

Intertwined Global Optimization based Reachability Analysis of Analog and Mixed-Signal Designs

Ibtissem Seghaier and Sofiène Tahar

Department of Electrical and Computer Engineering,
Concordia University, Montreal, Canada
{seghaier,tahar}@ece.concordia.ca

Technical Report

January, 2018

Abstract

This work proposes a semi-formal reachability analysis technique based on global optimization. First, circuit transient behaviors in the presence of jitter are modeled as a system of stochastic recurrence equations. A simulation-assisted global optimization based reachability analysis approach is then adopted to compute the reachable sets under an interval of initial conditions and in light of 65nm induced process variability. The novelty of our reachability approach is in approximating the reachable bounds in an intertwined forward/backward manner. The backward corrections refine the obtained reachable bounds in the forward scheme and so reduce the high reachability overbounding (i.e., reduce verifications bias). It determines which regions in the state space are actually unreachable and eliminates those unreachable regions from the over-approximated forward reachable set. Subsequently, a Monte Carlo verification technique based on hypothesis testing is performed on the resultant reachable behavioral bounds to assure compliance with certain desired specifications. These failures are quantified in terms of parametric yield rate which reflects the sensitivity of the analog and mixed-signal circuits to variations in its parameters. We demonstrate the effectiveness of our proposed verification methodology by applying it to two analog and mixed-signal design benchmarks, namely a Tunnel diode oscillator and a frequency synthesizer based Phase Locked Loop. Experimental results show that our approach gives an accurate estimation of the yield rate while accounting for a wide range of circuits disturbances/variations.

1 Introduction

The design of Analog and Mixed-Signal (AMS) based ICs has been in the scale of 65nm and 60GHz with primary applications in high data rate communication system [1]. At such an extreme scale, the verification of the AMS front-end has become particularly challenging due to process parameters, noise and input fluctuations. In particular, process variation is emerging as a key reliability concern and is a primary limiter for more advanced deeper technologies. It is reported that more than 35% variation is observed in the effective channel length due to process variation in 130nm technology nodes and below [2]. This can cause over 20 times variations in the leakage current with particularly strong impact on AMS designs. By approaching technologies with fundamentally random behavior, the continued seemingly unending physical feature size scaling of CMOS transistors is expected to reach its boundary at size of 10nm as foretasted in the International Technology and Roadmap for Semiconductors (ITRS) by 2021 [3]. After over five decades of advances, the semiconductor industry is threatened to no more keeping pace with Moores Law as depicted in Figure 1.

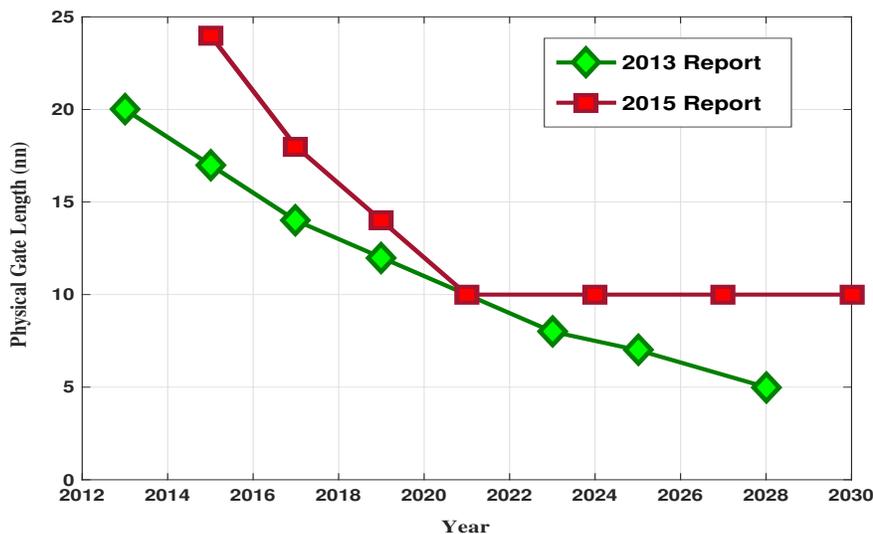


Fig. 1: Revised Moore’s Predictions: End of the Road [3]

As we approach the physical limits of device scaling, it is essential and imperative to develop accurate, efficient and reliable AMS verification methodologies for a sophisticated minimum cost VLSI design flow. Even though analog and mixed-signal blocks occupy small portions of todays’ chips, they may cause a lot of design problems. Successful AMS designs in terms of correctness, handling complexity, and time to market require large efforts in the development of verification methodologies. In fact, AMS designers have to deal with an inherited continuous state space of conservative quantities, highly nonlinear relationships, and a large number of uncertainties that worsen circuit behavior. The above mentioned hurdles present a key reliability issue and enlarges the number of possibly critical verification scenarios to infinity. To deal with these major limitations, designers often overdesign AMS circuits and consider excessive performance guard-bands using large safety margins. Such over-designing leads to circuits that are overly pessimistic in terms of power, area and speed specifications. However, the nano scale era will be much less forgiving for this design practice. Moreover, the fundamental heterogeneity of analog and digital blocks modes of operation in

AMS circuits has hindered the advancement of AMS verification methodologies compared to its digital counter part. Indeed, it is not sufficient to check the properties of AMS circuit behavior by simulations of single trajectories, e.g., when it is required to verify whether the specifications are not violated for all possible initial conditions and process parameters. Computing the set of all solutions is often referred to as reachability analysis, which spans from reachability algorithms of graphs to discrete systems, timed automata, and eventually to circuits with continuous and mixed- signal (mixed discrete/digital-continuous/analog) dynamics [4].

This technical report presents a unified modeling and verification methodology for AMS designs. The reachability analysis of nonlinear circuit dynamics is modeled as a system of stochastic recurrence equations (SSREs), a class of systems that has received only limited attention in the reachability analysis literature [5, 6]. Our ultimate goal is to extend the semi-formal reachability analysis scheme proposed in [7] by reducing the over-approximation of the reachable set. To do so, an intertwined forward/backward reachability refinement is proposed. The corrected reachable bounds are then used to verify whether the circuit withstands noise and process variation with regards to a predefined performance constraints. The robustness of the circuit against these uncertainties is quantified in terms of yield rate using a statistical hypothesis testing approach.

The rest of this report is organized as follows: Most relevant related works are sketched and compared in Section 2. Preliminaries are presented in Section 3. Details of the proposed methodology are described in Section 4. Thereafter, experimental results on two circuits are reported in Section 5. Finally, conclusions and future work are presented in Section 6.

2 Related Work

Except for simple circuits, verification of transient properties of analog and mixed-signal designs using paper and pencil proof method is rarely possible. Traditionally, simulation is the de facto standard for AMS circuits verification. Despite its generality and ease of use, simulation based verification is irreverently inaccurate due to the sampling based nature of the approach. Furthermore, running multiple simulations cannot guarantee 100% verification coverage owing to the presence of corner cases for which the required circuit performances are violated. Furthermore, with the technology scaling and the increasing circuits complexity, simulation techniques scale purely with the number of uncertain parameters in the circuit model and rapidly reach their limits in terms of simulation time [8]. In fact, verifying a PLL design for only few thousand traditional Monte Carlo simulation would take days or even weeks of a runtime to finish [9]. To accelerate the convergence of simulation techniques, several sampling techniques have been developed such as Latin Hypercube Sampling (LHS) [10] and Importance Sampling (IS) methods [11]. However, the number of required simulation runs is still prohibitive for verifying complex AMS circuits.

With the goal of generalizing the applicability and automating the verification process, formal methods for verifying such systems have been the subject of much study [12]. Formal methods have gradually become a necessity to cope with the difficulty of analytical methods and costly simulation. It ensures that the circuit exhibits the desired behavior through the conformance of a formal model of the circuit to certain design specifications. The latest advancement in formal verification of analog and mixed-signal circuits can be classified to theorem proving [13, 14], equivalence checking [15], statistical runtime verification [16], and reachability analysis [6]. Reachability analysis of a given system seeks to determine the tra-

jectories that the system -or more accurately, a mathematical model representing the system- can attain and the set of the states it can enter. Therefore, it can provide a complete coverage of the system dynamics that are intractable with a finite number of simulation runs and can give additional insights to falsify or approve the system properties. Reachability analysis can be conducted on two manners: *Forward reachability analysis*, and *Backward reachability analysis*. Various techniques to approximate reachable sets have preceded this work [17, 18]. Recently, the authors of [6] proposed a reachability analysis algorithm based on zonotopes and a continuization method. The proposed method is formulated to verify the lock time property of a charge-pump PLL design. Although their proposed algorithm shows a comparable speed-up over traditional Monte Carlo simulations, the applicability of this method is limited to circuits exhibiting linear behavior. Furthermore, their PLL design was verified under ideal conditions (i.e., no initial conditions nor noise was considered), which does not reflect the real need of verifying such designs. To overcome these issues, the authors of [5] extended the work in [6] by introducing substrate and power supply noise induced jitter. In addition, they proposed a backward correction scheme with a stability calibration method along with the forward zonotope method. However, this approach suffers from the curse of dimensionality and scaling. Indeed, the analysis of systems with more than a few continuous variable is very complex. To cope with the nonlinear behavior of most of the AMS designs, the authors of [7] proposed a semi-formal reachability analysis approach based upon qualitative simulations. Despite enclosing all possible behaviors and thus reaching a good verification coverage, the proposed approach highly over-approximates the circuit reachable states.

In this report, we present a technique that : (1) is faithful to the nonlinear nature of analog and mixed-signal circuits and so does not resort to linearization; and (2) reduces the reachable set over-approximation errors computed during a forward reachability analysis approach. We compute the refined more accurate reachable set boundaries using backward reasoning.

3 Preliminaries

In this section, we define the terminology that will be used for analog and mixed-signal modeling. We also present some background on the Latin Hypercube sampling technique, and statistical hypothesis testing along with their definitions.

3.1 Analog and Mixed-Signal Uncertainties

The design of analog and mixed-signal circuits is very challenging due to the significant impact of non-idealities such as process variation and noise on circuit performance.

3.1.1 Process Variation Uncertainty

Process variation can be defined as the deviation of parameters from their desired nominal values due to limited controllability of design and fabrication process. Random dopant fluctuation, annealing effects, lens aberration, chemical-mechanical polishing and lithographic limitation are some of the factors contributing to process variations. These process variations manifest into variations in device parameters such as threshold voltage, oxide thickness, and length of a transistor [19]. These variations can be broadly classified depending on the mechanism by which the variation is engendered into:

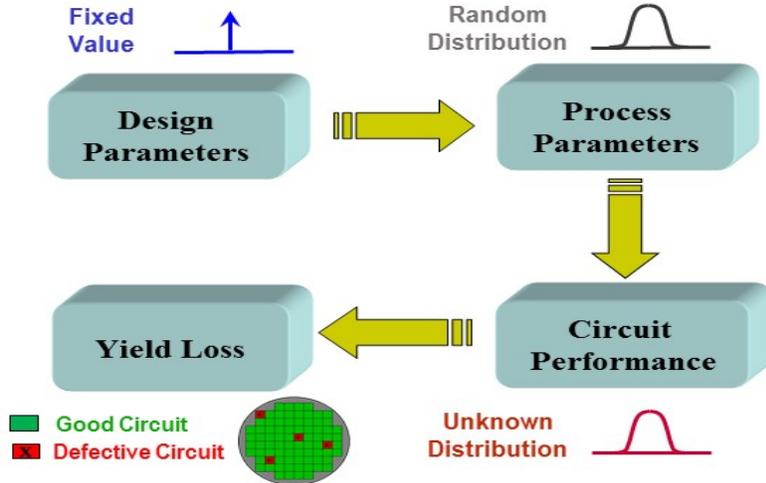


Fig. 2: Illustration of the Process Variation Effect on Circuit Yield

1. *Stochastic variations*: The manufactured parameter values p_i possess some inherent randomness determined by a random variable with a standard variation σ_{p_i} around the mean value $\mu_i = p_{nom}$ with p_{nom} being the nominal parameter value according to the following equation:

$$p_i = p_{nom} + \sigma_{p_i} \cdot \phi(\theta_i) \quad (1)$$

where $\phi(\theta_i)$ represents the stochastic process corresponding to parameter p_i .

2. *Systematic variations*: On the contrary to random variations, systematic variations affects circuit parameters equally. It has deterministic values and the process variation values are conclusively determined.

In this work, we will focus on stochastic device variations which are very hard to eliminate and verify. This type of variation significantly affects performance metrics of analog and mixed-signal circuits. Moreover, the impact of process variation on circuit performance is unpredictable. Thereby, it is very difficult to infer the relation process variation/performance variation. In other words, a predefined random distribution in the circuit parameters does not in general lead to performance variations with same random distributions but it has rather an unknown distribution that is hard to infer. A direct consequence of the circuit performance variation is a loss in post design parametric yield due to unmet specifications. For most of the analog circuits, circuit performances such as gain and oscillation frequency have an upper and/or lower stringent specification limit, and the circuits that exceed the tolerated specification limit during fabrication must be discarded. In other words, these circuits are failible and does not meet satisfy their prescribed specification, Hence, they are considered to be defective resulting in a higher dies probability of failure (i.e., lower yield). Figure 2 summarizes the process variation concept and visualizes the relationship between process variation and the yield loss.

3.1.2 Jitter Uncertainty

In an ideal oscillatory behavior, the spacing between transitions is constant as shown below.

$$Y_{out} = A.f(2\pi f_0 t + \phi(t)) \quad (2)$$

where the function f is periodic in 2π and $\phi(t)$ denotes the deviation in phase due to different noise sources such as supply and substrate noise. In real designs, however, the transition spacing is variable and inversely proportional to the oscillation frequency f_0 . This is known as jitter uncertainty effect. It can be defined as the deviations in the circuit output transition from their ideal positions. These deviations can either be leading or lagging the ideal position. Mathematically, it is the integral of spectral phase density $\mathcal{L}(f)$ between two frequency limits f_1 and f_2 as given in Equation (3).

$$\sigma = \frac{1}{2\pi f_0} \sqrt{\int_{f_1}^{f_2} 2\mathcal{L}(f) df} \quad \text{with} \quad \mathcal{L}(f) = \frac{\text{Power Density}}{\text{Carrier Power}} \quad (3)$$

Jitter is present in every active and resistive component, but it is most severe in oscillators. The most prominent factors that contribute to random timing jitter include broadband noise (a combination of white and flicker noise), phase noise, spurs, slew rate, and bandwidth. Both phase and broadband noise are random, whereas spurs have deterministic responses caused by various identifiable interference signals, such as crosstalk and power supply coupling. Accordingly, jitter can be classified to random and deterministic categories. Based on the nature of the behavior governing the circuit, jitter can also be classified to Phase Modulation (PM) jitter exhibited by driven circuits such as Phase Frequency Detector in PLLs, and Frequency Modulation (FM) jitter exhibited by autonomous systems such as analog oscillators [20].

3.2 Analog and Mixed-Signal Modeling

Analog and mixed-signal circuits contain two different types of components: those with continuous dynamics (analog components) and those with discrete dynamics (digital components). Despite their heterogeneous nature, a careful time domain discretization allows a unified description of all the AMS circuit components. Due to the statistical behavior that AMS circuits exhibit in the presence of uncertainties (such as noise and parameter variability), we are interested in modeling AMS circuits transient behavior as a System of Stochastic Recurrence Equations (SSRE) [21], which is a formalism that allows to capture the statistical properties of the circuit behavior in a unified discrete-time description. Moreover, the temporal properties of these mixed components and their interactions can be expressed as SSRE. In what follows, we explain the SSRE notations and detail the conversion process of circuit equations and properties to SSREs. A system of recurrence equations is a set of relations between consecutive elements of a sequence. The notion of recurrence equations to describe circuits using the normal form: *generalized If-formula* was first proposed by Al-Sammam [22]. In addition, a stochastic recurrence equation can be generated for the case of continuous circuits using the discrete version of their Stochastic Differential Equation (SDEs) [23]. In the following, we briefly present the SSRE theory. An SSRE is a set of SREs with stochastic processes. Let us consider the following Itô process $\{X_t, 0 \leq t \leq T\}$ SDE [24]:

$$dX_t(\omega) = f(X_t(\omega))dt + \sigma(X_t(\omega))dW_t(\omega) \quad (4)$$

where the stochastic variable W_t is a Brownian motion [25] (see Definition 1), the initial condition ($X_{t_0} = X_0$) and the diffusion coefficient σ are deterministic variables.

Definition 1. (*Brownian Motion*) A scalar standard Brownian process, or standard Winer process over $[0, T]$ is a random variable W_t that depends continuously on $t \in [0, T]$ and satisfies the following conditions:

Condition 1 $W(0) = 0$ with probability 1.

Condition 2 For $0 \leq s < t \leq T$ the random variable given by the increment $W_t - W_s$ is normally distributed with mean zero and variance $(t - s)$ ($W_t - W_s \sim \sqrt{t - s}\mathcal{N}(0, 1)$).

Condition 3 For $0 \leq s < t < u < v \leq T$ the increments $W_t - W_s$ and $W_v - W_u$ are independent.

By integrating Equation (4) between s and $s + \Delta s$, we will have:

$$dX_{s+\Delta s}(\omega) = X_s(\omega) + \int_s^{s+\Delta s} f(X_{s+\Delta s}(\omega))dt + \int_s^{s+\Delta s} f\sigma(X_{s+\Delta s}(\omega))dW_{s+\Delta s}(\omega) \quad (5)$$

The Euler scheme [26] consists in approximating the integral Equation (5) by the following iterative scheme:

$$\bar{X}_{s+\Delta s}(\omega) = \bar{X}_s(\omega) + f(\bar{X}_s(\omega))\Delta s + \sigma(W_{s+\Delta s}(\omega) - W_s(\omega)) \quad (6)$$

Definition 2. (*Generalized If-formula*) The generalized **If-formula** is a class of symbolic expressions that extend recurrence equations to describe discrete behavior of digital AMS blocks. Let i and n be natural numbers. Let \mathbb{K} be a numerical domain in $(\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}$ or $\mathbb{B})$, an **If-formula** is one of the following:

- A variable $X_i(n)$ or a constant C that takes values in \mathbb{K}
- Any arithmetic operation $\diamond \in \{+, \cdot, \times, \div\}$ between variables $X_i(n)$ that take values in \mathbb{K}
- A logical formula: any expression constructed using a set of variables $X_i(n) \in \mathbb{K}$ and logical operators: not, and, or, xor, nor, . . . , etc.
- A comparison formula: any expression constructed using a set of variables $X_i(n) \in \mathbb{K}$ and comparison operators $\Delta \in \{\neq, =, <, \leq, >, \geq\}$
- An expression $If(X, Y, Z)$, where X is a logical formula or a comparison formula and Y, Z are any generalized **If-formula**.

Here, $If(X, Y, Z) : \mathbb{B} \times K \times K \longrightarrow \mathbb{K}$ satisfies the axioms:

1. $If(\text{true}, X, Y) = X$
2. $If(\text{false}, X, Y) = Y$

Definition 3. (*SSRE*) Consider a set of variables $X_i(n) \in \mathbb{K}$, $i \in V = 1, \dots, k, \omega \in \mathbb{R}$, an SSRE is a system of the form:

$$X_i(\omega) = f_i(X_j(\omega)\gamma), (j, \gamma) \in \varepsilon_i, \forall \omega \in \mathbb{R} \quad (7)$$

where $f_i(X_j(\omega)\gamma)$ is a generalized **If-formula** of the recurrence stochastic differential equation given in Equation (6). The set ε_i is a finite non empty subset of $1, \dots, k \times \mathbb{N}$. The integer γ denotes the delay.

3.3 Latin Hypercube Sampling

To study parameter variation effects on the behavior of AMS circuits, an optimal exploration of the variation domain of the parameter values is very important in order to achieve a good accuracy and avoid non-informative verification runs. Traditional sampling techniques (e.g., Pseudo Random Sampling (PRS), Fractional Factorial, Central Composite, etc.) only arrange parameter values at some specific corners in the parameter space and can not handle multivariate stochastic parameters especially in terms of correlation. Consequently, when performing verification, it cannot mimic the circuit behavior in a global system parameter space. We first look at PRS as applied in the estimation of circuit failure in order to justify the use of Latin Hypercube Sampling (LHS). It has been demonstrated that the LHS technique gives samples that could reflect the integral distribution more effectively with a reduced samples variance [27]. Figure 3 illustrates the differences while using Monte Carlo PRS and Gaussian Monte Carlo LHS of a random normal parameter of transistor width for 1000 trials.

In the sequel, we explain the Latin Hypercube Sampling (LHS) main steps to generate a sample size N from n AMS circuit parameter variables $\xi = [\xi_1, \xi_2, \dots, \xi_n]$ with the probability distribution function $f_x(\cdot)$

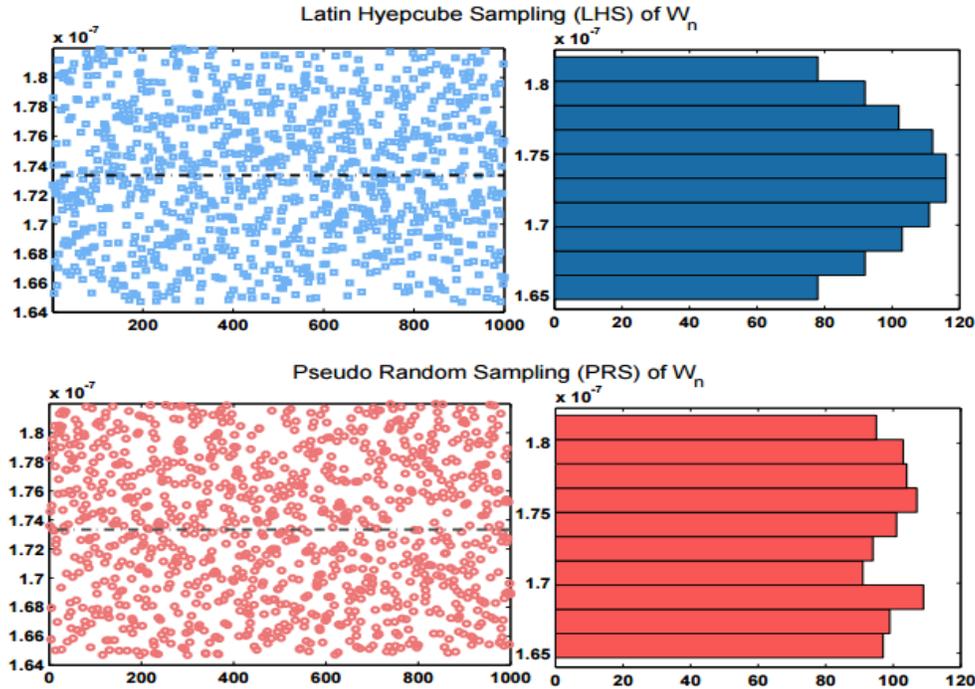


Fig. 3: Sampling Differences between Monte Carlo LHS and PRS

First, the approach involves the partitioning of the range of each circuit parameter variable into N non overlapping intervals on the basis of equally probability size $\frac{1}{N}$. One value from each interval is randomly selected w.r.t. the conditional probability density in the variation interval defined by the technology library. The N values thus obtained for ξ_1 , are paired in a random manner with the N values of ξ_2 . These N pairs are combined in a random manner with the N values of ξ_3 to form N triplets, and so on, until a set of $N \times n$ -tuples is formed. The choice of this sampling technique can be justified by its variance sampling

reduction, which results in a better sampling coverage and consequently a better verification coverage [28].

3.4 Hypothesis Testing

Hypothesis testing [29] uses statistics to make decisions about the acceptance or the rejection of some statements based on the data from random samples. In this technique, the property of interest is formulated as a null hypothesis (H_0) which is tested against an alternative hypothesis (H_1). If we reject H_0 , then the decision to accept H_1 is made.

Definition 4. *Given the property \mathcal{P} within the ambit of a null hypothesis H_0 , a significance level α , and a test statistic T , hypothesis testing is the process of verifying whether a circuit \mathcal{S} satisfies H_0 with a probability greater than or equal to α (i.e., $\mathcal{S} \models Pr(T) \geq \alpha$).*

As depicted in Figure 4, Hypothesis testing can be a one side test (upper test or lower tes) or two sided. In the case of a two sided test for example, we can verify if a variable X is within a bounded region $[x_1, x_2]$ as follows:

$$H_0 : P(x_1 < X < x_2) = P(X < x_2) - P(X < x_1) = 1 - \alpha \quad (8)$$

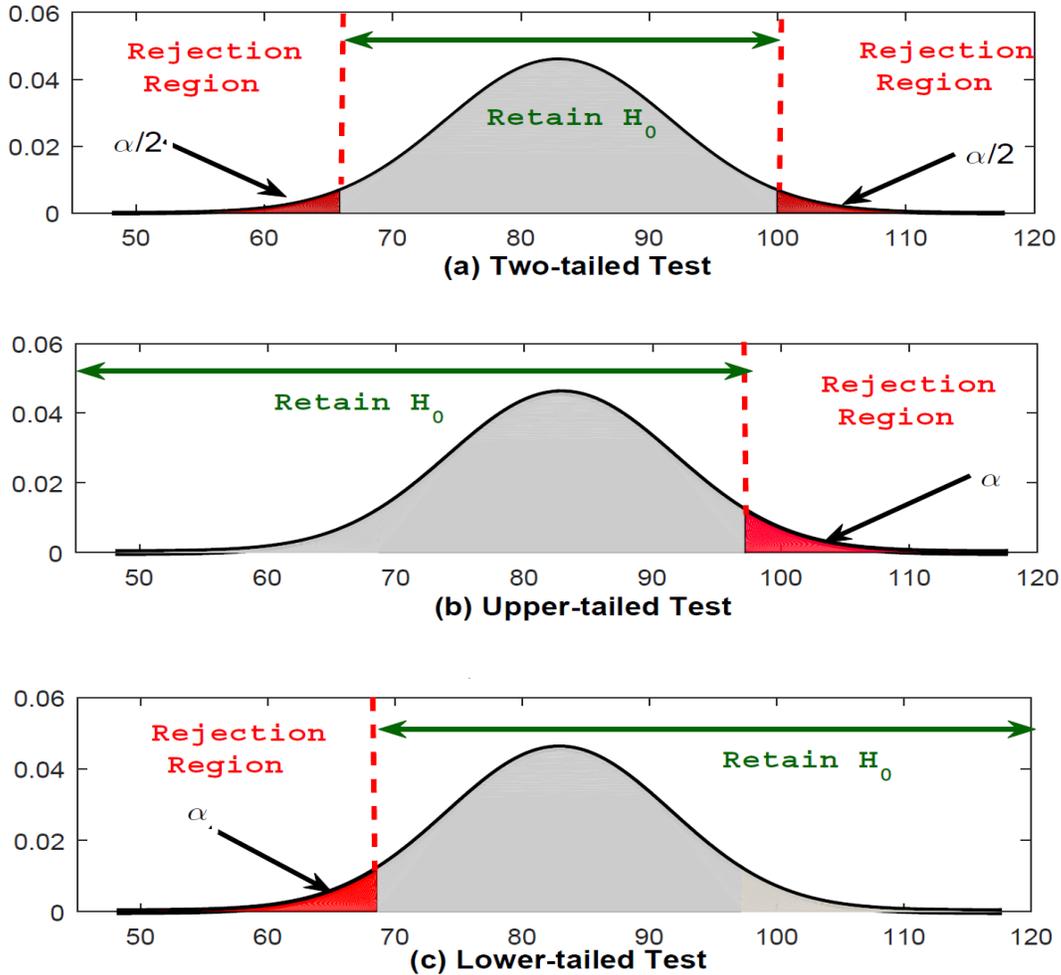


Fig. 4: Hypothesis Testing Concept

Following are the central steps to carry out hypothesis testing:

1. Elucidate the property to be verified and formulate it as H_0 and H_1 .
2. Specify the appropriate level of significance α and determine the type of the test, namely, upper test, lower test or two sided test.
3. Select the appropriate test statistic.
4. Compute the critical region or p-value of the test statistic.
5. Compute the test statistic of the observed value for the original data.
6. Make the decision of accepting or rejecting the null hypothesis H_0 . If the computed test statistic falls in the critical region, then the null hypothesis is rejected, otherwise H_0 is accepted.

The performance criteria of this approach is related to two types of errors as summarized in Table 1:

Table 1: Circuit Verification Classification

	Passed	Failed
Good Circuit	✓	Type I error
Defective Circuit	Type II error	✓

Type I error (α) or false positive, the null hypothesis H_0 is true but the decision based on the testing process erroneously rejected it. In other words, it represents the probability of accepting H_0 when H_1 holds.

Type II error (β) or false negative, the null hypothesis H_0 is false but the testing process concludes that it should be accepted. In other words, it corresponds to the probability of accepting H_1 when H_0 holds.

4 Proposed Methodology

An overview of the proposed methodology for intertwined forward/backward reachability analysis is shown in Figure 5. Given a nonlinear AMS circuit description, SSREs that express its stochastic behavior under noise perturbation are generated. The proposed SSRE formalism features a sound treatment of noise. It actually allows a consistent consideration of the noise effect to which the circuit is incurred during the reachability analysis process. Then, parameter values from a certain distribution of the parameter space are derived using the efficient LHS technique. Next, reachability bounds of the AMS circuit behavior for a continuous set of initial conditions, and under the derived circuit parameters are generated using a novel intertwined forward/backward reachability analysis technique. The reachability computed using SSRE circuit model with parameters selected by the LHS procedure and for initial conditions that are defined within intervals (n -cubes) is based on the global optimization theory. The SSRE is not solved for every initial condition value but it employs the reachability analysis algorithm to optimize the search for the global extremum. The output of this step is a refined reachability set generated from the backward reachability correction that includes all possible actual behaviors (trajectories) of the circuit transient behavior. The main advantage of the proposed verification scheme is its generality and scalability.

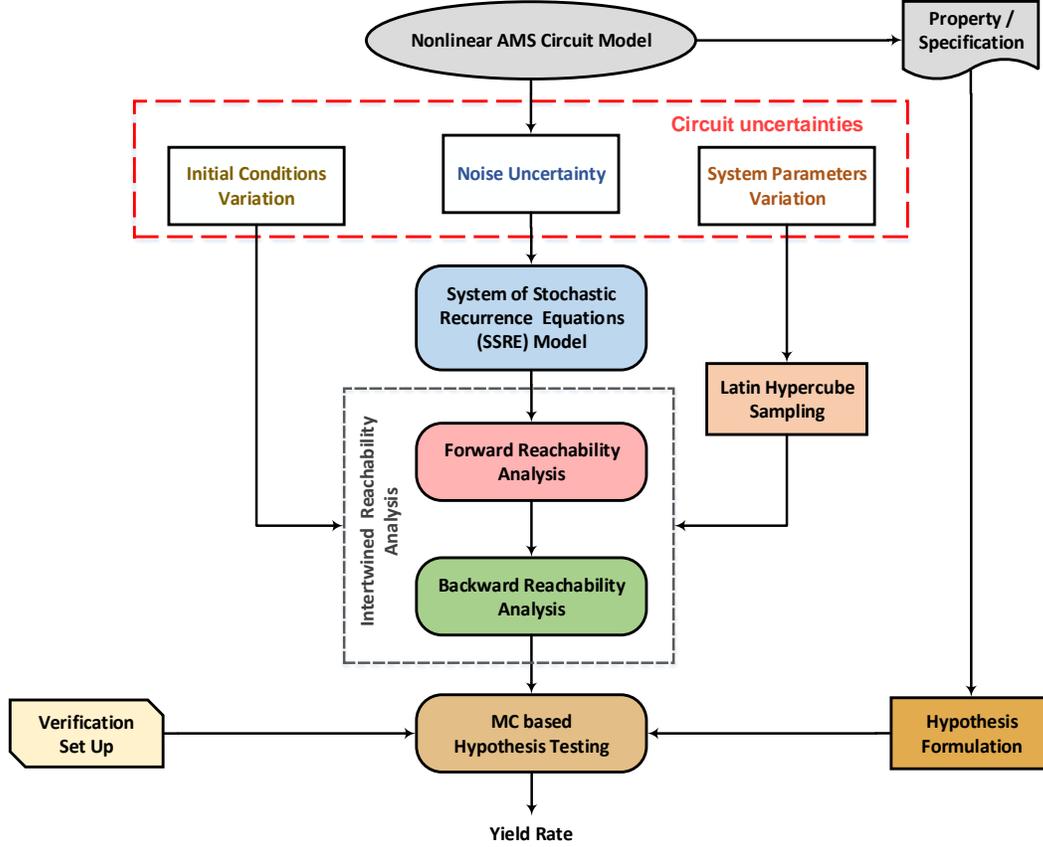


Fig. 5: Proposed Verification Methodology

In fact, it does not make any assumption about the nature of the AMS circuit dynamics: it works for any AMS design with linear and nonlinear behavior. Next, appropriate null and alternative hypotheses are formulated from a certain SSRE specification of the AMS circuit under verification. For each selected circuit parameters in the reachability iteration, Hypothesis Testing based Monte Carlo (MC) technique is conducted to estimate the circuit parametric failure which refers to failures caused by the deviation between manufactured circuit parameter values and intended parameter values. Each time the null hypothesis H_0 , which represents the desired circuit property/performance, is rejected, we draw a conclusion that the circuit fails to comply with its property and so we increment the number of circuit failures $N_{failure}$. Finally, the AMS circuit yield rate is computed based on the probability of failure $P_{Failure}$ as follows:

$$P_{Failure} = \frac{Nb. \text{ of Rejected } H_0}{Total \text{ Nb. of MC Trials}}$$

$$Yield = 1 - P_{Failure}$$

4.1 Forward-Backward Reachability Analysis

Definition 5. (*Reachability Analysis*) *Reachable set (or bounds) is the collection of all possible trajectories or states of the AMS circuit transient behavior originated from an interval of initial conditions. Mathematically, this can be defined as follows:*

$$X_{Reachable_set} = \{x \in \mathbb{R}^{N_x} \mid \underline{X}_L \leq x \leq \overline{X}_U\} \quad (9)$$

where \underline{X}_L is the lower reachable bound of the reachable set (or region) and \overline{X}_U is the upper bound of the reachable set.

The proposed intertwined reachability analysis approach is shown in Figure 6. First, the forward reachability analysis trajectories are calculated starting from the initial condition uncertainty region (in this case it has rectangular geometrical form) at each time step and projected to a reachable set as depicted in Figure 6 (a). Second, the backward reachable set is computed wherein the obtained forward reachable set is considered as the initial region of uncertainty. Trajectories starting from all points in the final backward reachable set are simulated in reverse time in order to screen out erroneous over-approximated reachable sets as illustrated in Figure 6 (b).

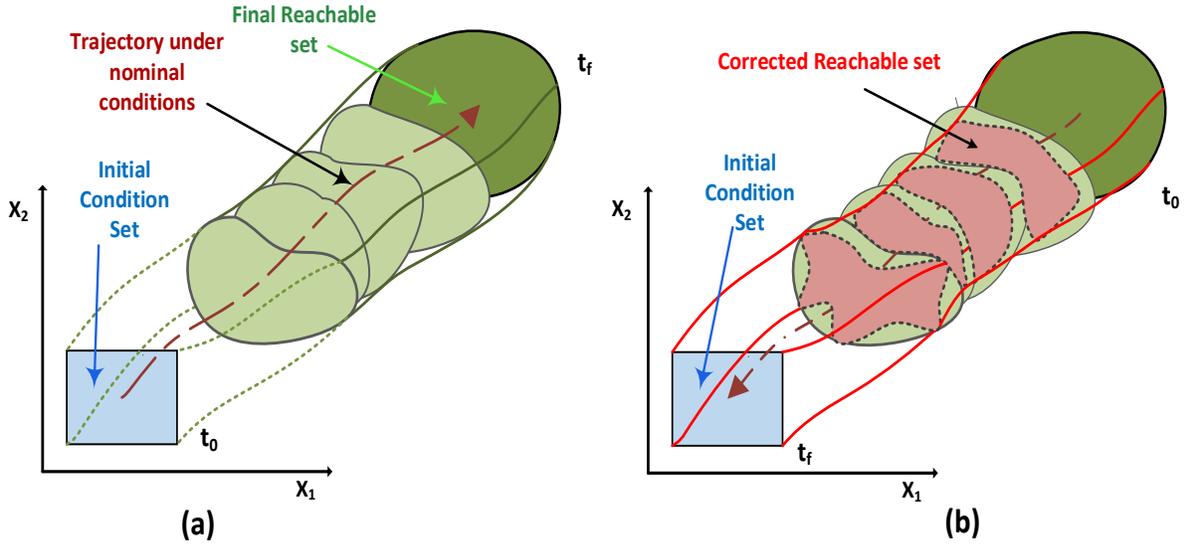


Fig. 6: Intertwined Reachability Analysis Concept

The definition of reverse time dynamics of the SSRE model allows the forward/backward reachability exchange. The detailed implementation of the intertwined reachability analysis approach is summarized in Algorithm 1. Given an interval system of stochastic differential equations (an SSRE whose initial conditions are intervals), the algorithm defines the region of uncertainty of the circuit output as an hypercube (n-cube) at time t_0 (Lines 3 and 18). Hence, the reachability analysis problem at a given simulation time point t^* for each circuit output (or state space) is equivalent to finding the maximum and minimum bounds of the SSRE model. In the proposed algorithm, the reachability analysis problem is so cast into a constrained multivariable nonlinear global optimisation problem. It was proven that under continuity condition, it is sufficient to compute the evolution of the external surface of the uncertainty region [30]. This means that to calculate the reachable bounds, it is sufficient to compute the trajectories emanating from the external surface of the region of the uncertainty region. The extreme functions (Max and Min) at a specific time t^* of the system equations $SSRE(t^*, j, X_{ext}), \forall j = 1, \dots, N_x$, which bound the circuit behavior, are first computed using the forward reachability analysis. We used the MATLAB Optimization solver [31] based on trust regions (Lines 1 to 15) to get these extreme functions of $SSRE(t^*, j, X_{ext}), \forall j = 1, \dots, N_x$ by fixing the time variable to t^* and constraining the circuit transient behavior to evolve over the external uncertainty region (Line 7). The

computed optimization point is then passed to the SSRE model, which uses X_{ext} as initial conditions and generates a partial derivatives (gradient) values that are used to control the stability of the reachability analysis (Line 8). The algorithm terminates if the optimisation method considers $SSRE(t^*, j, X_{ext}), \forall j = 1, \dots, N_x$ as an extremum; Otherwise the gradient values are used to select new points from the external uncertainty region X_{ext} and the above described steps are repeated. Although this step guarantees the completeness of the reachability set, the upper and lower obtained reachable sets are highly overbounded due to the *wrapping effect*. One way to tighten the reachability space is to conduct a backward reachability (Lines 16 to 30). Starting from the final computed set (Line 18), the backward optimization algorithm is now performed on the AMS circuit SSRE reversed in time in order to compute backwards the reachability bounds and consequently correct the overbounded forward reachability set.

Algorithm 1 Intertwined Forward/Backward Reachability Analysis

Require: $SSRE$: AMS Circuit Model, X_0 : Interval of Initial Conditions, P : Circuit parameters, N_x : Number of state variables, t_0 : Initial time, t_f : Final time

```

1: for  $t_1^* \leftarrow t_0$  to  $t_f$  do
2:   for  $j \leftarrow 1$  to  $N_x$  do
3:      $X_{ext}(t_1^*) = \text{Generate}(X_0)$  ▷ external surface of the uncertainty region
4:      $X_{max}(t_1^*, j) = -\infty$ 
5:      $X_{min}(t_1^*, j) = \infty$ 
6:     for each state variable  $X_{ext}(j) \in X_{ext}$  do
7:        $Const = \text{UpdateConstar}(j, SSRE, P, X_{ext})$ 
8:        $Grad = \text{UpdateGrad}(j, t_1^*, SSRE, P, X_{ext})$ 
9:        $[X_{max}(t_1^*), X_{min}(t_1^*)] = \text{Global\_Opt}(SSRE, j, t_0, t_1^*, P, X_{ext}), Grad, Constr)$ 
10:    end for
11:     $B_{L_{Forward}}(t_1^*) \leftarrow X_{min}(t_1^*)$ 
12:     $B_{U_{Forward}}(t_1^*) \leftarrow X_{max}(t_1^*)$ 
13:     $\text{update\_forward}(t_1^*, \Delta_t)$ 
14:  end for
15: end for
16: for  $t_2^* \leftarrow t_f$  to  $t_0$  do
17:   for  $j \leftarrow 1$  to  $N_x$  do
18:      $X_{ext}(t_2^*) = \text{Generate}(B_{L_{Forward}}(t_2^*), B_{U_{Forward}}(t_2^*))$  ▷ external surface of the approximate reachability bounds
19:      $X_{max}(t_2^*, j) = B_{U_{Forward}}(t_2^*, j)$ 
20:      $X_{min}(t_2^*, j) = B_{L_{Forward}}(t_2^*, j)$ 
21:     for each state variable  $X_{ext}(j) \in X_{ext}$  do
22:        $Const = \text{UpdateConstar}_B(j, SSRE, P, X_{ext})$ 
23:        $Grad = \text{UpdateGrad}_B(j, t_2^*, SSRE, P, X_{ext})$ 
24:        $[X_{max}(t_2^*), X_{min}(t_2^*)] = \text{Global\_Opt}_B(SSRE, j, t_f, t_2^*, P, X_{ext}), Grad, Constr)$ 
25:    end for
26:     $B_{L_{corrected}}(t_2^*) \leftarrow X_{min}(t_2^*)$ 
27:     $B_{U_{corrected}}(t_2^*) \leftarrow X_{max}(t_2^*)$ 
28:     $\text{update\_backward}(t_2^*, \Delta_t)$ 
29:  end for
30: end for

```

5 Applications

In this section, we validate our proposed intertwined forward reachability analysis with backward correction methodology on a Tunnel Diode Oscillator (TDO), and Phase Locked Loop (PLL) mixed-signal design. All computation and AMS circuit models were integrated in MATLAB environment and were run on a 64-bit Windows 7 machine with 2.8 GHz processor and 24 GB memory. The hypothesis testing is conducted for a level of significance $\alpha=5\%$.

5.1 Nonlinear Analog Circuit: Tunnel Diode Oscillator

Oscillators are integral parts of today's Integrated Circuits (ICs) which require a time reference (clock). One main salient feature of a perfect oscillator is its ability to provide an accurate time reference even in an imperfect environment. As a first application for our proposed methodology, we consider the Tunnel Diode Oscillator (TDO) circuit shown in Figure 7. It exhibits an oscillatory behavior when operating in the negative resistance region of the diode V-I characteristic (see Figure 8). It was reported that its oscillation property is affected by the temperature, the conductance $G = 1/R$ and the initial conditions [32].

Our goal is to verify the oscillation property in the presence of process variation in $G = 1/R, L$, and C and under a range of initial conditions $X_0 = [X_{0_{min}}, X_{0_{max}}]$ lying in a specific continuous range of values at a nominal temperature ($T = 200K$). The component values R, L and C and the input voltage and current values have been chosen from [32]. The metric of interest in this experiment is the oscillation property with frequency f_{osc} . The desired specification for the TDO is:

$$H_0 : Oscillation \wedge f_{osc} \in [71, 74]MHz \quad (10)$$

$$H_1 : Lock_up \parallel f_{osc} \notin [71, 74]MHz$$

The circuit was simulated for different conductance values G . Figure 9 depicts the output voltage V_c variation in the case of $G = 5m\Omega^{-1}$. It can be seen that the circuit, in this case, generates a periodic signal between $0V$ and $0.5V$ (Figure 9 (a)). Moreover, the state space representation given in Figure 9 (b) confirms the successful oscillatory behavior of the TDO

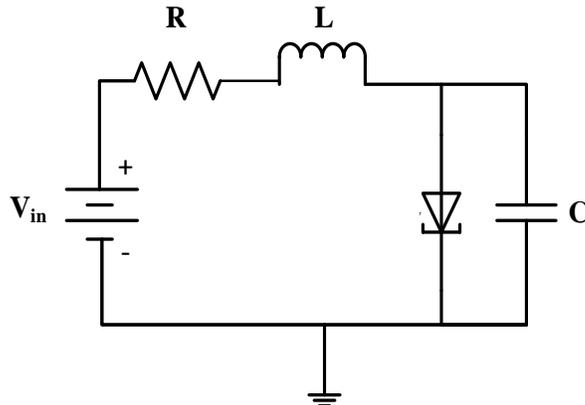


Fig. 7: Tunnel Diode Oscillator Schematic

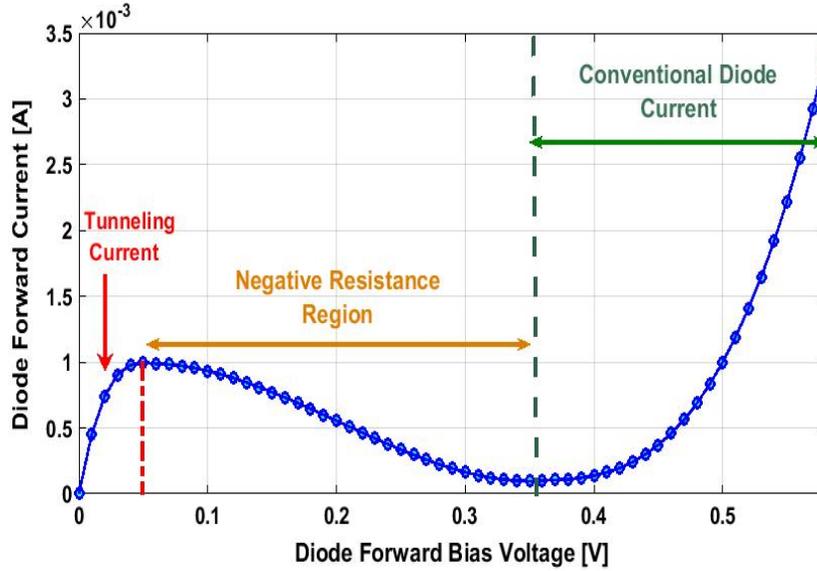


Fig. 8: Tunnel Diode V-I characteristic

circuit. Nonetheless, for a conductance value of $G = 4.13m\Omega^{-1}$, the TDO circuit fails to start-up and sustain oscillation (see Figures 9 (c),(d)). The voltage output actually settles to a fixed value which causes the circuit lock-up and hence violates the desired property given in (10). Figure 10 shows a state space representation of the reachable set as well as the corrected backward reachable set in the case of lock up ($G = 4.13m\Omega$). As it can be noticed, the resultant state space reachable bounds settle to a fixed region which holds up the free stable oscillatory behavior. This confirms the results shown in Figures 9 (c),(d). The oscillation verification performance constraint is therefore violated and the verification fails in this case.

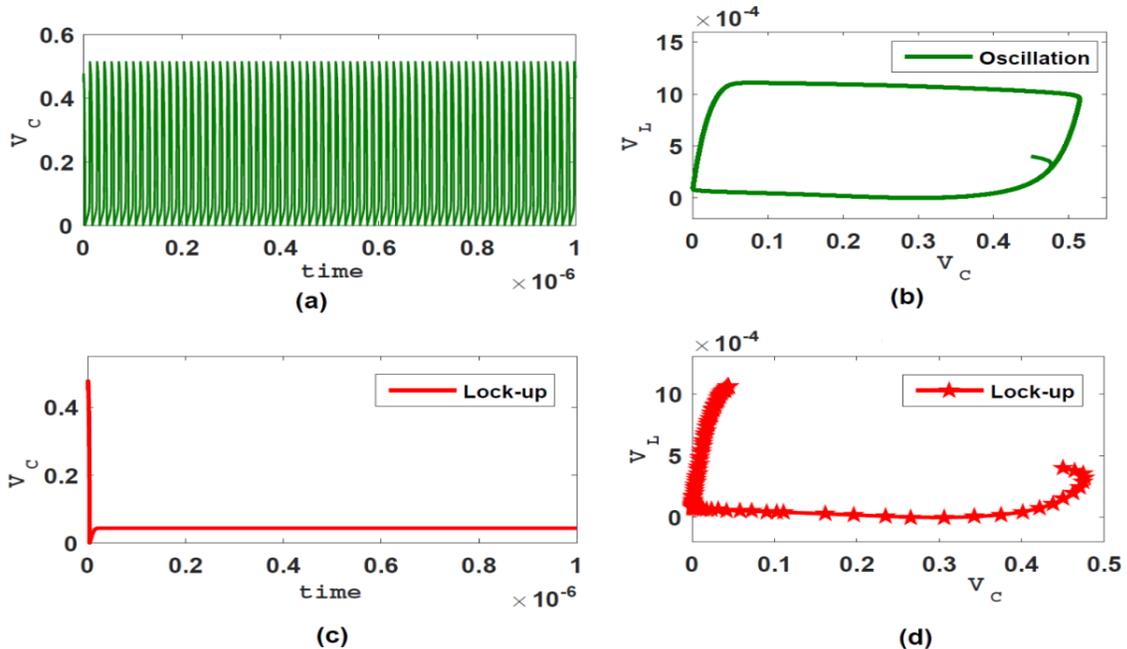


Fig. 9: Tunnel Diode Oscillator Output for Different Conductance G

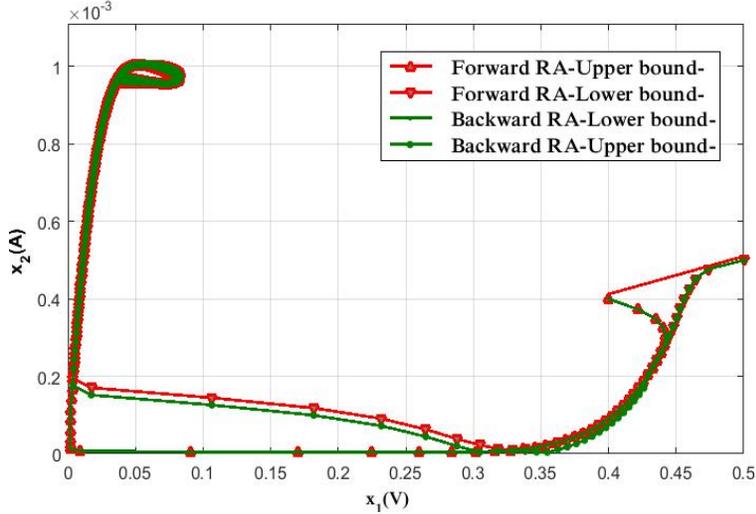


Fig. 10: Intertwined Reachability Analysis in the Lock-up case

The TDO is first verified using a variant of the Monte Carlo technique called Monte Carlo-Jackknife (MC-JK) [10], where the failure probability of the TDO property (see Equation (10)) is verified under process variation uncertainties. For the process variations in the circuits parameter, 1000 samples were drawn using LHS from the parameters space. For a fair comparison, these same parameters points were passed to both the MC-JK method and our intertwined forward/backward reachability analysis method. The results of the yield rate estimation are summarized in Table 2 in the case of ($G = 5m\Omega$).

Table 2: TDO Yield Estimation Comparison with Monte Carlo-Jackknife Method

	MC-JK [10] (%)	Proposed Methodology (%)	Relative Error (%)
IC Variation Only	92.1	87.7	4.4
PV & IC Variations	88.6	83.3	5.3
Jitter & IC Variations	82.5	76.1	6.4
Jitter & PV & IC Variations	79.9	72.8	7.1

It can be noticed that our proposed methodology gives better yield estimation by detecting failures that were not detected by the Monte Carlo-Jackknife (MC-JK) technique [10]. In this sense, the obtained parametric yield rate are over-estimated with up to 7% relative error in light of jitter, process variation and initial conditions uncertainties. In Table 3, we present the results of yield estimation for the same scenarios as in Table 2 yet for a forward only reachability scheme. The results show the better verification coverage offered by our proposed intertwined reachability technique.

Table 3: TDO Yield Estimation Comparison with Forward Solely Reachability Method

	Forward RA [7] (%)	Proposed RA (%)	Relative Error (%)
IC Variation Only	89.3	87.7	1.6
PV & IC Variations	85.8	83.3	2.5
Jitter & IC Variations	79.4	76.1	3.3
Jitter & PV & IC Variations	77.5	72.8	4.7

5.2 Mixed-Signal Circuit: PLL based Frequency Synthesizer

The PLL based frequency synthesizer is a basic and essential block of modern communication systems. It is basically a feedback circuit that tries to reduce the phase error between the input and the reference signals. In this case study, we consider a simple frequency synthesizer, that generates an output signal whose frequency is N times the frequency of the reference signal. We consider for this application a *Sine wave* reference signal with a frequency of ω_0 , the PLL output is a *Cosine wave* signal with frequency $N \times \omega_0$.

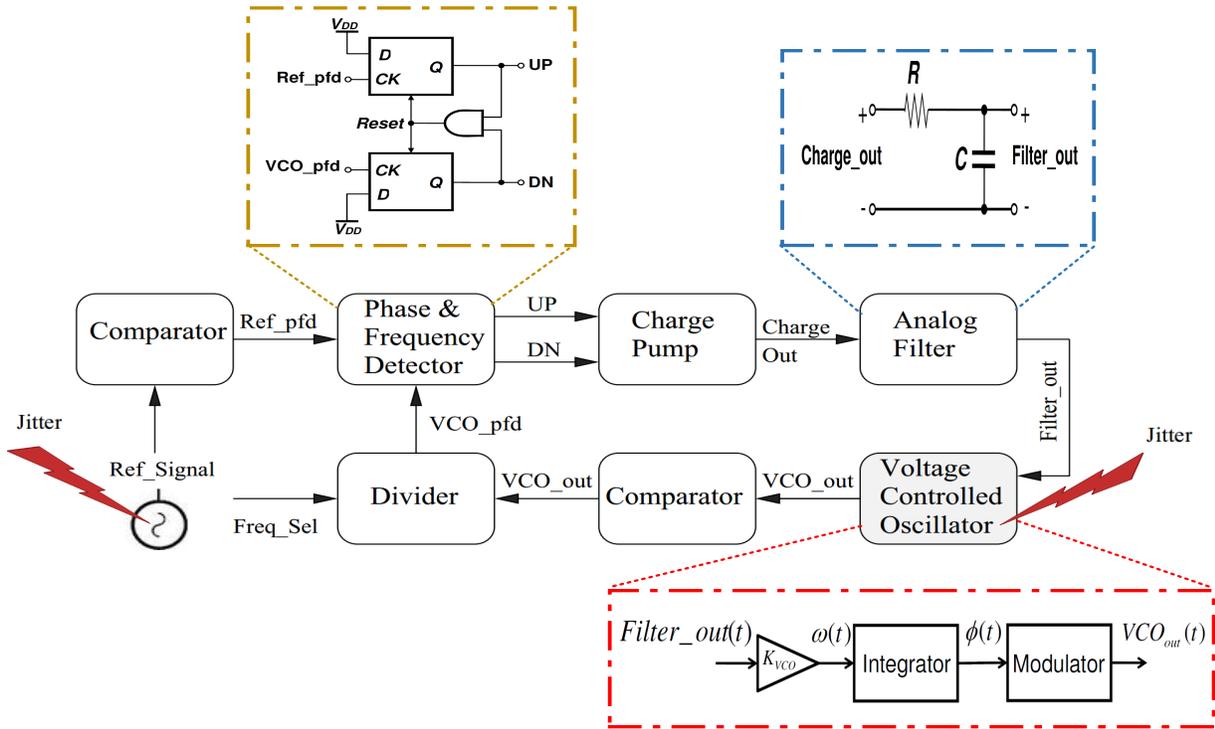


Fig. 11: PLL Design Block Diagram

Figure 11 shows a block based description of a second order PLL based frequency synthesizer. It consists of a reference oscillator, a Charge Pump (CP), a Low Pass Filter (LPF), and a Voltage Controlled Oscillator (VCO). In order to model this PLL using SSREs notation, we need to model each block separately and then link them according to the PLL architecture in Fig. 11. The noise considered in this case study is the random temporal variation of the phase (FM jitter as defined in Section 3.1.2) in the reference oscillator and the VCO block. It is well-known that jitter is the most dominant and critical noise metric in PLL because large jitter can modulate the oscillator signal both in frequency and amplitude.

These modulation effects can cause a deviation in the phase from targeted locking range and hence results in a design failure. The efficient verification of PLL for a certain design specification has always been a challenge for circuit designers. We apply the proposed methodology to verify the locking property of a second order PLL design shown in Figure 11. The lock time property is a safety property that expresses how fast the frequency synthesizer switches from one frequency to another. The verification of this property is achieved by checking that the PLL reaches the proper DC value within the lock time parameter range which is $\in [0.002, 0.0024]$ seconds.

This property is defined within the ambit of an SSRE model in Equation (11), where the SSRE concatenation operator (\wedge) indicates that the two Boolean expressions hold simultaneously.

$$\begin{aligned} \text{Property_PLL} = & \text{If}(\text{Filter_out}(\text{Lock_time}_{\min} + n) \in \text{DC_level_range} \wedge \\ & \text{Filter_out}(\text{Lock_time}_{\max} - n) \in \text{DC_level_range}, \text{true}, \text{false}) \end{aligned} \quad (11)$$

The verification property is “For a given confidence level α , and N Monte Carlo trials, what is the probability that the PLL meets the lock-time requirement?”.

In this case, the PLL has been designed with a lock-time in the range of $[0.002, 0.0024]$ sec. Hence, the null hypothesis H_0 and the alternative hypothesis H_1 of Property (11) can be, respectively, expressed as:

$$\begin{aligned} H_0 : & \text{lock_time} \in [0.002, 0.0024] \\ H_1 : & \text{lock_time} \notin [0.002, 0.0024] \end{aligned}$$

Figure 12 depicts a comparison between the locking property of the PLL design whose parameter values are listed in Table 5.2 with and without jitter.

A comparison of the same reachability algorithm without backward refinement [7] for the PLL design is given in Table 6. It can be remarked that in the case of jittery PLL (red dotted line), the low pass filter outputs do not stabilize to the tolerated DC level and keep fluctuating outside the tolerated range. As a result, the PLL locking property is violated and the verification fails.

Therefore, the verification of the PLL with consideration of jitter is very important when performing reachability analysis. Now, we validate our proposed intertwined forward/backward reachability technique on the jittery PLL design for an entire range of initial conditions and with consideration of parameter variations. The derived forward and backward reachable bounds are shown in Figure 13, in which the forward reachability bound is painted in

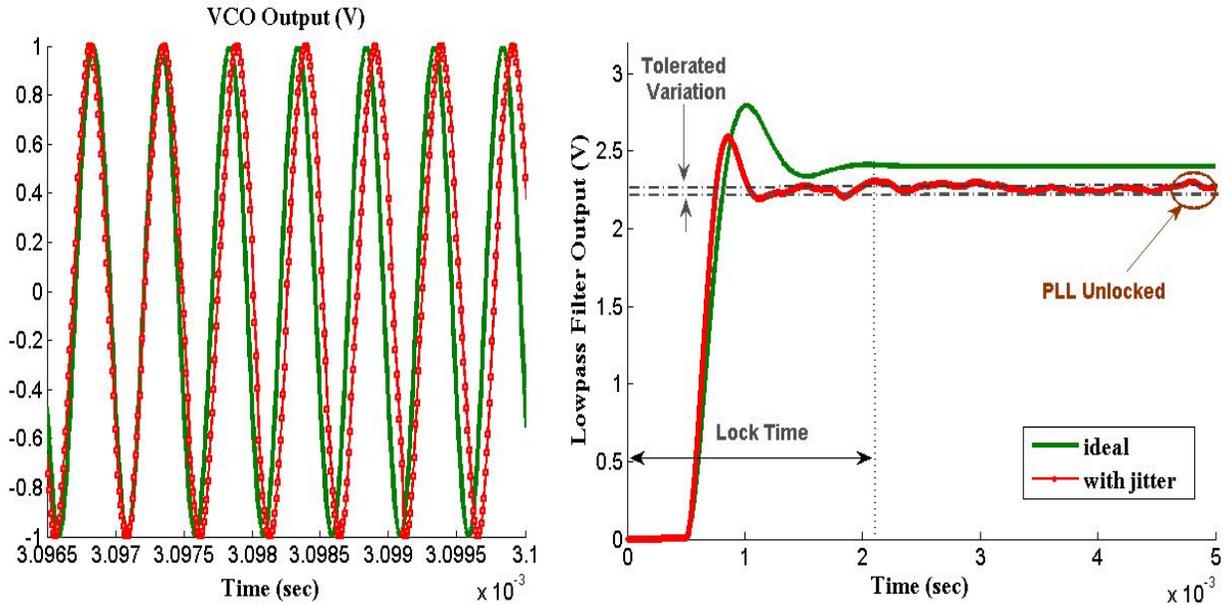


Fig. 12: PLL Output with and without Jitter

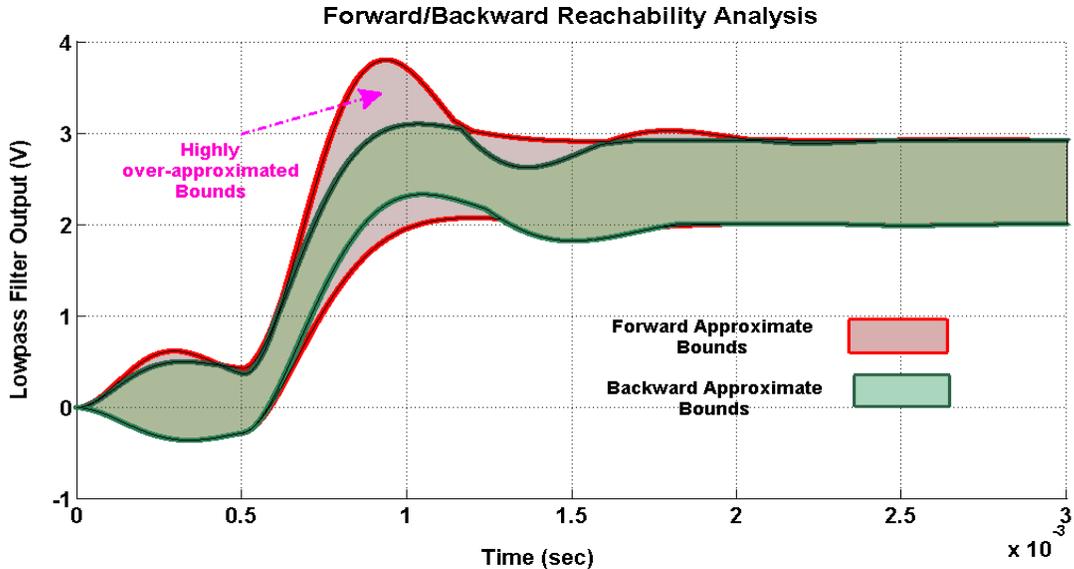


Fig. 13: Intertwined Forward/Backward Reachability Analysis of PLL under Jitter

red and the backward reachability bound in green.

In the forward iteration, the reachable set is highly over-approximating the PLL behavior. By performing the backward correction, we were able to tighten up this over-approximation and trace back the circuit dynamics down to the initial condition.

The results of the PLL yield estimation using a variant of statistical Monte Carlo technique [10] called Monte Carlo-Jackknife (MC-JK) and our proposed intertwined reachability technique are summarized in Table 5. It is worth mentioning that our technique converges in one iteration only while Monte Carlo technique requires thousands of runs. From Table 5, it can be noticed that our proposed method finds a lower yield percentage compared to the statistical Monte Carlo scheme in [10]. This can be explained by the fact that our verifica-

Table 4: PLL Parameters

Loop Parameter	Value	Unit
VCO gain (K_{VCO})	$4\pi/5$	rad.MHz/V
Loop Filter resistance (R)	10	$K\Omega$
Loop Filter capacitor (C)	10	nF
Charge time parameter	1.0001	–
Divider Ratio (N)	2	–
Natural frequency	1	MHz
VCO operating frequency	1	MHz
Damping Ratio (ξ)	0.05	–
Charge Pump current (I_{CP})	0.25	mA
LPF DC level	2.5	V
Supply voltage (V_c)	5	V

Table 5: Verification Results for the PLL Lock-Time Property

N=	Jitter Only			Parameter Variation Only			Jitter & P.V		
	[10]	Our method	RE	[10]	Our method	RE	[10]	Our method	RE
	Yield (%)	Yield(%)	(%)	Yield(%)	Yield (%)	(%)	Yield (%)	Yield (%)	(%)
1000	82.4	74.1	8.3	84.7	79.2	5.5	80.6	71.5	9.1
	83.3	71.7	11.6	80.9	76.3	4.6	78.2	68.9	9.3
	81.7	69.8	11.9	79.2	72.7	6.5	77.5	67.3	10.2
5000	83.6	73.1	10.5	85.8	81.6	4.2	81.8	72.3	8.7
	80.2	72.3	7.9	81.9	77.8	4.1	78.2	70.1	8.9
	79.8	70.8	9	80.7	74.4	6.3	78.2	68.6	9.6
10000	81.7	69.9	11.8	83.6	79.7	3.9	80.2	66.1	14.1
	79.6	67.1	12.5	80.3	74.4	6.1	78.1	62.6	15.3
	78.1	65.9	12.2	81.9	71.8	10.1	76.8	60.1	16.7

tion approach can weed out PLL locking failures that were not covered in [10]. In addition, the presence of combined jitter, initial conditions and process variations (Columns 8 – 10) have substantially decreased the PLL yield, meaning the PLL presents more probability of lock failure. The presence of jitter alone has shown a lower yield rate. This can be justified by the high sensitivity of the VCO block to jitter. The failure of the PLL is not due to lock up (non oscillation) of the VCO but, due to either an “ugly” (i.e., fluctuates outside the tolerated region) or delayed oscillation. The Relative Error (RE) between our proposed approach and the MC technique (Columns 4, 7 and 10) becomes more pronounced when the number of Monte Carlo trials is increased due to the high MC sampling variance.

We also performed a comparison between our proposed corrected forward reachability analysis technique and the forward solely scheme in [7]. The results of the comparison for different uncertainty scenarios are summarized in Table 6. It can be seen that the yield estimate is the lowest under the combined jitter and process variation effects (Columns 7-9 in Table 6). This confirms the importance of including jitter in the modeling and verification plan of the PLL design. As demonstrated, the proposed approach provides roughly 5% better accuracy in PLL parametric yield estimation, unlike the forward only reachability approach that highly over-approximates the reachable bounds and thus the yield rate.

Table 6: Comparison Forward/proposed Intertwined Reachability Analysis Schemes

Jitter Only			Parameter Variation Only			Jitter & P.V		
Forward [7]	Our method	RE	Forward [7]	Our method	RE	Forward [7]	Our method	RE
Yield (%)	Yield(%)	(%)	Yield(%)	Yield (%)	(%)	Yield (%)	Yield (%)	(%)
78.3	74.1	4.2	81.5	79.2	2.3	76.9	71.5	5.4
76.4	71.7	4.7	78.1	76.3	1.8	72.8	68.9	3.9
75.1	69.8	5.3	75.4	72.7	2.7	71.2	67.3	3.9

6 Conclusion

In this report, we have presented a novel methodology for modeling and verification of non-linear analog and mixed-signal circuits by computing reachable sets of possible state-space trajectories in the presence of uncertainties. In contrast to methods that use solely forward reachability, the refinement of the reachable state space is carried out in an intertwined forward/backward manner. The resulting set, which contains all periodic and aperiodic time bounded behaviors of the circuit under parameter variation and initial condition disturbance, can be used to verify critical properties such as bounds on voltages, currents, and cycle time (frequency) of embedded designs. Statistical verification based on hypothesis testing is then conducted on the resultant corrected reachable sets for an accurate parametric circuit failure estimation. Experimental results show that our intertwined forward/backward reachability analysis can succeed in accurately estimating the circuit failure rate (a.k.a yield) by reducing the high over-approximation of the forward scheme in the presence of noise and process variations. Experimental results of a second order PLL application, our algorithm outperforms existing methods by providing up to 17% more reliable yield estimation of the locking time property. However, the computational cost of the proposed methodology highly increases with the number of process parameters and design properties to be verified. In our future research, we will further investigate the possibility of adopting efficient heuristics and parallelization techniques that may address the computational time issue. We also plan to verify complex designs in presence of transient faults uncertainty [33] and that involve multiple performance metrics.

References

- [1] R. C. Daniels and R. W. Heath Jr, “60 GHz wireless communications: Emerging requirements and design recommendations,” *IEEE Vehicular Technology Magazine*, vol. 2, no. 3, pp. 41–50, 2007.
- [2] M. Orshansky, S. R. Nassif, and D. Boning, *Design for manufacturability and statistical design: A constructive approach*. Springer Science & Business Media, 2008.
- [3] International Technology Roadmap for Semiconductors, “ITRS 2.0 2015 edition executive report,” 2015. Available at <http://www.itrs2.net/itrs-reports.html>.
- [4] M. Althoff and B. H. Krogh, “Reachability analysis of nonlinear differential-algebraic systems,” *IEEE Transactions on Automatic Control*, vol. 59, no. 2, pp. 371–383, 2014.
- [5] Y. Song, H. Fu, H. Yu, and G. Shi, “Stable backward reachability correction for PLL verification with consideration of environmental noise induced jitter,” *Asia and South Pacific-Design Automation Conference*, pp. 755–760, 2013.
- [6] M. Althoff, A. Rajhans, B. H. Krogh, S. Yaldiz, X. Li, and L. Pileggi, “Formal verification of phase-locked loops using reachability analysis and continuization,” *Communications of the ACM*, vol. 56, no. 10, pp. 97–104, 2013.
- [7] I. Seghaier, H. Aridhi, M. H. Zaki, and S. Tahar, “A qualitative simulation approach for verifying PLL locking property,” *Great Lakes Symposium on VLSI*, pp. 317–322, 2014.
- [8] R. Drechsler, *Formal System Verification: State-of-the-Art and Future Trends*. Springer, 2017.

- [9] R. A. Saleh, S.-J. Jou, and A. R. Newton, *Mixed-mode simulation and analog multi-level simulation*, vol. 279. Springer Science & Business Media, 2013.
- [10] I. Seghaier, M. H. Zaki, and S. Tahar, “Statistically validating the impact of process variations on analog and mixed signal designs,” *Great Lakes Symposium on VLSI*, pp. 99–102, 2015.
- [11] J. Yao, Z. Ye, and Y. Wang, “Importance boundary sampling for SRAM yield analysis with multiple failure regions,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 33, no. 3, pp. 384–396, 2014.
- [12] V. Dubikhin, C. Myers, D. Sokolov, I. Syranidis, and A. Yakovlev, “Advances in formal methods for the design of analog/mixed-signal systems,” *Design Automation Conference*, pp. 36–41, 2017.
- [13] W. Denman, B. Akbarpour, S. Tahar, M. H. Zaki, and L. C. Paulson, “Formal verification of analog designs using MetiTarski,” *Formal Methods in Computer Aided Design*, pp. 93–100, 2009.
- [14] S. H. Taqdees and O. Hasan, “Formal verification of continuous models of analog circuits,” *Frontiers in Analog CAD, poster paper*, pp. 1–2, 2013.
- [15] A. Singh and P. Li, “On behavioral model equivalence checking for large analog/mixed signal systems,” *International Conference on Computer-Aided Design*, pp. 55–61, 2010.
- [16] R. Narayanan, I. Seghaier, M. H. Zaki, and S. Tahar, “Statistical run-time verification of analog circuits in presence of noise and process variation,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 21, no. 10, pp. 1811–1822, 2013.
- [17] G. Frehse, “PHAVER: algorithmic verification of hybrid systems past HyTech,” *International Journal on Software Tools for Technology Transfer*, vol. 10, no. 3, pp. 263–279, 2008.
- [18] G. Frehse, C. Le Guernic, A. Donzé, S. Cotton, R. Ray, O. Lebeltel, R. Ripado, A. Girard, T. Dang, and O. Maler, “SpaceEx: Scalable verification of hybrid systems,” *Computer Aided Verification*, pp. 379–395, 2011.
- [19] K. J. Kuhn, “Reducing variation in advanced logic technologies: Approaches to process and design for manufacturability of nanoscale CMOS,” *International Electron Devices Meeting*, pp. 471–474, 2007.
- [20] K. Kundert, “Modeling and simulation of jitter in PLL frequency synthesizers,” *Cadence white paper*, 2001.
- [21] G. N. Milstein, *Numerical integration of stochastic differential equations*, vol. 313. Springer Science & Business Media, 1994.
- [22] G. Al-Sammame, *Simulation symbolique des circuits décrits au niveau algorithmique*. PhD thesis, Université Joseph-Fourier-Grenoble I, France, 2005.
- [23] P. R. Kumar and P. Varaiya, *Stochastic systems: Estimation, identification, and adaptive control*. Society for Industrial and Applied Mathematics, 2015.
- [24] N. Ikeda and S. Watanabe, *Stochastic differential equations and diffusion processes*, vol. 24. Elsevier, 2014.

- [25] D. Revuz and M. Yor, *Continuous martingales and Brownian motion*, vol. 293. Grundlehren der mathematischen Wissenschaften, Springer, 1999.
- [26] G. N. Milstein and M. V. Tretyakov, *Stochastic numerics for mathematical physics*. Springer Science & Business Media, 2013.
- [27] H. Yu, C. Chung, K. Wong, H. Lee, and J. Zhang, “Probabilistic load flow evaluation with hybrid latin hypercube sampling and cholesky decomposition,” *IEEE Transactions on Power Systems*, vol. 24, no. 2, pp. 661–667, 2009.
- [28] K. Burrage, P. Burrage, D. Donovan, and B. Thompson, “Populations of models, experimental designs and coverage of parameter space by Latin hypercube and orthogonal sampling,” *Procedia Computer Science*, vol. 51, pp. 1762–1771, 2015.
- [29] W. L. Martinez and A. R. Martinez, *Computational statistics handbook with MATLAB*. CRC press, 2007.
- [30] A. Bonarini and G. Bontempi, “A qualitative simulation approach for fuzzy dynamical models,” *ACM Transactions on Modeling and Computer Simulation*, vol. 4, no. 4, pp. 285–313, 1994.
- [31] T. Coleman, M. A. Branch, and A. Grace, “Optimization toolbox,” *Users Guide for MATLAB 5, Version 2, Release II*, 1999.
- [32] K. Lata and H. Jamadagni, “Formal verification of tunnel diode oscillator with temperature variations,” *Asia and South Pacific Design Automation Conference*, pp. 217–222, 2010.
- [33] G. Bany Hamad, G. Kazma, O. A. Mohamed, and Y. Savaria, “Efficient and accurate analysis of single event transients propagation using SMT-based techniques,” *International Conference on Computer-Aided Design*, pp. 1–7, 2016.