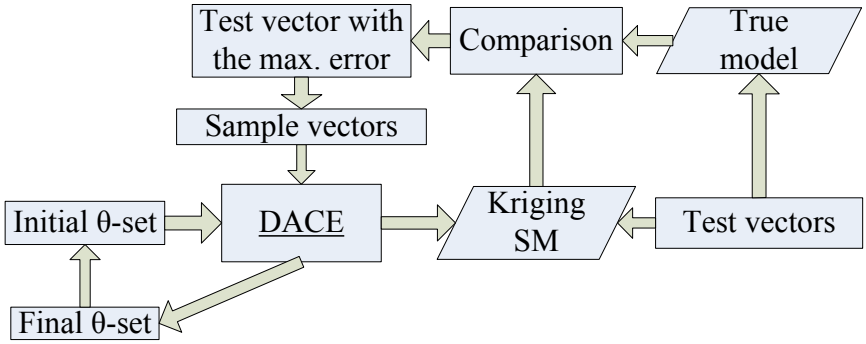
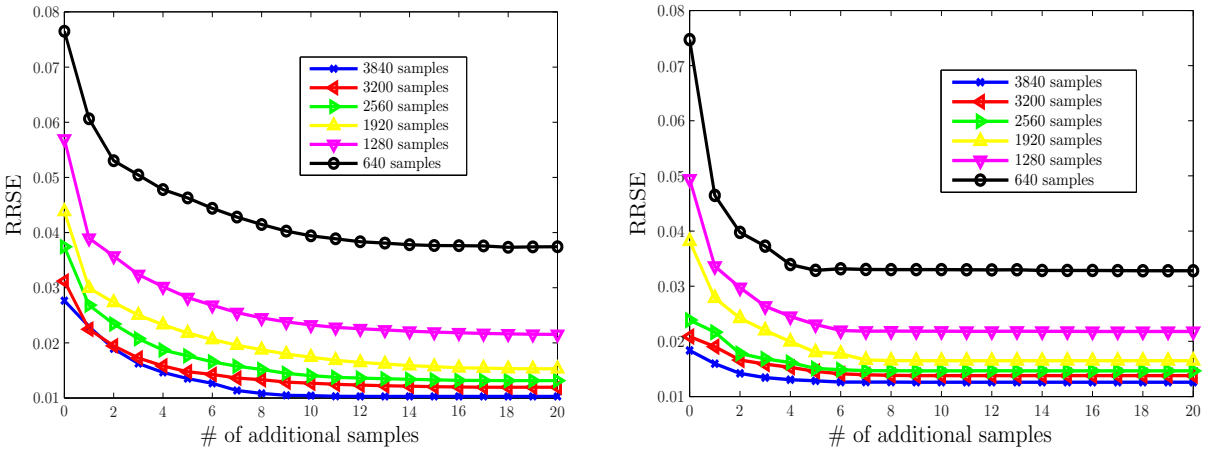


Figure 3.1: Flowchart explaining the RRSE reduction mechanism for I_{ds} using Kriging-based surrogate modeling.

The effect of doing this for up to 20 additional sample vectors is shown in Figure 3.2. The reduction in RRSE indicates that a significant enhancement of model accuracy is obtained with a minor increase in model size and thus model evaluation time.



(a): NMOS model (b): PMOS model

Figure 3.2: The reduction in RRSE for NMOS and PMOS devices.

3.3.2 Neural Network-Based Modeling

The neural network-based model extracted here has three layers, the input layer, the hidden neuron layer, and the output neuron layer (see Figure 3.3). The input layer has $k = 10$ elements. The neurons in the hidden layer have a nonlinear activation function ψ. Given that there are N_n hidden neurons, the connections from the input layer to the j^{th} neuron of the hidden layer are weighted by the weight vector $\mathbf{w}_j$. The outputs of the hidden neurons are combined at the output neuron in the output layer.

For a high level of model accuracy, the model should find the optimum weight vectors that will minimize the average error energy, E_{av}, which can be defined as

$$E_{\text{av}} = \frac{1}{N_s}\sum_{i=1}^{N_s} E\left(i\right) = \frac{1}{2N_s}\sum_{i=1}^{N_s}\left[d\left(i\right) - y\left(i\right)\right]^2, \tag{3.1}$$

where the $d\left(i\right)$s are the true model outputs and $y\left(i\right)$s are the generated model outputs

of the i^{th} sample vector [64]. The $y\,(i)$s are evaluated by the neural network as

$$y\,(i) = \sum_{j=1}^{N_n} \psi \left(\sum_{\ell=1}^{k} w_{j\ell}\,(i)\, x_\ell\,(i) \right) \tag{3.2}$$

where w_{jl} is the weight of the connection from the ℓ^{th} element of the sample vector, x_ℓ, to the j^{th} neuron. Neural networks generally use online learning which means that the weight vector $\mathbf{w}_j$ of every hidden neuron is updated with every sample vector in each iteration. The change in w_{jl}, denoted as $\Delta w_{j\ell}$, can be estimated using the back-propagation method by computing the successive degrees of partial derivatives $\partial E_{\text{av}}/\partial w_{j\ell}$. The first- and second-order derivatives are called the local gradient vector and local Hessian matrix which can be expressed for the i^{th} sample vector as follows [64]:

$$\mathbf{g}\,(i) = \left. \frac{\partial E_{\text{av}}\,(\mathbf{w}_j)}{\partial \mathbf{w}_j} \right|_{\mathbf{w}_j = \mathbf{w}_j(i)}, \tag{3.3}$$

$$\mathbf{H}\,(i) = \left. \frac{\partial^2 E_{\text{av}}\,(\mathbf{w}_j)}{\partial \mathbf{w}_j^2} \right|_{\mathbf{w}_j = \mathbf{w}_j(i)}. \tag{3.4}$$

The MATLAB® Neural Network toolbox was used with the Levenberg-Marquardt back-propagation method. Furthermore, the nonlinear activation function is chosen to be the hyperbolic tangent sigmoid function. The change in the weight vector $\Delta \mathbf{w}_j\,(i)$ in each iteration is found as

$$\Delta \mathbf{w}_j\,(i) = [\mathbf{H}\,(i) + \lambda \mathbf{I})]^{-1} \mathbf{g}\,(i)\,, \tag{3.5}$$

where $\mathbf{I}$ is the identity matrix and λ is a regularization parameter ensuring that the matrix in parenthesis is positive definite [64]. Thus the weight vectors are updated as

follows:

$$\mathbf{w}_j\,(i+1) = \mathbf{w}_j\,(i) + \Delta\mathbf{w}_j\,(i). \tag{3.6}$$

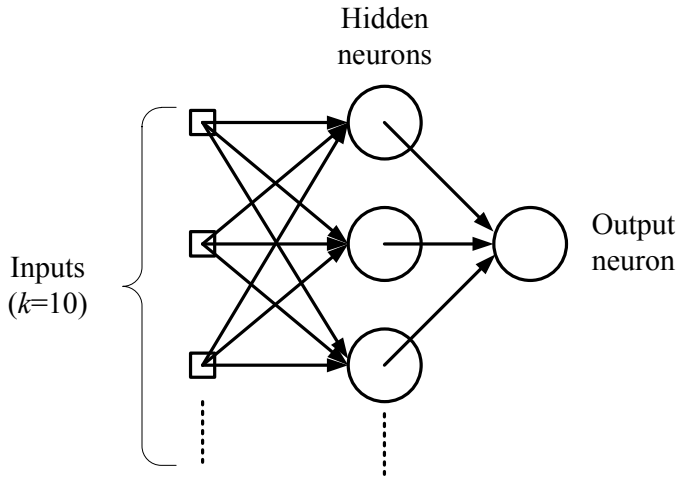


Figure 3.3: The structure of the neural network-based model with ten inputs.

The modeling of I_{ds} using a neural network involves two experiments. In the first, the effect of N_s is investigated. The results for various N_s values are tabulated in Table 3.2. As the number of samples increases, better accuracy is obtained at the expense of higher model generation time. The model size also rises with higher N_s though not at the same rate. It should be noted that the model evaluation time does not depend on N_s. In the second experiment, the number of hidden neurons, N_n, is varied and the results are recorded in Table 3.3. N_n should be sufficiently large to characterize the true model correctly since more than the necessary number results in additional modeling complexity. Table 3.3 also indicates that higher N_n rapidly raises the model establishment time but results in modest increases in model size. Again the model evaluation time remains constant, independent of N_s. It is worth noting that having 200 neurons instead of 50

or 100 neurons actually degrades the accuracy of PMOS models and increases model complexity considerably. Thus a neural network-based model with 100 hidden neurons in conjunction with 3840 samples yields a suitable model for circuit variability analysis.

Table 3.2: Neural network-based model results for I_{ds} with a constant $N_n = 100$.

N_s	Accuracy (RRSE)		Generation Time (s)		Model Size (kB)		Evaluation Time (ms)	
	NMOS	PMOS	NMOS	PMOS	NMOS	PMOS	NMOS	PMOS
640	18.16%	13.57%	5	4	49	49	7.6	7.5
1280	3.47%	3.91%	32	14	72	74	7.5	7.4
1920	2.29%	3.31%	55	42	95	98	7.5	7.4
2560	1.57%	1.68%	73	74	118	122	7.5	7.5
3200	1.45%	1.43%	73	121	141	146	7.5	7.3
3840	2.42%	1.18%	265	198	165	170	7.7	7.4

Table 3.3: Neural network-based model results for I_{ds} with a constant $N_s = 3840$.

N_s	Accuracy (RRSE)		Generation Time (s)		Model Size (kB)		Evaluation Time (ms)	
	NMOS	PMOS	NMOS	PMOS	NMOS	PMOS	NMOS	PMOS
10	3.58%	4.39%	24	23	146	152	7.5	7.8
20	3.27%	2.20%	22	40	148	154	7.5	7.5
50	1.47%	1.63%	94	58	155	160	7.5	7.4
100	1.06%	1.18%	265	198	165	170	7.3	7.4
200	0.91%	1.23%	812	759	185	191	7.6	7.4

3.3.3 Least Squares SVM-Based Modeling

Least squares SVM-based modeling solves the regression problem by attempting to ensure that for each sample the output of the generated model is within a tolerance of ε of the true model response surface. This objective is posed as an optimization problem in which the weight vectors $\mathbf{w}$ are to be minimized. The sample vectors that are located at the boundaries of this interval are called the support vectors. Minimization of the weight vectors is important for model generalization over test vectors with less complexity [38]. In order to solve this minimization problem, the sample vectors are treated by a kernel. The aim is to increase the dimensionality (as much as to infinity) through a nonlinear function so that the sample vectors become linearly separable (Cover's Theorem [64]). In this study, radial basis functions (RBF) are chosen for that purpose, which can be provided as

$$\varphi\left(\tilde{\mathbf{x}}, \mathbf{x}_j\right) = \exp\left(-\frac{\left\|\tilde{\mathbf{x}} - \mathbf{x}_j\right\|^2}{\sigma^2}\right), \tag{3.7}$$

with $\tilde{\mathbf{x}}$ being any input vector in the design space, $\mathbf{x_j}$ being a sample vector and σ representing the RBF variance [71]. Thus the modeling in the transformed dimensionality (also called the feature space) becomes

$$y\left(\tilde{\mathbf{x}}\right) = \mathbf{w^T}\varphi\left(\tilde{\mathbf{x}}\right) + \mu, \tag{3.8}$$

where μ is a bias term and $y\left(\tilde{\mathbf{x}}\right)$ is the corresponding model output to $\tilde{\mathbf{x}}$.

Based on this knowledge, the optimization primal problem can be set up as

$$\begin{aligned} &\min_{\mathbf{w},\mu,\mathbf{e}} J_P\left(\mathbf{w}, \mathbf{e}\right) = \frac{1}{2}\mathbf{w^T}\mathbf{w} + \frac{\gamma}{2}\sum_{i=1}^{N_s} e_i^2 \\ &\text{such that} \qquad y_i = \mathbf{w^T}\varphi\left(x_i\right) + \mu + e_i, \quad i = 1, \ldots, N_s \end{aligned} \tag{3.9}$$

where γ is a regularization parameter and $\mathbf{e}$ consists of N_s error variables [70]. It can be shown that the dual optimization problem can be written as linear Karush-Kuhn-Tucker system of equations from which the final expression for the function estimation is derived [70].

In this study, the MATLAB® toolbox LS-SVMlab is used to establish least square SVM models in conjunction with RBFs [77]. For the RBFs, γ and σ^2 values are chosen such that the maximum possible model accuracy is attained. All results are tabulated in Table 3.4. It can be observed that there is a slight increase in the model generation and model evaluation times as N_s increases. Also, the accuracy improves accompanied by larger model sizes with higher N_s values.

Table 3.4: Least square SVM-based model results for I_{ds}.

N_s	Accuracy (RRSE)		Generation Time (s)		Model Size (kB)		Evaluation Time (ms)	
	NMOS	PMOS	NMOS	PMOS	NMOS	PMOS	NMOS	PMOS
640	13.81%	13.35%	0.09	0.10	27	27	8.0	7.9
1280	10.92%	10.59%	0.48	0.51	44	44	9.5	8.7
1920	8.44%	7.90%	1.27	1.35	64	64	10.1	9.6
2520	7.35%	6.80%	2.98	2.97	84	84	11.4	10.6
3200	6.89%	6.32%	5.13	5.02	104	104	12.4	11.6
3840	6.37%	5.93%	8.77	8.00	122	122	13.2	12.7

3.3.4 Comparison of the Modeling Strategies

The metrics of the three types of models can now be compared by examining Tables 3.1, 3.2, 3.3, and 3.4. Neural network-based models have accuracy comparable to the Kriging-based surrogate models (See Tables 3.1 and 3.2). Nevertheless, least square SVM-based models cannot attain the same accuracy possibly because the strategy of optimizing the confidence interval around the response surface is not very effective with complex surfaces.

The model generation time is hard to compare because each modeling tool has a different stopping criterion. Nonetheless, among all three alternatives, the fastest to generate is the least square SVM-based model which lacks the desired accuracy. Conversely, for the same accuracy level, neural network-based models are generated faster than the Kriging-based models if $N_n \leq 100$. This partially originates due to the interpolation structure of Kriging-based models which enforces to build a correlation matrix Ψ that takes longer time to construct than $\mathbf{w}$ optimization adopted by the other modeling strategies.

Kriging-based models are much larger than the neural-network based models and least square SVM-based models because Ψ has to be stored for future model predictions. It should also be noted that SVM-based models are smaller in size compared to the neural network-based models thus being the minimum among all alternatives. The size of the model can be an issue when many of these models have to be developed and stored to evaluate different design quantities in a complete simulator. Compactness will be a preference of practical usage.

The model evaluation time comparison indicates Kriging-based surrogate models as the best candidate. For $N_s = 3840$, Kriging-based surrogate models are 7.5 times faster

than neural network-based models and 13 times faster than least square SVM-based models. This is a crucial advantage as it has a direct impact on the duration of the circuit variability analysis. However for large circuits with many devices, even 1 ms can result in uncompetitive results. As a remedy, rather than implementing successive single model evaluations, a batch of test vectors can be provided to the model prediction function. All three models have been tested by a test vector set of size $N_t = 3200$ and their evaluation times are recorded. Results are tabulated in Table 3.5. It can be seen that neural-network based models have the shortest implementation time much better than Kriging-based and least square SVM-based models. Also, for the latter two, the single model evaluation times rise proportionally with higher N_s, whereas neural network-based models become unaffected. The reason can be the implementation style of the model evaluation that takes into account the parallel structure of neural networks. As to the circuit variability analysis, one can conclude for smaller circuits (# of n- or p-type transistors ≤ 8), Kriging-based models should be employed whereas larger scale circuits have to be analyzed using neural-network based models.

To perform an accurate circuit variability analysis, RRSE of the generated model should not exceed 5%, thus in the forthcoming analysis sections only Kriging-based models and neural network-based models will be considered.

3.4 Summary

The chief aim of this chapter was to compare three types of reduced-order modeling approaches that are candidates to be used in circuit variability and reliability analysis. These approaches are Kriging-based surrogate model, a neural network-based model, and a least squares support vector machine-based model. All these methods use the same set

Table 3.5: Comparison of model evaluation times for a test vector batch with $N_t = 3200$.

N_s	Neural Network Model		Kriging Surrogate Model		Least Square SVM	
	NMOS	PMOS	NMOS	PMOS	NMOS	PMOS
640	28.4 ms	27.8 ms	0.94 s	0.90 s	0.13 s	0.12 s
1280	26.9 ms	26.5 ms	1.70 s	1.70 s	0.24 s	0.23 s
1920	27.2 ms	27.2 ms	2.42 s	2.32 s	0.35 s	0.34 s
2560	27.1 ms	27.6 ms	3.56 s	3.39 s	0.46 s	0.45 s
3200	27.9 ms	28.0 ms	3.90 s	3.82 s	0.56 s	0.54 s
3840	27.1 ms	27.6 ms	4.92 s	4.71 s	0.66 s	0.67 s

of sample vectors generated by the sampling methodology described in Chapter 2 with the same input parameters.

The comparison is based on model accuracy, model generation and evaluation times as well as the model size. I_{ds} modeling problem as before has been considered again since it creates the basis of circuit variability and reliability analysis introduced in the next chapters of the dissertation.

All modeling approaches have been implemented in MATLAB® environment by using different toolboxes. Although the least square SVM constructs the models with the least model size, the comparison has also shown that only Kriging-based surrogate models and neural network-based models can be generated with sufficient accuracy so that they can be used in circuit variability analysis. A methodology to increase accuracy of Kriging-based surrogate models was developed by adding sample vectors using an error-based strategy. Numerical experiments demonstrated that for smaller circuits, Kriging based-surrogate modeling yields results faster than the neural network-based models for the same accuracy whereas for larger circuits, neural network-based models are preferred

since, in all metrics, better performance is obtained.

Chapter 4

Circuit Variability Analysis Using Reduced-Order Models

4.1 Introduction

Variability and reliability analysis can be performed efficiently and simultaneously using surrogate models. The designer can independently choose in the input vector for each device the process variables as well as the operating temperature and the time frame. Voltages are initialized from the circuit schematic and updated during the analysis.

In this chapter variability analysis of the DC drain current characteristics is investigated. However this does not limit the generality of the surrogate model based approach. Surrogate models of other quantities (such as the small signal transconductance g_{m} or the input return loss S_{11} for the narrow-band operation of an RF transistor) could be developed to study the AC or RF variability. The focus on DC characteristics arises from the reliability analysis perspective as the time-based degradations predominantly degrade V_{th} and μ_{eff} leading to reductions in the drain current as will be discussed more

in detail in Chapter 5.

The chapter starts with a general discussion on the methodology pursued to do circuit analysis using surrogate models only in Section 4.2. Then three different variability analysis examples are provided. In the first example (Section 4.3) the effect of variability on DC operating conditions of a differential amplifier is described. DC biasing of differential amplifiers can impact other characteristics such as the voltage gain, linearity etc., thus the level of variability that will lead to undesired circuit performance can be determined via the analysis presented. As described in Chapter 3, both the Kriging-based surrogate and neural network-based I_{ds} models are employed in the variability analyses of two differential amplifiers in Section 4.3. Within-die variations are going to be analyzed in detail by randomly assigning different process corners to individual devices in the differential amplifier topology. Furthermore, in Section 4.3, the capabilities of surrogate modeling are demonstrated by applying Monte Carlo analysis based on derived surrogate models.

In the second example (Section 4.4) effects of within-die variations on voltage transfer characteristics for an XOR circuit are analyzed. Study results have shown that the nonlinear reduced-order models (Kriging-based surrogate and neural network-based models) developed can more effectively capture the within-die variations than the traditional process corner analysis. Finally, in the third example (Section 4.5), the impact of individual variations of some process parameters and temperature all which are sample vector elements, on the total I_{ds} difference of two NMOS transistors is discussed given they have the exact same biasing conditions.

4.2 Simulation Techniques using Surrogate Models

In DC circuit analysis using surrogate models, the intermediate voltages whose values are initially unknown are determined by equating the drain currents of neighboring transistors. Optimizing drain-source voltages brings the difference between successive drain currents to a minimum. A pseudo-code for a binary search algorithm to obtain V_{ds} of an n-channel device given its DC current $I_{ds,\text{act}}$, its width w_{n_1}, the input vector $\mathbf{NM}_{n_1}$ describing its process variables, temperature and age, and its associated device surrogate model SM_{nmos} is provided in Figure 4.1. While the function is designated for an NMOS transistor it can also be equivalently used for a PMOS given that the sign reversals for the terminal voltage differences are appropriately done.

In analog circuits, generally, intermediate nodes connect either one device directly to another (e.g. cascode configuration) or one device to two separate devices (e.g. differential pair). Thus most often, large circuits can be decomposed based on this grouping. Subsequently exhaustive search can be used to find the optimum node voltage (for these groupings). Exhaustive search is applied in two phases. In the first phase, a coarse region is located where the optimum voltage is expected. In the second phase, exhaustive search is repeated in this coarse region to find the finer solution of the intermediate voltage level. The algorithmic operation for a differential pair depicted in Figure 4.2 is shown in the pseudo-code of Figure 4.3.

Based on these two approaches, circuit analysis can be performed by using surrogate models only where binary search algorithms are used in conjunction with exhaustive searches. It is necessary to prepare a behavioral description of the given circuit with which the surrogate model-based circuit simulation is able to proceed. In this behavioral description the search for the critical node voltages are done by minimizing the current

```
function Vds_det (I_ds,act, w_n1, NM_n1, SM_nmos)
    V_d,up ← V_dd
    V_d,down ← 0
    V_ds,init ← (V_d,up + V_d,down) /2
    ΔI_ds ← M                                          ▷ M is a large number
    while (|ΔI_ds| ≥ ε) do
        if (ΔI_ds ≤ 0) then
            V_d,up ← V_ds,init
        else
            V_d,down ← V_ds,init
        end if
        V_ds,init ← (V_ds,up + V_ds,down) /2
        dev_param (V_ds) ← V_ds,init
        dev_param (V_gs) ← V_ds,init                   ▷ If diode-connected
        I_ds,pred ← w_n1 × predict (NM_n1, SM_nmos)
        ΔI_ds ← I_ds,act − I_ds,pred
    end while
    Vds_det ← V_ds,init
end function
```

Figure 4.1: Pseudocode to find the V_{ds} when the DC current is specified. ϵ should be chosen larger than the RMSE of the surrogate model to ensure reliable results. The function predict takes the transistor parameters and the surrogate model and finds the corresponding device current.

difference between the incoming and outgoing currents.

The search for optimized node voltages can also be accomplished with other single variable search algorithms, as well. However, it should be noted that since the surrogate model outcomes are discrete, no derivative information is available, thus the gradient-based search methods are not applicable in this problem. However, sequential search methods such as the golden section search or Fibonacci search methods can be used instead, that can easily outperform exhaustive search methods on the condition that the function is unimodal [78]. Unimodality of a function is defined as having only one minimum or maximum point in a given range of the input variable. This condition may

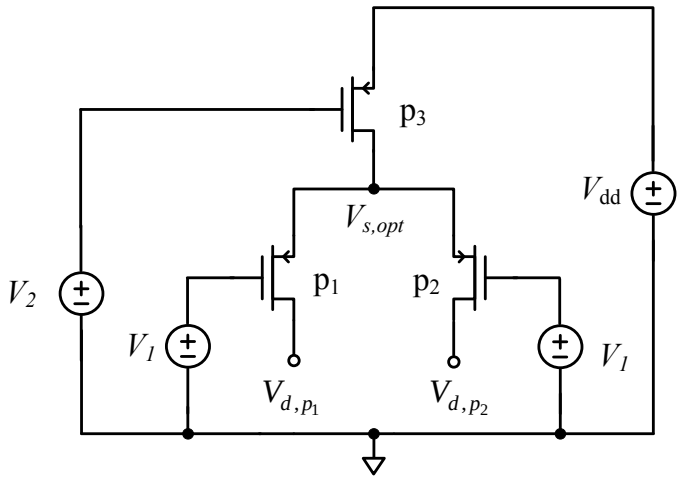


Figure 4.2: Circuit diagram for a basic differential pair.

not always be satisfied in the optimum node voltage search thus the results of these algorithms can be incorrect if a different minimum or maximum point within the range is chosen other than the desired minimum/ maximum.

4.3 Variability Analysis Using Surrogate Models

In this section the variability of the differential amplifier circuit of Figure 4.4 is analyzed. The aim here is to demonstrate how surrogate models can be used to determine the effects of process variations during the initial design phase. The external voltages and the widths and lengths of the transistors are provided in Table 4.1. The surrogate model-based simulation and SPICE simulations are compared in Table 4.2 for the nominal process, at the process corners and at temperature extremes. These process corners are nominal NMOS–nominal PMOS (NN), fast NMOS–fast PMOS (FF), slow NMOS–slow PMOS (SS), fast NMOS–slow PMOS (FS) and slow NMOS–fast PMOS (SF). Both the surrogate model-based variability analysis and SPICE simulations were done on an

1: **function** diff_amp_pmos $(V_{d-p_1,p_2}, w_{p_1}, w_{p_2}, w_{p_3}, \mathbf{PM}_{p_1}, \mathbf{PM}_{p_2}, \mathbf{PM}_{p_3}, \mathrm{SM}_{\mathrm{pmos}})$
2: **if** $(V_{d,p_1} \geq V_{d,p_2})$ **then**
3: $V_{s,\max} \leftarrow V_{d,p_1}$
4: **else**
5: $V_{s,\max} \leftarrow V_{d,p_2}$
6: **end if**
7: **for** $V_s \leftarrow V_{s,max}, V_{dd}$ **do**
8: $\mathbf{PM}_{-p_1,p_2,p_3}\left[V_{sb}, V_{sg}, V_{sd}\right] \leftarrow$ set_values $(V_{dd}, V_{d,p_1}, V_{d,p_2}, V_{g,p_1}, V_{g,p_2}, V_{g,p_3})$
9: $i_{d,p_1} \leftarrow w_{p_1} \times$ predict $(\mathbf{PM}_{-p_1}, \mathrm{SM}_{\mathrm{pmos}})$
10: $i_{d,p_2} \leftarrow w_{p_2} \times$ predict $(\mathbf{PM}_{-p_2}, \mathrm{SM}_{\mathrm{pmos}})$
11: $i_{d,p_3} \leftarrow w_{p_3} \times$ predict $(\mathbf{PM}_{-p_3}, \mathrm{SM}_{\mathrm{pmos}})$
12: **end for**
13: $(\Delta i_d, \mathrm{ind}_{\min,1}) \leftarrow$ min $(|i_{d,p_1} + i_{d,p_2} - i_{d,p_3}|)$
14: $V_{s,\mathrm{opt}} \leftarrow V_{s,\max} + (\mathrm{ind}_{\min,1} - 1)/100$
15: **for** $V_s \leftarrow (V_{s,\mathrm{opt}} - 0.1), (V_{s,\mathrm{opt}} + 0.1)$ **do**
16: $\mathbf{PM}_{-p_1,p_2,p_3}\left[V_{sb}, V_{sg}, V_{sd}\right] \leftarrow$ set_values $(V_{dd}, V_{d,p_1}, V_{d,p_2}, V_{g,p_1}, V_{g,p_2}, V_{g,p_3})$
17: $i_{dx,p_1} \leftarrow w_{p_1} \times$ predict $(\mathbf{PM}_{-p_1}, \mathrm{SM}_{\mathrm{pmos}})$
18: $i_{dx,p_2} \leftarrow w_{p_2} \times$ predict $(\mathbf{PM}_{-p_2}, \mathrm{SM}_{\mathrm{pmos}})$
19: $i_{dx,p_3} \leftarrow w_{p_3} \times$ predict $(\mathbf{PM}_{-p_3}, \mathrm{SM}_{\mathrm{pmos}})$
20: **end for**
21: $(\Delta i_{dx}, \mathrm{ind}_{\min,2}) \leftarrow$ min $(|i_{dx,p_1} + i_{dx,p_2} - i_{dx,p_3}|)$
22: $V_{s,\mathrm{opt}} \leftarrow V_{s,\mathrm{opt}} + (\mathrm{ind}_{\min,2} - 11)/100$
23: **end function**

Figure 4.3: Pseudo-code for the algorithm to yield the node voltage $V_{s,\mathrm{opt}}$ in Figure 4.2. The function set_values evaluates the voltages V_{sg}, V_{sd}, and V_{sb} for the devices p_1, p_2 and p_3 using the input quantities.

INTEL I7 Quad Core 2.66 GHz processor. A single drain current evaluation of one transistor using the surrogate models takes 0.5 ms and full circuit simulation using the search algorithms requires 1.4 s in MATLAB®. The SPICE simulations using Cadence Spectre® required 2 s. Note that if the surrogate models were incorporated in a circuit simulator no more than 5 iterations per cycle would typically be required so that each surrogate model-based simulation cycle would require approximately 2.5 ms. In addition, a full-scale Monte Carlo analysis including all statistical variations across the process

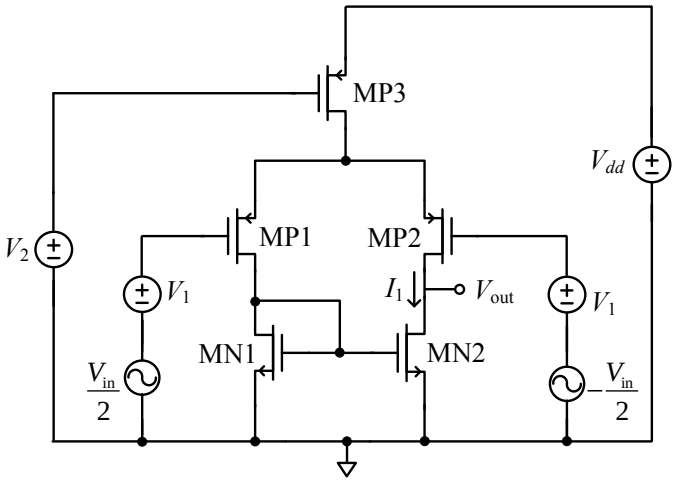


Figure 4.4: Actively-loaded PMOS-input differential amplifier. This circuit has the basic analog circuit building blocks: a current mirror, a cascode and a differential amplifier.

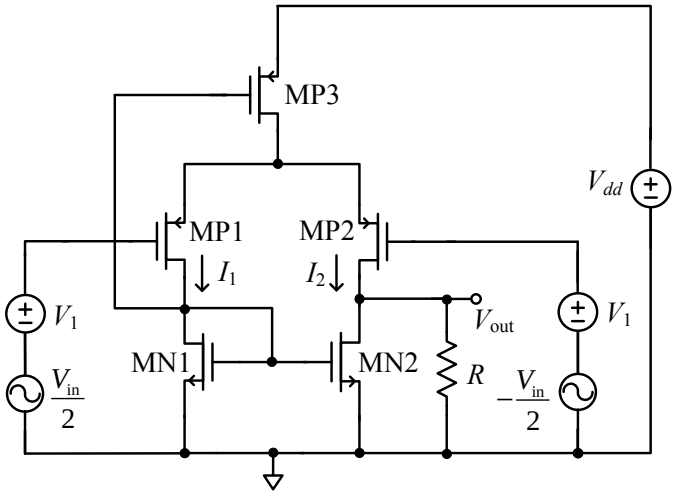


Figure 4.5: Another actively-loaded PMOS-input differential amplifier with the bias voltage for MP3 is supplied through a feedback structure.

corners was performed. Each cycle required 5.3 s. It can be seen that the surrogate model-based variability analysis is approximately four times faster than the Monte Carlo analysis. Monte Carlo analysis varies all process dependent parameters in the design kit,

thus yielding higher accuracy (RRSE $< 1\%$) than obtained with the surrogate models (RRSE $< 2\%$). Therefore, a small trade-off between the accuracy and analysis speed can be observed between surrogate model-based and Monte Carlo analysis methods.

Table 4.2 shows close agreement between the surrogate model-based analysis and SPICE results across the whole range with an average error of 3.5% with the maximum error of 6.7% recorded for the SS low temperature corner. The slight difference can be referred back to the minor deviation in the modeling phase as depicted in Figures 2.3(c) and 2.4(c).

Table 4.1: Circuit parameters for Figure 4.4 and corresponding values

Device	W/L (μm/μm)	Voltage name	Voltage Value
MN1	30/0.05	V_1	0.4 V
MN2	30/0.05	V_2	0.5 V
MP1	50/0.05	V_{dd}	1.2 V
MP2	50/0.05		
MP3	50/0.05		

The essential premise of variability analysis using surrogate modeling is that response evaluation is rapid enough that variability can be considered in the design phase. This is becoming increasingly important in sub 100 nm design where adjacent transistors can experience very different process parameters due to large manufacturing equipment tolerances and layout proximity effects as explained in Chapter 1. A direct consequence of these effects is the increased levels of within-die variations. Traditionally, within-die variations have been considered in smaller scales due to the spatial correlation between devices located close to each other. However, as the technology moves below 45 nm

technology, this assumption proves to be no longer valid and modeling each transistor in a circuit independently with its process parameters becomes necessary. The reduced-order models constructed here allow the designer to change the extent of process variations between devices. If the level of within-die variations for a particular device technology is small then the input vectors for the transistors can be made up with little variation in process parameters. Conversely, the variation may be increased to a higher degree in case within-die variations are widely observed in a particular device process.

Now the surrogate models will be used to examine the process and temperature related variability of the differential amplifier shown in Figure 4.5. In this case, the drain currents of the differential pair will not be equal because of the 500 Ω load resistance. The other circuit parameters are given in Table 4.1. Table 4.3 shows the results of the analysis. It can be seen again that the proposed surrogate model-based variability method yields relatively accurate results with respect to the results of SPICE simulations.

The same circuit has also been analyzed using neural network-based surrogate models. The results are tabulated in Table 4.4. Comparison of Tables 4.3 and 4.4 reveals that the performances of both reduced-order models here in terms of their accuracy are similar. It should be noted that this was expected since the true model employed to construct both of these reduced-order models is the same. Furthermore, their corresponding RRSE values are very close to each other indicating the fact that their responses should be close as well.

Table 4.2: Comparision of the surrogate model-based DC operating point analysis with the SPICE simulations for the circuit in Figure 4.4

CMOS Differential Amplifier (I_1 [mA], V_{out} [V])

Temp	Low T (-100°C)		Nominal T (27°C)		High T (100°C)	
Process	Surrogate Model	SPICE Simulation	Surrogate Model	SPICE Simulation	Surrogate Model	SPICE Simulation
NN	2.05/ 0.554	2.03/ 0.562	2.31/ 0.51	2.29/ 0.526	2.51/ 0.49	2.44/ 0.51
FF	2.46/ 0.545	2.58/ 0.547	2.72/ 0.504	2.81/ 0.512	2.95/ 0.483	2.96/ 0.494
FS	1.71/ 0.523	1.62/ 0.524	1.99/ 0.481	1.92/ 0.486	2.21/ 0.454	2.09/ 0.468
SS	1.72/ 0.559	1.61/ 0.57	1.96/ 0.514	1.89/ 0.535	2.18/ 0.493	2.05/ 0.52
SF	2.42/ 0.6	2.49/ 0.6	2.67/ 0.552	2.7/ 0.567	2.87/ 0.531	2.84/ 0.552

Table 4.3: Comparision of the surrogate model-based DC operating point analysis with the SPICE simulations for the circuit in Figure 4.5

CMOS Differential Amplifier (I_1 [mA] /I_2 [mA] /V_{out} [V])

Temp	Low T (−100°C)		Nominal T (27°C)		High T (100°C)	
Process	Surrogate Model	SPICE Simulation	Surrogate Model	SPICE Simulation	Surrogate Model	SPICE Simulation
NN	1.49/ 1.85/ 0.392	1.41/ 1.80/ 0.392	2.10/ 2.46/ 0.356	2.01/ 2.38/ 0.374	2.45/ 2.79/ 0.343	2.28/ 2.64/ 0.367
FF	1.92/ 2.32/ 0.416	2.02/ 2.42/ 0.423	2.57/ 2.95/ 0.384	2.61/ 2.99/ 0.398	2.95/ 3.30/ 0.369	2.88/ 3.25/ 0.387
FS	1.43/ 1.75/ 0.373	1.33/ 1.66/ 0.369	2.01/ 2.31/ 0.333	1.91/ 2.22/ 0.350	2.36/ 2.61/ 0.319	2.16/ 2.46/ 0.342
SS	1.19/ 1.57/ 0.343	0.99/ 1.36/ 0.355	1.77/ 2.09/ 0.330	1.59/ 1.94/ 0.348	2.10/ 2.39/ 0.323	1.85/ 2.19/ 0.344
SF	1.57/ 1.96/ 0.416	1.47/ 1.92/ 0.416	2.20/ 2.62/ 0.382	2.11/ 2.54/ 0.400	2.55/ 2.95/ 0.374	2.39/ 2.82/ 0.394

Table 4.4: Comparison of the Neural Network-Based DC Operating Point Analysis with the SPICE simulations for the circuit in Figure 4.5

CMOS Differential Amplifier (I_1 [mA] /I_2 [mA] /V_{out} [V])

Temp	Low T (-100°C)		Nominal T (27°C)		High T (100°C)	
Process	Neural Network	SPICE Simulation	Neural Network	SPICE Simulation	Neural Network	SPICE Simulation
NN	1.47/ 1.89/ 0.379	1.41/ 1.80/ 0.392	2.09/ 2.45/ 0.356	2.01/ 2.38/ 0.374	2.44/ 2.75/ 0.35	2.28/ 2.64/ 0.367
FF	1.86/ 2.34/ 0.405	2.02/ 2.42/ 0.423	2.58/ 2.95/ 0.386	2.61/ 2.99/ 0.398	2.98/ 3.31/ 0.373	2.88/ 3.25/ 0.387
FS	1.42/ 1.80/ 0.356	1.33/ 1.66/ 0.369	2.00/ 2.30/ 0.334	1.91/ 2.22/ 0.350	2.36/ 2.62/ 0.325	2.16/ 2.46/ 0.342
SS	1.18/ 1.56/ 0.348	0.99/ 1.36/ 0.355	1.73/ 2.05/ 0.330	1.59/ 1.94/ 0.348	2.05/ 2.34/ 0.317	1.85/ 2.19/ 0.344
SF	1.52/ 2.01/ 0.399	1.47/ 1.92/ 0.416	2.18/ 2.59/ 0.387	2.11/ 2.54/ 0.400	2.54/ 2.92/ 0.375	2.39/ 2.82/ 0.394

The variability of the circuit in Figure 4.5 has been characterized based on the idea that it is no longer sufficient to evaluate variability by considering all transistors to be in the same process conditions. Here, at a temperature T of 100°C, each of the five devices in the circuit are assigned to the five different process corners yielding a total of 3125 distinct combinations. The aim of this experiment is to evaluate the negative effects of within-die variations. The variability of I_2 is shown in Figure 4.6. In this graph the percentage variation of I_2 is

$$\eta = 100 \times (I_{2,i} - I_{2,\text{nom}}) / I_{2,\text{nom}}, \tag{4.1}$$

where $I_{2,\text{nom}}$ is the nominal value of I_2. The extreme values of I_2 obtained using conventional process corner analysis are also indicated. The graph reveals that the range between the maximum (3.67 mA) and minimum (1.94 mA) values can increase by 90% if within-die variations are considered. Examination indicates that the change in the maximum value stems from a slow MP1 and fast MP2, steering a certain percentage of I_1 to I_2. The reduction in the minimum value is due to slow NMOS transistors in conjunction with slow MP2 and MP3, and fast MP1. These results highlight the importance of extensive variability simulations by describing the transistors independently. That is, conventional variability analysis (which examines circuit performance with all transistors held at each of the process corners, plus the nominal process) significantly underestimates the variability experienced with within-die variations.

Monte Carlo analysis is also used in variability analysis [19]. With this technique individual BSIM process parameters are varied based on a certain distribution. This analysis can be performed using the surrogate modeling variability analysis tool. In Figure 4.7 the BSIM parameters `VTH0` and `XL` are chosen using Gaussian distributions

with the standard deviation set to 6% and 20% of their nominal values, respectively. All other input variables are held at their nominal values with $T = 100°\text{C}$. A total of 3600 input vector responses are evaluated. As expected, a Gaussian distribution of I_2 around its nominal value is obtained with the maximum and minimum values being less than the extremes detected previously. This lower variability results because in Monte Carlo analysis the same BSIM parameter set is used with devices of the same type (n- or p-channel).

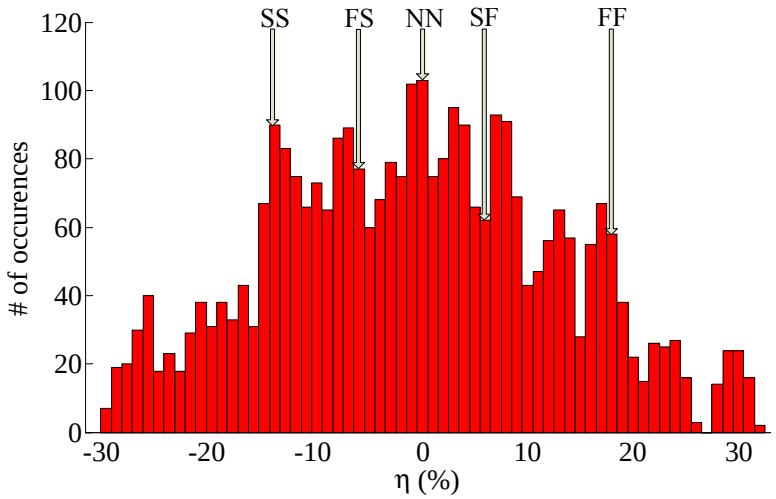


Figure 4.6: Histogram of I_2 in Figure 4.5 with respect to all combinations of transistors at the process corners with $T = 100°\text{C}$.

The variability of the voltage gain A_v of the differential amplifier must also be examined. Here

$$|A_v| \approx \frac{1}{2} \left(g_{m,\text{MP1}} + g_{m,\text{MP2}} \right) R, \tag{4.2}$$

where it is assumed that the output resistances of MP2 and MN2 are much higher than

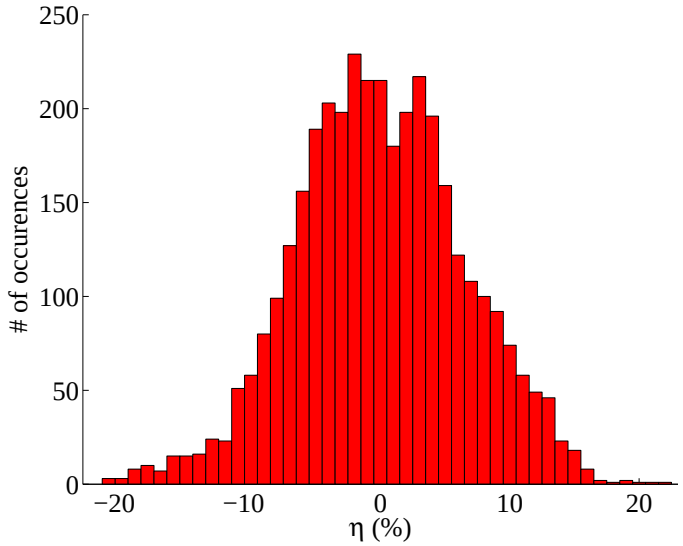


Figure 4.7: The Monte Carlo analysis of the circuit in Figure 4.5 using the surrogate model-based variability analysis tool. The standard deviation corresponds to the 6.4% of I_2 in nominal conditions.

R. The transconductances of MP1 and MP2 are

$$g_{m,\mathrm{MP1}} = \sqrt{2\mu_{\mathrm{eff,MP1}}\, C_{\mathrm{ox,MP1}}\, (W_{\mathrm{eff}}/L_{\mathrm{eff}})\, I_1} \tag{4.3}$$

$$g_{m,\mathrm{MP2}} = \sqrt{2\mu_{\mathrm{eff,MP2}}\, C_{\mathrm{ox,MP2}}\, (W_{\mathrm{eff}}/L_{\mathrm{eff}})\, I_2}, \tag{4.4}$$

respectively. The distribution of A_v is shown in Figure 4.8 with the horizontal axis being

$$\zeta = 100 \times (A_{v,i} - A_{v,\mathrm{nom}})\, /A_{v,\mathrm{nom}}, \tag{4.5}$$

the percentage variation of A_v with respect to the nominal value $A_{v,\mathrm{nom}}$. The increased variability with respect to process corner analysis is 7.5%. This degree of increased variability is less than was observed with the drain current (Figure 4.6). The reason for

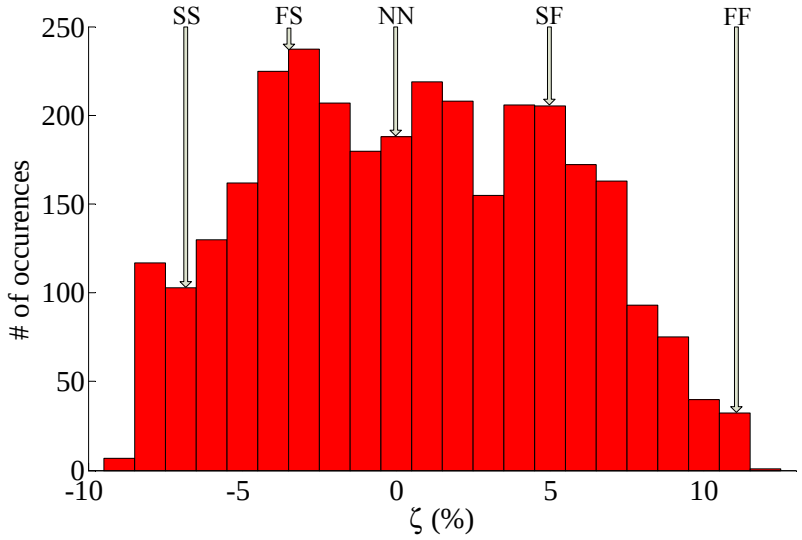


Figure 4.8: Histogram of A_v of the circuit in Figure 4.5 with respect to all combinations of transistors at the process corners with $T = 100°C$.

this can be traced back to (4.2). Since the average value of two transconductances is used in the gain calculation, locating the two transistors of the differential pair at opposite process corners has little impact on A_v.

4.4 A case study: Analysis of variability in the VTC of an XOR circuit

The produced I_{ds} models have been used to analyze the variability in the voltage transfer characteristics of the XOR circuit shown in Figure 4.9 [79]. The XOR circuit here is composed of two inverters and a transmission gate with the nominal channel width and length of each device in the XOR circuit provided in Table 4.5. The inverter comprising MN1 and MP1 is employed to find the complement of the voltage V_A. When V_A is high, the other inverter comprising MN2 and MP2 begins to be active and the output voltage

of the XOR circuit V_{out} becomes the inverse of V_A. Conversely, when V_A is low, then the transmission gate comprising MN3 and MP3 makes V_{out} equal to V_B. These relations can be summarized in the form of the truth table provided in Figure 4.10.

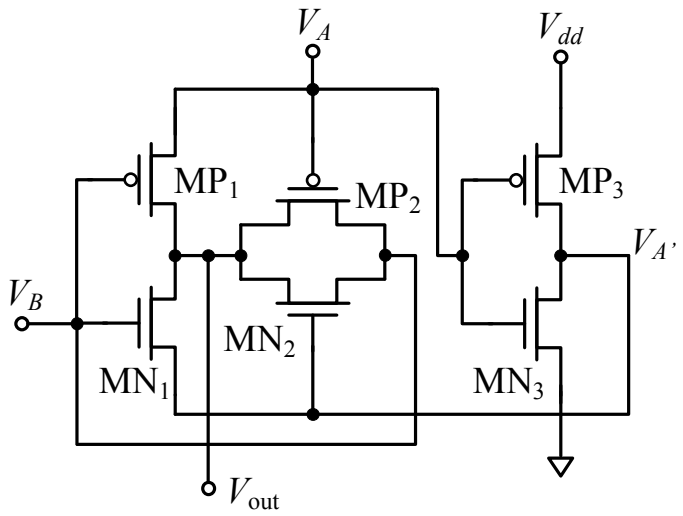


Figure 4.9: The circuit diagram for the XOR operation.

V_A	V_B	V_{out}
0 V	0 V	0 V
0 V	1.2 V	1.2 V
1.2 V	0 V	1.2 V
1.2 V	1.2 V	0 V

Figure 4.10: The truth table for the XOR operation.

Four cases characterize the voltage transfer response of a two-input XOR circuit.

Table 4.5: Device sizes of the circuit in Figure 4.9.

Device	W/L (μm/μm)	Device	W/L (μm/μm)
MN1	10/0.05	MP1	20/0.05
MN2	20/0.05	MP2	40/0.05
MN3	10/0.05	MP3	20/0.05

These are $V_A = 0$ V and V_B changes from 0 V to 1.2 V, $V_A = 1.2$ V and V_B changes from 0 V to 1.2 V, $V_B = 0$ V and V_A changes from 0 V to 1.2 V, and $V_B = 1.2$ V and V_A changes from 0 V to 1.2 V. The variability is captured by considering the fast-fast (FF) process at low temperature ($T = -100°$C) and the slow-slow (SS) process at high temperature ($T = 100°$C) in addition to the nominal process (NN) at room temperature $T = 27°$C.

In order to analyze the voltage transfer characteristics of an XOR circuit using the produced models, a behavioral description of the circuit functionality has to be developed in MATLAB environment. The approach taken here is to employ a two-stage exhaustive search for V_{out} using the given values of V_A and V_B. The coarse exhaustive search which is the first stage has ten times larger step sizes than the fine exhaustive search. In each step, the voltages (V_{bs}, V_{gs}, V_{ds}) of the input vectors corresponding to each device in the circuit are updated to find the correct V_{out} for the particular choices of V_A and V_B.

The graphs for all of the twelve different voltage transfer characteristics are shown in Figures 4.11 and 4.12. In each graph, the Kriging-based surrogate model results and neural network-based model results from the behavioral variability analysis are compared with the circuit SPICE simulation. Both the Kriging-based and the neural network-based models have $N_s = 3840$. N_n is chosen to be 100 for the neural network-based model. A good match is accomplished with the maximum RMSE of 32.6 mV and the average RMSE

of 15.9 mV for the Kriging-based models and the maximum RMSE of 34.5 mV and the average RMSE of 15.1 mV for the neural network-based models. Also, the time elapsed for each numerical experiment to complete has been compared for both models. Given that the behavioral description evaluates the device currents one at a time, Kriging-based models take 0.782 s for a single V_{out} evaluation, whereas this duration rises to 5.427 s for the neural network-based model. For a different behavioral description in which the currents of all devices are evaluated in batch, a single V_{out} evaluation is completed in 0.979 s using Kriging-based models whereas neural network-based models find the same quantity in 3.625 s. These outcomes corroborate the previous conclusions that for small circuits Kriging-based models outperform neural network-based models which tend to become more advantageous in larger circuits given that the input vectors are evaluated in batch.

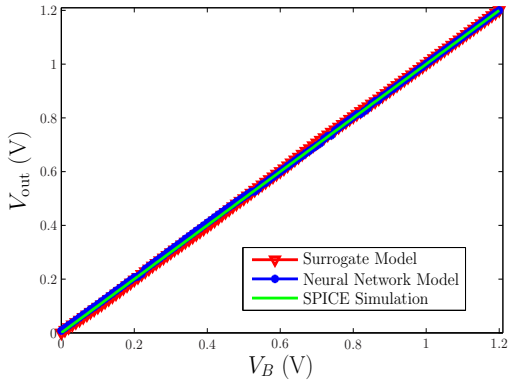
It is not practical to use SPICE transistor-level simulations to model within-die variations because all transistors are assumed to be at the same process corner having the same process parameters. Here the behavioral models are used to examine within-die variations. Specifically, the process parameters of the six transistors in the XOR circuit were randomly chosen to be one of the fast-fast (FF), slow-slow (SS), fast-slow (FS), and slow-fast (SF) process corners. The resulting curves (see Figures 4.13(a) and 4.13(b)) have been compared with the outcomes of the regular process corner analysis which assumes each transistor to be in the same corner (see Figures 4.14(a) and 4.14(b)). It can be seen that the maximum variation between the voltage transfer curves estimated using the process corner analysis has been extended once the within-die variations are taken into account. More quantitatively, the RMS variation has increased by 11.5% in the case of Figure 4.13(a) and 7% in the case of 4.13(b). Furthermore, this increase in the variability affects also performance metrics of the XOR circuit such as the noise immunity.

The noise immunity is characterized by the noise margins for the high and low inputs of a digital circuit. Traditionally, the noise margins are the points in a voltage transfer curve where the slope becomes -1 [79]. Increased RMS variation due to within-die variations can cause a decrease in the worst-case noise margin of the analyzed XOR circuit by 30 mV. This is a clear indication that even though the channel widths are chosen to be large (see Table 4.5) which is a common practice to avoid circuit variability, within-die variations can make a sizable difference in the design outcomes.

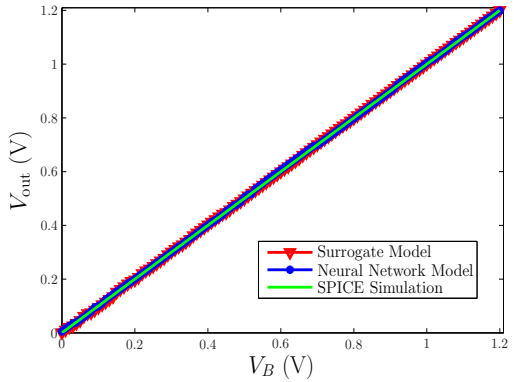
In Table 4.6, the impact of the model accuracy on the behavioral analysis of the XOR circuit is characterized. For this purpose Kriging-based surrogate models of various N_s are used to find out the voltage transfer characteristics. Then, the resulting curves are compared with the SPICE circuit simulations. It can be observed in Table 4.6, that generally, the analysis accuracy drops with smaller N_s although the degradation is minor. This is an important result since one can significantly reduce model complexity by using less sample vectors. For Kriging-based surrogate models, this also means shorter model generation and evaluation times as well as smaller sizes.

Table 4.6: Comparison of the accuracy of different model with varying N_s.

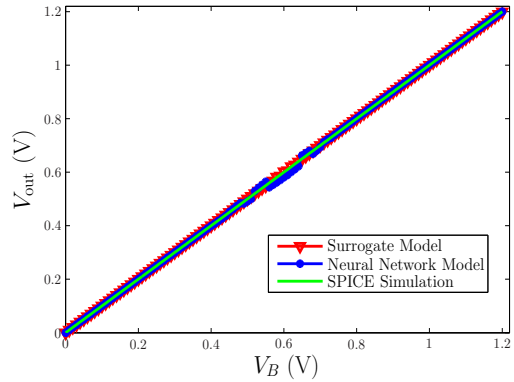
Accuracy (RMSE (mV)) of voltage transfer characteristics analysis	N_s					
	640	1280	1920	2560	3200	3840
$V_A = 1.2$ V and $0 \leq V_B \leq 1.2$ V	13.5	11.8	8.6	10.4	7.2	5.5
$V_B = 1.2$ V and $0 \leq V_A \leq 1.2$ V	27.3	21.8	21.8	25.1	24.5	24.2



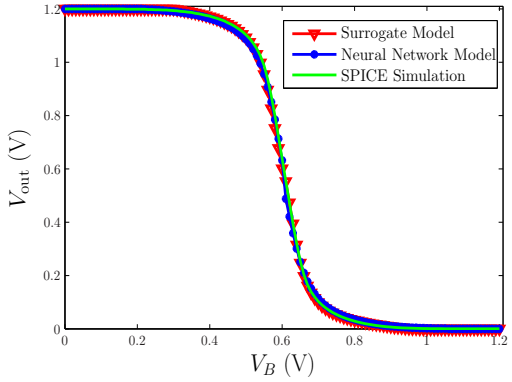
(a): SS, $T = 100°\text{C}$, $V_A = 0$ V



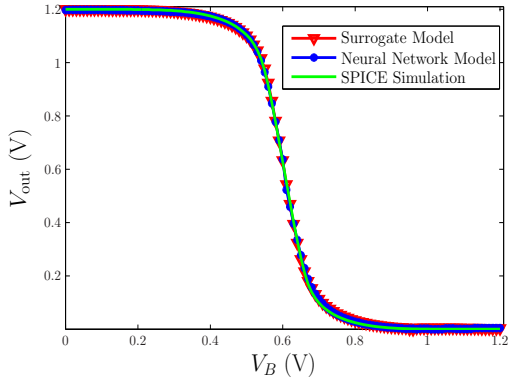
(b): NN, $T = 27°\text{C}$, $V_A = 0$ V



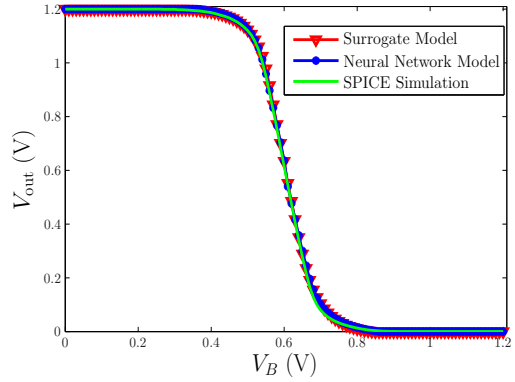
(c): FF, $T = -100°\text{C}$, $V_A = 0$V



(d): SS, $T = 100°\text{C}$, $V_A = 1.2$ V

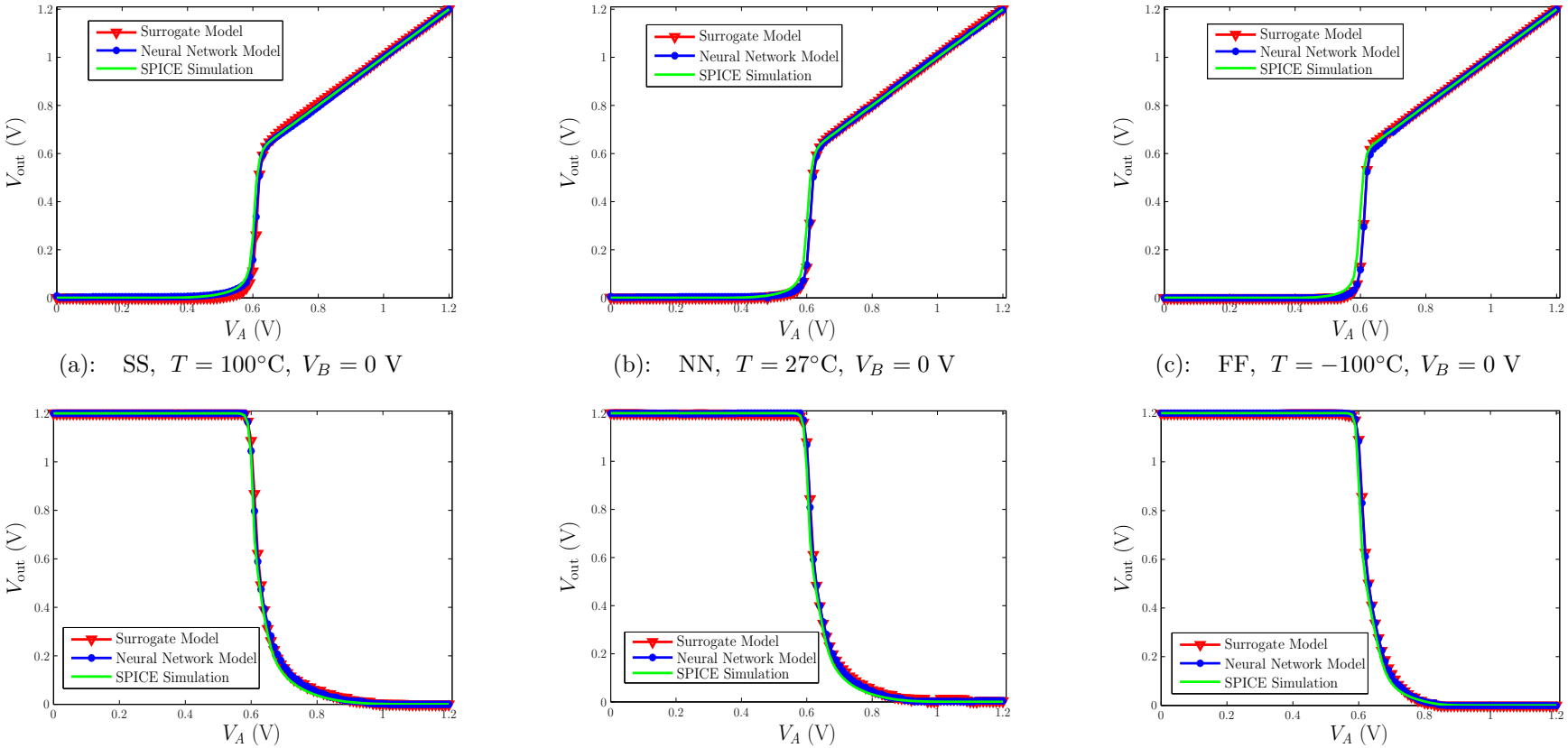


(e): NN, $T = 27°\text{C}$, $V_A = 1.2$ V



(f): FF, $T = -100°\text{C}$, $V_A = 1.2$ V

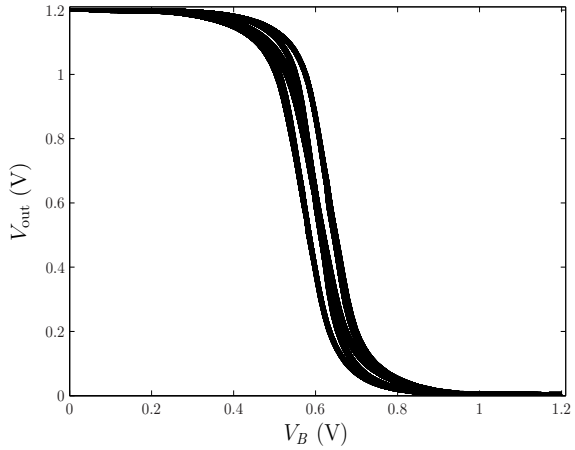
Figure 4.11: Comparison of the different voltage transfer characteristics curves, produced through Kriging-based I_{ds} surrogate models, neural network-based I_{ds} models and SPICE simulations.



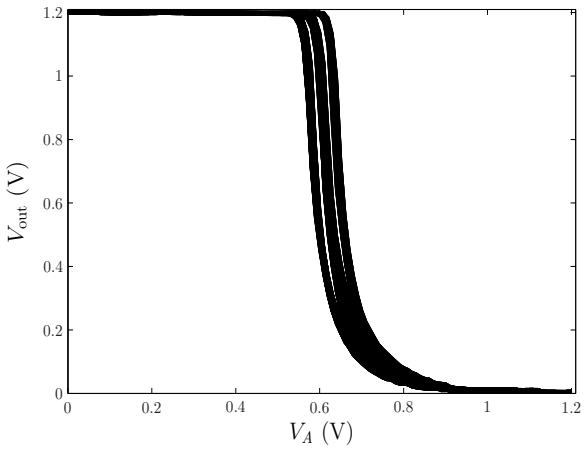
(a): SS, $T = 100°C$, $V_B = 0$ V (b): NN, $T = 27°C$, $V_B = 0$ V (c): FF, $T = -100°C$, $V_B = 0$ V

(d): SS, $T = 100°C$, $V_B = 1.2$ V (e): NN, $T = 27°C$, $V_B = 1.2$ V (f): FF, $T = -100°C$, $V_B = 1.2$ V

Figure 4.12: Comparison of the different voltage transfer characteristics curves, produced through Kriging-based I_{ds} surrogate models, neural network-based I_{ds} models and SPICE simulations.

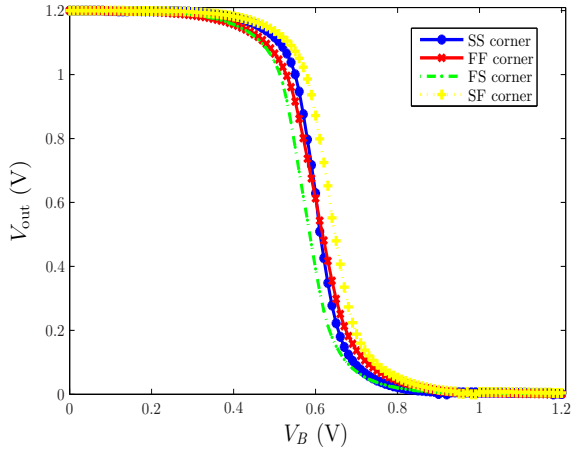


(a): $T = 27°\text{C},\ V_A = 1.2$ V

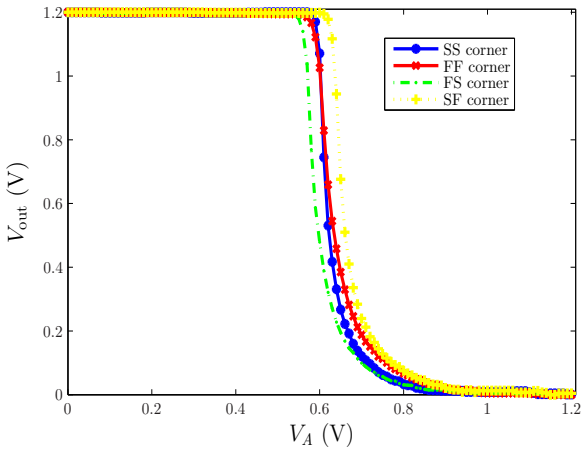


(b): $T = 27°\text{C},\ V_B = 1.2$ V

Figure 4.13: Within-die variability of the voltage transfer characteristics of the XOR circuit using the Kriging-based I_{ds} surrogate models.



(a): $T = 27°\text{C},\ V_A = 1.2$ V



(b): $T = 27°\text{C},\ V_B = 1.2$ V

Figure 4.14: Voltage transfer characteristics of the XOR circuit at the process corners using the Kriging-based I_{ds} surrogate models.

4.5 Mismatch Analysis of NMOS and PMOS transistor pairs

A final example of the variability analysis using reduced-order models is the mismatch analysis. The mismatch of two transistors can create large variations of DC currents in a current mirror. It can also lead to undesired DC offset voltages in differential amplifiers. Generally, the mismatch variations can be reduced significantly by choosing the width and lengths of the transistors in a circuit larger than the minimum sizes so that the percentage of variations is decreased. However, with the nanoscale transistors, other factors such as the process parameters included in the vector describing each transistor (such as $\mathbf{NM}_{n_1}$) may have large variations around their nominal values. This will lead to substantial differences in the design quantities such as V_{th} or I_{ds}, thus causing the mismatch between transistors in a current mirror or a differential amplifier.

One particular issue in mismatch analysis that has to be characterized is to identify the process parameters that lead to the highest variations in the desired quantity. A traditional process corner analysis changes multiple process parameters at the same time thus it is not possible to visualize the individual contributions of important process parameters. Conversely, in Monte Carlo analysis, a mismatch parameter is introduced that changes the device parameters with respect to each other. However, this does not provide a direct understanding of how big the variation in the process parameters should be in order to generate a given amount of mismatch in the output quantity. Reduced-order modeling approach described in this dissertation can be used to find a solution to this problem by making changes in the input vector elements, where each vector is corresponding to a separate transistor.

The study in this section considers two NMOS and PMOS transistors that have the

exact same biasing configuration. V_{gs}, V_{ds}, and V_{bs} for NMOS devices (V_{sg}, V_{sd}, and V_{sb} for PMOS devices) are assumed to be 0.6 V, 0.8 V to 0 V, respectively. The ages of all devices are taken to be zero. Both transistor pairs are assumed to be in the nominal process corners initially. Then, variations of $\pm 2.5\%$ and $\pm 5\%$ in five of the process parameter (t_{ox}, $V_{\text{th},0}$, L_{eff}, μ_0, and $R_{ds,0}$) as well as $\pm 7.5^{\circ}$C and $\pm 15^{\circ}$C variations in temperature (T) have been presented. In particular, one NMOS (PMOS) transistor is set to have the change $+2.5\%$ whereas the other one is made subject to the change -2.5%. These changes are made one at a time so that the variation amount is a direct consequence of the change made in the process parameter or the temperature.

I_{ds} values of both transistor pairs are recorded for both levels of variations in all six parameters using the Kriging-based surrogate models for I_{ds}. Results are tabulated in Tables 4.7 and 4.8. $\min\left(\Delta I_{ds}/I_{ds,\text{nom}}\right)$ and $\max\left(\Delta I_{ds}/I_{ds,\text{nom}}\right)$ indicate the changes in I_{ds} due to a reduction and increase of 2.5% (or 5%) in any of the six parameters, respectively. Total $\Delta I_{ds}/I_{ds,\text{nom}}$ represents the sum of the absolute values of these two changes.

The results show that the largest contribution to the mismatch variation comes from L_{eff} and $V_{\text{th},0}$. These two are followed by the μ_0 change. Thus it should be inferred the line edge roughness (LER), random dopant fluctuation (RDF), and shallow trench isolation (STI) that significantly affect the values of L_{eff}, $V_{\text{th},0}$, and μ_0 should carefully be modeled. Noting that these phenomena mostly originate randomly in nanoscale device technologies, reduced-order modeling proves to be a very suitable and useful tool for visualizing mismatch variations in deep submicron analog circuits.

As a concluding remark to this section, a different aspect of the mismatch analysis should be discussed. It is not unusual to observe that different process parameters present correlation with each other. This can easily be modeled within the reduced-order modeling scheme described here. Once the exact nature of the correlation between the two

Table 4.7: Analysis of I_{ds} variation of two NMOS transistors due to mismatches in process parameters and temperature

Mismatch Analysis				
Parameter	Mismatch Range	min $\frac{\Delta I_{ds}}{I_{ds,\text{nom}}}$	max $\frac{\Delta I_{ds}}{I_{ds,\text{nom}}}$	total $\Delta I_{ds}/I_{ds,\text{nom}}$
ΔT	$\pm 7.5°C$	-1.33%	1.31%	2.64%
	$\pm 15°C$	-2.68%	2.60%	5.28%
Δt_{ox}	$\pm 2.5\%$	-0.79%	0.7%	1.49%
	$\pm 5\%$	-1.66%	1.32%	2.98%
ΔL_{eff}	$\pm 2.5\%$	-3.47%	4.13%	7.6%
	$\pm 5\%$	-6.43%	9.12%	15.55%
$\Delta V_{\text{th},0}$	$\pm 2.5\%$	-3.62%	3.69%	7.31%
	$\pm 5\%$	-7.15%	7.45%	14.6%
$\Delta \mu_0$	$\pm 2.5\%$	-1.90%	1.82%	3.72%
	$\pm 5\%$	-3.87%	3.56%	7.43%
$\Delta R_{ds,0}$	$\pm 2.5\%$	-0.09%	0.08%	0.17%
	$\pm 5\%$	-0.17%	0.18%	0.35%

parameters (e.g. μ_0 and $V_{\text{th},0}$) is understood, one of the parameters can be expressed in terms of the other one using the correlation expression. Subsequently, the input vectors representing the transistors can be updated with this new expression instead of the independent parameter. The variability analysis can continue in the same way as described throughout the chapter, then.

4.6 Summary

This paper demonstrated that compact behavioral response surfaces captured as a surrogate model can be efficiently used in circuit variability analysis. As gate length scaling

Table 4.8: Analysis of I_{ds} variation of two PMOS transistors due to mismatches in process parameters and temperature

Mismatch Analysis

Parameter	Mismatch Range	min $\frac{\Delta I_{ds}}{I_{ds,\text{nom}}}$	max $\frac{\Delta I_{ds}}{I_{ds,\text{nom}}}$	total $\Delta I_{ds}/I_{ds,\text{nom}}$
ΔT	$\pm 7.5°C$	-1.27%	1.27%	2.54%
	$\pm 15°C$	-2.55%	2.57%	5.12%
Δt_{ox}	$\pm 2.5\%$	-0.66%	0.72%	1.38%
	$\pm 5\%$	-1.38%	1.11%	2.49%
ΔL_{eff}	$\pm 2.5\%$	-3.79%	4.05%	7.84%
	$\pm 5\%$	-7.35%	8.41%	15.76%
$\Delta V_{\text{th},0}$	$\pm 2.5\%$	-3.57%	3.60%	7.17%
	$\pm 5\%$	-7.09%	7.24%	14.33%
$\Delta \mu_0$	$\pm 2.5\%$	-0.92%	0.91%	1.83%
	$\pm 5\%$	-1.83%	1.80%	3.63%
$\Delta R_{ds,0}$	$\pm 2.5\%$	-0.50%	0.50%	1.00%
	$\pm 5\%$	-0.99%	0.98%	1.97%

continues beyond 65 nm, variation of transistor parameters will increase complicating the circuit design as variability must be considered during the design process. Other variability simulation methods can determine the amount of variation but it is much more difficult for the designer to make a clear assessment of the individual contributions of each device. The techniques described here assign separate input vectors that characterize process variables, temperature and device age individually in DC circuit analysis. During the design process, the designer can identify possible problems due to process variations and create solutions to remedy them. This essentially merges design for the nominal process with within-die process-corner analysis. Additionally the techniques

presented are flexible and can be extended beyond DC circuit analysis. Parameter distribution profiles can be other than that considered here including the use of mismatch distributions, if known, of the parameters of adjacent transistors.

Furthermore, using a behavioral description, the variability in the voltage transfer characteristics of an XOR circuit is investigated. Results reveal that that nonlinear reduced-order modeling based approach successfully presents the possible effects of within-die variations that cannot be captured by regular process corner analysis. Kriging-based surrogate models and neural network-based models performed equally well in terms of accuracy. Consequently, this study bears a strong hope that reduced-order modeling techniques can be used to characterize IC variability with sufficient accuracy.

Chapter 5

Circuit Reliability Analysis using Reduced-Order Models

5.1 Introduction

The reliability of integrated circuits has become a focus of concern due to extreme scaling of device sizes. The time-based degradation of transistors, also called aging, significantly changes certain device characteristics that affect the performance of critical circuit parameters. Nanoscale circuits are prone to appreciable degradation and the variability of process parameters due to aging is a substantial percentage of their nominal values over time.

Transistor aging reduces the reliability of both digital and the analog circuits [80], [81]. Recent literature revealed that the aging of transistors is not just a result of bias voltages but also a strong function of temperature as well as process parameters of individual devices [14]. Each transistor undergoes a different amount of stress so the aging rate is not the same for all devices. This provides a strong reason to formulate simulation

methodologies for reliability analysis of circuits. It should be noted that process variations must be included within the framework developed because they can have a significant impact on the aging characteristics. Simultaneous analysis for variability and reliability of circuits is critical for successful design in advanced design technologies.

This chapter demonstrates that the surrogate modeling techniques which were developed to perform circuit variability analysis in the previous chapter can be extended such that an integrated variability and reliability analysis tool is realized. This tool is used in this paper to identify how the process parameters of individual transistors can affect the degradation of circuit performance due to aging. The analysis is used to create design methods that combat the adverse effects of time-based degradation yielding design for reliability guidelines. In particular, biasing circuits of analog building blocks are shown to impact aging resilience.

In Subsections 5.2.1 and 5.2.3, reliability degradation processes are theoretically formulated and then, earlier reliability simulation frameworks are reviwed in Subsection 5.2.4. A surrogate model-based reliability simulation methodology is described in Subsection 5.2.5. Subsection 5.3.1 demonstrates the variability and reliability analysis on a differential amplifier. Finally, design for reliability approaches are explained in Subsection 5.3.2.

5.2 Device Reliability

The reliability of integrated circuits derives from time-based degradations of certain electrical characteristics of transistors. Transistor scaling accelerates aging because the formation of traps due to high electrical fields across and along the gate dielectric is increased as oxide thickness and effective channel length continue to shrink as the technol-

ogy node advances. These effects have been experimentally studied but their generation mechanisms are not fully explained. However, several studies have provided a physical background as well as analytical compact formulas for the quantification of the resulting effects [14], [82]. In this section, theoretical knowledge about the main reliability concerns will be presented in conjunction with the compact analytical aging formulas that are used in this study.

5.2.1 Negative Bias Temperature Instability (NBTI)

The NBTI is significant at elevated temperatures when a negative bias ($V_{gs} < 0$) is applied to transistors and affects mostly p-channel devices. The origin of this phenomenon is breakdown of the Si-H bonds formed during passivation at the interface of the crystalline device channel and the amorphous oxide layer [82]. The bonds break apart after being subject to a large negative bias of long duration especially at elevated temperatures. Also, as the oxide thickness becomes thinner accommodating short channel lengths, the inversion layer holes tunnel more easily through the dielectric [82]. These holes contribute to the deconstruction of the Si-H bonds yielding dangling Si bonds at the channel-dielectric interface and H or H_2 which essentially diffuse through the dielectric back to the gate. An analytic framework called the reaction-diffusion mechanism is developed to find the number of dangling bonds which effectively act as interface traps [14], [82]. The number of traps generated due to NBTI determines the threshold voltage (V_{th}) increase. In addition to the V_{th} increase the traps also adversely affect the effective mobility of the channel carriers (μ_{eff}). So, NBTI manifests itself as an increase in the device V_{th} and a drop in μ_{eff} over time.

NBTI has a distinguishing property compared to other sources of aging: switching

activity on the NBTI-affected device can change the level of NBTI-induced degradation. When a PMOS transistor is off, i.e. the gate voltage becomes V_{dd}, the generation of traps ceases since no new Si-H bonds are broken. Additionally, some of the hydrogen species stop diffusing to the gate and instead passivate damaged Si-H bonds. Thus some of the traps formed during the stress phase disappear in the recovery phase. This phenomenon is called dynamic NBTI and should be taken into account for digital circuits. However, in analog circuits, constant application of DC voltages result in static NBTI where no recovery phase occurs. Therefore, in this study, static NBTI is considered for the degradation of the PMOS transistors in analog circuits.

Using the reaction-diffusion framework analytical expressions for the V_{th} increase and μ_{eff} drop of p-channel devices have been developed [14]:

$$(\Delta V_{\text{th}})_{\text{NBTI}} = \left[\left(\frac{q_{\text{e}} t_{\text{ox}}}{\epsilon_{\text{ox}}}\right)^3 \exp\left(\frac{2\left(V_{gs} - V_{\text{th}}\right)}{E_{01} t_{\text{ox}}}\right)\right]^{2n_{\text{NBTI}}} \times \left\{\left[K_1^2 C_{\text{ox}}\left(V_{gs} - V_{\text{th}}\right)\right]^2 \left[\frac{\exp\left(\frac{-E_{\text{a}}}{k_{\text{B}} T}\right)}{T_0}\right] t\right\}^{n_{\text{NBTI}}} \tag{5.1}$$

$$(\Delta \mu_{\text{eff}})_{\text{NBTI}} = \frac{\mu_{\text{eff},0}}{\left(1 + \alpha_{\text{NBTI}} \frac{C_{\text{ox}}}{q_{\text{e}}} \Delta V_{\text{th}}\right)^{m_{\text{NBTI}}}} \quad . \tag{5.2}$$

The expressions in (5.1) and (5.2) were developed from experiment and depend on the process parameters and the particular terminal voltage configuration of the p-channel device under test. Here, q_{e} stands for unit electronic charge, ϵ_{ox} for the dielectric constant of the oxide layer, C_{ox} for the oxide capacitance, k_{B} for the Boltzmann constant, and E_{A} for the Arrhenius activation energy. Other parameters K_1, E_{01} and T_0 are introduced for 65 nm technology. These analytic compact formulas, where n_{NBTI}, α_{NBTI} and m_{NBTI} are equal to 0.16, 5, and 1.6 respectively, have been verified through extensive measurements

of test circuits in 65 nm technology [14].

In this study, (5.1) and (5.2) must be embedded in the analytical expression for the transistor drain current. In order to do this, certain variables should be replaced with their counterparts in the BSIM model terminology. Here the expression $(V_{gs} - V_{\text{th}})$ is changed to $V_{gst,\text{eff}}$ for this purpose. Once the degradation levels are calculated during the analysis steps, all related electrical quantities including $V_{gst,\text{eff}}$ will be updated.

5.2.2 Positive Bias Temperature Instability (PBTI)

PBTI in NMOS is the counterpart of the NBTI effect in PMOS devices. As the name suggests, the onset of PBTI depends on the application of a positive bias to the gate of an NMOS at higher temperatures for long time. This effect becomes more prominent with the introduction of the high-κ dielectric with metal gate transistors and it increases the V_{th} of the device under stress. Experimental studies suggest that the already existing electron traps in the high-κ dielectric layer lead to the acceleration of the PBTI degradation at higher biasing voltages [83]. A secondary component of PBTI at lower biasing voltages requires temperature activation for the increased PBTI degradation. Although some empirical models have been established as in [83], the physical origins of PBTI are still not very clear. Since the models developed here are based on a technology which does not involve high-κ dielectric with metal gate devices, PBTI is not incorporated into the reliability analysis presented.

5.2.3 Hot Carrier Injection (HCI)

As the channel length shrinks, the lateral electrical field increases thereby yielding high-energy charge carriers along the channel. When these carriers acquire sufficient energy

they are injected into the gate oxide where they create interface traps at or near the channel-dielectric boundary [17]. HCI occurs mostly near the drain end of the channel since device electrons, the charge carriers for an NMOS (correspondingly holes for PMOS) transistors, travel most of the length of the channel before having the necessary energy for the injection process. The effect of HCI is to increase V_{th} and reduce μ_{eff}. HCI is more significant with n-channel devices compared to their p-channel counterparts.

Two basic differences can be noted between HCI and NBTI. Firstly, unlike HCI, in general NBTI occurs homogeneously along the channel-oxide layer interface. Secondly, the HCI problem grows with shorter gate length since the lateral electric field, which is the main cause of HCI, is escalated with channel length reduction. Conversely, NBTI mechanisms are not related to the gate-length scaling thus NBTI effects are mostly independent of the device channel length [14].

HCI is modeled by the same reaction-diffusion framework as NBTI where the recovery time has been shown to be negligible compared to the stress time. The V_{th} increase and the μ_{eff} drop have been quantified as follows [14]:

$$(\Delta V_{\text{th}})_{\text{HCI}} = \frac{q_e}{C_{\text{ox}}} K_2 \sqrt{C_{\text{ox}}(V_{gs} - V_{\text{th}})} \exp\left[\frac{(V_{gs} - V_{\text{th}})}{t_{\text{ox}} E_{02}}\right] \times \exp\left(-\frac{\phi_{\text{it}} \ell}{q_{\text{e}} \lambda \left(V_{ds} - V_{d,\text{sat}}\right)}\right) t^{n_{\text{HCI}}} \quad (5.3)$$

$$(\Delta \mu_{\text{eff}})_{\text{HCI}} = \frac{\mu_{\text{eff},0}}{\left(1 + \alpha_{\text{HCI}} \frac{C_{\text{ox}}}{q_{\text{e}}} \Delta V_{\text{th}}\right)^{m_{\text{HCI}}}} \quad . \quad (5.4)$$

In (5.3) and (5.4), ℓ is the length of the pinch-off region, ϕ_{it} is the minimum impact ionization energy for the Si-SiO_2 interface, λ is the mean free path of a charge carrier, and $V_{d,\text{sat}}$ is the saturation value of the drain-source potential. K_2 and E_{02} are technology dependent parameters specifically found for 65 nm devices. A major change in (5.3)

compared to (5.1) is that $n_{\mathrm{HCI}} = 0.45$ and $n_{\mathrm{NBTI}} = 0.16$, whereas α_{HCI} and m_{HCI} have the same value as their counterparts for NBTI. Again, (5.3) and (5.4) are experimentally developed expressions [14]. To embed these equations in the BSIM model equations, $V_{d,\mathrm{sat}}$ is replaced by the term $V_{ds,\mathrm{eff}}$ and ℓ is approximated by the BSIM parameter `litl`.

5.2.4 Previous Reliability Simulation Framework

Initial circuit reliability simulation tools targeted estimating the circuit degradation due to hot carrier injection and computing the failure possibility due to electromigration or oxide breakdown [84], [85]. As the scaling continued, NBTI effects become more prominent so the scope of the developed analysis techniques increased commensurately. These circuit reliability simulation techniques were compared in [86] where two main categories were identified. These were the SPICE-based and behavioral reliability simulation methods. SPICE-based reliability simulators separate the design and reliability analyses. One example of this approach consists of simulating the circuit to be analyzed using nominal parameters followed by circuit-failure analysis where the transistor degradation models are applied to the nominal simulation results [87], [16]. In the analog domain this technique was used in the reliability analysis of an analog-to-digital converter [88]. Although this approach successfully provides the level of degradation over time, the effects of process variations are difficult to consider since each process corner requires a separate reliability simulation. A similar method is the Monte Carlo-based reliability analysis as given in [89]. In this work, samples of the analyzed circuit are generated using the Monte Carlo simulation. Subsequently, transient SPICE simulations are performed on these samples. The resulting voltage waveforms are then put in a degradation model to produce the degraded circuit netlist. The problem with this method is the computational

cost associated with the Monte Carlo simulations which also makes it hard to include novel process variation data. Later the authors proposed to perform statistical screening on the given circuit to determine the important circuit and process parameters, create samples, and then apply the reliability simulation technique. This step is followed by response surface modeling to interpolate the final samples [90]. Authors also advocated to include time-dependent dielectric soft-breakdown which can be modeled by probabilistic degradation models (see [91]) within their framework [92]. Even though the methodology is accurate, it aims to detect local variations in the vicinity of the initial circuit design including process parameters and voltages. Thus it provides less feedback to the designer about the behavior of the circuit in response to extreme variations such as the within-die variations. In addition, since each circuit modeled by different parameters, analysis consistency cannot be maintained.

The second category, the behavioral reliability simulation framework, consists of the methods that try to unify electrical and aging simulations, i.e. no separate simulator is needed to determine the extent of the time-based degradation. This category can be further divided into two subgroups which are the behavioral reliability simulation at the circuit and at the transistor levels. At the circuit level the aim is to visualize the change in design parameters over time through the construction of a behavioral model for the given circuit. In [86] this goal was achieved by creating VHDL-AMS models of desired circuits combined with the aging expressions, which may be less accurate with the presence of large process variations. The other subcategory, transistor-level behavioral methods, seem to be more promising in that regard since they evaluate the effects of aging behaviorally on the basis of the changes in the transistor electrical parameters. In [93] unaged NMOS and PMOS devices are degraded in time and at different time points of the degraded waveform BSIM parameters `U0` and `VTH0` are modified to fit the given

data. Subsequently the distributions of these parameters are calculated with respect to the aging time and their expected values at that time point are used in the simulation of a differential amplifier. This approach integrates the aging analysis into circuit simulation but the effects of other process parameters such as the oxide thickness or temperature have not been accounted for.

The aim of the sections following is to use the transistor-level variability analysis based on surrogate modeling developed in Chapter 4 for time-based degradation analysis. By including aging effects during electrical simulation realized with the surrogate models, a complete integration of variability and reliability analyses can be achieved enabling simultaneous electrical design and design for variability and reliability.

5.2.5 Reliability Simulation using Surrogate Models

In Chapter 4, simulation techniques for variability analysis using surrogate models for drain current were introduced. Each device was described by 11 parameters which were the temperature T, electrical gate equivalent oxide thickness t_{ox} (`toxe`), channel doping parameter N_{ch} (`NDEP`), intrinsic mobility μ_0 (`U0`), the variation due to mask/etch processes of effective gate length ΔL_{eff} (`XL`), intrinsic threshold voltage at zero bulk-source potential $V_{\mathrm{th},0}$ (`VTH0`), the zero-bias lightly doped drain-source resistance $R_{ds,0}$ (`RDSW`), the absolute values of the gate-source voltage ($|V_{gs}|$), drain-source voltage ($|V_{ds}|$), bulk-source voltage ($|V_{bs}|$) and the transistor age t_{age}. Mathematical expressions that account for time-based degradation should be incorporated to the drain current equations for which the surrogate models will be constructed. Thus the process can be summarized as in Figure 5.1.

This framework allows the integration of variability analysis with reliability simulation. For any given t_{age}, the degradation in V_{th} and μ_{eff} can be calculated for all of

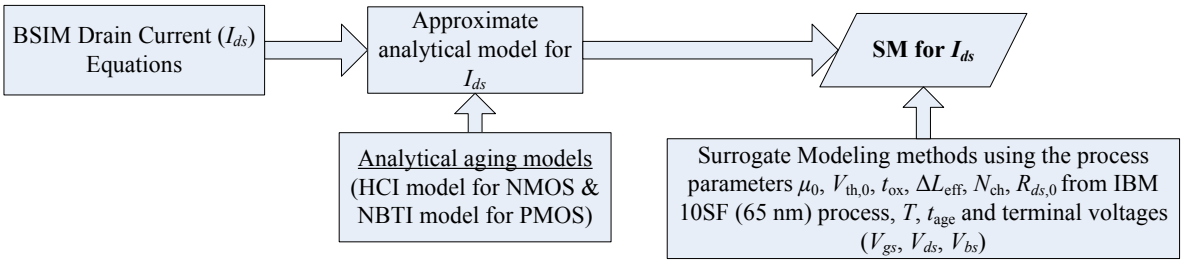


Figure 5.1: Flowchart describing the steps in the construction of drain current surrogate models. This scheme captures the effect of process variations and aging on the drain current of NMOS and PMOS devices.

the transistors in the circuit. Subsequently, the device models are updated during DC analysis until convergence of the intermediate node voltages is achieved. Hence the need in previous studies [87] for an initial DC simulation to find the operating conditions —to which the changes in ΔV_{th} and μ_{eff} are applied— can now be eliminated. A secondary advantage of the scheme developed here stems from the circuit analysis viewpoint. Assuming a static ΔV_{th} for a certain target age, t_{age}, will only be accurate if the transistor V_{gs} remains at the same level for all time. However, if the gate voltage depends on a different node voltage in the circuit, then due to the effects of aging, the gate voltage will not retain its original value thus the degradation rates should be recalculated. Additionally, both NBTI and HCI depend on the transistor V_{ds} values which will change as well. Since the surrogate model contains the response surface for all cases of the input parameters, the intermediate node voltages computed through the integrated variability/reliability simulation will yield the exact deteriorations on the transistor drain current. Finally, the effect of process variations on time-based degradations are made clear to the designers by handling t_{age} together with the other process and temperature parameters.

Drain current surrogate models are constructed from the evaluated device responses to a number of sample vectors which are different combinations of 11 variables. The

Table 5.1: Properties of constructed drain current surrogate models

Model Name	# of sample vectors	# of test vectors	RMSE	RRSE
Aged NMOS drain current surrogate model ($0 \leq t_{\text{age}} \leq 8$ months)	2880	3200	46 μA	1.7%
Aged NMOS drain current surrogate model (8 months $\leq t_{\text{age}} \leq 10$ years)	2880	3200	69.3 μA	2.11%
Aged PMOS drain current surrogate model ($0 \leq t_{\text{age}} \leq 8$ months)	2880	3200	38.3 μA	2.62%
Aged PMOS drain current surrogate model (8 months $\leq t_{\text{age}} \leq 10$ years)	2880	3200	25.6 μA	1.85%

surrogate models produced can be classified based on the modeled ranges of t_{age}. The first model covers the time span 0–8 months whereas the second model is used to find the drain current in a time frame from 8 months to 10 years. This kind of a division is a natural result of the reliability phenomena characteristics that has power-law characteristics. In the first year a large ΔV_{th} is observed leading to large changes in the transistor drain current. After the first year, the degradation rate is reduced to a small yet steady value. Detailed information about both the models for NMOS and PMOS devices is provided in Table 5.1. Root mean square error (RMSE) and root relative square error (RRSE) values here indicate that accurate surrogate models can be established that are capable of visualizing the reliability degradations.

5.3 Results

5.3.1 Reliability Simulation Example

The proposed reliability analysis will be applied here to the differential amplifier in Figure 5.2 which has the circuit parameters in Table 4.1. The results of the reliability simulation is presented in Figure 5.3 where five different process corners —nominal NMOS–nominal PMOS (NN), fast NMOS–fast PMOS (FF), slow NMOS–slow PMOS (SS), fast NMOS–slow PMOS (FS) and slow NMOS–fast PMOS (SF)— are compared with respect to the time-based degradation of I_1 in the circuit of Figure 5.2. These process corners are implemented by selecting the individual components of the NMOS and PMOS drain current surrogate models derived from the BSIM device model. The main aim of this exercise was showing the capabilities of the proposed variability/reliability simulation methodology.

The two surrogate models representing the short-term and long-term degradation responses were employed in the reliability analysis. The short-term span (0–8 months) model was characterized with five simulations equally spaced in the time domain. Similarly, 31 equally spaced time points were used to determine the behavior of I_1 over the long-term span. The simulation results are then applied to a nonlinear regression analysis to create the final continuous curves. All simulations were at 100°C to account for the worst case degradation as both NBTI and HCI effects become more dominant at elevated temperatures. Each simulation takes on average 2 s to complete in the MATLAB® environment on an INTEL I7 Quad Core processor clocking at 2.66 GHz.

Initial simulation results have suggested that, compared to NBTI, HCI does not play a critical role. This situation arises mainly from the circuit topology. The variability simulations reveal that the V_{ds} value of MN2, V_{out}, stays at 0.5 V which is far below the

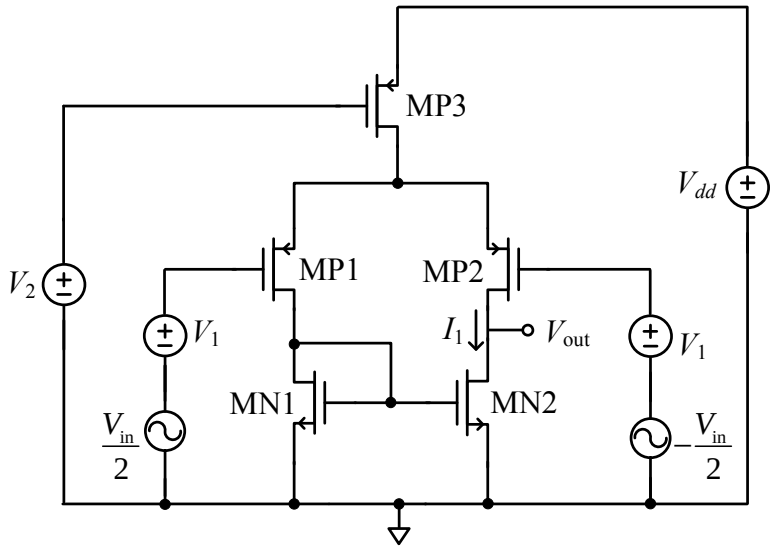


Figure 5.2: Actively loaded PMOS-input differential amplifier. This circuit was analyzed in Chapter 4 from the variability perspective.

V_{dd} value. HCI becomes more of a concern as the drain voltage approaches V_{dd}, where electrons have sufficient energy to induce interface traps. Based on this, only NBTI aging effects were considered and this resulted in substantial changes of the V_{th} values of the PMOS transistors. Several important conclusions can be made based on Figure 5.3: First, the level of degradations remains between 89.5% to 93% even after 10 years of operation. This indicates that 65 nm is relatively safe for analog circuit operation from the reliability point of view. Second, there are small differences of degradation in I_1 for the different process corners. The V_{th} increase of MP3 after 10 years of operation was found to be 18.3 mV for nominal process condition, varying from 16.6 mV for the SS corner to 20.2 mV for the FF corner.

Although some of the previous art only considered effects due to the largest degradation contributor [88], this work considers all contributors expressed as a ratio of the largest degradation term. The ratio of the V_{th} increase of MP1 and MP2 —which have

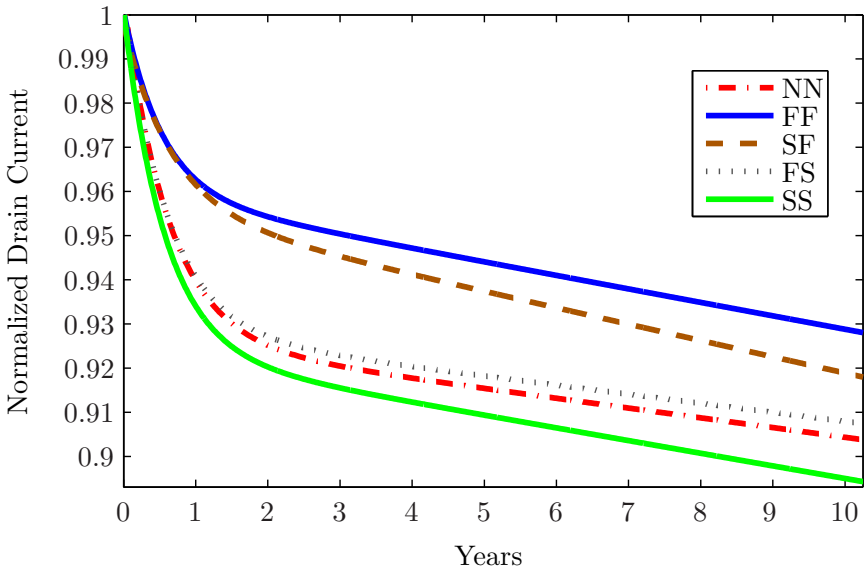


Figure 5.3: The degradation of I_1 for the circuit in Figure 5.2 at different process corners over the time.

the same biasing conditions— to the V_{th} increase of MP3 has been computed to be between 0.35 and 0.4 for different process corners over the 10 year period. This ratio will be utilized to develop techniques in the next section that enable the design for reliability.

Finally, the gate voltage of MP3, V_2 was varied to provide insight into the biasing circuit design. MP3 determines the DC current of the differential amplifier. The V_{sg} level for MP3 establishes how much the DC current will be degraded through NBTI effects. Three choices are compared where V_2 is selected to be 0.3 V, 0.5 V and 0.65 V. The analysis is conducted for nominal process conditions and 100°C operation temperature. The results shown in Figure 5.4 indicate that, as expected, for a lower V_{sg} voltage which is the case when V_2 is 0.65 V, the deterioration in the drain current saturates early and the current remains almost at the same level. However the impact of aging continues further for higher V_{sg} levels as can be seen for V_2 equal to 0.3 V. To combat aging the

gate voltage should follow a pattern that will reverse the effect of aging on the drain current. This can only be achieved with a suitable biasing network.

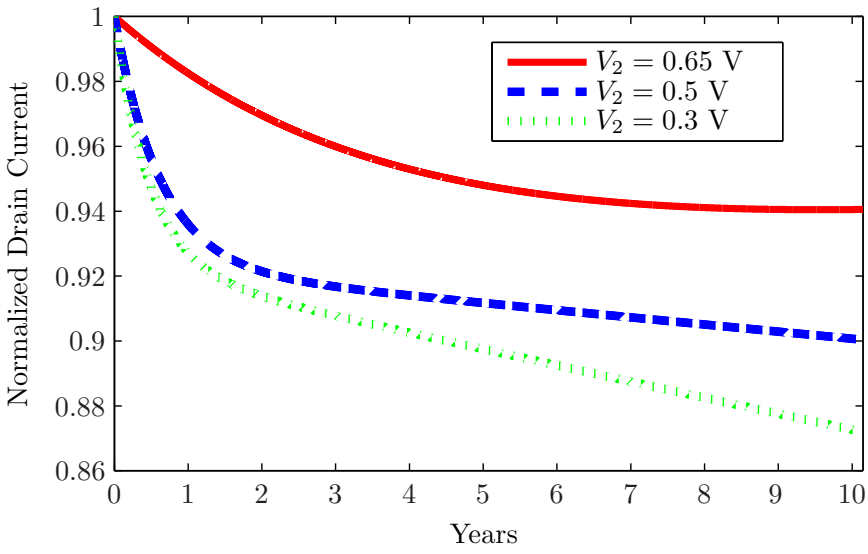


Figure 5.4: The drain current degradation of the circuit in Figure 5.2 for different gate voltages of MP3.

5.3.2 Design For Reliability Approaches

In this section, the variability/reliability analysis framework is utilized to draw conclusions with respect to the design for reliability concept. Design for reliability guidelines have been reported to some extent in the literature specifically for digital circuits. A typical approach is increasing the sizes of the transistors in a gate [94]. All design constraints are ensured to satisfy the design specifications even with the worst case degradations of devices are assumed. Besides, adaptive methods attracted attention of designers. Tuning the supply voltage V_{dd} where V_{dd} is gradually augmented as time progresses can avoid

performance degradations. Similarly, adaptive body biasing aims to gradually increase the magnitude of V_{bs} so that the threshold voltage is kept at a constant value. It has been employed in several design examples [95]. Multiple techniques can also be simultaneously applied to reduce the impact of NBTI on a circuit. In [96], adaptive V_{dd} increase, adaptive clock frequency reduction (to relax the design specification for the increased device V_{th}) and adaptive cooling (to reduce the accelerating effect of temperature on transistors) techniques are combined to increase the lifetime computational efficiency. This concept gives a measure of how many clock cycles per consumed energy are successfully realized during the lifetime of the analyzed circuit [96]. The other solution methods in digital circuit design for reliability include path reordering and duty cycle tuning [80]. For analog circuits, design for reliability guidelines have been proposed mostly on the design of analog-to-digital converter components, such as the preamplifier and the comparator [88]. In [97], switching of the reference and input voltages applied to the gates of a differential pair is suggested for a comparator design to minimize the imbalance of aging degradation.

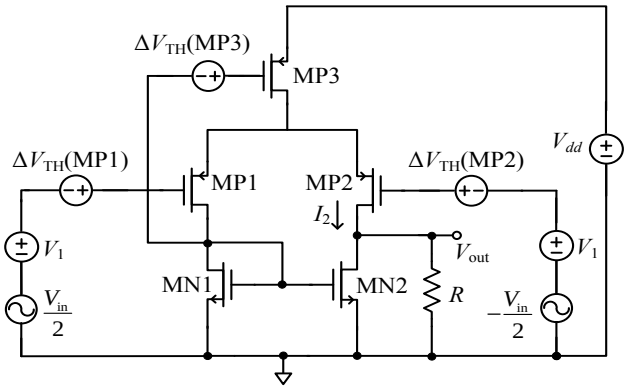
As pointed out at the end of the previous subsection, the biasing network for the circuit in Figure 5.2 should be designed such that the gate voltage of MP3 is modulated over the course of time exploiting time-based degradations. To illustrate this point, three different biasing structures shown in Figure 5.5 are considered. In Figure 5.5(a), the gate voltage of MP3 is connected to the gate voltage of MN1 creating a feedback structure. Due to the current degradation, the diode-connected MN1, which essentially acts as a resistor, will have a smaller drain voltage. This will be reflected directly to the gate of MP3, trying to augment the V_{sg} value. Obviously, the increase in V_{sg} would boost the current but the rate of degradation will slightly increase as well. These counteracting mechanisms can balance and reduce the rate of current reduction. The biasing circuit

in Figure 5.5(b) is the typical means of biasing a differential amplifier. In this case, the current mirror formed by MP1 and MP3 creates a similar balancing mechanism for drain current degradation. This phenomenon has been observed for NMOS current mirrors in the literature [98], and an analogous effect will occur for PMOS current mirrors as well. The degradation of the drain current of MP1 reduces the drain voltage of MN1 which in turn is connected to the gate of MP3. It should be noted that the drain currents of MP1 and MP3 are boosted due to this mechanism.

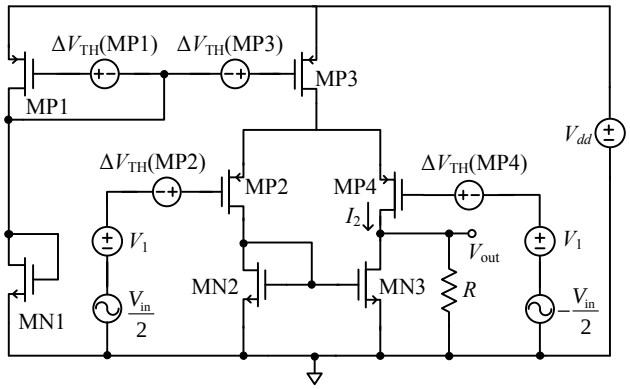
Another remark is that both circuits in Figures 5.5(a) and 5.5(b) have a similar feedback structure to reduce the effects of the aging. The circuit in Figure 5.5(c), however, has a biasing network consisting of only NMOS devices. In this circuit the gate voltage of MP3 remains almost constant. Whether the devices MN1 and MN2 will suffer appreciably from HCI depends on the gate voltage of MP3. As long as this voltage remains within 0.3 V to 0.9 V for a V_{dd} level of 1.2 V, the V_{ds} voltages for both MN1 and MN2 become sufficiently smaller than V_{dd} and thus the HCI degradation of NMOS transistors can be neglected. Similar conclusions can be drawn based on experimental data from [99], [100]. It can also be assumed that the gate voltage of MP3 will remain the same in this topology over the time period analyzed.

In the next step, the reliability aspect of the three circuits with respect to AC gain is investigated. The circuits shown in Figure 5.5 are analyzed using the surrogate model-based variability/reliability method and the corresponding V_{th} changes are found which modify the effective gate voltages of the transistors. Since each transistor is represented with a different input vector, the V_{th} changes can reflect all within-die process variations and temperature differences.

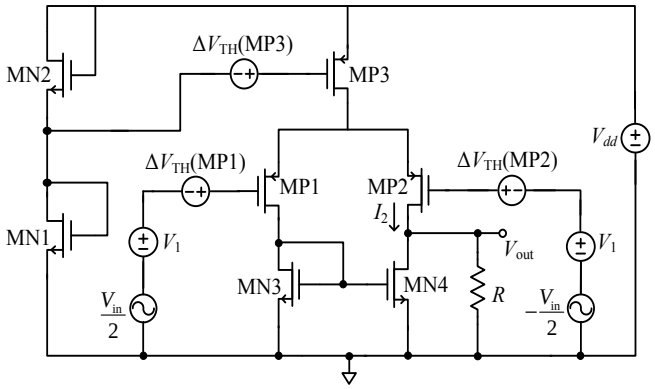
The analysis here represents the AC voltage gain results for the three circuits in three different process corners at an operating temperature of 100°C. The results are



(a): Circuit I



(b): Circuit II



(c): Circuit III

Figure 5.5: The same differential amplifier topology in Figure 5.2 with three different biasing structures. The added ΔV_{th} sources are used as a means of visualizing the adverse effect of NBTI on I_1, I_2 and V_{out}.

presented in Figure 5.6. Several observations can be made from this figure. Firstly, the prediction regarding the circuits having a biasing structure that increases aging resilience is confirmed. In all subfigures, the degradation performance of the circuits Figures 5.5(a) and 5.5(b) are better than that of the circuit in Figure 5.5(c). It is also noteworthy that for different process corners the rate of aging changes. Moreover, for the FF corner, the voltage gain seems to be increasing for a certain time period. The underlying reason of this behavior stems from the choice of the load resistance R and the channel lengths of the devices in the circuits. In all these circuits the value of R is set to 500 Ω, transistors are from IBM 10SF process (65 nm) and they have a minimum channel length. For the FF corner, the output resistance r_o is smaller than for SS corner. It is known that r_o increases as the drain current drops which is the case here due to the time-based degradation. Although some literature claims that there is an increase in g_{ds} due to aging effects [101] competing with the increase in r_o due to drain current drop, to the best of our knowledge, no reported data on the increase in g_{ds} for PMOS transistors in 65 nm has appeared yet.

At this point it should also be remembered that the gain of differential amplifiers are calculated using the formula $g_m\,(r_{o,\mathrm{p}}||r_{o,\mathrm{n}}||R)$ where $r_{o,\mathrm{p}}$ and $r_{o,\mathrm{n}}$ are of the devices attached to the load resistance R. For the FF corner, the values of $r_{o,\mathrm{p}}$ and $r_{o,\mathrm{n}}$ at the minimum channel length are lower than the load resistance thus they impact gain. For all process corners g_m decreases due to time-based degradation. That is why these two counteracting mechanisms shape the situation in Figure 5.6(c). Until the equivalent resistance of $r_{o,\mathrm{p}}$ and $r_{o,\mathrm{n}}$ surpasses R, the gain increases a little, but then it stops increasing and begins to decrease due to the reduction in g_m. The gain increase in amplifiers over the time that has been observed in the literature can also be linked to this reasoning [102]. This effect is not observed at NN and SS corners since the equivalent

resistances of $r_{o,\mathrm{p}}$ and $r_{o,\mathrm{n}}$ are higher than R.

Another remark can be made from Figure 5.6 comparing the performance of Circuit I, (see Figure 5.5(a)), and Circuit II, (see Figure 5.5(b)). It can be inferred that the aging resilience of Circuit I is a little higher than that of Circuit II under these conditions. However, Circuit I is not as flexible as Circuit II in terms of the range of bias currents, thus there is a design trade-off between these two biasing topologies.

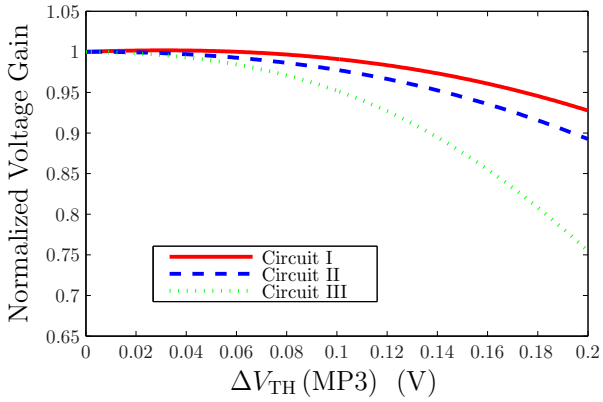
The evolution of the V_{th} increase in MP1, MP2 and MP3 for the nominal process corner of the circuit in Figure 5.5(a) is illustrated in Figure 5.7. The ΔV_{th} value becomes higher when the biasing conditions of a transistor make it susceptible to NBTI degradation. This can be visualized when the curves of MP3 and MP1 (or MP2) are compared. The slight difference between the curves of MP1 and MP2 which have the same V_{sg} arises due to the differences of their V_{sd} values.

Finally, the reliability analysis has been applied to the three circuits to visualize how the drain current I_2 (of the p-channel device on the output side of the differential pair) degrades over time. Nominal process conditions along with 100°C operation temperature have been used. The results are depicted in Figure 5.8. It can be seen that the circuits of Figures 5.5(a) and 5.5(b), which benefit from a feedback structure to balance the aging effects, perform better than the circuit of Figure 5.5(c) where the gate voltage of MP3 is kept constant. This observation also highlights the design for reliability practice of modulating the biasing transistor for a certain circuit topology where the deteriorating effects of aging phenomena are used as the input of this modulation.

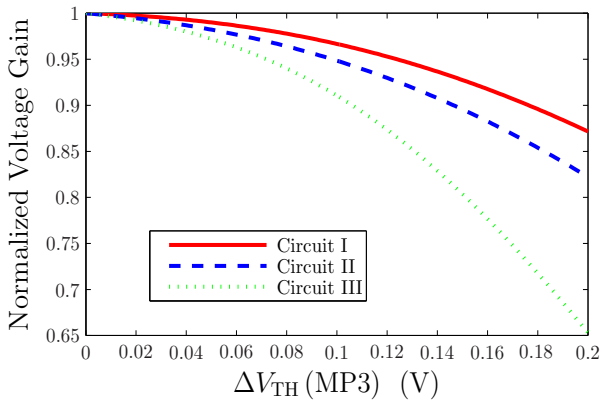
5.4 Summary

This chapter presented a reliability simulation framework that has been integrated with variability analysis. Since the relationships of reliability and variability phenomena will continue to become more complex in future technology nodes, they need to be analyzed simultaneously. Surrogate modeling has proven to be a useful tool for this purpose greatly reducing the dimensionality of the problem to be modeled.

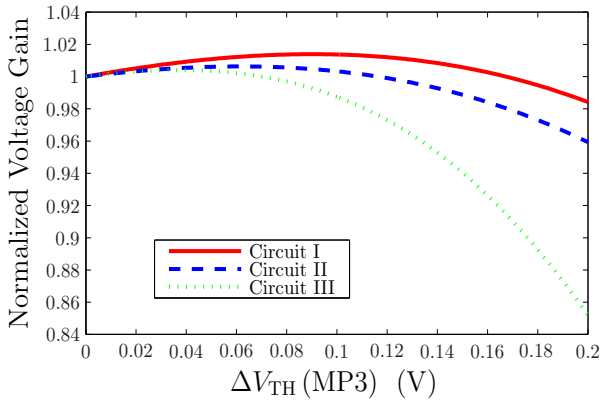
The effect of NBTI on analog circuits has been characterized for a differential amplifier using the reliability simulation methodology introduced. The drain current is shown to decrease by about 10% over 10 years. This result indicated that at 65 nm, reliability problems pose a minor concern. The paper also reveals that the biasing of analog circuits will play a crucial role in determining the aging response of a circuit. DC bias voltages need to be modulated through feedback mechanisms for better aging resilience which will be of more importance with the advancement to newer technologies where the reliability concerns are more significant compared to 65 nm node.



(a): NN Process Corner



(b): SS Process Corner



(c): FF Process Corner

Figure 5.6: The time-based variation of voltage gain at different process corners for the circuits depicted in Figure 5.5. The increase of gain in Figure 5.6(c) is related to the increase of the output resistance of r_o of the transistors where the r_o value is still less than the load resistance R.

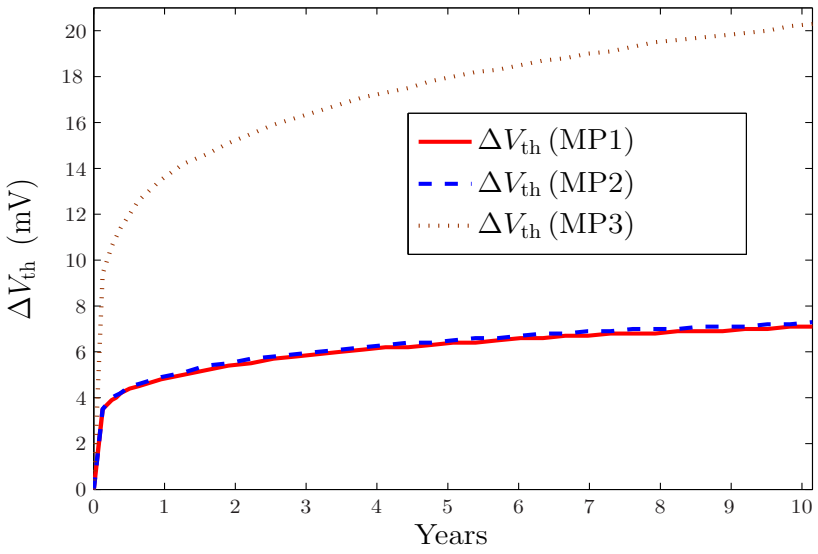


Figure 5.7: The increase in the threshold voltages of the PMOS transistors in the circuit of Figure 5.5(a). Since the effective gate-source potential of MP3 is the highest it undergoes the most severe NBTI degradation.

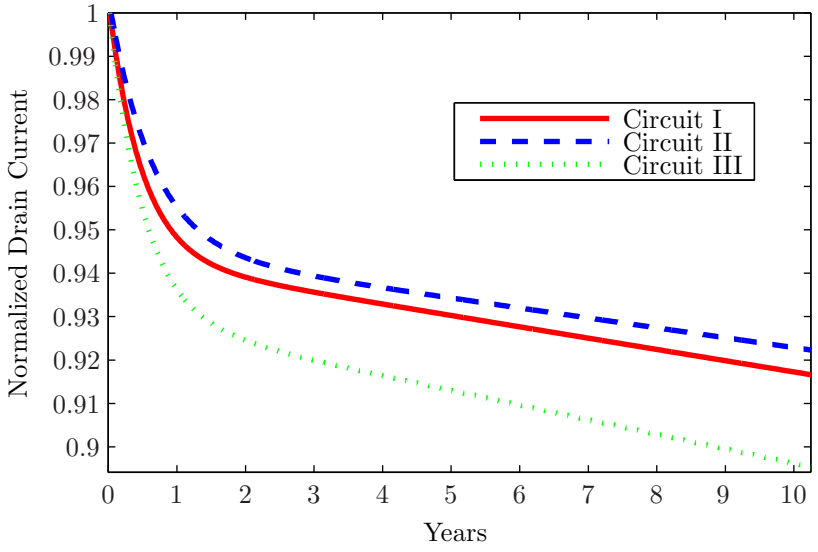


Figure 5.8: Comparison of the degradation in I_2 of the three differential amplifiers with different biasing topologies depicted in Figure 5.5. As expected, the biasing structures where the adverse effects of NBTI have been utilized as a means of feedback yield less degradation.

Chapter 6

Monitoring and Characterization of Circuit Reliability

This chapter is devoted to the experimental studies that investigate reliability degradation effects in ICs. To realize accelerated aging of fabricated circuits, an experimental setup has been developed that allows a combined electrical and thermal (electrothermal) stress to be applied circuits under test. A monitoring circuit for NBTI is presented with details of its structure, implementation and measurements. Then a differential amplifier described in Chapters 4 and 5 is characterized based on experimental results. Finally, the changes in voltage transfer characteristics of two-input NAND and NOR gates that have been subjected to electrothermal stress, and thus degraded are explained.

6.1 Implementation and Experimental Setups

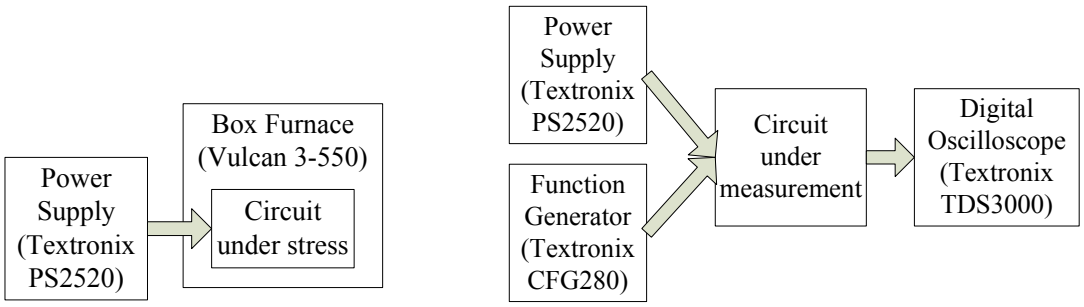
An important aspect of investigating the variability and reliability effects on nanoscale circuits is monitoring and characterization. Experimental studies should be conducted

to provide realistic data for model development. In this study a test chip was fabricated in IBM 65 nm device technology (10 LPE). The die photo for the complete test chip is shown in Figure 6.1.

Figure 6.1: The die photo.

The chip contains an analog NBTI monitor, a differential amplifier, and several digital building blocks. It is directed at demonstrating the degradation level of different performance metrics of fabricated circuits by applying electrothermal stress. The electrothermal stress is defined here as the electrical stress applied to the circuits in a furnace at a temperature of 100°C. In particular NBTI effects are investigated. The stress experiment setup is shown in Figure 6.2(a). The circuit is simultaneously stressed thermally and electrically using the programmable furnace and the power supply, respectively. Pictures of the stress experiment setup are provided in Figures 6.3 and 6.4.

The measurement experiment setup is given in Figure 6.2(b). Here, the signal generator is used to create the small signal potentials (if needed), and the power supply generates the desired DC biasing condition. Results are captured through the digital oscilloscope. A picture of the measurement setup is provided in Figure 6.5.



(a): Stress test setup (b): Measurement setup

Figure 6.2: Experimental setups to test the reliability degradation of fabricated circuits.

6.2 Analog NBTI Monitoring Circuit

This section describes the design and implementation of an NBTI monitor based on an analog circuit structure in 65 nm CMOS technology. As described in Chapter 5, NBTI constitutes a major concern for analog and digital circuits since it increases V_{th} and decreases μ_{eff}. These consequences of NBTI lead to degradations in circuit performance metrics. Sensor circuits have been developed in order to quantify these degradations. The aim of the circuit is determining the time point at which the circuit under consideration is no longer functionally correct or does not meet design specifications. Emphasis has been placed on monitoring digital circuits and digital building blocks have been shown to be very susceptible to NBTI degradations [103]. Techniques proposed for NBTI monitoring circuits include phase comparison of a stressed and reference ring oscillator [104], [105], delay locked loop-based approaches [106], observation of changes in the slew rate of rising edge in ring oscillators [107], delay-element based approaches [108], [109], comparison of the performance of differently sized (and thus stressed) inverters [110], and observation of voltage glitch generation, and detection circuits [111]. To the knowledge of the authors, an analog circuit-based NBTI monitor has not been developed yet. There is a significant

Figure 6.3: Electrothermal stress setup for the test circuits.

need for such a monitor since analog circuits built using deep sub-micron technologies are more susceptible to aging effects than their larger counterparts. As can be observed in (5.1), transistors with larger (more negative) gate-source potential, V_{gs}, thinner oxide thickness, t_{ox}, and subject to higher temperatures are degraded more. Analog circuits designed with advanced technologies, especially at the 65 nm technology and below, have lower t_{ox} and can operate at higher temperatures with relatively comparable voltages to

Figure 6.4: Interior of the furnace with the circuits under degradation shown connected to the stress voltages.

the previous device technologies. Thus there is a significant need for an NBTI monitoring circuit that tracks actual circuit operating conditions.

The degradation that NBTI causes affects both the DC and AC characteristics of a transistor. The AC performance, including voltage gain and frequency response, depends on the DC characteristics as well as other parameters such as parasitic capacitances. The effect of aging on parasitic capacitances is not clear. Thus, prior approaches that consider sensing the AC performance of a stressed p-channel transistor have assumed that the AC degradation solely relates to the DC degradation. The work presented here eliminates this assumption by presenting an NBTI monitor that uses a stressed device (and thus

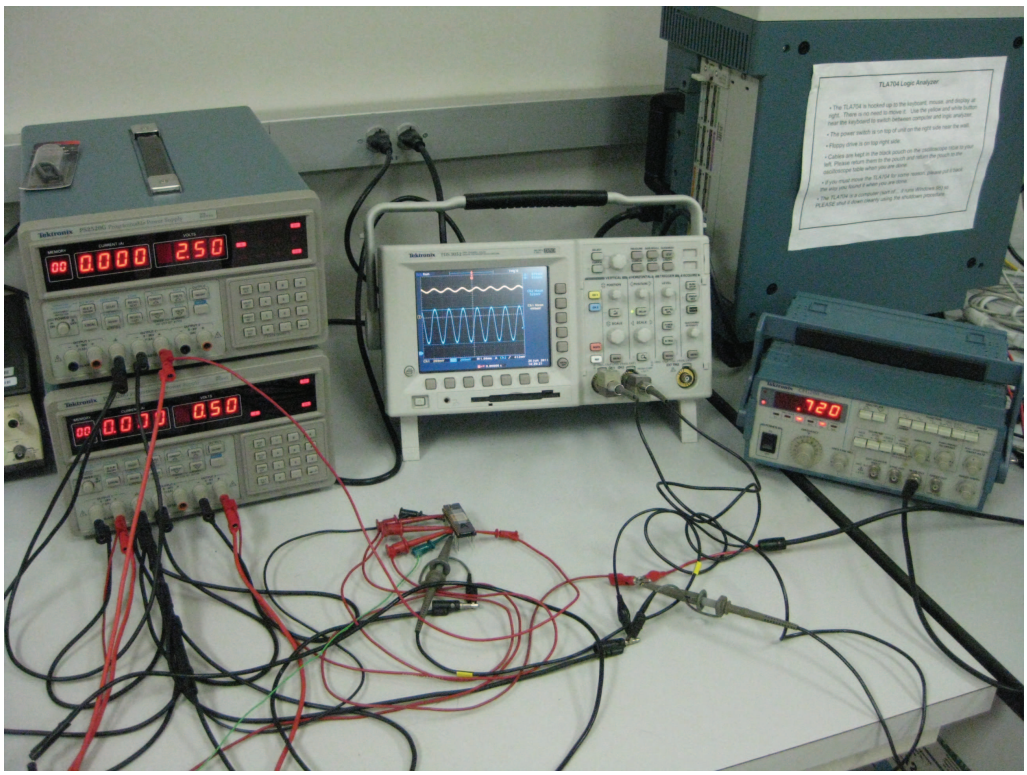


Figure 6.5: Setup for measuring the performance of test circuits before (after) the electrothermal stress.

subject to the NBTI-based degradation) in the biasing of a circuit that is not subject to NBTI-based degradation. This is the key idea behind the construction of the proposed NBTI monitor and is supported by the results of Chapter 5 confirming that biasing is important in the reliability of circuits.

The NBTI monitor circuit comprises a common-gate amplifier whose biasing circuit is subject to degradation. By altering the operating point of the amplifier, a drop in voltage gain is presented that varies linearly with respect to the V_{th} change. Moreover this monitor can easily be integrated with the analog circuits that will be monitored.

6.2.1 Circuit Design

The NBTI monitor circuit is shown in Figure 6.6 and comprises a single stage common gate amplifier and a biasing circuit (a PMOS transistor, MP1 and a diode-connected NMOS transistor, MN1). The switch selects the reference voltage V_{ref} at times $t = nt_{\text{meas}}$ (where n is an integer). The stress voltage V_{str} is applied to the gate of MP1 except for very short time intervals during measurement when V_{ref} is applied. V_{str} is chosen to produce the NBTI degrading effect being monitored for a particular circuit. By designing the circuit such that $V_{\text{str}} = V_{\text{ref}}$, the need for the switch is eliminated. The response of the monitor circuit depends on V_{str} since biasing of the common gate amplifier depends on the operating condition of MP1. Initially, the common gate amplifier is designed to have a reasonable voltage gain A_v that forces MN2 to operate near the boundary of the saturation and triode regions. Once MP1 is aged, the biasing conditions change such that MN1 is pushed into the triode region which drops the voltage gain. This drop should be linear so that the variation of the amplifier gain can be linearly correlated with the V_{th} change of MP1. Appropriate choices of R_1 and R_2 realize this. R_s is the Thévenin's resistance of a signal source.

The design presented relies on NMOS transistors not being affected by the degradation due to NBTI. However, the degradation of NMOS transistors due to HCI could potentially interfere with the biasing changes due to NBTI. Thus NM1 should have sufficient immunity to HCI degradation. To accomplish this, the lengths of all transistors were chosen to be 3 times the minimum length L_{min}, thus minimizing the impact of HCI. To see this, consider the analytical expression for HCI, see (5.3). $V_{d,\text{sat}}$ can be shown to be proportional to L_{eff} for external and internal drain-source resistance models of a transistor [47]. Thus as L_{eff} increases, so does $V_{d,\text{sat}}$. This causes $V_{ds} - V_{d,\text{sat}}$ to drop and

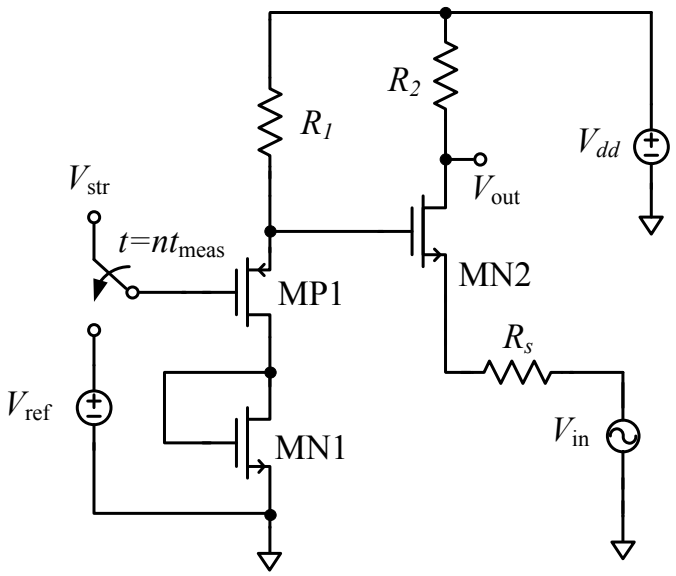


Figure 6.6: NBTI monitor circuit.

the first term in the second line of (5.3) diminishes exponentially. Therefore, $(\Delta V_{\text{th}})_{\text{HCI}}$ becomes smaller which in turn means that the impact of HCI becomes insignificant.

6.2.2 Measurement results

The circuit of Figure 6.6 was implemented in the IBM 65 nm process (10 LPE) with a size of 7 μm $\times$ 7 μm. The design parameters are provided in Table 6.1. A proof of the design concept is achieved by demonstrating a linear relationship between the voltage gain and the threshold voltage change ΔV_{th} due to aging. The latter is characterized by an effective gate-source voltage change $|\Delta V_{\text{gs}}|$ of MP1 since $|\Delta V_{\text{th}}|$ directly reduces the biasing voltage of MP1. The change in A_v is measured with respect to the changes in the externally-applied gate voltage of MP1. V_{dd} is 1.2 V and the gate biasing of MP1, V_{str}, is initially set to 0 V. An 11 MHz function generator creates the small signal voltage V_{in} and an oscilloscope is used to measure A_v. The measured power consumption of the

monitor is in the range of 1.6 mW to 1.77 mW depending on the biasing condition of the amplifier as modulated by the V_{th} regulation of MP1. The effective power consumed by the NBTI monitor circuit can be reduced by merging it with the functional analog circuits to be monitored.

Table 6.1: Circuit parameters of the proposed NBTI monitor circuit.

Device	W/L (μm/μm)	Resistance name	Resistance Value
MN1	10/0.18 μm	R_1	2.9 kΩ
MN2	50/0.18 μm	R_2	575 Ω
MP1	10/0.18 μm	R_s	50 Ω

Figure 6.7 shows that the voltage gain drops linearly with $|\Delta V_{\text{gs}}|$ as desired. To illustrate this point further, a linear fitting function to the responses of the monitoring circuit is introduced. The slope of the linear fitting function yields a sensitivity of -3.15 V^{-1}. The coefficient of the determination, R^2, is 0.997 indicating a strong linear relation between A_v and $|\Delta V_{\text{gs}}|$. This linear relation is very important particularly during the operation of the monitored circuit to easily estimate the level of maximum degradation.

The response of the NBTI monitoring circuit to stress was investigated by operating the circuit in a condition of electrothermal stress. The electrothemal tests were conducted by applying a constant voltage of 0 V to the gate of MP1 with the circuit held at a temperature of 100°C (in a box furnace) and measuring the voltage gain of the monitoring common gate amplifier. Results are provided in Figure 6.8 where each unit of electrothermal stress, σ_T, is a 30 minutes time interval. Thus Figure 6.8 represents 2

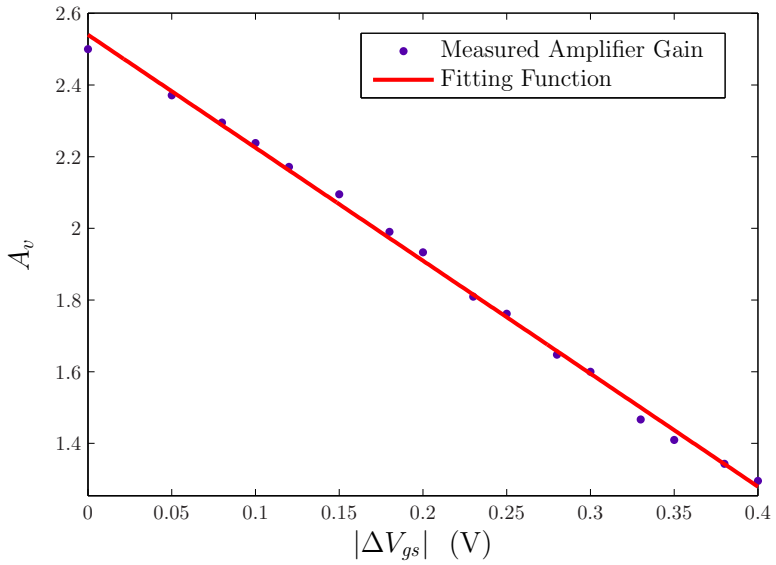


Figure 6.7: The measured voltage gain and the fitting function.

hours of electrothermal stress. A cooling period is left between successive measurement periods. This is required to correctly estimate the level of degradation. In particular, when the circuits are just removed from the furnace, a lower voltage gain is measured compared to the gain at room temperature. Cooling periods prevent overestimation of the NBTI effect and provide the necessary context for a fair comparison of voltage gains. Thus Figure 6.8 presents the voltage gain reduction due to electrothermal stress that induces NBTI degradation monitored by the common gate amplifier. These results also justify the outcomes of Figure 6.7, which demonstrates that a permanent voltage drop is induced to the monitor through the NBTI degradation of the biasing PMOS.

All in all, the idea that the gain of an amplifier can be linearly changed through the NBTI degradation of a p-channel transistor in the biasing of the amplifier has been experimentally validated. Through careful design of the circuit, the desired linear decay of voltage gain is achieved. Since many analog designs do not include digital parts in

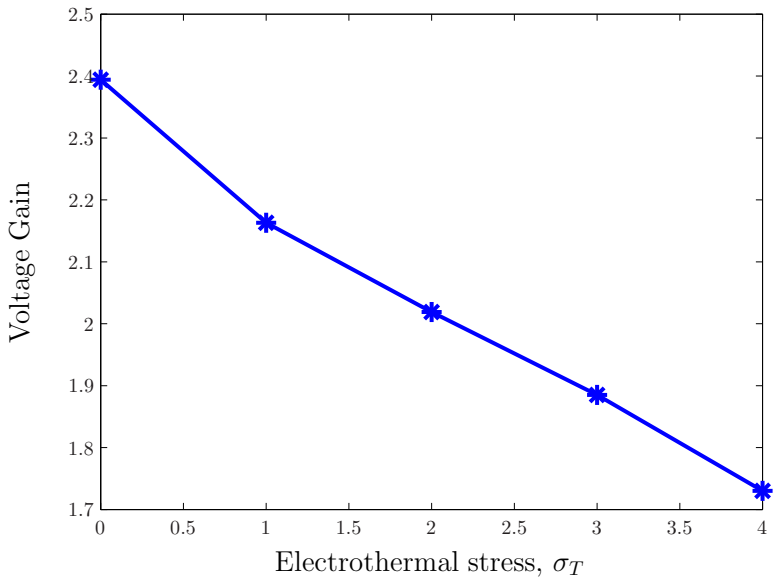


Figure 6.8: Change of the measured voltage gain as the electrothermal stress is applied. One unit of σ_T is 30 min.s at 100°C with 0 V applied to the gate of MP1.

which the digital circuit-based NBTI monitors can be included, the monitoring circuit developed here can easily be integrated into larger circuits for monitoring aging of p-channel devices that are subject to large V_{gs} potentials or electrothermal stress.

6.3 Characterization of Reliability Degradations

Although considerable research effort has been spent on characterizing the reliability degradation of the individual transistors, few studies have been undertaken to observe the aging degradation at the level of circuit building blocks. In [112] inverters in 130 nm technology were experimentally analyzed with respect to NBTI and HCI aging effects. Both DC and AC stress were applied. A rightward shift in the voltage transfer curve is observed due to HCI and leftward shift occurred due to NBTI. It has been determined

that HCI degradation in NMOS transistors becomes worse with switching activity, i.e. AC stress. In a similar study provided in [113], aging of inverters causes a decrease in the noise margin of an inverter.

Aging experiments were also carried out on power amplifiers where the drain source voltage can be excessively high causing HCI damage to the devices under stress. A power amplifier in 65 nm technology operating at 60 GHz was tested for its different specifications [114]. Experimental results revealed that more than 15% of the power gain and output power are lost after 50 hours of stress. Additionally, degradations of input and output return ratios (S_{11} and S_{22}, respectively) are recorded.

In this section, the characterization of reliability degradation on a differential amplifier as well as two-input NAND and NOR gates are analyzed. In particular, the changes in the voltage gain of a differential amplifier due to NBTI and in the voltage transfer characteristics of NAND and NOR gates are measured based on the electrothermal stressing of fabricated test circuits.

6.3.1 Characterization of the Reliability of a Differential Amplifier

The reliability of a differential amplifier was discussed in Chapter 5. One of the circuits discussed (see Figure 6.9) is fabricated with the transistor parameters provided in Table 6.2. It should be noted that unlike the designs provided in Chapter 5, the lengths of transistors are chosen to be three times the minimum lengths. Longer channel lengths result in larger output resistances which increase the voltage gain. They also reduce the variability observed. Finally, HCI degradation can also be diminished by a longer channel length selection. Nevertheless, this choice also impacts the frequency response of

the amplifier and reduces the bandwidth leading to lower operation speed of the circuit.

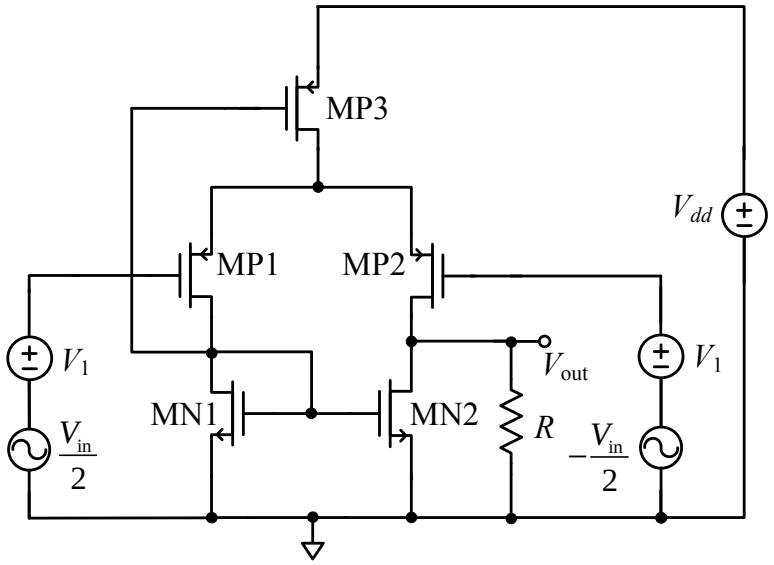


Figure 6.9: Circuit diagram of the fabricated differential amplifier.

Table 6.2: Circuit parameters of the fabricated differential amplifier.

Device	W/L (μm/μm)
MN1	30/0.18 μm
MN2	30/0.18 μm
MP1	60/0.18 μm
MP2	60/0.18 μm
MP3	120/0.18 μm

The testing of the circuit is made in two stages. In the first stage, the effective gate-source voltage of the differential pair is changed through the power supply. This is an

attempt to observe the adverse effects of V_{th} increase due to NBTI on the voltage gain of the differential amplifier. At the same time, the same procedure is simulated with the SPICE circuit simulator. Results are given in Figure 6.10. It can be seen that for a change of 0.1 V in the effective gate-source voltage, no appreciable drop can be recorded for both the simulated and measured voltage gain. A constant decrease is observed from 0.1 V onward. This result can be compared with the gain reductions shown in Figure 5.6(a). The difference between the Figures 5.6(a) and 6.10 is that the x-axis of the former figure is the V_{th} degradation of the tail transistor whereas the x-axis in the latter figure is the V_{th} degradation of a transistor of the differential pair. Recalling that the V_{th} degradation of the differential pair is about one third of the V_{th} degradation of the tail transistor (i.e. a V_{th} reduction of 0.2 V of the tail transistor corresponds to 0.07–0.08 V of V_{th} reduction for a transistor of the differential pair), it can be seen that simulated gain reductions depicted in both graphs coincide with each other. Thus, the differential amplifier here is assumed to be subject to a large degradation that extends beyond the typical degradation through its lifetime which the results of Figure 5.6 are based on.

In the second stage electrothermal stress tests are conducted which corroborated the results above. Here, a voltage of -0.5 V is applied to the gates of the differential pair and a V_{dd} of 2.5 V is chosen to increase the magnitude of the gate-source voltages. The test is conducted at a temperature of 100°C for one hour. The results yield a minor reduction of gain from 8.45 to 7.9. Thus, it has been experimentally demonstrated that the differential amplifier having a feedback structure to increase aging resilience as explained in Chapter 5, indeed presents a better reliability performance.

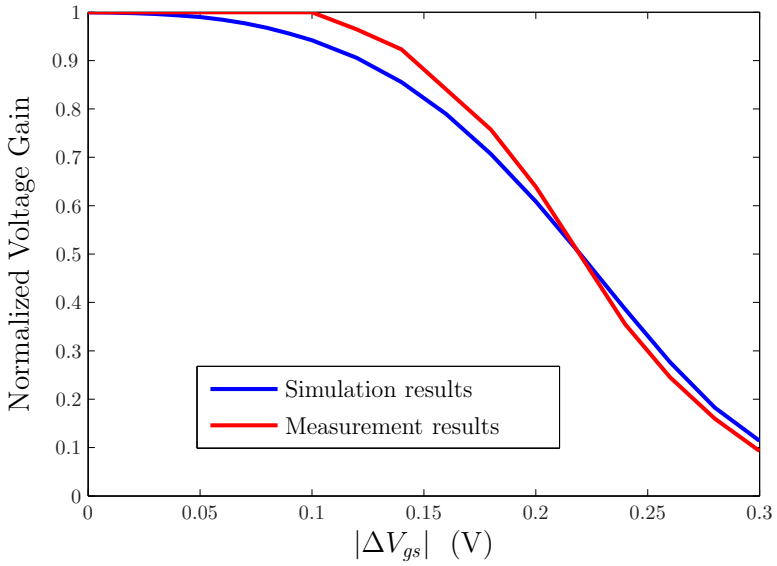
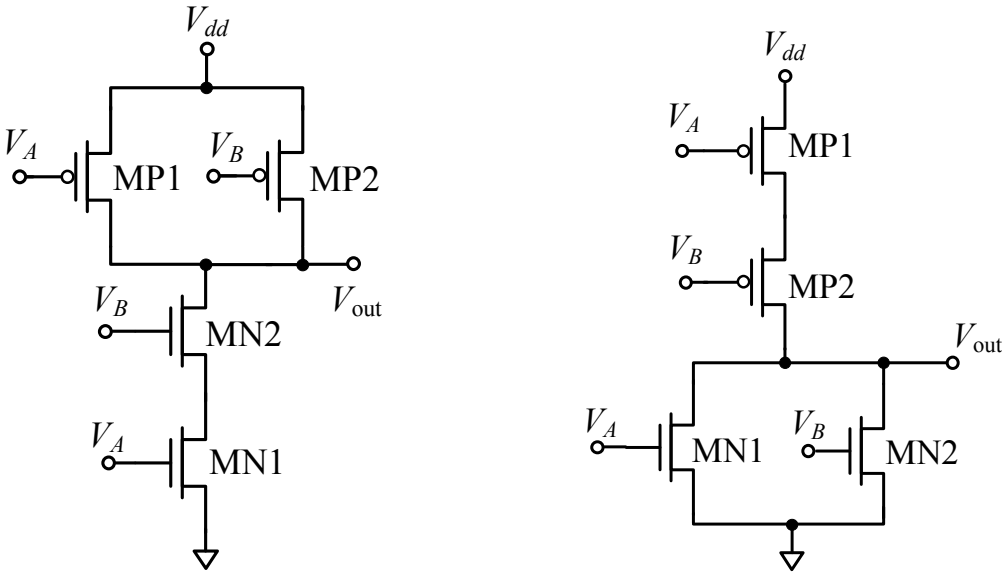


Figure 6.10: Change of the voltage gain of the fabricated differential amplifier in response to the changes in the effective gate-source voltages of the differential amplifier.

6.3.2 Characterization of the Reliability of two-input NAND and NOR gates

Two-input NAND and NOR gates are fabricated and their voltage transfer characteristics are analyzed before and after an electrothermal stress testing. The circuit diagrams for these gates are given in Figures 6.11(a) and 6.11(b). The widths and lengths of the transistors in the circuit diagrams of Figure 6.11 are provided in Table 6.3. Voltage transfer curves are one of the most critically affected quantities by the reliability degradation mechanisms and as it has been demonstrated in the case of the XOR circuits, changes induced can result in reduced noise margins of the digital circuits, particularly for the high inputs.

Firstly, the reliability performance of the NAND gate will be analyzed. In the electrothermal testing of the NAND gate, V_A is set to 0 V and V_{dd} is set to 2 V. This choice



(a): 2-input NAND gate (b): 2-input NOR gate

Figure 6.11: Circuit diagrams for two-input NAND and NOR gates.

Table 6.3: Circuit parameters for the two-input NAND and NOR gates.

NAND		NOR	
Device	W/L (μm/μm)	Device	W/L (μm/μm)
MN1	12/0.06 μm	MN1	6/0.06 μm
MN2	12/0.06 μm	MN2	6/0.06 μm
MP1	12/0.06 μm	MP1	24/0.06 μm
MP1	12/0.06 μm	MP2	24/0.06 μm

is expected to degrade MP1 with its V_{sg} value equaling 2 V. The temperature is held at 100°C for one-and-half hours along with the application of the voltages to the circuit. The voltage transfer characteristics of the NAND gate while V_A is changing before and after MP1 degraded is shown in Figure 6.12. A shift to the left can be noticed which is in agreement with the simulation results given in Figure 6.13. This can also be interpreted

as the strength of the p-channel device pulling up to V_{dd} is reduced. The simulation for the degraded MP1 is realized by changing the effective gate-source voltage of MP1 such that similar conditions to the aftermath of the experimental degradation are obtained.

Another aim of characterization is determining the threshold voltage shift due to the electrothermal stress applied to MP1. This has been estimated by using the simulation results in conjunction with measured data. However, one has to be careful in extracting this value since the simulated and measured voltage transfer curves cannot be overlapped. A more robust method of estimation is based on the calculation of mean square variations between the fresh and degraded curves. The difference between the voltage transfer curves with fresh and degraded MP1 for measured and simulated data are calculated for several voltage settings, then their squares are summed up and divided by the number of voltage settings employed. According to the results of this procedure, the experimental V_{th} change of MP1 is estimated to be 25 mV. A visual comparison of Figures 6.12 and 6.13 indicate that the V_{th} shift is closely estimated.

The NOR gate is characterized from two different perspectives. The first of them considers the degradation of MP1 and then recording the voltage transfer curve with V_A changing. The electrothermal stress setup is planned to be the same of that applied to the NAND gate. Voltage transfer curves of the NOR gate with the fresh and degraded MP1 where the signal V_A is changing is presented in Figure 6.14. As expected a shift of the voltage transfer curve to the left is observed. A similar approach is taken to determine the V_{th} degradation of MP1. The results show that the V_{th} change of MP1 is about 50 mV. This has been justified again by the simulation results depicted in Figure 6.15.

In the next experiment, the signal V_B is set to 0 V —thus MP2 is degraded— and temperature is held at 100°C for one hour. Afterwards, the voltage transfer curve as V_B changes is measured and compared with the curve recorded when MP2 was fresh.

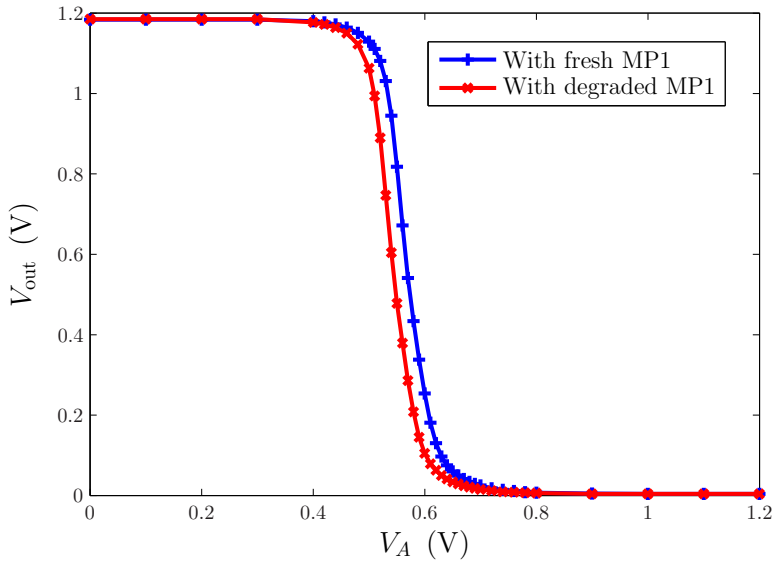


Figure 6.12: Voltage transfer characteristics of the NAND gate while the signal V_A is changing, before and after MP1 undergone the electrothermal stress.

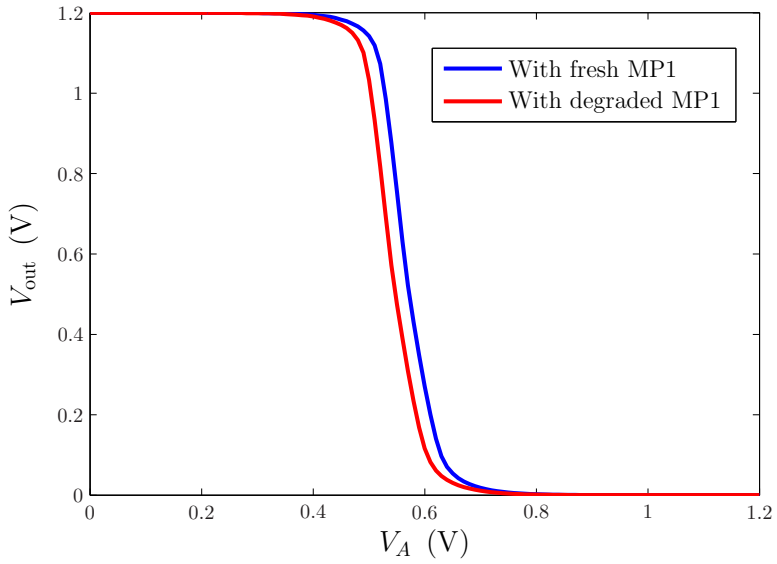


Figure 6.13: Simulated NAND voltage transfer characteristics with and without a 25 mV change in the effective gate-source voltage of MP1.

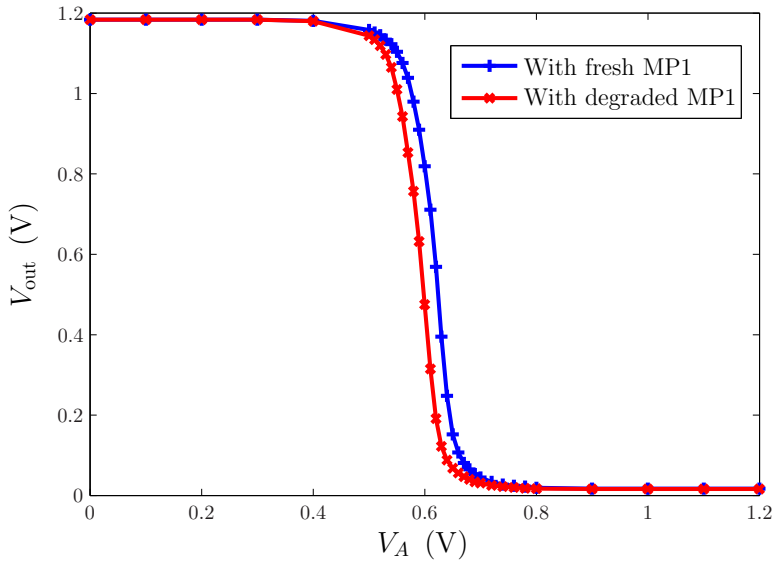


Figure 6.14: Voltage transfer characteristics of the NOR gate while V_A is changing, before and after MP1 undergone the electrothermal stress.

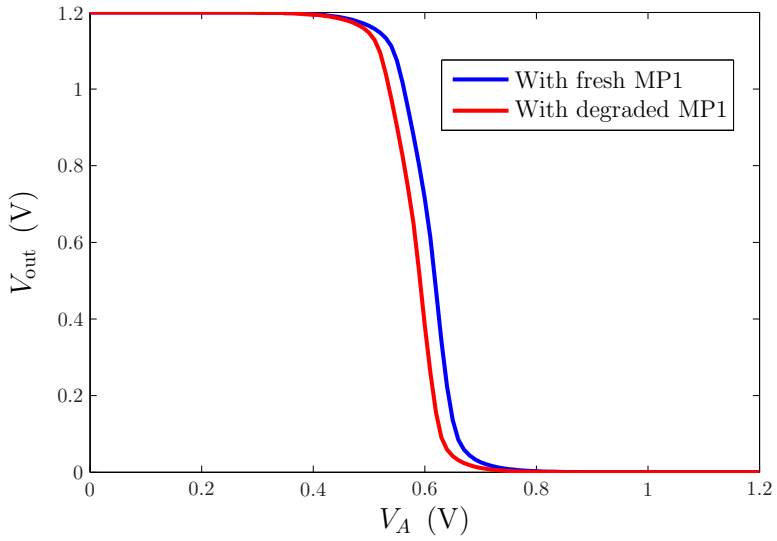


Figure 6.15: Simulated NOR voltage transfer characteristics with and without a 50 mV change in the effective gate-source voltage of MP1.

The results are shown in Figure 6.16. The degradation is much smaller (around 16 mV) compared to that of the voltage transfer curve when MP1 was degraded. The reason for that can be explained based on the body effect that MP2 was subject to. The body effect increases the threshold voltage and causes the V_{th} change to be smaller than that of MP1 where the body effect does not exist [94]. In addition, it should be noted that MP2 will only be affected by the NBTI degradation if MP1 is turned on. Otherwise, the effective gate-source voltage of MP2 becomes very small, which also prevents any possible NBTI degradation.

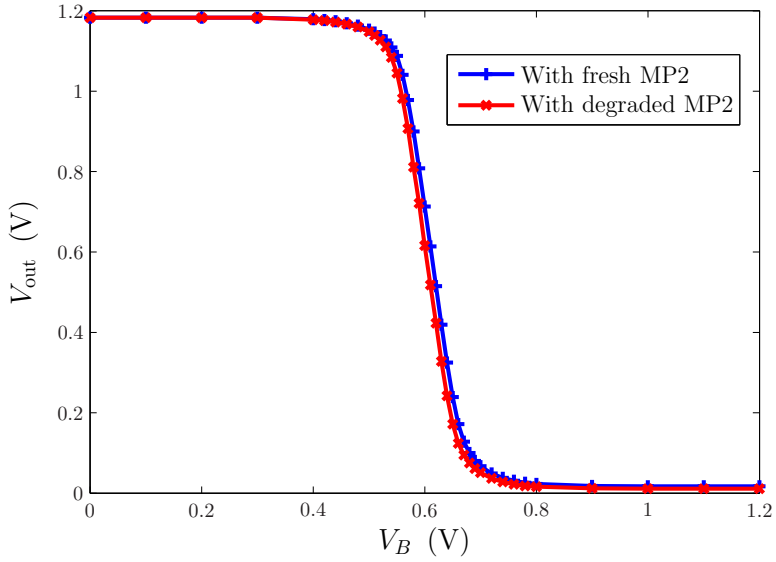


Figure 6.16: Voltage transfer characteristics of the NOR gate while V_B is changing, before and after MP2 undergone the electrothermal stress.

6.4 Summary

In this chapter, an NBTI monitor was introduced and the reliability characterization of a differential amplifier as well as two-input NAND and NOR gates were presented. The designed NBTI monitor is based up on the idea of having a common gate amplifier where its biasing is based on a p-channel device subject to NBTI degradation. The fabricated monitor is tested with electrothermal stress and the results have shown that the monitor operates as initially planned. A sensitivity of magnitude of 3.15 V^{-1} has been recorded as the gate-source voltage of the biasing PMOS device is changed. The gain of the common gate amplifier is decreasing linearly with increasing gate-source voltages.

Reliability characterization of the proposed differential amplifier demonstrated that the suggested topology is resilient to aging effects based on the electrothermal stress test results. The changes in the voltage transfer curves of NAND and NOR gates were comparable with the simulation results as well as prior work. It has also been demonstrated that the V_{th} increases due to the same electrothermal stressing procedure can be different based on the biasing conditions of the applied transistor as well as the topology of the circuit under testing.

Chapter 7

Conclusion and Future Work

In this chapter, basic conclusions to be drawn from the dissertation are provided in conjunction with the possible future work that may follow the studies presented in the dissertation.

7.1 Conclusion

This dissertation presented a complete overview to the variability and reliability concerns in nanoscale transistors. It made contributions in simulation, design, monitoring and characterization. All these components of the topic are important and they cannot be treated independently. Thus an integrated analysis methodology was chosen to provide the broadest perspective.

Reduced-order modeling techniques were described in Chapter 2. The flow of model development was shown for the case study of the model for the drain-source current, I_{ds}, of a MOSFET transistor. The surrogate modeling concept was introduced which established an accurate macromodel for a complex system with multiple inputs. A multidimensional

sampling scheme required for the surrogate model development was derived using a novel technique based on orthogonal Latin hypercube sampling.

Another outcome of this dissertation is the demonstration of reduced-order models used in the analysis and simulation of IC variability and reliability. In particular, the extent of within-die variations which were defined as mostly random fluctuations of major device process parameters can be well investigated by representing each transistor with a vector of crucial process parameters, terminal voltages, temperature and device age. The examples provided in Chapter 4 demonstrate that the reduced-order modeling approaches ranging from surrogate modeling to neural network-based modeling can successfully be applied in variability analysis of analog and digital circuits yielding sufficient accuracy. Different modeling strategies have been compared in Chapter 3 and have employed to perform circuit electrical simulation while variability and reliability effects on I_{ds} are quantified.

The reliability degradation of integrated circuits depend on many factors including the device age, and operational history of the transistor as well as the device process parameters. Existing models in the literature for 65 nm MOSFET transistors have been used to consolidate reduced-order model-based variability analysis. Design for reliability guidelines proposed in Chapter 5 aim to increase the durability of ICs to increased levels of electrical and thermal (electrothermal) stress.

The NBTI monitor described in Chapter 6 was based on an analog circuit concept —unlike the many previous digital counterparts—where the biasing voltage variation in a common gate amplifier is used to monitor the change in the voltage gain. The monitor was implemented and measured under electrothermal stress to validate the design concept. The characterization of basic building blocks attempted to quantify how much the high-level design quantities such as the voltage gain and voltage transfer characteristics change

with respect to a given level of electrothermal stress.

One important conclusion of the dissertation is how much the process variations and reliability degradations can change the design quantities of an analog or digital circuit building block. It has been shown that increased variability, in particular within-die variations could have a significant impact on the performance metrics of a circuit. Thus a crucial need for new modeling approaches is observed. These approaches should satisfy two conditions: Firstly, individual devices must be identified as separate vectors. This will allow an accurate description of within-die variations among individual devices. Secondly, the interactions of multiple parameters including process parameters, voltages, temperature and ages of devices should be considered together in this framework. Hence the reduced-order modeling is focused throughout the dissertation that has these capabilities.

The work in this dissertation also provided important insight in several aspects of variability and reliability of nanoscale circuits. A new application area for the reduced-order modeling techniques has been added where it can yield promising results. Comparison of Monte Carlo methods and reduced-order modeling approaches shows that the latter has significant advantages in estimating the extent of process variations including the interactions of individual device parameters. This property overlaps with the need of an accurate reliability analysis of nanoscale circuits since each transistor ages separately based on its process parameters, thermal characteristics and biasing. In analog circuits, the biasing can be used to improve the performance of the circuit which would be reduced due to aging. In particular, conventional circuit design techniques can be modified so that they can compensate for reliability degradations. Furthermore, the same idea makes it possible to construct aging monitors, which employ single stage amplifiers. The NBTI degradation results in changes in the biasing of these amplifiers leading to large drops

of their voltage gains. This observation indicates the importance of the biasing for the analog circuit reliability.

7.2 Future Work

There are several topics that require further research. Firstly, the reduced-order models implemented within the scope of this dissertation can be integrated to a full-scale circuit simulator instead of the search algorithms developed in MATLAB® environment. In this case, there will be a significant speed-up as evident from a single evaluation of the Kriging-based surrogate models shown in Chapter 4. Circuit simulators can be modified such that they use the surrogate models instead of lengthy analyses where the real speed-up can be observed. The model generation will still take place in MATLAB® with the developed methodology in Chapter 1. Then, the models can be transferred to the simulator environment and run there.

The models established in this dissertation are based on the DC drain current of a transistor. However other quantities can also be modeled using reduced-order techniques in a similar way. For example, reduced-order models of any design quantity that is subject to a large variance due to process characteristics of deep submicron devices may be built with the methodology described in this dissertation. These include harmonic intermodulation terms, noise figure, etc. of analog circuits. Building such models will contribute to the design of variability- and reliability-aware circuits.

As the modeling and characterization of time-dependent dielectric soft-breakdown and PBTI advance, analytical models developed should be incorporated into the analytical equations for I_{ds} as it has been done here for NBTI and HCI. Alternatively, surrogate modeling methodology can be used to construct models for several of these phenomena

as some of the characteristics observed cannot be explained by physics-based analytical expressions as done for NBTI and HCI. This enforces an empirical approach for predictive modeling. In this case, based on measurement data, reduced-order models can be generated that can anticipate the level of degradation with respect to input parameters. These models will be accurate and quick to evaluate so that they can be either integrated into a reliability simulator or used standalone.

The NBTI monitor can also be developed further such that the gain of the designed common gate amplifier is more sensitive to the V_{th} changes of the PMOS transistor in its biasing. This in turn would result in a higher magnitude of slope compared to the one presented in Figure 6.7. The next stage for the NBTI monitor is integrating it into different analog and mixed-signal circuits that are subject to major reliability degradation such as phase locked loops and analog-to-digital converters. Finally, the characterization of analog and digital building blocks should be undertaken in more advanced technology nodes, such as 32 nm and 22 nm, where the reliability degradations can be substantially large.

REFERENCES

[1] (2010) International technology roadmap for semiconductors, ITRS, 2010 update. [Online]. Available: http://www.itrs.net/links/2010itrs/2010Update/ToPost/2010_Update_Overview.pdf

[2] A. M. Pappu, X. Zhang, A. V. Harrison, and A. B. Apsel, "Process-invariant current source design: Methodology and examples," *IEEE J. Solid-State Circuits*, vol. 42, no. 10, pp. 2293–2302, Oct. 2007.

[3] C. Chiang and J. Kawa, *Design for Manufacturability and Yield for Nano-Scale CMOS*. Dordrecht, the Netherlands: Springer, 2007.

[4] S. Deyati and P. Mandal, "An automated design methodology for yield aware analog circuit synthesis in submicron technology," in *IEEE International Symposium on Quality of Electronic Design, ISQED, 2009*, Santa Clara, CA, USA, Mar. 2011, pp. 229–235.

[5] H. Masuda, S. Ohkawa, A. Kurokawa, and M. Aoki, "Challenge: Variability characterization and modeling for 65- to 90-nm processes," in *IEEE Custom Integrated Circuits Conference, CICC, 2005*, San Jose, CA, USA, Sep. 2005, pp. 593–599.

[6] N. Drego, A. Chandrakasan, and D. Boning, "All-digital circuits for measurement of spatial variation in digital circuits," *IEEE J. Solid-State Circuits*, vol. 45, no. 3, pp. 640–651, Mar. 2010.

[7] M. Orshansky, S. R. Nassif, and D. Boning, *Design for Manufacturability and Statistical Design: A Constructive Approach*. New York, USA: Springer, 2008.

[8] J. Cain and C. Spanos, "Electrical linewidth metrology for systematic CD variation characterization and causal analysis," in *Proceedings of SPIE*, 2003, pp. 350–361.

[9] M. Erckken, L. H. A. Leunissen, I. Pollentier, G. P. Patsis, V. Constantoudis, and E. Gogolides, "Effects of different processing conditions on line-edge roughness for 193 nm and 157 nm resists," in *Proceedings of SPIE*, 2004, pp. 266–275.

[10] N. Drego, A. Chandrakasan, and D. Boning, "Lack of spatial correlation in MOSFET threshold voltage variation and implications for voltage scaling," *IEEE Trans. Semicond. Manuf.*, vol. 22, no. 2, pp. 245–255, May 2009.

[11] S. S. Sapatnekar, "Overcoming variations in nanometer-scale technologies," *IEEE Transactions on Emerging and Selected Topics in Circuits and Systems*, vol. 1, no. 1, pp. 5–18, Mar. 2011.

[12] Y. M. Sheu, C. S. Chang, H. C. Lin, S. S. Lin, C. H. Lee, C. C. Wu, M. J. Chen, and C. H. Diaz, "Impact of STI mechanical stress in highly scaled MOSFETs," in *IEEE International Symposium on VLSI Technology, Systems, and Applications, 2003*, Hsinchu, Taiwan, Oct. 2003, pp. 76–79.

[13] M. Y. Kwong, R. Kasnavi, P. Griffin, J. D. Plummer, and R. W. Dutton, "Impact of lateral source/drain abruptness on device performance," *IEEE Trans. Electron Devices*, vol. 49, no. 11, pp. 1882–1890, Nov. 2002.

[14] W. Wang, V. Reddy, A. T. Krishnan, R. Vattikonda, S. Krishnan, and Y. Cao, "Compact modeling and simulation of circuit reliability for 65-nm CMOS technology," *IEEE Trans. Device Mater. Rel.*, vol. 7, no. 4, pp. 509–517, Dec. 2007.

[15] H. Wang, M. Miranda, F. Catthoor, and W. Dehaene, "Impact of random soft oxide breakdown on SRAM energy/delay drift," *IEEE Trans. Device Mater. Rel.*, vol. 7, no. 4, pp. 581–591, Dec. 2007.

[16] X. Li, J. Qin, and J. B. Bernstein, "Compact modeling of MOSFET wearout mechanisms for circuit-reliability simulation," *IEEE Trans. Device Mater. Rel.*, vol. 8, no. 1, pp. 98–121, 2008.

[17] J. B. Bernstein, M. Gurfinkel, X. Li, J. Walters, Y. Shapira, and M. Talmor, "Electronic circuit reliability modeling," *Elsevier Microelectronics Reliability*, vol. 46, pp. 1957–1979, 2006.

[18] S. K. Saha, "Modeling process variability in scaled CMOS technology," *IEEE Des. Test. Comput.*, vol. 27, no. 2, pp. 8–16, 2010.

[19] *Virtuoso Analog Design Environment XL User Guide Product Version 6.1.2*, Cadence Design Systems, Oct. 2007.

[20] A. A. Mutlu and M. Rahman, "Statistical methods for the estimation of process variation effects on circuit operation," *IEEE Trans. Electron. Packag. Manuf.*, vol. 28, no. 4, pp. 364–375, Oct. 2005.

[21] S. Shadabale, H. Ramakrishnan, G. Russell, A. Yakovlev, and S. Chattopadhyay, "Statistical modelling of the variation in advanced process technologies using a multi-level partitioned response surface approach," *IET Circuits, Devices & Systems*, vol. 2, no. 5, pp. 451–464, Oct. 2008.

[22] D. Basu, J. Guha, P. Hatab, P. Vaidyanathan, C. Mouli, and S. K. Groothuis, "Using TCAD, response surface model and Monte Carlo methods to model processes and reduce device variation," in *IEEE International Conference on Simulation of Semiconductor Process and Devices, SISPAD, 2009*, San Diego, CA, USA, Sep. 2009, pp. 1–4.

[23] X. Li, Y. Zhan, and L. T. Pileggi, "Quadratic statistical *MAP* approximation for parametric yield estimation of analog/RF integrated circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 27, no. 5, pp. 831–843, May 2008.

[24] V. Wang and D. Markovic, "Linear analysis of random process variability," in *IEEE/ACM International Conference on Computer-Aided Design, ICCAD, 2008*, San Jose, CA, USA, Nov. 2008, pp. 292–296.

[25] E. Rémond, E. Nercessian, C. Bernicot, and R. Mina, "Mathematical approach based on a 'design of experiment' to simulate process variations," in *Design, Automation & Test in Europe Conference & Exhibition, DATE, 2011*, Grenoble, France, Mar. 2011, pp. 1–5.

[26] H. Filiol, I. O'Connor, and D. Morche, "A new approach for variability analysis of ICs," in *Proc. IEEE NorthEast Workshop on Circuits and Systems — Analog Information and Signal Processing, and its Applications, NEWCAS-TAISA, 2009*, Toulouse, France, Jun. 2009, pp. 1–4.

[27] M. Chen, W. Zhao, F. Liu, and Y. Cao, "Finite-point-based transistor model: A new approach to fast circuit simulation," *IEEE Trans. VLSI Syst.*, vol. 17, no. 10, pp. 1470–1480, Oct. 2009.

[28] B. Cheng, A. R. Brown, S. Roy, and A. Asenov, "PBTI/NBTI-related variability in TB-SOI and DG MOSFETs," *IEEE Electron Device Lett.*, vol. 31, no. 5, pp. 408–410, May 2010.

[29] D. Boning, K. Balakrishnan, H. Cai, N. Drego, A. Faranchi, K. M. Gettings, D. Kim, A. Somani, H. Taylor, D. Truque, and X. Xie, "Variation," *IEEE Trans. Semicond. Manuf.*, vol. 21, no. 1, pp. 63–71, Feb. 2008.

[30] P.-I. Mak and R. P. Martins, "High-/mixed-voltage RF and analog CMOS circuits come of age," *IEEE Circuits Syst. Mag.*, vol. 10, no. 4, pp. 27–39, Nov. 2010.

[31] S. Sen and A. Chatterjee, "Design of process variation tolerant radio frequency low noise amplifier," in *IEEE International Symposium on Circuits and Systems, ISCAS*, Seattle, WA, USA, May 2008, pp. 392–395.

[32] D. Gómez, M. Sroka, and J. L. G. Jiménez, "Process and temperature compensation for RF low-noise amplifiers and mixers," *IEEE Trans. Circuits Syst. I*, vol. 57, no. 6, pp. 1204–1211, Jun. 2010.

[33] J. S. Yuan and H. Tang, "CMOS RF design for reliability using adaptive-gate source biasing," *IEEE Trans. Electron Devices*, vol. 55, no. 9, pp. 2348–2353, Sep. 2008.

[34] N. Kupp, H. Huang, Y. Makris, and P. Drineas, "Improving analog and RF device yield through performance calibration," *IEEE Des. Test. Comput.*, vol. 28, no. 3, pp. 64–75, 2011.

[35] G. Yu and P. Li, "Hierarchical analog/mixed-signal circuit optimization under process variations and tuning," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 2, pp. 313–317, Feb. 2011.

[36] G. Gielen, E. Maricau, and P. D. Wit, "Analog circuit reliability in sub-32 nanometer CMOS: Analysis and mitigation," in *Design, Automation & Test in Europe Conference & Exhibition, DATE, 2011*, Grenoble, France, Mar. 2011, pp. 1–6.

[37] X. Zhou, G. J. Zhu, S. H. Lin, Z. H. Chen, M. K. Srikanth, Y. F. Yan, R. Selvakumar, W. Chandra, J. B. Zhang, C. Q. Wei, and P. Bathla, "Subcircuit approach to inventive compact modeling for CMOS variability and reliability," in *IEEE* 12^{th} *International Symposium on Integrated Circuits, ISIC, 2009*, Singapore, Singapore, Dec. 2009, pp. 133–136.

[38] A. Forrester, A. Sobester, and A. Keane, *Engineering Design via Surrogate Modelling*. West Sussex, UK: John Wiley & Sons, 2008.

[39] D. Gorissen, L. D. Tommasi, W. Hendrickx, J. Croon, and T. Dhaene, "RF circuit block modeling via Kriging surrogates," in 17^{th} *International Conference on Microwaves, Radar and Wireless Communications, MIKON, 2008*, Wroclaw, Poland, May 2008, pp. 1–4.

[40] S. Koziel and J. Bandler, "Accurate modeling of microwave devices using kriging-corrected space-mapping surrogates," *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, 2011, to be published.

[41] J. W. Bandler, Q. S. Cheng, S. A. Dakroury, A. S. Mohamed, M. H. Bakr, K. Madsen, and J. Søndergaard, "Space mapping: The state of the art," *IEEE Trans. Microw. Theory Tech.*, vol. 52, no. 1, pp. 337–361, Jan. 2004.

[42] S. Koziel and J. Bandler, "Recent advances in space-mapping-based modeling of microwave devices," *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 23, no. 6, pp. 425–446, 2010.

[43] F. Declercq, I. Couckuyt, H. Rogier, and T. Dhaene, "Complex permittivity characterization of textile materials by means of surrogate modeling," in *IEEE Antennas and Propagation Society International Symposium, APSURSI, 2010*, Toronto, ON, Canada, 2010, pp. 1–4.

[44] T. Zhu and P. D. Franzon, "Application of surrogate modeling to generate compact and PVT-sensitive IBIS models," in *IEEE* 18^{th} *Conference on Electrical Performance of Electronic Packaging and Systems, EPEPS, 2009*, Portland, OR, USA, Oct. 2009, pp. 77–80.

[45] B. Liu, Y. He, P. Reynaert, and G. Gielen, "Global optimization of integrated transformers for high frequency microwave circuits using a Gaussian process based surrogate model," in *Design, Automation & Test in Europe Conference & Exhibition, DATE, 2011*, Grenoble, France, Mar. 2011, pp. 1–6.

[46] J. E. Ráyas-Sanchez, J. Aguliar-Torrentera, and J. A. Jasso-Urzúa, "Surrogate modeling of microwave circuits using polynomial function interpolants," in *IEEE International Microwave Symposium, IMS, 2010*, Anaheim, CA, USA, May 2010, pp. 197–200.

[47] T. H. Morshed, W. Yang, M. V. Dunga, X. Xi, J. He, W. Liu, K. Yu, M. Cao, X. Jin, J. J. Ou, M. Chan, A. M. Niknejad, and C. Hu. (2009, Apr.) BSIM4.6.4 MOSFET model- User's manual. [Online]. Available: http://www-device.eecs.berkeley.edu/~bsim3/BSIM4/BSIM464/BSIM464_Manual.pdf/

[48] N. V. Queipo, R. T. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, and P. K. Tucker, "Surrogate-based analysis optimization," *Elsevier Progress in Aerospace Sciences*, vol. 41, pp. 1–28, 2005.

[49] Y. Tsividis and C. McAndrew, *Operation and Modeling of the MOS Transistor*, 3rd ed. New York, USA: Oxford University Press, 2010.

[50] R. J. Baker, *CMOS Circuit Design Layout and Simulation*, 2nd ed. Hoboken, NJ, USA: IEEE Wiley Press, 2008.

[51] A. S. Roy, S. P. Mudanai, and M. Stettler, "Mechanism of long-channel drain-induced barrier lowering in halo MOSFETs," *IEEE Trans. Electron Devices*, vol. 58, no. 4, pp. 979–984, Apr. 2011.

[52] C.-H. Lin, M. V. Dunga, D. D. Lu, A. M. Niknejad, and C. Hu, "Performance-aware corner model for design for manufacturing," *IEEE Trans. Electron Devices*, vol. 56, no. 4, pp. 595–600, Apr. 2009.

[53] M. Orshansky, J. Chen, and C. Hu, "Direct sampling methodology for statistical analysis of scaled CMOS technologies," *IEEE Trans. Semicond. Manuf.*, vol. 12, no. 4, pp. 403–408, Nov. 1999.

[54] S. Realov, W. McLaughlin, and K. L. Shepard, "On-chip transistor characterization arrays with digital interfaces for variability characterization," in *IEEE International Symposium on Quality of Electronic Design, ISQED, 2009*, San Jose, CA, USA, Mar. 2007, pp. 167–171.

[55] C. Zhuo, K. Agarwal, D. Blauuw, and D. Sylvester, "Active learning framework for post-silicon variation extraction and test-cost reduction," in *IEEE/ACM International Conference on Computer-Aided Design, ICCAD, 2010*, San Jose, CA, USA, Nov. 2010, pp. 508–515.

[56] M. D. McKay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979.

[57] M. D. Morris and T. J. Mitchell, "Exploratory designs for computer experiments," *J. Statist. Plann. Inference*, vol. 43, no. 3, pp. 381–402, 1995.

[58] M. E. Johnson, L. M. Moore, and D. Ylvisaker, "Minimax and maximin distance designs," *J. Statist. Plann. Inference*, vol. 26, no. 2, pp. 131–148, 1990.

[59] B. Tang, "Orthogonal array-based latin hypercubes," *J. Amer. Statist. Assoc.*, vol. 88, no. 424, pp. 1392–1397, 1993.

[60] G. Yu and P. Li, "Yield-aware analog integrated circuit optimization using geostatics motivated performance modeling," in *IEEE/ACM International Conference on Computer-Aided Design, ICCAD, 2007*, San Jose, CA, USA, Nov. 2007, pp. 464–469.

[61] H. You, M. Yang, D. Wang, and X. Jia, "Kriging model combined with Latin hypercube sampling for surrogate modeling of analog integrated circuit performance," in *IEEE International Symposium on Quality of Electronic Design, ISQED, 2009*, San Jose, CA, USA, Mar. 2009, pp. 554–558.

[62] J. D. Martin, "Computational improvements to estimating Kriging metamodel parameters," *ASME Journal of Mechanical Design*, vol. 131, no. 084501, pp. 1–7, 2009.

[63] S. N. Lophaven, H. B. Nielsen, and J. Søndergaard. (2002, Aug.) A MATLAB Kriging toolbox version 2.0. [Online]. Available: http://www2.imm.dtu.dk/~hbn/dace/dace.pdf

[64] S. Haykin, *Neural Networks and Learning Machines*, 3rd ed. New Jersey, USA: Pearson, 2009.

[65] N. Kahraman and T. Yildirim, "Technology independent circuit sizing for fundamental analog circuits using artificial neural networks," in *Ph.D. Research in Microelectronics and Electronics, PRIME, 2008*, Istanbul, Turkey, Jun. 2008, pp. 1–6.

[66] F. Mkadem and S. Boumaiza, "Physically inspired neural network model for RF power amplifier behavioral modeling and digital predistortion," *IEEE Trans. Microw. Theory Tech.*, vol. 59, no. 4, pp. 913–923, Apr. 2011.

[67] Z. N. Mevawalla, G. S. May, and M. W. Kiehlbauch, "Neural network modeling for advanced process control using production data," *IEEE Trans. Semicond. Manuf.*, vol. 24, no. 2, pp. 182–189, May 2011.

[68] Q. Zhang, J. Bandler, S. Koziel, H. Kabir, and L. Zhang, "ANN and space mapping for microwave modelling and optimization," in *IEEE International Microwave Symposium, IMS, 2010*, Anaheim, CA, USA, May 2010, pp. 980–983.

[69] L. Zhang, Q.-J. Zhang, and J. Wood, "Statistical neuro-space mapping technique for large-signal modeling of nonlinear devices," *IEEE Trans. Microw. Theory Tech.*, vol. 56, no. 11, pp. 2453–2467, Nov. 2008.

[70] J. A. K. Suykens, T. V. Gestel, J. D. Brabanter, B. D. Moor, and J. Vandewalle, *Least Squares Support Vector Machines*. Singapore: World Scientific, 2002.

[71] T. Kiely and G. Gielen, "Performance modeling of analog integrated circuits using least-squares support vector machines," in *Design, Automation & Test in Europe Conference & Exhibition, DATE, 2004*, Paris, France, Feb. 2004, pp. 448–453.

[72] M. Ding and R. Vemuri, "A combined feasibility and performance macromodel for analog circuits," in 42^{nd} *ACM/IEEE Design Automation Conference, DAC, 2005*, Anaheim, CA, USA, Jun. 2005, pp. 63–68.

[73] V. Čeperić and A. Barič, "Modeling of analog circuits by using support vector regression machines," in 11^{th} *IEEE International Conference on Electronics, Circuits and Systems, ICECS, 2004*, Tel Aviv, Israel, Dec. 2004, pp. 391–394.

[74] X. Ren and T. Kazmierski, "Performance modelling and optimization of RF circuits using support vector machines," in 14^{th} *International Conference Mixed Design of Integrated Circuits and Systems, MIXDES, 2007*, Ciechocinek, Poland, Jun. 2007, pp. 317–321.

[75] D. Gorissen, L. D. Tommasi, K. Crombecq, and T. Dhaene, "Sequential modeling of a low noise amplifier with neural networks and active learning," *Springer Neural Computing and Applications*, vol. 18, no. 5, pp. 485–494, 2009.

[76] D. W. Stephens, D. Gorissen, K. Crombecq, and T. Dhaene, "Surrogate based sensitivity analysis of process equipment," *Springer Applied Mathematical Modelling*, vol. 35, no. 4, pp. 1676–1687, Apr. 2011.

[77] K. D. Brabanter, P. Karsmakers, F. Ojeda, C. Alzate, J. D. Brabanter, K. Pelckmans, B. D. Moor, J. Vandewalle, and J. A. K. Suykens. (2010, Sep.) LS-SVMlab toolbox version 1.7. [Online]. Available: http://www.esat.kuleuven.be/sista/lssvmlab/

[78] D. G. Luenberger and Y. Ye, *Linear and nonlinear programming*, 3rd ed. New York City, NY, USA: Springer, 2008.

[79] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits — A Design Perspective*, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2003.

[80] W. Wang, S. Yang, S. Bhardwaj, S. Vrudhala, F. Liu, and Y. Cao, "The impact of NBTI effect on combinational circuit: Modeling, simulation, and analysis," *IEEE Trans. VLSI Syst.*, vol. 18, no. 2, pp. 173–183, Feb. 2010.

[81] L. L. Lewyn, "Physical design and reliability issues in nanoscale analog CMOS technologies," in 27^{th} *The Nordic Microelectronics Event, NORCHIP Conference, 2009*, Trondheim, Norway, Nov. 2009, pp. 1–10.

[82] H. Kufluoglu and M. A. Alam, "Theory of interface-trap-induced NBTI degradation for reduced cross section MOSFETs," *IEEE Trans. Electron Devices*, vol. 53, no. 5, pp. 1120–1130, May 2006.

[83] D. P. Ioannou, S. Mittl, and G. L. Rosa, "Positive bias temperature instability effects in nMOSFETs with HfO_2/TiN gate stacks," *IEEE Trans. Device Mater. Rel.*, vol. 9, no. 2, pp. 128–134, Jun. 2009.

[84] R. Tu, E. Rosenbaum, W. Chan, C. Li, E. Minami, K. Quader, P. Ko, and C. Hu, "Berkeley reliability tools-BERT," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 12, no. 10, pp. 1524–1534, Oct. 1993.

[85] X. Xuan, A. Chatterjee, A. D. Singh, N. P. Kim, and M. T. Chisa, "IC reliability simulator ARET and its application in design-for-reliability," in *IEEE 12th Asian Test Symposium, ATS, 2003*, Xian, China, Mar. 2003, pp. 18–21.

[86] F. Marc, B. Mongellaz, C. Bestory, H. Levi, and Y. Danto, "Improvement of aging simulation of electronic circuits using behavioral modeling," *IEEE Trans. Device Mater. Rel.*, vol. 6, no. 2, pp. 228–234, Jun. 2006.

[87] X. Li, J. Qin, B. Huang, X. Zhang, and J. B. Bernstein, "A new SPICE reliability simulation method for deep submicrometer CMOS VLSI circuits," *IEEE Trans. Device Mater. Rel.*, vol. 6, no. 2, pp. 247–257, 2006.

[88] B. Yan, J. Qin, J. Dai, Q. Fan, and J. B. Bernstein, "Reliability simulation and circuit-failure analysis in analog and mixed-signal applications," *IEEE Trans. Device Mater. Rel.*, vol. 9, no. 3, pp. 339–347, Sep. 2009.

[89] E. Maricau and G. Gielen, "Efficient reliability simulation of analog ICs including variability and time-varying stress," in *Design, Automation & Test in Europe Conference & Exhibition, DATE, 2009*, Nice, France, Apr. 2009, pp. 1238–1241.

[90] ——, "Variability-aware reliability simulation of mixed-signal ICs with quasi-linear complexity," in *Design, Automation & Test in Europe Conference & Exhibition, DATE, 2010*, Dresden, Germany, Mar. 2010, pp. 1094–1099.

[91] E. Y. Wu and J. Suñé, "Power-law voltage acceleration: A key element for ultra-thin gate oxide reliability," *Elsevier Microelectronics Reliability*, vol. 45, no. 12, Nov.

[92] E. Maricau and G. Gielen, "Stochastic circuit reliability analysis," in *Design, Automation & Test in Europe Conference & Exhibition, DATE, 2011*, Grenoble, France, Mar. 2011, pp. 1–6.

[93] J. Martin-Martinez, R. Rodriguez, M. Nafria, and X. Aymerich, "Time-dependent variability relayed to BTI effects in MOSFETs: Impact on CMOS differential amplifiers," *IEEE Trans. Device Mater. Rel.*, vol. 9, no. 2, pp. 305–310, Jun. 2009.

[94] R. Vattikonda, W. Wang, and Y. Cao, "Modeling and minimization of PMOS NBTI effect for robust nanometer design," in 43^{rd} *ACM/IEEE Design Automation Conference, DAC, 2006*, San Francisco, CA, USA, Jul. 2006, pp. 1047–1052.

[95] Z. Qi and M. R. Stan, "NBTI resilient circuits using adaptive body biasing," in *Proc.* 18^{th} *ACM Great Lakes Symposium on VLSI, GLSVLSI, 2008*, Orlando, FL, USA, May 2008, pp. 285–290.

[96] E. Mintarno, J. Skaf, R. Zheng, J. B. Velamela, Y. Cao, S. Boyd, R. W. Dutton, and S. Mitra, "Self-tuning for maximized lifetime energy-efficiency in the presence of circuit aging," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 5, pp. 760–773, May 2011.

[97] A. Ghosh, R. Franklin, and R. B. Brown, "Analog circuit design methodologies to improve negative-bias temperature instability degradation," in 23^{rd} *International Conference on VLSI Design, VLSID, 2010*, Bangalore, India, Jan. 2010, pp. 464–469.

[98] B. Dubois, J.-B. Kammerer, L. Hebrard, and F. Braun, "Modelling of hot-carrier degradation and its application for analog design for reliability," *Elsevier Microelectronics Journal*, vol. 40, no. 9, pp. 1274–1280, Sep. 2009.

[99] S. Cimino, L. Pantisano, M. Aoulaiche, R. Degraeve, D. H. Kwak, F. Crupi, and G. Groeseneken, "Hot carrier degradation on n-channel HfSiOn MOSFETs: Effects on the device performance and lifetime," in 43^{rd} *IEEE International Reliability Physics Symposium, IRPS, 2005*, San Jose, CA, USA, Apr. 2005, pp. 275–279.

[100] E. Amat, T. Kauerauf, R.Degraeve, R. Rodriguez, M. Nafria, X. Aymerich, and G. Groeseneken, "Simulation of the hot-carrier degradation in short channel transistor with high-k dielectric," *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 23, no. 4-5, pp. 315–323, Jul. 2010.

[101] E. Maricau, P. D. Wit, and G. Gielen, "An analytical model for hot carrier degradation in nanoscale CMOS suitable for the simulation of degradation in analog IC applications," *Elsevier Microelectronics Reliability*, vol. 48, pp. 1576–1580, 2008.

[102] N. K. Jha, P. S. Reddy, D. K. Sharma, and V. R. Rao, "NBTI degradation and its impact for analog circuit reliability," *IEEE Trans. Electron Devices*, vol. 52, no. 12, pp. 2609–2615, Dec. 2005.

[103] X. Li, J. Qin, B. Huang, X. Zhang, and J. B. Bernstein, "A new SPICE reliability simulation method for deep submicrometer CMOS VLSI circuits," *IEEE Trans. Device Mater. Rel.*, vol. 6, no. 2, pp. 247–257, 2006.

[104] P. Singh, E. Karl, D. Sylvester, and D. Blaauw, "Dynamic NBTI management using a 45 nm multi-degradation sensor," in *Custom Integrated Circuits Conference, CICC, 2010*, San Jose, CA, USA, Sep. 2010, pp. 1–4.

[105] K. K. Kim, W. Wang, and K. Choi, "On-chip aging sensor circuits for reliable nanometer MOSFET digital circuits," *IEEE Trans. Circuits Syst. II*, vol. 57, no. 10, pp. 798–802, Oct. 2010.

[106] J. Keane, T.-H. Kim, and C. H. Kim, "An on-chip NBTI sensor for measuring pMOS threshold voltage degradation," *IEEE Trans. VLSI Syst.*, vol. 18, no. 6, pp. 947–956, Jun. 2010.

[107] A. Ghosh, R. B. Brown, R. M. Rao, and C.-T. Chuang, "A precise negative bias temperature instability sensor using slew-rate monitor circuitry," in *IEEE International Symposium on Circuits and Systems, ISCAS*, Taipei, Taiwan, May 2009, pp. 381–384.

[108] M. Omaña, D. Rossi, N. Bosio, and C. Metra, "Self-checking monitor for NBTI due degradation," in *IEEE* 16^{th} *International Mixed-Signals, Sensors and Systems Test Workshop, IMS3TW*, La Grande Motte, France, 2010, pp. 1–6.

[109] J. Vazquez, V. Champac, A. Ziesemer, R. Reis, J. Semiao, I. Teixeria, M. Santos, and J. Texeira, "Predictive error detection by on-line aging monitoring," in *IEEE 15th On-Line Testing Symposium, IOLTS, 2009*, Sesimbra, Lisbon, Jun. 2009, pp. 9–14.

[110] A. C. Cabe, Z. Qi, S. N. Wooters, T. N. Blalock, and M. Stan, "Small embedddable NBTI sensors (SENS) for tracking on-chip performance decay," in *IEEE International Symposium on Quality of Electronic Design, ISQED, 2009*, San Jose, CA, USA, Mar. 2009, pp. 1–6.

[111] H. F. Dadgour and K. Banerjee, "Built-in aging monitoring for safety-critical applications," in 48^{th} *IEEE International Reliability Physics Symposium, IRPS, 2005*, Anaheim, CA, USA, May 2010, pp. 822–825.

[112] J. Martin-Martinez, S. Gerardin, E. Amat, R. Rodriguez, M. Nafria, X. Aymerich, A. Paccagnella, and G. Ghidini, "Channel-hot-carrier degradation and bias temperature instabilities in CMOS inverters," *IEEE Trans. Electron Devices*, vol. 56, no. 9, pp. 2155–2159, Sep. 2009.

[113] N. Berbel, R. Fernandez, and I. Gil, "Modelling and experimental verification of the impact of negative bias temperature instability on CMOS inverter," *Elsevier Microelectronics Reliability*, vol. 49, pp. 1048–1051, 2009.

[114] T. Quémerais, L. Moquillon, V. Huard, J.-M. Fournier, P. Benech, N. Corrao, and X. Mescot, "Hot-carrier stress effect on a CMOS 65-nm 60-GHz one-stage power amplifier," *IEEE Electron Device Lett.*, vol. 31, no. 9, pp. 927–929, Sep. 2010.

APPENDICES

Appendix A

MATLAB® Codes for Building A Surrogate Model

A.1 Introduction

In this appendix, the codes for developing a surrogate model are presented. All codes have been provided with extensive comments which describe the methodology pursued in individual functions.

The main program to run is given in the beginning. All the subsequent functions are following the main program in alphabetical order. Also, in order to run the functions properly, the MATLAB® toolbox DACE should be installed and set up correctly.

The function "ids_nmos" is left to the choice of the user since it mostly depends on the specific details of the device model file. So, it is sufficient to write a program with that name and put it in the same directory with the functions given in this appendix.

Section A.2 provides the main function as well as the functions "assigner", "assigner_new", "bestplan", "chan", "comparator", "evaluater_gm", "lhc", "ortcheck", "ortnum",

"packet_n", "RRSE", "samplecreate", and "summa". Section A.3 presents two screen dumps while the codes provided in Section A.2 are operated in MATLAB® environment.

A.2 Commented MATLAB® Codes

```
%This is the main program to run for the surrogate model development.

clear;
clc;

%The theta values which can be described as the Kriging basis function parameter for each
    variable is limited by the upper and lower bounds. The initial value vector for the
    theta set is a vector of 1s.

LowerTheta=[1e-4 1e-4 1e-4 1e-4 1e-4 1e-4 1e-4 1e-4 1e-4 1e-4 1e-4];
theta0=[1 1 1 1 1 1 1 1 1 1 1];
UpperTheta=[20 20 20 20 20 20 20 20 20 20 20];

theta(1,:) =theta0;

%Initalization
resultat =zeros(3,3);

%Lower and upper limits of the variables.

%First element: Temperature T= T (in K)/10. ex. For room temperature use 30.
%Second element: Oxide thickness tox= tox (in nm) * 1e10 ex. 5nm corresponds to 50
%Third element: Channel doping Nch= Nch (in 1/cm^3) * 1e-16
%Fourth element: Effective length offset xl= xl (in nm) * 1e9
%Fifth element: Vbs = Vbs (in V) * 10 (for pfet Vsb)
%Sixth element: Vth0= Vth0 (in V) * 100 (for pfet |Vth0|)
%Seventh element: Intrinsic mobility for nfet u0= u0 (in cm^2(Vs)^-1)* 0.1
%for pfet u0= u0 (in cm^2(Vs)^-1)
%Eighth element: Vgs= Vgs (in V) * 10 (for pfet Vsg)
%Ninth element: Vds= Vds (in V) * 10 (for pfet Vsd)
%Tenth element: Drain-source resistance Rds= Rds (in Ohm) (for pfet Rsd)
%Eleventh element: Aging time
%for 0 to 8 months -> 0 to 2000 (model multiplies it with 1e4 to convert to seconds)
%for 8 months to 10 years -> 1 to 33 (model multiplies it with 1e7 to convert to seconds)

lower_limit =[T_ll, tox_ll, Nch_ll, xl_ll, Vbs_ll, Vth0_ll, u0_ll, Vgs_ll, Vds_ll,
    RDSW_ll, time_age_ll];
upper_limit=[T_ul, tox_ul, Nch_ul, xl_ul, Vbs_ul, Vth0_ul, u0_ul, Vgs_ul, Vds_ul,
    RDSW_ul, time_age_ul];
```

```
%‘‘N” is the number of sample vectors
N=2880;
%‘‘M” is the number of small groups that build up the whole list of sample vectors.
    Generally, it is preferred to chose N=M*k where k is an integer.
M=64;
%Maximum number of loops until the final transistor drain current model is created.
th=100;
%‘‘div” corresponds to k described above.
div=N/M;
%The ‘‘samplefile” will hold all input samples.
samplefile =[];

%This is the number of Latin squares to be compared using the orthogonality number and
    the minimax theorem. More info can be found in the function of ‘‘bestplan.m”.
z=5;
%Total number of existing variables.
dim=11;

%‘‘ dil_i ” and ‘‘ dil_j ” are the number of pieces in x and y directions of a Latin square.
    It should be kept in mind that M needs to be chosen such that M>=dil_i*dil_j is
    satisfied .
dil_i =4;
dil_j =4;

%Number of sample vectors that represents the variation in each piece of the created Latin
    square, i.e. each piece in a Latin square of length n has to have nominally
    x=round(N/(dil_i*dil_j)) number of samples. However to ease the process of finding a
    Latin square, x can vary from x−v to x+v.
v=1;
direc=1;

%This parameter keeps track of the the number of loops where no improvement in terms of
    the model accuracy is observed when a new sample from the list of test vectors is
    allocated . The criteria for the model accuracy is chosen to be the maximum error
    between the surrogate model outputs and the true model outcomes for the randomly
    selected test vectors.
sonmax=80;

turn_flag =1;

%Sample vector space is constructed.
```

```
for divpar=1:div
    tempfile=packet_n(M,dim,z,dil_i,  dil_j ,  v);
    if (divpar>1)
        comparison=comparator(tempfile, samplefile, dim);
        while (comparison==0)
            tempfile=packet_n(M,dim,z,dil_i,  dil_j ,  v);
            comparison=comparator(tempfile, samplefile, dim);
        end
    end
    samplefile=[samplefile;tempfile];
end

s=N;

%Sample vectors are evaluated and the corresponding drain current values are  calculated.

[x,  y]=assigner(M, N, dim, lower_limit,  upper_limit,  samplefile);

inp=x';
otp=y';

cnt=1;

%The parameter ''kor" can go from 1 to 3 where 1 is corresponding to constant factors
    regression, 2 is corresponding to linear factors regression, and 3 is corresponding to
    the second-order regresssion.

%Here constant factors regression is considered only.

for kor=1:1:1
    for p=1:length(theta(:,1))
        theta0=theta(p,:);

        %''kez" and ''son" keep the track of the current loop number and the number of
            non-improvement loops, respectively and they go from 1 to ''th" and 1 to
            ''sonmax"

        kez=1;
        son=1;

        %''maxmax" will hold the maximum error value between the true model and
            surrogate model outputs for each loop.
        maxmax=zeros(1,th);
```

```
inp=x';
otp=y';

while (kez<th) && (son<sonmax)

    if (kez==1)
        sigma=s;
    end

    %Model fitting for constant(regpoly0), linear (regpoly1) and second-order
        (regpoly2) regression functions.

    if (kor==1)
        [dmodel, perf]= dacefit(inp, otp, @regpoly0, @corrgauss, theta0,
            LowerTheta, UpperTheta);
    end

    if (kor==2)
        [dmodel, perf]= dacefit(inp, otp, @regpoly1, @corrgauss, theta0,
            LowerTheta, UpperTheta);
    end

    if (kor==3)
        [dmodel, perf]= dacefit(inp, otp, @regpoly2, @corrgauss, theta0,
            LowerTheta, UpperTheta);
    end

    % The established models are checked for their accuracy.

    [diff, posvec]=evaluater_gm(M, N, dim,dmodel, samplefile, lower_limit,
        upper_limit);

    h=length(diff);

    %The maximum of the error vector is found.

    maxvalue=max(max(max((max(max(max(diff)))))));

    %In the first loop no comparison is done and all the results are directly
        accepted.

    if (kez==1)
```

```
        diff_opt =diff;
        dmodel_opt=dmodel;
        inp_opt=inp;
        otp_opt=otp;
    end

    %After the first loop, the maximum value of the error vector is compared with
        the ones of the previous runs. If it is better (smaller) than all, that
        group of samples are written to be optimum -thus ''inp_opt", ''otp_opt",
        etc.-. ''son" is reduced to 1 and starts counting from that group of
        sample vectors. If the maximum of the error vector is larger than any of
        the previously recorded maximum error values, than that loop becomes a
        non-improvement loop and ''son" is increased by 1.

    if (kez>1)
        if (maxvalue< min(maxmax))
            inp_opt=inp;
            otp_opt=otp;
            diff_opt =diff;
            dmodel_opt=dmodel;
            son=1;
        else
            son=son+1;
        end
    end

    %The test sample that results in the maximum error is added as a new input to
        the existing list of sample vectors.

    [inp_new, otp_new]=assigner_new(kez, dim, lower_limit, upper_limit,
        samplefile, posvec,N);

    sigma=sigma+1;
    inp(sigma,:)=inp_new;
    otp(sigma)=otp_new;

    maxmax(kez)=maxvalue;
    theta0=dmodel.theta;
    kez=kez+1;
end

maxdiff=max(diff_opt);
```

```
        %Average and root mean square error between the surrogate model and true model
            results are computed and recorded.

        avg_error=0;
        rms_error=0;

        for i=1:N
            avg_error=avg_error+diff_opt(i);
            rms_error=rms_error+diff_opt(i)^2;
        end

        avg_error=avg_error/(N);

        rms_error=sqrt(rms_error/(N));

        resultat(cnt,1)=avg_error;
        resultat(cnt,2)=rms_error;
        resultat(cnt,3)=maxdiff;

        %Out of constant, linear and the second-order regression options the one that has
            the least maximum error is chosen.

        if (cnt==1)
            best_option=resultat(cnt,3);
            best_dmodel2=dmodel_opt;
            best_inp=inp_opt;
            best_otp=otp_opt;
            best_maxdiff=maxdiff;
        else
            if (resultat(cnt,3)<=best_option)
                best_option=resultat(cnt,3);
                best_dmodel2=dmodel_opt;
                best_inp=inp_opt;
                best_otp=otp_opt;
                best_maxdiff=maxdiff;
            end
        end
        cnt=cnt+1;
    end
end

%Root relative square error is calculated for the ''best_option" chosen above. Similar to
    the concept of ''M" and ''N", ''mrrse" and ''nrrse" are employed.
```

```
mrrse=64;
nrrse=3200;
[RRSE,avg_otp_true]=RRSE(mrrse, nrrse, dim,lower_limit, upper_limit, best_dmodel2);
```

```
function [x, y]=assigner(M, N, dim, lower_limit, upper_limit, samplefile)

% This function takes a list of input (sample) vectors of length N and does the mapping
    on to the values of input parameters values based on the sampling resolution determined
    by M.

%Initialization

dat=zeros(1,dim);
input=zeros(1,dim);
x=zeros(dim,N);
y=zeros(1,N);

%The vector containing the differences between the upper and lower limits of variables is
    computed.

difference=upper_limit-lower_limit;

s=1;

%First the input vector components are determined then the corresponding drain current
    values are computed through the function ''ids_nmos". The list of sample vectors ''x"
    and the output vector ''y" are returned.

for b1=1:N
    for b2=1:dim
        dat(b2)=samplefile(b1, b2);
        input(b2)=lower_limit(b2)+difference(b2)/M*(dat(b2)-1);
    end
    x(:,s)=input;
    y(s)=ids_nmos(input);
    s=s+1;
end
```

```
function [inp_new, otp_new]=assigner_new(kez, dim, lower_limit, upper_limit, samplefile,
    posvec,N)

%This function takes the test vector that results in the highest error among all test
    vectors and does the mapping on to the input parameter values.

% Initializations
dat=zeros(1,dim);
input=zeros(1,dim);

%The variable "kez" has been sent to indicate the current loop number.

difference =upper_limit-lower_limit;

%The test vector of interest is transferred to a new vector.

for b1=1:dim
    datmax(kez,b1)=posvec(b1);
end

%The associated values of input vector elements are calculated and then the corresponding
    value of the transistor drain current is determined. Both outcomes are returned to the
    main program.

for b2=1:dim
    dat(b2)=samplefile(b1, b2);
    input(b2)=lower_limit(b2)+difference(b2)/N*(datmax(kez,b2)-1);
end
inp_new(:,1)=input;
otp_new=ids_nmos(input);
```

```
function bestplan=bestplan(n,z, dil_i ,  dil_j ,  v)

%This function chooses the best sampling plan out of a certain number of sampling plans
    based on the minimax and orthogonality measures.

% Initializations
points=zeros(z,n,n);
m=zeros(z,n,n);
ortnumber=zeros(1,z);
bestplan=zeros(n,n);
sum_dist=zeros(1,z);
su1=zeros(1,z);
quotient=zeros(1,z);
dist=zeros(z,n*(n-1)/2);

bndr=floor(n/(dil_i* dil_j ));

if bndr>0
    bndr=bndr-1;
end

%z represents the number of constructed Latin squares that will be compared with each
    other. Their orthogonality number is found using the ''ortnum" function. The more the
    ''ortnumber" the higher the chance of that Latin square to be chosen.

for i=1:z
    m(i ,:,:) =lhc(n, dil_i ,  dil_j ,  v);
    ortnumber(i)=ortnum(m(i,:,:),n, dil_i , dil_j ,v,bndr);
end

t=zeros(n);

for k=1:z
    s=1;
    for i=1:n
        for j=1:n
            if (m(k,j,i)==1)
                points(k,s,1)=i;
                points(k,s,2)=j;
                s=s+1;
            end
        end
```

```
    end
end

%Here the minimax theorem is used which suggests minimizing the number of same distance
    values from sample to sample, while maximizing the distances between distinct samples.
    This theorem increases the sampling efficiency.

for k=1:z
    u=1;
    for i=1:n-1
        for j=i+1:n
            dist(k,u)= (points(k,j,1)-points(k,i,1))^2+ (points(k,j,2)-points(k,i,2))^2;
            u=u+1;
        end
    end
end

%"quotient" represents the quality of a certain Latin square in terms of the sampling
    efficiency criteria. Out of z different Latin squares the one with the highest
    "quotient" values is chosen.

for k=1:z
    sum_dist(k)=sum(dist(k,:));
    su1(k)=length(dist(k,:))-length(unique(dist(k,:)));
%    quotient(k)=sum_dist(k)/su1(k);
   quotient(k)=ortnumber(k)*sum_dist(k)/su1(k);
end

[maxnum, maxind]=max(quotient);

for i=1:n
    for j=1:n
        bestplan(i,j)=m(maxind,i,j);
    end
end
```

```
function X=chan(n,arr)

%This function goes through each element of ‘‘arr” and creates another array ‘‘X” where
    ‘‘X(k)” is set to 1 with ‘‘arr(k)=n”. All other elements of ‘‘X” are set to 0. ‘‘X” is
    returned as the output of the function ‘‘chan”

% Initialization
X=zeros(1,n);

for q=1:n
    if (arr(q)==n)
        X(q)=1;
    else
        X(q)=0;
    end
end
return
```

```
function comparator=comparator(smfold, grfold, dim)

%This function compares the sampling sets so that the repetition of sample vectors can be
    prevented. The big set of samples of length N have been divided in to small sets of
    length M. When a group of sample vectors of length M is generated it is compared with
    the previously generated lists of sample vectors. If it does not contain any repating
    sample vector it is appended to the list of sample vectors otherwise it is discarded
    and a new one is created. This process continues until the desired length of sample
    vectors is reached which is given by the number n.

%The lengths of the sample sets are determined first.

n=length(smfold);
N=length(grfold);

comparator=1;

%Comparison is done in these loops.

for i=1:n
    for j=1:N
        flag=0;
        for k=1:dim
            if (smfold(i,k)==grfold(j,k))
                flag=flag+1;
            end
        end
        if (flag==dim)
            comparator=0;
            break;
        end
    end
    if (comparator==0)
        break;
    end
end
```

```
function [evaluater,posvec]= evaluater_gm(M, N, dim, dmodel, samplefile, lower_limit,
    upper_limit)

%This function takes the generated surrogate model for the transistor drain current and
    evaluates its validity by creating a set of test vectors and comparing the
    corresponding outputs with the true model results. The test vector that yields the
    highest error is identified and returned such that it can be added to the list of
    sample vectors.

sum=0;

%The group of test vectors is created at this step.
tempfile=samplecreate(M, N, dim);
%The vector that will hold the difference between the surrogate model predictions and the
    true model outputs is initiated.
diff=zeros(1,N);

%A check is performed here to prevent the selection of any input vectors that show up both
    as a sample vector and as a test vector. In case of overlap, a new list of test
    vectors is generated.
k=1;
while (k<=N)
    i=1;
    while (i<=N)
        for j=1:dim
            if (tempfile(k,j)==samplefile(i,j))
                sum=sum+1;
            end
        end
        if (sum==dim)
            tempfile=samplecreate(N, dim);
            i=1;
            k=1;
            sum=0;
        else
            i=i+1;
            sum=0;
        end
    end
    k=k+1;
end
```

```
%The input vector elements corresponding to the test vectors are found here.

[x_ev, y_ev]=assigner(M, N, dim, lower_limit, upper_limit, tempfile);

%The vectors are transposed for the correct usage in the next function.
inp_ev=x_ev';
otp_ev=y_ev';

%The ''predictor" function is encompassed in DACE toolbox. Surrogate model outputs
    corresponding to the test vectors are evaluated using this function.

yhat=predictor(inp_ev, dmodel);

h=length(yhat);

%The difference between the surrogate model output and the true model outcome is
    calculated for each sample vector.

for i=1:h
    diff(i)=abs(yhat(i)-otp_ev(i));
end

%The difference vector is returned.
evaluater= diff;

mxd=diff(1);

%The test vector that incurs the highest difference is found and returned as well. It will
    be added to the already existing list of sample vectors.

for i=1:h
    if diff(i)>=mxd
        mxd=diff(i);
        posvec=tempfile(i,:);
    end
end
```

```
function lhc=lhc(n, dil_i , dil_j , v)

%This function generates an orthogonal Latin square of length n for two variables. At the
    same time, the generated square is divided into pieces to check orthogonality. ''n" is
    divided in to dil_i and dil_j pieces in x- and y-directions, respectively. A total of
    n samples are chosen out of n^2 selections. The parameter v represents the amount of
    variation in the number of samples of each Latin square piece so that the generation of
    Latin squares becomes easier.

%ort is the result of the orthogonality checking function ''ortcheck".

ort=0;

while (ort==0)
    %''a1" is initialized as an ''n by n" square matrix. It is aimed to choose n samples
        out of n^2 selections where each row and column get one sample. A random
        permutation of numbers from 1 to n is generated and stored in the array called
        ''arr". The function ''chan" takes ''arr" and returns another array which contains
        only one '1' located at the index k with ''arr(k)=n". All other elements of the
        output array of the function ''chan" are 0.

    a1=zeros(n);
    arr=randperm(n);
    a1(:,1)=chan(n,arr);

    %The same procedure is applied to create other columns while a row-by-row check is
        realized if more than two samples per column are chosen. In that case a new column
        is created.

    for i=2:n
        arr=randperm(n);
        a1(:,i)=chan(n,arr);
        k=1;
        while (k<=i-1)
            j=1;
            while (j<=n)
                if (a1(j,i)==1) && (a1(j,k)==1)
                    arr=randperm(n);
                    a1(:,i)=chan(n,arr);
                    j=1;
                    k=1;
                else
```

```
                    j=j+1;
                end
            end
            k=k+1;
        end
    end

    %Finally, an orthogonality check is applied.
    ort=ortcheck(a1,n,dil_i , dil_j ,v);
end

%The resulting Latin square is returned.
lhc=a1;
```

```
function ortcheck=ortcheck(a,n,dil_i, dil_j ,v)

%This function is used for the implementation of orhogonality in the Latin hypercube
    sampling. Here, it is aimed that each piece of the (n by n) square has a certain
    number of samples. This ensures that samples are selected from each region of the
    sample space.

%a is the Latin square under investigation.
%n is the side length of the given Latin square.
%dil_i and dil_j are the number of divisions in x- and y-directions, respectively.
%v is the amount of variation in the number of samples of each Latin square pieces (on top
    of the nominal number "bo" -see definition below- so that the generation of Latin
    squares becomes easier.

%These bounds are determined so that loops which aim recording the number of samples in
    each piece can work properly.

bq_i=floor(n/dil_i);
bq_j=floor(n/dil_j);

%This is the nominal number of samples that is expected to be contained in each piece of
    the Latin square.
bo=round(n/(dil_j*dil_i));

%Initialization
qu=zeros(1,dil_i* dil_j );

%p is a loop parameter.
p=1;

%In these loops, each region is visited and the number of samples there is counted. The
    results are collected in an array called "qu".

for i=1:dil_i
    for j=1:dil_j
        if (i==dil_i)
            i_uplim=n;
        else
            i_uplim=i*bq_i;
        end
        if (j==dil_j)
            j_uplim=n;
```

```
        else
            j_uplim=j*bq_j;
        end
        qu(p)=summa(a, 1+(i-1)*bq_i, i_uplim, 1+(j-1)*bq_j, j_uplim);
        p=p+1;
    end
end

p=p-1;

%Here it is checked whether the variation of number of samples in each Latin square piece
    is beyond the given limit ''v". If it is so, then the Latin square is discarded
    resulting in ''ortcheck"=0. Otherwise, ''ortcheck"=1 and the generated Latin square
    passes the test.

for i=1:p
    if ~((qu(i)>=bo-v) && (qu(i)<=bo+v))
        ortcheck=0;
        return
    end
end

ortcheck=1;
```

```
function ortnum=ortnum(ax,n,dil_i,dil_j,~,bndr)

%This function is used for the implementation of the orhogonality in the Latin hypercube
    sampling.

% Initialization
a=zeros(n,n);
qu=zeros(1, dil_i* dil_j );

for i=1:n
    for j=1:n
        a(i,j)=ax(1,i,j);
    end
end

bq_i=floor(n/dil_i);
bq_j=floor(n/dil_j);

ortnum=0;

%bo=round(n/(dil_j*dil_i));

p=1;

%The function ''summa" is used to find the total sum of the samples within each piece of
    the big Latin square. The aim here is to detect the pieces that have less than the
    standard amount of samples per piece which is ''bndr".

for i=1:dil_i
    for j=1:dil_j
        if (i==dil_i)
            i_uplim=n;
        else
            i_uplim=i*bq_i;
        end
        if (j==dil_j)
            j_uplim=n;
        else
            j_uplim=j*bq_j;
        end
        qu(p)=summa(a, 1+(i-1)*bq_i, i_uplim, 1+(j-1)*bq_j, j_uplim);
        p=p+1;
```

```
    end
end

p=p-1;

%The Latin squares which are more evenly distributed easily satisy the requirement and get
    a higher ''ortnum". This makes them more advantageous to be chosen in the ''bestplan"
    function.

for i=1:p
    if qu(i)>bndr
        ortnum=ortnum+1;
    end
end
```

```
function packet_n=packet_n(N,dim,z,dil_i, dil_j , v)

%This function creates a k−dimensional Latin hypercube sampling scheme out of k−1 Latin
    squares (dimension of 2). It should be noted that the number of rows correspond to the
    total number of samples whereas each column is a separate dimension. This scheme
    enables a huge flexibility .

%Initial memory allocation is performed.

datas=zeros((dim−1)∗N,3);

s=1;

%Latin squares are produced that satisfy orthogonality requirement as well as the minimax
    theorem.

for i=1:dim−1
    x=bestplan(N,z, dil_i , dil_j , v);
    for j=1:N
        for k=1:N
            if (x(j,k)==1)
                datas(s,1)=j;
                datas(s,2)=k;
                s=s+1;
            end
        end
    end
end

%The resulting squares are randomly permutated to increase the sampling efficiency. An
    example is given below.
%For the case of N=5:
%
%12   3    4    5
%
%12 | 13 | 14 | 12
%25 | 25 | 23 | 25
%34 | 32 | 35 | 31
%43 | 41 | 42 | 44
%51 | 54 | 51 | 53
%
%If t=3 then the final permutated sampling matrix would look like:
```

```
%
% 4 1 3 5 1
% 1 2 5 1 3
% 5 3 2 3 5
% 2 4 1 4 2
% 3 5 4 2 4

final_datas =zeros(N,dim);

t=randperm(dim);

param=t(1);

%Latin hypercube sampling scheme generation is realized here.

if (param==1)
    for k=1:N
        final_datas (k,1)=datas(k,1);
    end
    for k=1:N
        for i=1:dim-1
            final_datas (k,i+1)=datas((i-1)*N+k,2);
        end
    end
end


if (param~=1)
    for k=1:N
        sekpar=datas((param-2)*N+k,2);
        final_datas (k,param)=sekpar;
        for i=1:dim-1
            if (i<=param-1)
                for q1=1:N
                    if (datas((i-1)*N+q1,2)==sekpar)
                        final_datas (k,i)=datas((i-1)*N+q1,1);
                    end
                end
            end
            if (i>param-1)
                for q1=1:N
                    if (datas((i-1)*N+q1,1)==sekpar)
                        final_datas (k,i+1)=datas((i-1)*N+q1,2);
```

```
                    end
                end
            end
        end
    end
end

packet_n=final_datas;
```

```
function [RRSE,avg_otp_true]=RRSE(M,N,dim,lower_limit, upper_limit, best_dmodel)

%This function calculates the root relative square error (RRSE) for a given model.

%A test vector is created first and the corresponding mapping to the input parameters are
    done.

tempfile2=samplecreate(M, N, dim);

[tempfile3, otp_true]=assigner(M, N, dim, lower_limit, upper_limit, tempfile2);

tempfile4=tempfile3';

% Initialization
otp_model=zeros(1,length(tempfile4));

%Corresponding output values are computed at this step.

for i=1:length(tempfile4)
    otp_model(i)=predictor(tempfile4(i ,:) ,best_dmodel);
end

%The average of true model outputs are calculated for the RRSE formula. They also yield a
    measure for the sampling efficiency.

avg_otp_true=0;
for i=1:length(tempfile3)
    avg_otp_true=avg_otp_true+otp_true(i);
end
avg_otp_true=avg_otp_true/length(otp_true);

%Calculation of RRSE is done here.

sum1=0;
sum2=0;
for i=1:length(tempfile3)
    sum1=sum1+(otp_model(i)-otp_true(i))^2;
    sum2=sum2+(otp_true(i)-avg_otp_true)^2;
end
RRSE=sqrt(sum1/sum2);
```

```
function samplecreate= samplecreate(M, N, dim)

%This function is used for the generation of a group of test vectors.

%An empty array of ''N by dim" is created
samplecreate=zeros(N,dim);

% Initialization
temp=zeros(1,dim);

%An initial random vector of length of M is created
init_rnd=randperm(M);

%The first elements of '' init_rnd" of length M are recorded as the first row.
for i=1:dim
    samplecreate(1,i)=init_rnd(i);
end

%For the successive rows, the same operation is applied. However, in order to prevent the
    double selection of the same vector a control parameter ''flag" is incorporated to the
    system. At the end of this process, a group of test vectors called ''samplecreate" is
    returned.

k=2;
while (k<=N)
    init_rnd=randperm(M);
    for i=1:dim
        temp(i)=init_rnd(i);
    end
    for j=1:k-1
        if (temp==samplecreate(j))
            flag=0;
            break;
        else
            flag=1;
        end
    end
    if (flag)
        for s=1:dim
            samplecreate(k,s)=temp(s);
        end
        k=k+1;
```

```
    end
end
```

```
function summa=summa(a,x_down_lim, x_up_lim, y_down_lim, y_up_lim)

%This function is used for the implementation of orhogonality in the Latin hypercube
    sampling. All samples (0 or 1) in the small pieces of the Latin square are summed up.

summa=0;
for i=x_down_lim:1:x_up_lim
    for j=y_down_lim:1:y_up_lim
        summa=summa+a(i,j);
    end
end
```

A.3 Screen Shot of the Program Operation

Figure A.1 is a screen shot that has been made during a run of the main function given in Section A.2. In this figure, an I_{ds} surrogate model for NMOS devices is developed that will be used in the long-term time span reliability analysis.

Figure A.2 is captured during a run of the orthogonal Latin hypercube sampling. Here, a total of 15 sample vectors of five dimensions are created in three batches of five sample vectors each. x_{div} and y_{div} are both chosen to be two.

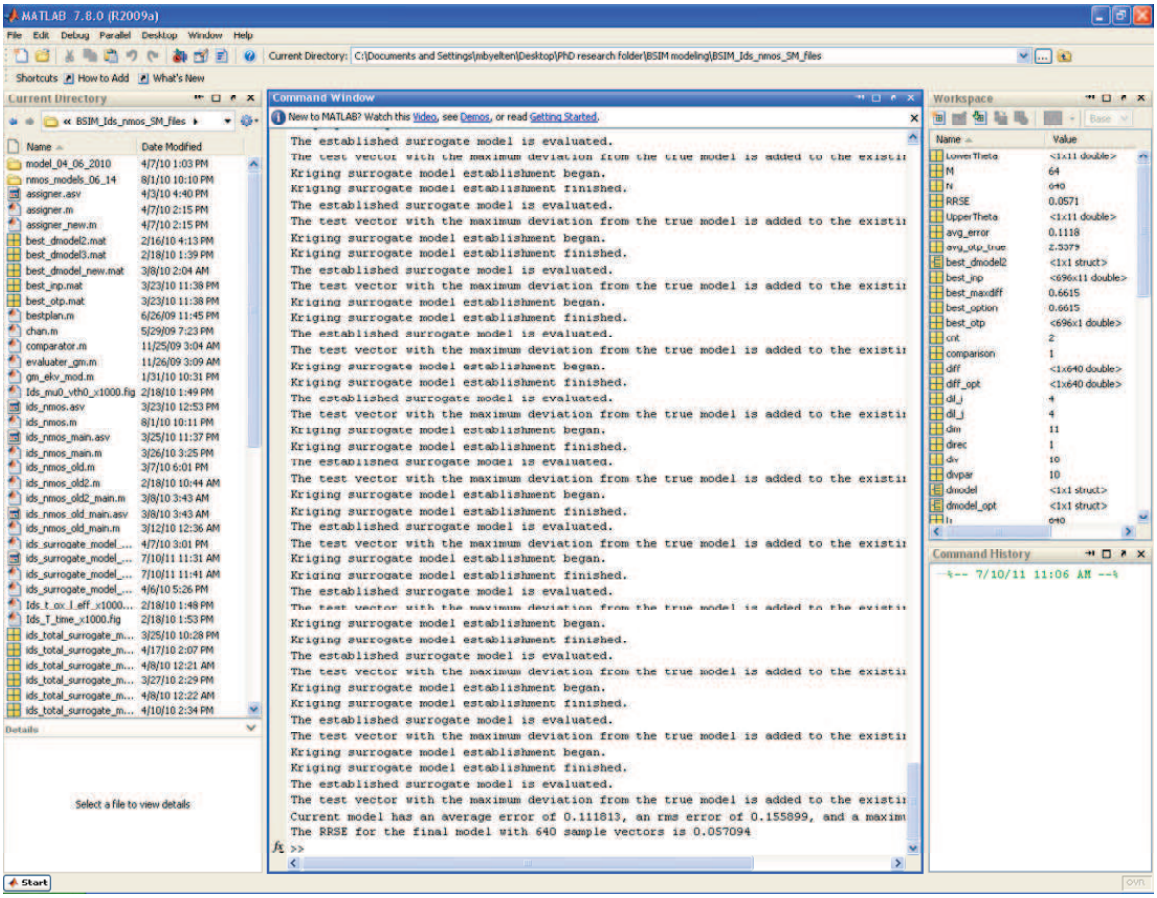


Figure A.1: The screen shot demonstrating the operation of the surrogate modeling development codes given in Section A.2.

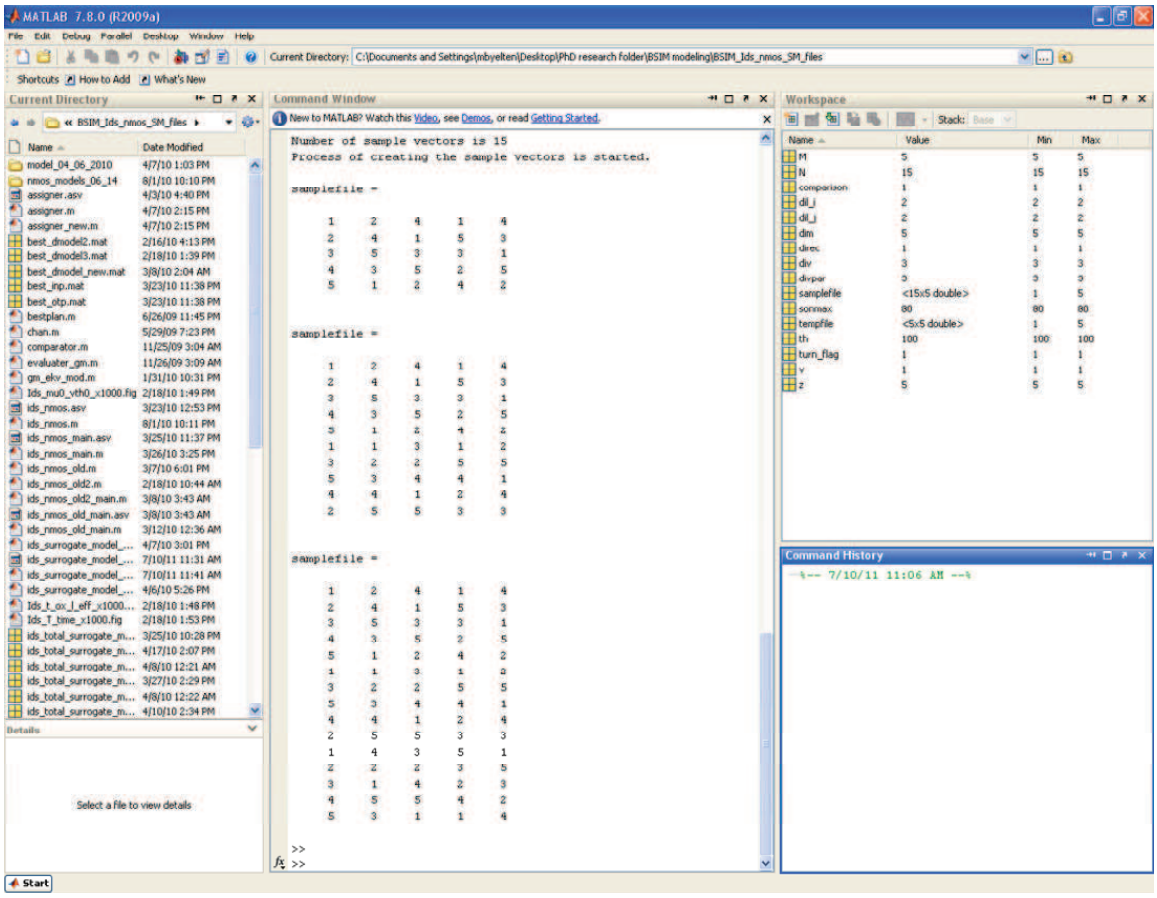


Figure A.2: The screen shot showing the operation of the multidimensional orthogonal Latin hypercube sampling code given in Section A.2.

Appendix B

MATLAB® for the Surrogate Model-Based Circuit Variability and Reliability Analysis

B.1 Introduction

In this appendix, the codes to perform simultaneous circuit variability and reliability analysis on the circuit provided in Figure 4.5 using Kriging-based surrogate model of I_{ds} are presented. All codes have been provided with extensive comments which describe the methodology pursued in individual functions.

The main program to run is given in the beginning. All the subsequent functions are following the main program in alphabetical order. Also, in order to run the functions properly, the MATLAB® toolbox DACE should be installed and set up correctly.

Section B.2 provides the main function, also the functions "diff_amp_junc_pmos_func", "diff_amp_res_active_load2_pmos_func3", "gain_stage", "pmos_nmos_cascode_func", and

"Vgd_determine". Section B.3 presents a screen dump while the main function is operated in MATLAB® environment.

B.2 Commented MATLAB® Codes

```
%Main function to run circuit variability and reliability analysis on the circuit of
    Figure 4.5

clear;
clc;

tic

%Adding the paths of associated folders
addpath(genpath('C:\Users\Mustafa_Berke\Desktop\Simulation_Files'));

%Surrogate model for NMOS
load ids_nmos_L_age_fresh_surrogate_model.mat;
nmos_model=best_dmodel2;

%Surrogate model for PMOS (device age (t_age)= 0 to 8 months)
%load ids_pmos_L_age_0_8_months_model.mat;
%pmos_model=best_dmodel2;

%Surrogate model for PMOS (t_age= 8 months 10 years)
load ids_pmos_L_age_8months_10years_model.mat;
pmos_model=best_dmodel2;

%Initialization of some matrices
op_point_table_nm=zeros(4,5);
op_point_table_pm=zeros(5,6);

%Power supply. Vdd= Vdd (in V) * 10
global Vdd;
Vdd=12;

%Setting the initial voltages, widths (All lengths are Lmin).
Vg_p1=0;
Vg_n1=0;
w_n1=25e-6;
w_p1=50e-6;
w_n_bias=11.42e-6;
w_p_bias=6e-6;
```

```
%Diode connection flags for different devices 1-yes 0-no
diode_con1=[1 1];

%Resistances for the biasing units and the differential core
R_given=66;
res=500;

%Age mode: first element: 0, second element: for time scale 0 to 8 months use 0, for time
    scale 8 months to 10 years use 1
age_mod=[0 1];

%t_age: for 0 to 8 months -> 0 to 2000 (model multiplies it with 1e4 to convert to
    seconds) for 8 months to 10 years -> 1 to 33 (model multiplies it with 1e7 to convert
    to seconds)
time_p=29;

%Setting temperature: T= T (in K)/10. ex. For room temperature use 30.
T=37.3;

%Normalization of the widths
w_p1_con=w_p1/10e-6;
w_n1_con=w_n1/10e-6;

%Bias transistor entities for nominal process coordinates. Adjustments made to increase
    numerical accuracy of the surrogate models whereby matrix calculation based errors are
    avoided.
%First element: Temperature (explained above)
%Second element: Oxide thickness tox= tox (in nm) * 1e10 ex. 5nm corresponds to 50
%Third element: Channel doping Nch= Nch (in 1/cm^3) * 1e-16
%Fourth element: Effective length offset xl= xl (in nm) * 1e9
%Fifth element: Vbs = Vbs (Vsb for PMOS)(in V) * 10
%Sixth element: Vth0= Vth0 (in V) * 100
%Seventh element: Intrinsic mobility for nfet u0= u0 (in cm^2(Vs)^-1)* 0.1
%for pfet u0= u0 (in cm^2(Vs)^-1)
%Eighth element: Vgs= Vgs (Vsg for PMOS)(in V) * 10
%Ninth element: Vds= Vds (Vsd for PMOS) (in V) * 10
%Tenth element: Drain-source resistance Rds= Rds (Rsd for PMOS)(in Ohm)
%Eleventh element: t_age
%for 0 to 8 months -> 0 to 2000 (model multiplies it with 1e4 to convert to seconds)
%for 8 months to 10 years -> 1 to 33 (model multiplies it with 1e7 to
%convert to seconds)

design_nm1=[T, tox_nm1, Nch_nm1, xl_nm1, 0, Vth0_nm1, u0_nm1, 0, 0, Rds_nm1, 0];
```

```
design_pm1=[T, tox_pm1, Nch_pm1, xl_pm1, 0, Vth0_pm1, u0_pm1, 0, 0, Rsd_pm1, time_p];

%PMOS-NMOS cascode biasing
[id_bias ,Vbias_opt]=pmos_nmos_cascode_func(Vg_n1,Vg_p1,w_n_bias,w_p_bias,diode_con1,
    design_nm1, design_pm1,nmos_model,pmos_model);

%Voltage initializations
design_pm1(5)=0;
design_pm1(8)=Vdd-Vbias_opt;
design_pm1(9)=Vdd-Vbias_opt;

%Degradation calculation
[delta_VTH_pm1, mu_coeff_pm1]=thr_volt_det(design_pm1, age_mod);

%Operating point recording
op_point_table_nm(1,:)=[0, 0, Vdd-bias_volt_nm/10, Vdd-bias_volt_nm/10, 1];
op_point_table_pm(1,:)=[ mu_coeff_pm1, delta_VTH_pm1, bias_volt_pm/10, Vdd/10,
    (Vdd-bias_volt_pm)/10, (Vdd-bias_volt_pm)/10];

%Biasing voltage for p2 and p3
Vg_p2=4;
Vg_p3=4;

%Choose the biasing for p4
Vg_p4=Vbias_opt;

%Transistor widths
w_n2=30e-6;
w_n3=30e-6;
w_p2=50e-6;
w_p3=50e-6;
w_p4=50e-6;

%Transistor entities for the core differential amplifier
design_nm2=[T, tox_nm2, Nch_nm2, xl_nm2, 0, Vth0_nm2, u0_nm2, 0, 0, Rds_nm2, 0];
design_nm3=[T, tox_nm3, Nch_nm3, xl_nm3, 0, Vth0_nm3, u0_nm3, 0, 0, Rds_nm3, 0];
design_pm2=[T, tox_pm2, Nch_pm2, xl_pm2, 0, Vth0_pm2, u0_pm2, 0, 0, Rsd_pm2, time_p];
design_pm3=[T, tox_pm3, Nch_pm3, xl_pm3, 0, Vth0_pm3, u0_pm3, 0, 0, Rsd_pm3, time_p];
design_pm4=[T, tox_pm4, Nch_pm4, xl_pm4, 0, Vth0_pm4, u0_pm4, 0, 0, Rsd_pm4, time_p];

%Transistor entities for the core differential amplifiers, the widths of all transistors,
    external voltages, and surrogate models for n- and p-channel devices are sent to the
    function "diff_amp_res_active_load2_pmos_func3" to perform circuit variability and
```

reliability analysis.

```
[ id1_final , id2_final , Vs0_opt, Vs2_opt, Vsx_opt]=
    diff_amp_res_active_load2_pmos_func3(Vg_p2,Vg_p3,Vg_p4,w_n2,w_n3,w_p2,w_p3,w_p4,
    res, design_nm2, design_nm3, design_pm2, design_pm3, design_pm4,
    nmos_model,pmos_model);

%Operating point voltages
design_pm2(5)=Vsx_opt−Vdd;
design_pm2(8)=Vsx_opt−Vg_p2;
design_pm2(9)=Vsx_opt−Vs0_opt;
design_pm3(5)=Vsx_opt−Vdd;
design_pm3(8)=Vsx_opt−Vg_p3;
design_pm3(9)=Vsx_opt−Vs2_opt;
design_pm4(5)=0;
design_pm4(8)=Vdd−Vs0_opt;
design_pm4(9)=Vdd−Vsx_opt;

%Degradation calculation
[delta_VTH_pm2, mu_coeff_pm2]=thr_volt_det(design_pm2, age_mod);
[delta_VTH_pm3, mu_coeff_pm3]=thr_volt_det(design_pm3, age_mod);
[delta_VTH_pm4, mu_coeff_pm4]=thr_volt_det(design_pm4, age_mod);

%Operating point recording
op_point_table_nm(2,:)=[0, 0, Vs0_opt/10, Vs0_opt/10, 1];
op_point_table_nm(3,:)=[0, 0, Vs2_opt/10, Vs2_opt/10, 1];
op_point_table_pm(2,:)=[mu_coeff_pm2, delta_VTH_pm2,Vs0_opt/10, Vdd/10,
    (Vsx_opt−Vs0_opt)/10,(Vsx_opt−Vg_p2)/10];
op_point_table_pm(3,:)=[mu_coeff_pm3, delta_VTH_pm3,Vs2_opt/10, Vdd/10,
    (Vsx_opt−Vs2_opt)/10,(Vsx_opt−Vg_p3)/10];
op_point_table_pm(4,:)=[mu_coeff_pm4, delta_VTH_pm4, Vsx_opt/10, Vdd/10,
    (Vdd−Vsx_opt)/10,(Vdd−Vg_p4)/10];
```

toc

```
function [id1xx_opt,id2xx_opt,
    Vs_opt]=diff_amp_junc_pmos_func(Vd_p1,Vd_p2,Vg_p1,Vg_p2,Vg_p3,w_p1,w_p2,w_p3,
    design_pm1,design_pm2, design_pm3, pmos_model)

%This function takes a three transistor differential core (a differential pair and the
    tail transistor) and calculates the distribution of the currents and the intermediate
    voltage

%Upper/lower voltage limits depending whether it is an NMOS/PMOS differential pair
    respectively
if (Vd_p1>Vd_p2)
    Vs_max=ceil(10*Vd_p1);
else
    Vs_max=ceil(10*Vd_p2);
end

%Initialization
id1x=zeros(1,121-Vs_max);
id2x=zeros(1,121-Vs_max);
id3x=zeros(1,121-Vs_max);
id1xx=zeros(1,21);
id2xx=zeros(1,21);
id3xx=zeros(1,21);

%Power supply
global Vdd;
Vdd=12;

%Width normalization
w_p1_con=w_p1/10e-6;
w_p2_con=w_p2/10e-6;
w_p3_con=w_p3/10e-6;

%Setting the gate voltage of the tail transistor
design_pm3(8)=Vdd-Vg_p3;

i=1;

%External loop to find out the approximate level of the intermediate voltage that will
    balance the three drain currents.
for Vs=Vs_max:1:120
    Vst=Vs/10;
```

```
    %Setting the voltages to the appropriate levels for the differential core topology
    design_pm1(5)=Vst-Vdd;
    design_pm1(8)=Vst-Vg_p1;
    design_pm1(9)=Vst-Vd_p1;
    design_pm2(5)=Vst-Vdd;
    design_pm2(8)=Vst-Vg_p2;
    design_pm2(9)=Vst-Vd_p2;
    design_pm3(9)=Vdd-Vst;

    %Evaluating the currents for each case
    id1x(i)=w_p1_con*predictor(design_pm1, pmos_model);
    id2x(i)=w_p2_con*predictor(design_pm2, pmos_model);
    id3x(i)=w_p3_con*predictor(design_pm3, pmos_model);
    i=i+1;
end

%Finding the minimum current difference and the associated voltage level for the evaluated
    cases
[ id_diff_min_opt ,ind]=min(abs(id1x+id2x-id3x));
Vs_opt=Vs_max+ind-1;
ind1=10*(Vs_opt-1);
i=1;

%Inner loop to determine more exactly the minimum current difference and the associated
    voltage level
for Vx=10*(Vs_opt-1):1:10*(Vs_opt+1)
    Vsdt=Vx/100;
    design_pm1(5)=Vsdt-Vdd;
    design_pm1(8)=Vsdt-Vg_p1;
    design_pm1(9)=Vsdt-Vd_p1;
    design_pm2(5)=Vsdt-Vdd;
    design_pm2(8)=Vsdt-Vg_p2;
    design_pm2(9)=Vsdt-Vd_p2;
    design_pm3(9)=Vdd-Vsdt;
    id1xx(i)=w_p1_con*predictor(design_pm1, pmos_model);
    id2xx(i)=w_p2_con*predictor(design_pm2, pmos_model);
    id3xx(i)=w_p3_con*predictor(design_pm3, pmos_model);
    i=i+1;
end

%Finding the minimum current difference and the associated voltage level for the evaluated
    cases
[ id_diff_min_opt , ind_fin]=min(abs(id1xx+id2xx-id3xx));
```

```
%Recording the outcomes of the searches
id1xx_opt=id1xx(ind_fin);
id2xx_opt=id2xx(ind_fin);
id3xx_opt=id3xx(ind_fin);
Vs_opt=(ind1+ind_fin-1)/100;
```

```
function [id1_final , id2_final , Vs1_opt, Vs2_opt,
    Vsx_opt]=diff_amp_res_active_load2_pmos_func3(Vg1,Vg2, Vg3,
    w_n1,w_n2,w_p1,w_p2,w_p3, res, design_nm1, design_nm2, design_pm1, design_pm2,
    design_pm3, nmos_model,pmos_model)

%This function takes the transistor entities for the core differential amplifiers along
    with the widths of all transistors , external voltages , and surrogate models for n-
    and p-channel devices and performs circuit variability and reliability analysis .

%Initialization
min_opt=zeros(20,8);

%Normalization of the widths
w_n1_con=w_n1/10e-6;
w_n2_con=w_n2/10e-6;

%Power supply
global Vdd;
Vdd=12;

%Upper and lower limits
Vs1_up=12;
Vs2_up=12;
Vs1_down=0;
Vs2_down=0;

%Loop counter specifications
run=0;
run_max=50;

can=0;
can_max=200;

flag=1;

%Initial high value for the current difference of neighboring circuit elements
id1_diff_min =100;
id2_diff_min =100;

rp=1;
kez=1;
```

```
rVs_up=randperm(11);

%Outer loop for the initial search of optimum node voltage
while ((abs(id1_diff_min)>0.05) || (abs(id2_diff_min)>0.05)) && (can<can_max)

    if (can>0)
        if id1_diff_min >0.05
            if (abs(Vs1_up-Vs1_down)<1e-1)
                if (rp<12)
                    Vs1_up=rVs_up(rp);
                    Vs1_down=0;
                    rp=rp+1;
                end
            end
        end
        if id2_diff_min >0.05
            if (abs(Vs2_up-Vs2_down)<1e-1)
                if (rp<12)
                    Vs2_up=rVs_up(rp);
                    Vs2_down=0;
                    rp=rp+1;
                end
            end
        end
    end

    %Midpoint is taken
    Vs1=(Vs1_up+Vs1_down)/2;
    Vs2=(Vs2_up+Vs2_down)/2;

    Vs1_init=Vs1;
    Vs2_init=Vs2;

    %Setting the appropriate voltages for the choice
    design_nm1(8)=Vs1;
    design_nm2(8)=Vs1;
    design_nm1(9)=Vs1;
    design_nm2(9)=Vs2;

    run=0;

    flag=1;
```

```
%Current calculations for the initial node voltage choice
[itar1_opt, itar2_opt, Vsx_val]=
    diff_amp_junc_pmos_func(Vs1,Vs2,Vg1,Vg2,Vs1,w_p1,w_p2,w_p3,design_pm1,
    design_pm2, design_pm3,pmos_model);

id_nm1=w_n1_con*predictor(design_nm1, nmos_model);
id_nm2=w_n2_con*predictor(design_nm2, nmos_model);

%Inner loop: Once the initial starting point is chosen now a localized search is
    conducted for the optimum node voltage
while ((abs(id1_diff_min)>0.05) || (abs(id2_diff_min)>0.05)) && (run<run_max)

    id_nm1=(itar1_opt+id_nm1)/2;
    id_nm2=(itar2_opt+id_nm2)/2;

    %The topology based current calculations.
    Vds1=Vgd_determine(id_nm1,w_n1_con,design_nm1,nmos_model);
    [id_nm2_real, i_res_real , Vds2]=gain_stage(id_nm2, res, Vs1, w_n2_con,
        design_nm2, nmos_model);

    %Updated voltage levels voltage settings
    design_nm1(8)=Vds1;
    design_nm1(9)=Vds1;
    design_nm2(8)=Vds1;
    design_nm2(9)=Vds2;

    Vs1=Vds1;
    Vs2=Vds2;

    %Updated current values
    [itar1_opt, itar2_opt, Vsx_val]=
        diff_amp_junc_pmos_func(Vs1,Vs2,Vg1,Vg2,Vs1,w_p1,w_p2,w_p3,design_pm1,
        design_pm2, design_pm3,pmos_model);

    %Updated current difference
    id1_diff =id_nm1-itar1_opt;
    id2_diff =id_nm2-itar2_opt;

    %Checking for the local/global minimum
    if (run==0) && (rp==1)
        id1_diff_min=id1_diff;
        id2_diff_min=id2_diff;
        id1_final =itar1_opt;
```

```
        id2_final =itar2_opt;
        id1_final_nm1=id_nm1;
        id2_final_nm2=id_nm2;
        Vsx_opt=Vsx_val;
        Vs1_opt=Vs1;
        Vs2_opt=Vs2;
    else
        if (abs(id1_diff)<abs(id1_diff_min)) && (abs(id2_diff)<abs(id2_diff_min))
            id1_diff_min=id1_diff;
            id1_final =itar1_opt;
            id1_final_nm1=id_nm1;
            Vs1_opt=Vs1;
            id2_diff_min=id2_diff;
            id2_final =itar2_opt;
            id2_final_nm2=id_nm2;
            Vs2_opt=Vs2;
            Vsx_opt=Vsx_val;
        end
    end
    run=run+1;
end

%If the threshold level is not reached, initial voltage points are updated based on
    the sign of the current difference
if id1_diff_min<0 && abs(id1_diff_min)>0.05
    Vs1_up=Vs1_init;
else if id1_diff_min>0 && abs(id1_diff_min)>0.05
        Vs1_down=Vs1_init;
    end
end

if id2_diff_min<0 && abs(id2_diff_min)>0.05
    Vs2_up=Vs2_init;
else if id2_diff_min>0 && abs(id2_diff_min)>0.05
        Vs2_down=Vs2_init;
    end
end

%Recording the local minimum values- Optional
if (flag)
    id1_diff_min_opt=id1_diff_min;
    id2_diff_min_opt=id2_diff_min;
    id1_final_min=id1_final;
```

```
        id2_final_min=id2_final;
        min_opt(kez,1)=id1_diff_min;
        min_opt(kez,2)=id2_diff_min;
        min_opt(kez,3)=id1_final;
        min_opt(kez,4)=id2_final;
        min_opt(kez,5)=id1_final_nm1;
        min_opt(kez,6)=id2_final_nm2;
        min_opt(kez,7)=(min_opt(kez,3)+min_opt(kez,5))/2;
        min_opt(kez,8)=(min_opt(kez,4)+min_opt(kez,6))/2;
        kez=kez+1;
        flag=0;
    else
        if (abs(id1_diff_min)<abs(id1_diff_min_opt)) &&
            (abs(id2_diff_min)<abs(id2_diff_min_opt))
            id1_diff_min_opt=id1_diff_min;
            id2_diff_min_opt=id2_diff_min;
            min_opt(kez,1)=id1_diff_min;
            min_opt(kez,2)=id2_diff_min;
            min_opt(kez,3)=id1_final;
            min_opt(kez,4)=id2_final;
            min_opt(kez,5)=id1_final_nm1;
            min_opt(kez,6)=id2_final_nm2;
            min_opt(kez,7)=(min_opt(kez,3)+min_opt(kez,5))/2;
            min_opt(kez,8)=(min_opt(kez,4)+min_opt(kez,6))/2;
            kez=kez+1;
        end
    end

    %Printing the accomplished minimum current difference values for that particular loop
        - Optional
    id1_diff_min=id1_diff_min;
    id2_diff_min=id2_diff_min;

    can=can+1;
end
```

```
function [id2xx_opt, i_resxx_opt, Vs_opt]=gain_stage(i_given, res, Vs1, w_n2_con,
    design_nm2, nmos_model)

%This function is used to find the current distribution due to the topology of one
    transistor in parallel with a resistor.

% Initialization
id2x=zeros(1,121);
i_resx=zeros(1,121);
id2xx=zeros(1,21);
i_resxx=zeros(1,21);

%External loop to find out the approximate node voltage level that will balance the drain
    currents and the current over the resistor
for Vs=1:1:120
    Vst=Vs/10;

    %Setting the voltages to the appropriate levels for the differential core topology
    design_nm2(5)=0;
    design_nm2(8)=Vs1;
    design_nm2(9)=Vst;

    %Evaluating the currents for each case
    id2x(Vs)=w_n2_con*predictor(design_nm2, nmos_model);
    i_resx(Vs)=1e2*Vst/res;
end

%Finding the minimum current difference and the corresponding voltage level for the
    evaluated cases
[id_diff_min_opt,ind]=min(abs(id2x+i_resx-i_given));
Vs_opt=ind;
ind1=10*(Vs_opt-1);
i=1;

%Inner loop to find out more exactly the node voltage that will balance the drain currents
    and the current over the resistor
for Vx=10*(Vs_opt-1):1:10*(Vs_opt+1)
    Vsdt=Vx/100;
    design_nm2(5)=0;
    design_nm2(8)=Vs1;
    design_nm2(9)=Vsdt;
    id2xx(i)=w_n2_con*predictor(design_nm2, nmos_model);
```

```
    i_resxx(i)=1e2*Vsdt/res;
    i=i+1;
end

%Finding the minimum current difference and the associated voltage level for the evaluated
    cases
[id_diff_min_opt , ind_fin]=min(abs(id2xx+i_resxx-i_given));

%Recording the outcomes of the searches
id2xx_opt=id2xx(ind_fin);
i_resxx_opt=i_resxx(ind_fin);
Vs_opt=(ind1+ind_fin-1)/100;
```

```
function [id1xx_opt,Vs_opt]=pmos_nmos_cascode_func(Vg_n,Vg_p,w_n1,w_p1,diode_con,
    design_nm1, design_pm1, nmos_model, pmos_model)

%This function calculates the biasing voltage established by an NMOS transistor and a
    PMOS transistor.

% Initialization
id1x=zeros(1,120);
id2x=zeros(1,120);
id1xx=zeros(1,21);
id2xx=zeros(1,21);

%Power supply
global Vdd;
Vdd=12;

%Width normalization
w_n1_con=w_n1/10e-6;
w_p1_con=w_p1/10e-6;

%Setting the appropriate voltages for the transistor entity.
design_nm1(8)=Vg_n;
design_pm1(8)=Vdd-Vg_p;

%First loop to detect the approximate voltage level to return.
for Vs=1:1:120
    Vst=Vs/10;

    %Setting the appropriate voltages for the transistor entity.
    design_nm1(9)=Vst;
    design_pm1(9)=Vdd-Vst;

    %If the transistors are diode connected, gate-source and drain-source voltages are
        equalized.
    if (diode_con(1))
        design_nm1(8)=design_nm1(9);
    end
    if (diode_con(2))
        design_pm1(8)=design_pm1(9);
    end

    %Drain currents are evaluated for each intermediate voltage case.
    id1x(Vs)=w_n1_con*predictor(design_nm1, nmos_model);
```

```
    id2x(Vs)=w_p1_con*predictor(design_pm1, pmos_model);
end

%Finding the minimum current difference and the corresponding voltage level.
[ id_diff_min_opt ,ind]=min(abs(id1x-id2x));
ind1=10*(ind-1);
i=1;

%Second loop to detect the more exact intermediate voltage level to return.
for Vx=10*(ind-1):1:10*(ind+1)
    Vsdt=Vx/100;
    design_nm1(9)=Vsdt;
    design_pm1(9)=Vdd-Vsdt;
    if (diode_con(1))
        design_nm1(8)=design_nm1(9);
    end
    if (diode_con(2))
        design_pm1(8)=design_pm1(9);
    end
    id1xx(i)=w_n1_con*predictor(design_nm1, nmos_model);
    id2xx(i)=w_p1_con*predictor(design_pm1, pmos_model);
    i=i+1;
end
[ id_diff_min_opt , ind_fin]=min(abs(id1xx-id2xx));
id1xx_opt=id1xx(ind_fin);
id2xx_opt=id2xx(ind_fin);
Vs_opt=(ind1+ind_fin-1)/100;

%Final value is recorded
pmos_nmos_cascode_func=id1xx_opt;
```

```
function Vgd_determine=Vgd_determine(Ids_act,w_con,design,best_dmodel2)

%This function determines the drain-source voltage of a diode-connected transistor given
    its drain current, width and the associated surrogate model.

global Vdd;

flag=0;

%Setting voltage upper and lower limits
Vd_up=Vdd;
Vd_down=0;

%Taking the midpoint as the initial guess
Vds_init=(Vd_up+Vd_down)/2;
run=0;
%Setting a high number as the initial difference
delta_Ids=1000;

%Loop for searching the optimum Vds using binary search
while(abs(delta_Ids)>1e-3)
    if (flag==5)
        break;
    end
    %Updating the voltage limits based on the sign of the current difference
    if (run>0)
        if( delta_Ids<0)
            Vd_up=Vds_init;
        else
            Vd_down=Vds_init;
        end
    end
    Vds_init=(Vd_up+Vd_down)/2;
    design(8)=Vds_init;

    %Updating the transistor entity
    design(9)=Vds_init;

    %Calculating the predicted current
    Ids_pred=w_con*predictor(design, best_dmodel2);

    %Finding the difference between the given and the predicted currents.
```

```
    delta_Ids_new=Ids_act-Ids_pred;

    %If the difference is below the threshold the program exits otherwise it will retry 5
        times.
    if (abs(delta_Ids_new-delta_Ids)<1e-4)
        flag=flag+1;
    else
        flag=0;
    end

    %Updating the current difference
    delta_Ids=delta_Ids_new;
    run=run+1;
end

%Recording the final predicted Vds value.
Vgd_determine=Vds_init;
```

B.3 Screen Shot of the Program Operation

Figure B.1 is a screen shot that has been made during a run of the main function given in Section B.2. In particular, the process corners of the five transistors in Figure 4.5 are randomly assigned to the available corners, NN, SS, FF, FS, and SF. Then the long-term time span reliability analysis is applied for 32 different time points at each selection. Each time, the current through MP2 and the run-time of the code for every iteration are written to the screen. The mean of the runtime of 1920 simulations is calculated to be 1.8473 s.

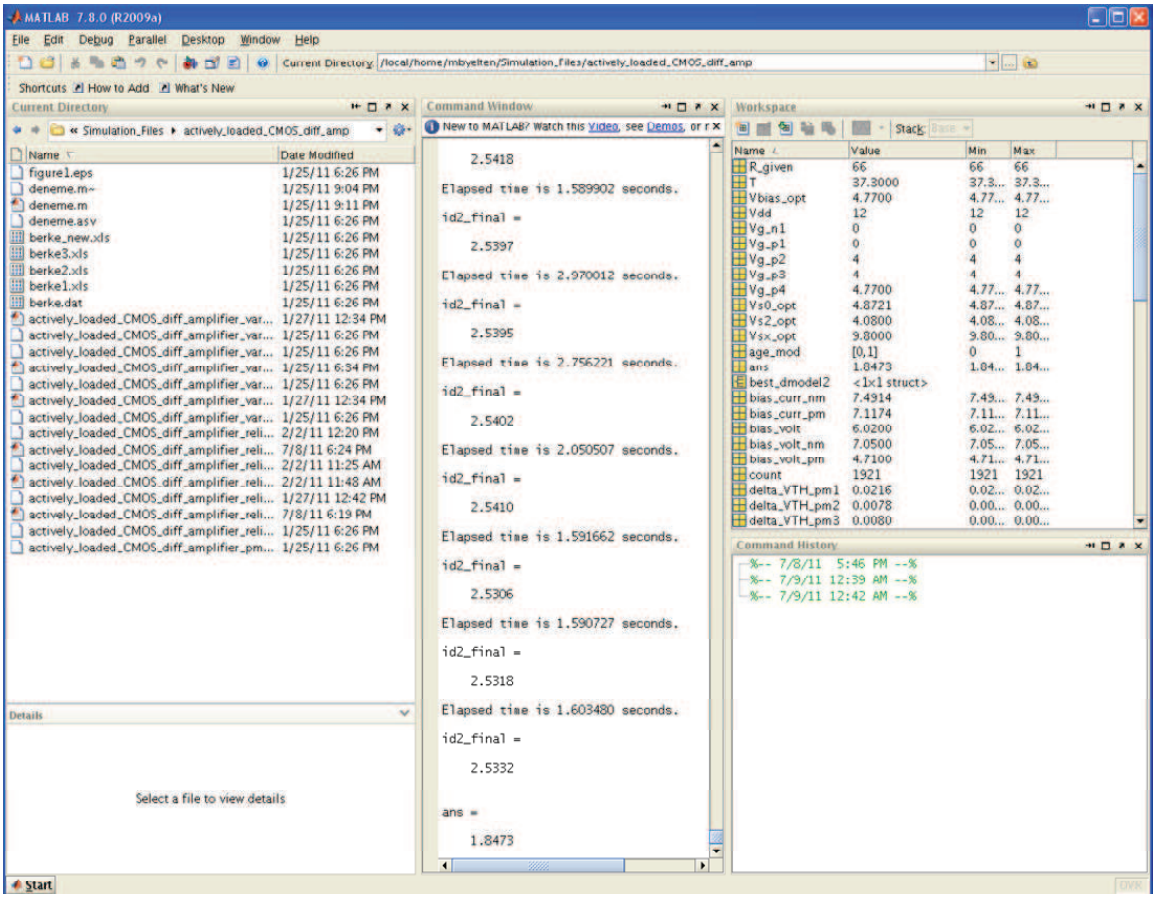


Figure B.1: The screen shot demonstrating the operation of the codes given in Section B.2.