Analog Integr Circ Sig Process (2015) 85:379–394
DOI 10.1007/s10470-015-0588-x

Fast statistical analysis of nonlinear analog circuits using model order reduction

Henda Aridhi[1] · Mohamed H. Zaki[1] · Sofiène Tahar[1]

Received: 4 February 2015 / Revised: 25 May 2015 / Accepted: 15 June 2015 / Published online: 3 July 2015

Abstract The reduction of the computational cost of statistical circuit analysis, such as Monte Carlo (MC) simulation, is a challenging problem. In this paper, we propose to build macromodels capable of reproducing the statistical behavior of all repeated MC simulations in a single simulation run. The parameter space is sampled similarly to the MC method and the resulting nonlinear models are reduced simultaneously to a small macromodel using nonlinear model order reduction method based on projection, perturbation theory and linearization techniques. We demonstrate the effectiveness of the proposed method for three applications: a current mirror, an operational transconductance amplifier, and a three inverter chain under the effect of current factor and threshold voltage variations. Our experimental results show that our method provides a speedup in the range 100–500 over 1000 samples of MC simulation.

Keywords Clustering · Monte Carlo · Model order reduction · Nonlinear analog circuits · Perturbation theory · Projection · Statistical simulation

1 Introduction

Process, voltage, and temperature (PVT) variations have a huge impact on circuit performance, yield, and reliability [1]. Circuit parameters are no longer truly deterministic and are considered as probability distributions on their infinite space. The problem of predicting circuit behavior and performance for their entire parameter space is compulsory to catch most of their undesired behavior prior to their fabrication.

Traditional corner case verification methods are not accurate and cannot guarantee that a circuit will always behave according to its specification. Also, the methods, which compute circuits performance bounds in presence of parameters variability using affine interval arithmetic [2] or global optimization [3, 4], are expensive, scale poorly with circuit complexity, and often lead to over-conservative results. Sampling-based methods such as MC simulation [5] methods are easy to implement but are computationally expensive. The enhancement of MC space sampling schemes by performing importance sampling [6] or by reducing the sampling discrepancy [7–9] does not work for all circuits and scales poorly with circuit sizes. Stochastic spectral methods [10–12], which model parameters as stochastic processes and avoid repeated simulations, require sophisticated solvers and quickly hit the computation limits for nonlinear circuits with correlated parameters.

Model order reduction (MOR) [13] is a promising technique that reduces the size and complexity of large mathematical models. It builds compact models that reproduce the simulated behavior of an original model in a smaller amount of time. MOR methods for circuit simulation [14] can be a key to address the challenging problem of alleviating the computational cost of MC methods. They can be effectively used to reduce the number of differential

✉ Henda Aridhi
aridhi.henda@gmail.com; h_aridh@ece.concordia.ca

Mohamed H. Zaki
mzaki@ece.concordia.ca

Sofiène Tahar
tahar@ece.concordia.ca

[1] Department of Electrical and Computer Engineering, Concordia University, Montreal, QC, Canada

equations which have to be solved in order to estimate circuit performances. The preliminary application of MOR methods to address process variation for linear networks [12, 15, 16] proves their potential. Unfortunately, the key challenge, which has been preventing further progress in this direction, resides in the limitations of MOR methods for the case of nonlinear circuits.

In this paper, we propose a novel simulation method which makes use of the recently developed MOR methods for nonlinear circuits [17–19] and reduces the computational cost of statistical circuit analysis under process variation. The parameter space is sampled similarly to the MC method and the resulting nonlinear models are reduced simultaneously to a small macromodel capable of reproducing the statistical behavior of all repeated MC simulations in a single simulation run. The efficiency of the proposed method is proved for three applications: a current mirror, a three-stage inverter chain, and an operational transconductance amplifier. We analyze the performance of each circuit under the effect of device mismatch and show that they are accurate in terms of mean and standard deviation measurement compared to a MC simulation analysis of the original circuit model. Also, we show that the simulation traces retain the required accuracy (less than 5 % relative error).

The remainder of this paper is organized as follows. Related work is briefly reviewed in Sect. 2. Preliminaries are introduced in Sect. 3. The proposed approach is described in detail in Sect. 4. Section 5 discusses experimental results and Sect. 6 concludes the paper.

2 Related work

In order to address the problem of circuit parameters uncertainty and circuit yield estimation, a few methods have been used in practice.

MC simulation is a leading method that consists in N repetitive simulations of a circuit model for randomly generated parameter values. Then, the statistical distributions of circuit performance metrics are predicted from the N obtained simulation traces. While this method is straightforward to implement, its run time scales poorly with the number of parameter samples and the length of transient simulations [5].

To accelerate the convergence of the MC method, several techniques have been developed, for example, Latin hypercube sampling (LHS) based methods [7, 8]. The LHS based method controls the generation of the random samples which reduces the number of required samples and provides superior convergence rates over the MC method. However, the LHS based method might require a preprocessing time and a large memory, and its performance becomes comparable to the MC method as the number of random parameters increase. The quasi MC method [9], a.k.a low divergence sampling, is a generalization of the LHS method for the multivariate case. The LHS and the Quasi MC methods do not always reduce the simulation cost even though they simulate a smaller number of experiments.

Stochastic spectral methods [10–12], which have the same objective as the MC method, model the circuit parameters as continuous stochastic processes and compute the circuit response in terms of polynomial chaos (PC), in a Hilbert space. The stochastic circuit behavior is obtained by solving the obtained stochastic circuit model. The PC based methods show remarkable speedup over the MC method for RC or RLC interconnect model analysis. However, they result in extremely large models when random parameters are correlated and they require robust stochastic solvers especially for nonlinear circuit models. MC methods are typically more feasible in these situations.

In [20], the authors presented an intrusive-type stochastic solver, named ST, to quantify the uncertainties in transistor-level circuit analysis. The simulator is based on generalized PC (gPC) expansions and therefore can handle Gaussian and non-Gaussian random parameters. The efficiency of the ST method is enhanced by allowing decoupled numerical simulation and adaptive step size control.

The use of stochastic spectral circuit simulators based on gPC to handle parameters uncertainties has many limitations. For example, the number of gPC expansions scales poorly with the parameter space size. Also, it is difficult to transform correlated non Gaussian parameters to uncorrelated ones in order to easily construct gPC basis, as discussed in [21]. However, it is important to notice that in [22], an efficient framework to reduce the computational cost associated with the stochastic simulation based on gPC of complex systems with a large number of parameters, such as MEMS problems, is proposed. As detailed in [23], complex systems are decomposed hierarchically into subsystems which are simulated using a sparse stochastic testing simulator based on the adaptive anchored analysis of the variance method [24]. Then, the system level stochastic simulation is accelerated by the use of the tensor-train decomposition [25].

In [26], the authors extended the DC sensitivity-based mismatch analysis [27, 28] for analyzing mismatch effects on transient characteristics. The accuracy of the pseudo noise based mismatch analysis, which only computes the variance of a Gaussian performance variation, relies on the assumptions of a linear perturbation model and small mismatches.

Recently, statistical simulation methods started to benefit from linear circuits MOR methods. For example, in

[29] the authors predicted the impact of process variation using linear MOR methods over intervals of parameters (the asymptotic waveform evaluation method [30] and the passive reduced order interconnect macromodeling algorithm [31]). However, these methods are limited to linear interconnect circuits, may have more than 5 % estimation error for some cases and require robust numerical methods to avoid the interval range estimation error explosion. Also, MOR and stochastic spectral methods have been employed to enhance the extraction and the simulation of linear interconnect models [15, 16] and mismatch analysis [12]. However, these methods inherit the complexity and limitations of stochastic methods.

In [32], the authors proposed to first represent the performance bound of analog circuits under the effect of multiple interval valued parameter variations. They then generate a reduced differential model using the polytope representations [33] of circuits uncertain states and the nonlinear systems MOR method [34]. The models are solved using computational geometry. Unfortunately, due to the computational cost, this method is impractical for large nonlinear circuit models with correlated state variables.

In this work, we are taking advantage of the recent progress in the MOR methods which apply to the case of nonlinear circuit models [17, 18] and using them to reduce the computational cost of solving the large number of nonlinear equations needed to analyze process variation effects. We aim to use a single simulation run of a reduced model to approximate a large number of MC simulations without degrading the analysis accuracy. Therefore, our method computes the statistical circuit performances in a much smaller amount of time and can make them converge faster to their theoretical values by increasing the number of samples of the parameters space.

3 Preliminaries

3.1 Circuit model formulation in presence of process Variation

Analog circuit differential models are often obtained using the modified nodal analysis [35] method. In general this method leads to the model given in Eq. (1)

$$\dot{x} = f(x, u, p) \tag{1}$$

where $\dot{x}$ is the time derivative of the vector x, which represents the circuit voltage and current state variables, and f is a nonlinear vector function of the state vector x, the input u, and the circuit device model parameters p. The accuracy of the model in Eq. (1) is directly related to the accuracy of the circuit device models which take into consideration the performance variation due to process, voltage and temperature (PVT) variations. The effect of process variations (PV) on a circuit device is the mathematical sum of the effect of two variation types: (1) the inter-die variation that affects all the devices similarly; and (2) the intra-die variation which affect different devices differently [36]. PV leads to variations in attributes of devices (length, width, or oxide thickness, etc.) when integrated circuits are fabricated. It affects the yield and performance (bandwidth, gain, rise time, delay, etc.) of the produced circuits and its effect becomes prevalent at smaller manufacturing technology processes and lower power supply voltages. For example, device mismatches, which refers to the small random variations in the characteristics of identically designed devices, is a major concern in the design of analog circuits such as digitally controlled analog circuits, oscillators, current mirrors, or amplifiers, etc. The Pelgrom's model [37] for MOS transistors is used to relate the local mismatch variance of electrical device parameters $\sigma(\Delta p)$, device width W and length L, and technology constants A_p, as given in Eq. (2). It is widely used to express the threshold voltage v_t and the current factor $\beta = \mu C_{ox} \frac{W}{L}$ mismatches.

$$\sigma^2(\Delta p) = \frac{A_p^2}{W \cdot L} + S_p \cdot D \tag{2}$$

where D is the distance between two transistors, S_p describes the variation of the parameter p with spacing. In [38], a different mismatch model, which is proposed for semiconductor devices (diodes, bipolar, etc.), is based on the propagation of variance in Eq. (3), where e is an electrical property and p_l are the process and geometry parameters.

$$\sigma_{\delta e}^2 = \sum_l \left(\frac{\partial e}{\partial p_l} \right)^2 \sigma_{\delta p_l}^2 \tag{3}$$

Except for very small circuits, it is difficult to analytically predict the behavior of a circuit due to the combination of mismatches of individual devices [1]. The impact of these random parameter variations on circuit behavior is rather studied with MC simulation [5] by repeating circuit simulations for randomly varied devices.

3.2 Sampling-based statistical methods

In sampling-based statistical simulation methods, the statistical characteristics of a state variable x due to a variation of the parameter p is obtained by solving the model in Eq. (1) N times $(x_{p_1}, x_{p_2}, \ldots, x_{p_N})$ for random generations of the parameter p $(p_1, p_2, \ldots, p_N)$. In order to accurately capture the effect of the variation of the parameter p, the entire parameter space has to be covered with a very large number N based on the variance of the output of interest and the required accuracy. For example, a MC simulation may require 1000 to 10,000 randomly generated values to get a good confidence level on the conducted simulation results [5]. The main challenges of the sampling based methods are: (1) how to efficiently sample the high dimensional parameters space for a high coverage of a model behavior and (2) how to efficiently or simultaneously solve the subsequent large number of nonlinear dynamical models. Methods like quasi MC [9], LHS MC [7, 8], sparse sampling grids [39] and importance sampling algorithms [6] address the first challenge. The second challenge has been approached in different ways but is still problematic given the increasing complexity and size of integrated circuits. The differences of these approaches are lead by the way how the PV effect is modeled. The stochastic methods model PV as a stochastic process and employ stochastic solvers to estimate circuits statistical behavior [11, 12, 20, 21]. MOR methods can be used to address both challenges by reducing the size of the parameters space or reducing the models which have to be solved iteratively for each parameter sample, respectively. The application of MOR to address PV for linear networks [12, 15, 16] proves their potential. In this paper, our contribution lies in taking advantage of the recent progress in MOR methods for nonlinear circuits and applying them to reduce the computational cost of solving the large number of nonlinear equations to analyze PV effects, mainly mismatch of identically designed devices.

3.3 Nonlinear projection based model order reduction

In the literature, model order reduction (MOR) is the transformation of a large dynamical model described by Eq. (1) into a smaller model which mimics its behavior while it can be simulated in a considerably smaller amount of time. MOR has been previously applied to the class of nonlinear circuit models and led to faster circuit models with acceptable accuracy levels [18, 19, 40, 41]. We consider in this paper the method proposed in [17] that can be briefly described by Algorithm 1. This MOR method requires a differential model (Line 1) and constructs reduced models in an iterative way until a target speedup and accuracy requirements are checked (Line 11). The original model is simulated (in Line 4, DC and transient simulations) which results in a collection of trajectories for different inputs and initial conditions. These trajectories are captured in the form of three matrices; the state variable X, its time derivative F and the input U. The obtained behavior snapshot X is clustered into a set of k centroids C in Line 5 using the k-means algorithm [42]. Then, the model in Eq. (1) is linearized at each element of the set of clusters C. The number k of clusters is set initially to a minimal value and is increased iteratively until the linearized model becomes accurate. The linearization matrices are used to compute a unitary Krylov space projection matrix V $(VV^t = I_n)$ using either the block Arnoldi or block Lanczos algorithms [13]. The linearized model in Line 6 is reduced via projection in Line 8 which leads to Eq. (4), where $\hat{F} = V^t \cdot F$, $\hat{J}_z = V^t \cdot \frac{\partial f}{\partial x|x} \cdot V$, $\hat{J}_u = V^t \cdot \frac{\partial f}{\partial u|u}$, and $Z = V^t \cdot C$. The matrices and vectors are dynamically evaluated using the weights $w(i) = \frac{\|z - Z(i)\|_2^{-1}}{\left(\sum_{i=1}^{m} \|z - Z(i)\|_2\right)^{-1}}, i = 1, \ldots, k$ in order to approximate the behavior of the original model.

$$\dot{z} = \sum_{i=1}^{m \leq k} w(i) \cdot \left(\hat{F}(i) + \hat{J}_{zi} \cdot (z - Z(i)) + \hat{J}_{ui} \cdot (u - U(i))\right) \tag{4}$$

The simulation of the reduced model in Line 9 is required to check that it yields an acceptable speedup and accuracy conformance criteria. The speedup is evaluated as the simulation time ratio $S = T(z)/T(x)$ where $T(z)$ and $T(x)$ are the simulation times of the reduced and the original models, respectively. The accuracy of the reduced model is checked by measuring the relative error between the state variable x and its approximation $\hat{x}$. If the speedup and accuracy goals are not met in Line 11, the MOR process is iteratively restarted with a refinement of the parameters until the reduced model is accepted.

ALGORITHM 1: MOR Algorithm

```
1:  **Input:** $Equation\ (1)$
2:  **Output:** $\hat{F}, \hat{J}_z, \hat{J}_u, Z, V$
3:  while $< status == 0 >$ do
4:      $[F, X, U, T(x)] = Simulate(Equation\ (1), U, T)$
5:      $[C, U] = Cluster(X, F, U, k)$
6:      $[F, J_x, J_u] = Linearize\ (X, U, Equation\ (1))$
7:      $V = GenerateProjection(J_x, J_u)$
8:      $[\hat{F}, \hat{J}_z, \hat{J}_u, Z] = Reduce(F, J_x, J_u, X)$
9:      $[z, T(z)] = Simulate(Equation\ (4))$
10:     $\hat{x} = V \cdot z$
11:     $status = Check(x, \hat{x}, T(x), T(z))$
12: end while
```

4 Proposed methodology

We propose to transform the problem of statistical simulation into a problem of reducing the size of a large nonlinear dynamical model using nonlinear circuits MOR methods [17]. Instead of using a traditional statistical analysis approach of performing repeated simulations of a circuit model for a large number of samples of uncertain parameters, we propose to reduce a larger differential model built with different instances of the circuit model each of which corresponding to different samples of the uncertain parameters. Then, the obtained reduced model is simulated only once to perform the job of N-points MC simulation. Figure 1 depicts the four main steps of the methodology. First is the model replication step where we build a large differential model out of N instances of the circuit model *Model*(p, x) for the randomly generated parameter samples ($Model(x, u, p_1, p_2, \ldots, p_N)$). Then, we reduce the obtained large model using the MOR method described in Sect. 3.3. After that in the reduced model simulation step, we simulate the resulting reduced model. Finally, we perform a backward projection of the reduced model simulation traces into the state space of the N circuit instances and use them in the statistics generation step to compute the statistical behavior of *Model*(p, x). The details of each step are provided in the sequel.

4.1 Model replication

In this first step of the statistical simulation methodology, we build a large differential model out of N instances of the circuit model *Model*(p, x) for the randomly generated parameter samples ($Model(x, u, p_1, p_2, \ldots, p_N)$). First, a number of N parameter samples ($p_1, p_2, \ldots, p_N$)) are generated according to the circuit technology specification or some PV estimation formulas such as the Pelgrom's model [37] provided in Eq. (2). The parameter distribution is used in N instances of the circuit model, as shown in Eq. (5). This system of differential models ($Model(x, u, p_1, p_2, \ldots, p_N)$) can be viewed as a single differential model with a large state vector formed by the states vectors of all the N instances of the circuit model.

$$\begin{array}{lcl} \dot{x}_{p_1} & = & f(x_{p_1}, u, p_1) \\ \dot{x}_{p_2} & = & f(x_{p_2}, u, p_2) \\ \vdots & \vdots & \quad \vdots \\ \dot{x}_{p_N} & = & f(x_{p_N}, u, p_N) \end{array} \tag{5}$$

where $x_{p_i}, i = 1, \ldots, N$ is the state vector of the original circuit *Model*(x, u, p) when the parameter p is set to the sample p_i. The model replication step is implemented using a script that copies N times the original circuit model while it sets the corresponding parameter sample according to the perviously generated random parameter distribution.

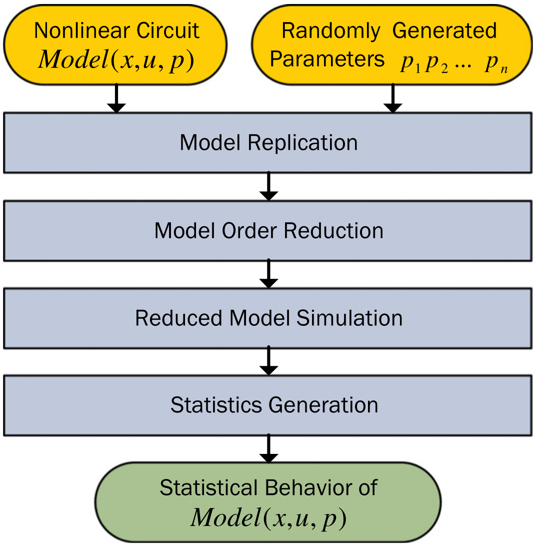


Fig. 1 Fast statistical simulation method

4.2 Model order reduction

The MOR method described in Algorithm 1 is modified and customized for an efficient reduction of the large model in Eq. (5), as described in the following three steps.

4.2.1 *Linearization points generation*

The main steps for generating the linearization points for the N circuit model instances are summarized in Algorithm 2. In Algorithm 1, the linearization points are selected by clustering a snapshot of the original circuit model simulation. In this case, performing a simulation of the N circuit instances to generate a snapshot from which we can select linearization points is computationally expensive and the main objective of this work is to avoid it. Consequently, as shown in Algorithm 2, first we simulate only one instance of the original circuit model in Line 3 and use clustering to generate the necessary linearization points of the simulated model in Line 4. The snapshot of the circuit model simulation (DC and transient simulations) when the parameter p is set to the mean value $\mu_p = \frac{1}{N}\sum_{i=1}^{N} p_i$ is divided into clusters using the agglomerative hierarchical clustering method in MATLAB [43]. As a result, a number of k clusters is obtained and it leads to an accurate piecewise linear approximation of $Model(x, \mu_p)$ in each cluster $c = 1, 2, \ldots, k$, as given in Eq. (6).

$$\dot{x} = f(x(c), u(c), \mu_p) + J_x \cdot (x - x(c)) + J_u \cdot (u - u(c)) \tag{6}$$

where $x(c)$ is the cluster centroid, $u(c)$ is the input that corresponds to $x(c)$, $J_x = \frac{\partial f}{\partial x}$, and $J_u = \frac{\partial f}{\partial u}$. Then, in Lines 5–14, we employ a small perturbation model around each cluster centroid in Line 9, that is given in Eq. (7), to find the linearization points needed for all the N instances of the model subject to PV. Finally, it is also possible that the parameter variation changes the DC operating behavior and in this case we must verify that the linear approximation is still valid. Basically, we solve the DC equation of the circuit model again with the new parameter value as shown in Lines 10–12. Therefore, any DC behavior change of the circuit instances due to PV is captured into the set of clusters.

$$x(c, \mu_p + \delta p) = x(c, \mu_p) + \delta p \frac{\partial x}{\partial p} \tag{7}$$

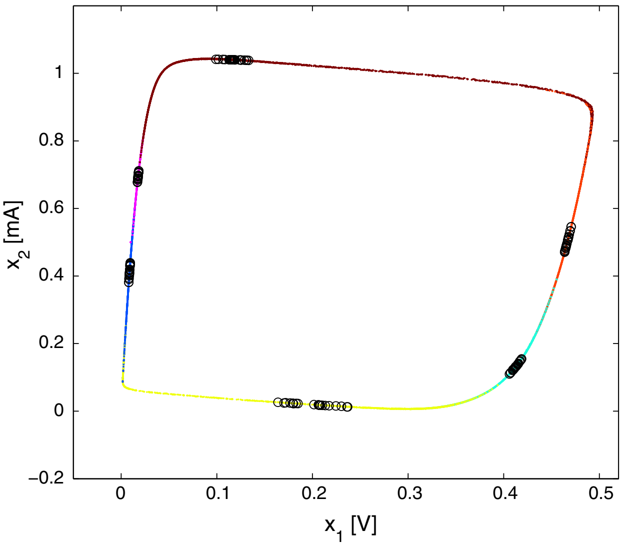


Fig. 2 Example of perturbed clusters centroids

```
ALGORITHM 2: Linearization Points Generation
1: Input: Equations (1) and (7)
2: Output: $x(c, p_i),\ c = 1 \ldots k,\ i = 1 \ldots N,$
3: $X_{\mu_p} = Simulate(Model(x, u, \mu_p))\{Equation\ (1)\}$
4: $x(c, \mu_p) = Cluster(X_{\mu_p}),\ for\ c = 1 \ldots k$
5: for $i = 1 \ldots N - 1$ do
6:     $\delta p = p_i - \mu_p$
7:     $\frac{\partial x}{\partial p} = \frac{\partial \dot{x}}{\partial p} \cdot \frac{\partial x}{\partial \dot{x}}$
8:     for $c = 1 \ldots k$ do
9:         $x(c, p_i) = Perturb(x(c, \mu_p), \delta p, \frac{\partial x}{\partial p})\{Equation\ (7)\}$
10:        if $(Model(x(c, \mu_p), u(c), \mu_p) = 0)$ then
11:            $x(c, p_i) = Solve(Model(x, u(c), p_i) = 0)\{Equation\ (1)\}$
12:        end if
13:    end for
14: end for
```

Figure 2 illustrates the clusters generation step for a two state variable circuit (a tunnel diode oscillator). It describes six clusters ($k = 6$) and their perturbed centroid points which together form the centroid of twenty circuit instances ($N = 20$). The perturbation effect is shown as a deviation the nominal parameter circuit model cluster while always being within the real trajectories of the perturbed circuit models. In the case where the parameter variation highly affects the DC circuit behavior, we must verify that the linear approximation is still valid. Basically, we solve the DC equation of the circuit model again with the new parameter value as shown in Lines 10–12. Therefore, all DC behavior variation of the circuit instances due to PV is captured into the set of clusters.

4.2.2 *Linearization of system of equations*

In this step the set of differential models in Eq. (5) is reformulated as follows:

$$\dot{y} = f^{*}(y, u, p) \tag{8}$$

where $y = [x_{S_1}, \ldots, x_{S_m}]$ is the new state variable that consists of groups of state variables. The state variables in each group exhibit almost the same dynamical behavior range and have the same order of magnitude. For example, currents are grouped together, and voltages are divided into groups based on their range and the sign of the $\frac{\partial x_i}{\partial u}$. As a result, the state variable of each group have a similar behavior and can be reduced efficiently using the proper

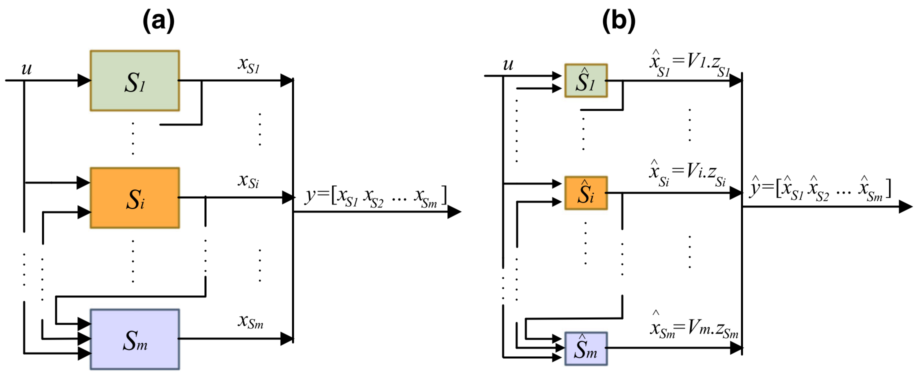

(a)
u
S_1
x_{S1}
S_i
x_{Si}
S_m
x_{Sm}
$y=[x_{S1}\, x_{S2} \ldots x_{Sm}]$
(b)
u
$\hat{S}_1$
$\hat{x}_{S1}=V_1.z_{S1}$
$\hat{S}_i$
$\hat{x}_{Si}=V_i.z_{Si}$
$\hat{S}_m$
$\hat{x}_{Sm}=V_m.z_{Sm}$
$\hat{y}=[\hat{x}_{S1}\, \hat{x}_{S2} \ldots \hat{x}_{Sm}]$

Fig. 3 Block subdivision. (**a**) Original model. (**b**) Reduced model

orthogonal decomposition (POD) reduction method described in [13]. The step of grouping state variables is performed manually, in this work, however a classification algorithm [43] can perform it automatically based on the circuit simulation traces.

The function f^* in Eq. (8) represents m system of equations labeled $S_1, \ldots, S_m$. Each system S_i is represented with equations that are functions of the state variable x_{S_i}, the new input u_{S_i} (the original circuit input u as well as other state variables from the remaining systems $S_1, \ldots, S_m$), and a subset of the randomly generated parameters $p_1, \ldots, p_N$. Thereby, the new model in Eq. (8) can be described by Fig. 3(a). Then, each system S_i is locally linearized using the linearization points generated in the previous step. As a result, each linearized system S_i is described by Eq. (9).

$$\dot{x} = \sum_{c=1}^{l} W(x) \cdot [f^*_{S_i}(x(c), u_{S_i}(c), p) + A_{S_i} \cdot (x - x_{S_i}(c)) + B_{S_i} \cdot (u_{S_i} - u_{S_i}(c))] \tag{9}$$

where $A_{S_i} = \frac{\partial f^*}{\partial x_{S_i}}$, $B_{S_i} = \frac{\partial f^*}{\partial u_{S_i}}$, and $W(x)$ are the weights computed using Eq. (10) that enable the aggregation of $l \leq k$ linearized models over the clusters boundaries.

$$W(x) = \frac{\|x - x_{S_i}(c)\|^{-1}}{\left(\sum_{c=1}^{l} \|x - x_{S_i}(c)\|\right)^{-1}} \tag{10}$$

4.2.3 Reduction of system of equations

A reduction basis V_i of size $n_{S_i} \times q_i$ is computed using the POD reduction method described in [13]. This step corresponds to Line 7 of Algorithm 1 where $q_i \ll n_{S_i}$. Then, V_i is used to reduce the matrices that appear in the multiple input linear systems in Eq. (9). It results in the reduced systems $\hat{S}_i$ for $i = 1\ldots m$, as given in Eq. (11).

$$\dot{z} = \sum_{c=1}^{l} W(z) \cdot [\hat{f}^*_{S_i}(x(c), u_{S_i}(c), p) + \hat{A}_{S_i} \cdot (z - z(c)) + \hat{B}_{S_i} \cdot (u_{S_i} - u_{S_i}(c))] \tag{11}$$

where z is the state variable of the reduced system $\hat{S}_i$, $\hat{f}^*_{S_i}(x(c), u_{S_i}(c), p) = V_i^T \cdot f^*_{S_i}(x(c), u_{S_i}(c), p)$, $\hat{A}_{S_i} = V^t \cdot A_{S_i} \cdot V$, $\hat{B}_{S_i} = V^t \cdot B_{S_i}$, and $z(c) = V^t \cdot x(c)$. The local reduced linear models are also weighted to enable models aggregation in the reduced state space using the weight function in Eq. (10).

Figure 3(b) depicts the reduced systems of equations $\hat{S}_i$ and how the backward projection of their state variables is used to form the state variable y of the original problem in 3(a). In fact, the reduced model has a size $q = \sum_i^m q_i$ and a state variable $[z_{S_1}, z_{S_2}, \ldots, z_{S_m}]$. The full order state variable that approximate the state vector y is $\hat{y} = [\hat{x}_{S_1}, \hat{x}_{S_2}, \ldots, \hat{x}_{S_m}]$, where $\hat{x}_{S_i} = V_i \cdot z_{S_i}, \quad i = 1, \ldots, m$.

4.3 Reduced model simulation and statistics generation

Algorithm 3 provides a description of the steps for the reduced model simulation, the statistics generation of a circuit performance P_f, and the comparison with the MC simulation method. In Lines 3-8, the reduced model is simulated and the state vector $\hat{y}$ that can be compared with MC simulation traces is reconstructed via the backward projection $\hat{y} = [V_1 \cdot z_{S_1}, V_2 \cdot z_{S_2}, \ldots, V_m \cdot z_{S_m}]$. The circuit behavior performance P_f statistics using the reduced model are generated in Line 6 and the runtime T_{RM} is saved in Line 7. In Lines 8–14, the MC simulation is conducted for the original circuit model $Model(x, u, p_1, p_2, \ldots, p_N)$ in Eq. (5). The circuit behavior performance P_f statistics using the MC method are generated in Line 13 and the runtime T_{RM} is saved in Line 14. Finally, the reduced model speedup over the MC method and its accuracy are evaluated in Lines 15 and 16, respectively.

ALGORITHM 3: Statistics Generation

```
Input: $Equations$ (1) and (11), $P_f, V_1, \ldots, V_m, p_1, \ldots, p_m$
Output: $[\mu_{RM}(P_f), \sigma_{RM}(P_f), \mu_{MC}(P_f), \sigma_{MC}(P_f), Speedup, Accuracy]$
%% Reduced Model Simulation
$T_{RM} = current\_time$
$[z_{S_1}, z_{S_2}, \ldots, z_{S_m}, U] = Simulate(Equation\ (11))$
$\hat{y} = [V_1 \cdot z_{S_1}, V_2 \cdot z_{S_2}, \ldots, V_m \cdot z_{S_m}]$
$[\mu_{RM}(P_f), \sigma_{RM}(P_f)] = ComputeStats(P_f, \hat{y}, U)$
$T_{RM} = current\_time - T_{RM}$
%% MC Simulation
$T_{MC} = current\_time$
for $i = 1 \ldots N$ do
    $x_{MC}(i) = Simulate(Equation\ (1), U, p_i)$
end for
$y = [x_{MC}(1), \ldots, x_{MC}(N)]$
$[\mu_{MC}(P_f), \sigma_{MC}(P_f)] = ComputeStats(P_f, y, U)$
$T_{MC} = current\_time - T_{MC}$
$Speedup = \frac{T_{MC}}{T_{RM}}$
$Accuracy = Sim\_Error(y, \hat{y})$
```

5 Applications

We apply the proposed statistical simulation method on three circuits; a current mirror, an operational transconductance amplifier and a three inverter chain under the current factor (β) and threshold voltage (v_t) process variation. For all applications, N values of the variations $\delta\beta$ and δv_t are generated based on the Pelgrom's simplified model [37]. The mean values of β and v_t are set to the 180 nm technology nominal values and their standard deviation is computed using Eq. (12).

$$\sigma^2(\Delta V_t) = \frac{A_{v_t}^2}{W \cdot L} \qquad \frac{\sigma^2(\Delta\beta)}{\beta} = \frac{A_\beta^2}{W \cdot L} \tag{12}$$

where the terms A_{v_t} and A_β are proportionality constants for 180 nm technology and are taken from [44], W and L refer to the width and the length of the transistors, respectively.

Figure 4 provides an example that uses $N = 1000$ Gaussian distributed samples of the current factor and the threshold voltage mismatch ($\delta\beta$ and δv_t for *NMOS* ($\frac{W}{L} = \frac{360}{180}$) and *PMOS* ($\frac{W}{L} = \frac{720}{180}$) transistors) using the proportionality constant values provided in Table 1.

In what follows, we describe and compare the statistical circuit performance obtained from the N points MC simulations and the ones obtained through the application of the proposed method. All simulations were performed in the MATLAB environment [43], on a *Windows* 7 operating system with an Intel core *i*7 CPU, 2.8 GHz with 24 GB of RAM.

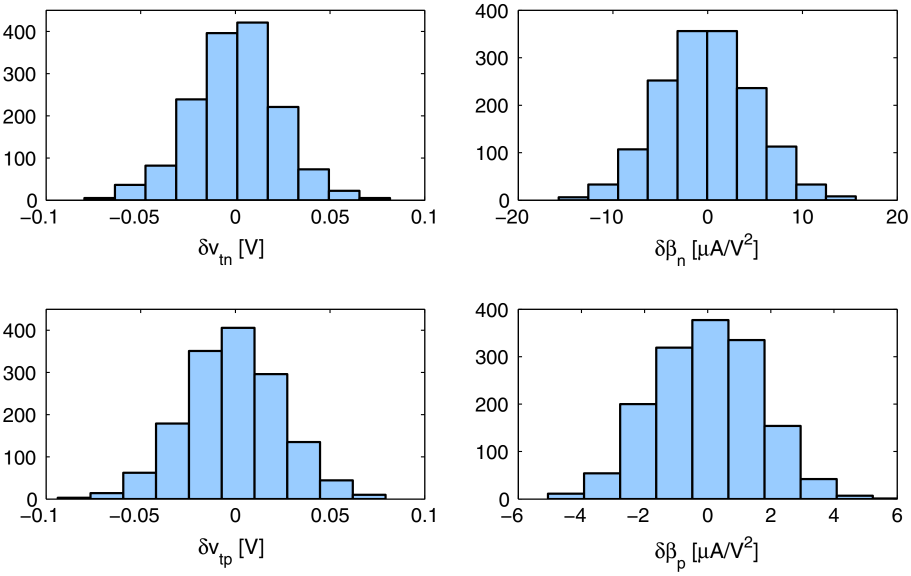


Fig. 4 Threshold voltage and current factor variation distributions for NMOS and PMOS transistors

Table 1 180 nm matching proportionality constants for size dependence

	NMOS	*PMOS*
A_{v_t} (mV µm)	5	5.09
A_β (% µm)	1.04	0.99
v_t (mV)	475.23	−449.21
$\beta = \mu C_{ox} \frac{W}{L}$ (µA/V^2)	180	−120

5.1 Current mirror

We consider the current mirror shown in Fig. 5 which functions by replicating the current produced in one active device into a second active device. The main feature of a current mirror is the high output impedance which guarantees a stable output current regardless of the load conditions. The output currents $I_2, \ldots, I_M$ are proportional to I_1 as shown in Eq. (13). The current ratios depend on the transistors sizes, their drain-source voltages and the early voltage. When the transistors are not perfectly matched there is always a systematic current gain error ε [45].

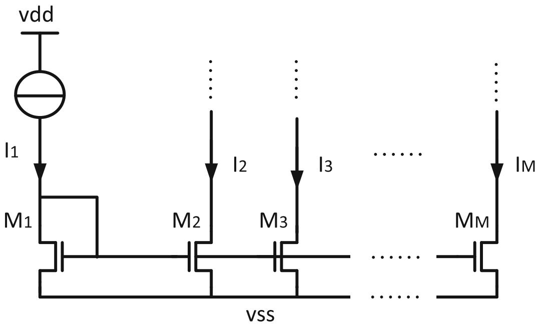


Fig. 5 Current mirror circuit

$$I_j = \frac{(W/L)_j}{(W/L)_1} I_1(1+\varepsilon) \quad for \quad j = 2, \ldots, M \tag{13}$$

In this application, we apply our method to analyze the effect of threshold voltage mismatch on the copying capability of the current I_1 where the transistors M_1 and M_2 have the same width and length while the rest of the transistors have different sizes. The size of the original problem is $n \times N = 4 \times 1000$; 4 state variables and 1000 sample points Gaussian threshold voltage distribution. The state variables are divided into 4 systems of 1000 equations based on their order of magnitude which is different since the mirroring capability of the transistors is different. The reduction procedure is performed using 4 reduction basis of size 5 × 1000 which makes the total reduction size $q = 20$.

Figure 6 illustrates the current mirror statistical distribution of the currents I_1, and I_2 for the 4 state original model, in the left column, obtained through 1000 points MC simulation and the 20 state reduced model, in the right column, obtained through the proposed method. The currents I_1 and I_2, in the left column, have similar distribution since the transistors M_1 and M_2 have equal sizes. The slight variation of their mean and standard deviation is due to the effect of their mismatch. The comparison of the currents, in the left and the right columns, shows that the distributions are the same which illustrates that the 20 state reduced model provides the same statistics of the 1000 points MC simulation.

Fig. 6 Current mirror currents distributions

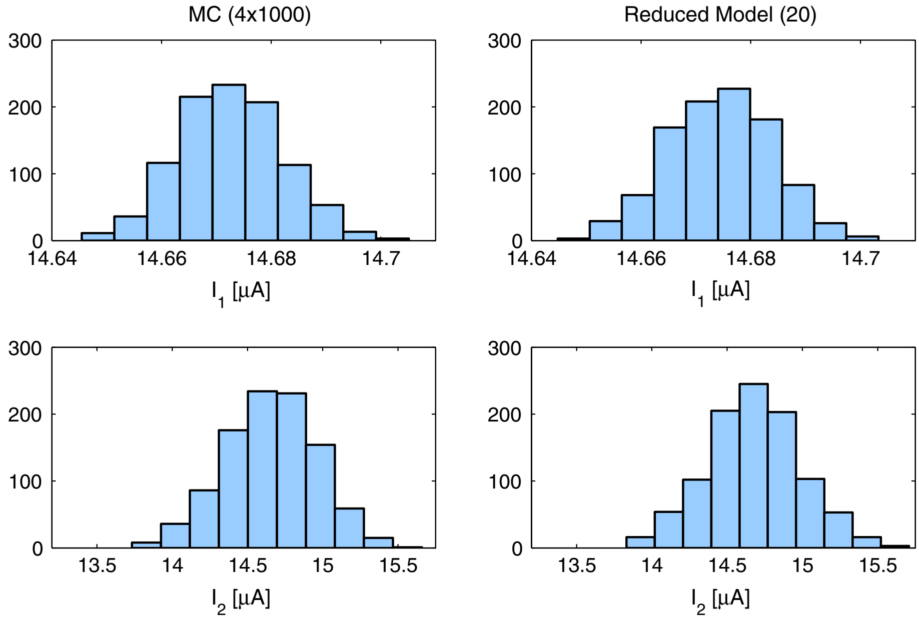


Table 2 Comparison of the simulated current mirror performance using the MC method ($n \times N = 4 \times 1000$) and the reduced model ($q = 20$)

Clusters	*MC*	Reduced model		
	–	30	20	10
$\mu(I_1)$ (μA)	14.67	14.66	14.66	14.68
$\sigma(I_1)$ (μA)	0.09	0.09	0.09	0.09
$\mu(I_2)$ (μA)	14.65	14.65	14.68	14.67
$\sigma(I_2)$ (μA)	0.29	0.29	0.30	0.31
Speedup	–	211	358	456

Table 2 provides the numerical values of the mean and the standard deviation of currents I_1 and I_2 for MC simulations (column 2) and the 20 state reduced model obtained using a different number of clusters, i.e., linearization points (columns 3, 4, and 5). The last row of Table 2 shows that the simulation speedup when using the reduced model ranges from 211 to 456 compared with the MC simulation while their statistical behaviors are almost the same.

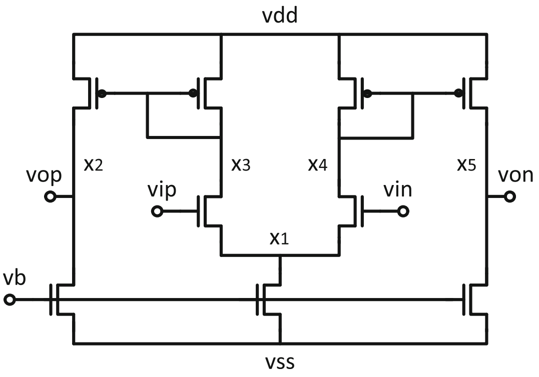


Fig. 7 Fully differential operational transconductance amplifier circuit

5.2 Operational transconductance amplifier

In this application, we consider an Operational Transconductance Amplifier (OTA) shown in Fig. 7. It is one of the most basic and versatile circuits in analog IC design for which performance is affected in the presence of PV. If the symmetrical devices of the OTA circuit are not identical, the differential gain, common mode rejection ratio, and offset voltage are affected [1].

We use our method to analyze the effect of threshold voltage and current factor mismatches on the differential gain $A_d = \frac{vop-von}{vip-vin}$ and the output offset voltage V_{os} (the differential output $(vop - von)$ when the inputs are tied together $(vip - vin = 0V)$). The input common mode voltage of the OTA is set to 0.5 V. The size of the original problem is $n \times N = 5 \times 1000$; 5 state variables and 1000 sample points for threshold voltage v_t and current factor β assumed to have Gaussian distributions. The state variables are divided into 2 systems: the first system has 1000 equations and corresponds to the state x_1. The second system has 4000 equations and corresponds to the states x_2, x_3, x_4, x_5, as shown in Fig. 7. The reduction procedure is performed using 2 projection basis of size 5×1000 and 20×4000 which makes the total reduction size $q = 25$. Figure 8 shows that the DC behavior under PV obtained with the reduced OTA model and MC simulation overlap. The continuous line DC behavior corresponds to one parameter sample s from 1000 points MC simulation. The circle marked line DC behavior was generated by solving only 25 state DC equations, backward projection of the resulting solution vector to the original state space using the projection matrix transpose V^t, and the selection of the DC solution that corresponds to the same parameter sample s.Figure 9 compares the OTA transient behavior under PV obtained with the reduced OTA model and the MC simulations. The continuous line transient behavior corresponds to one parameter sample s from 1000 points MC simulation. The circle marked line transient behavior was

Fig. 8 OTA DC characteristic sample

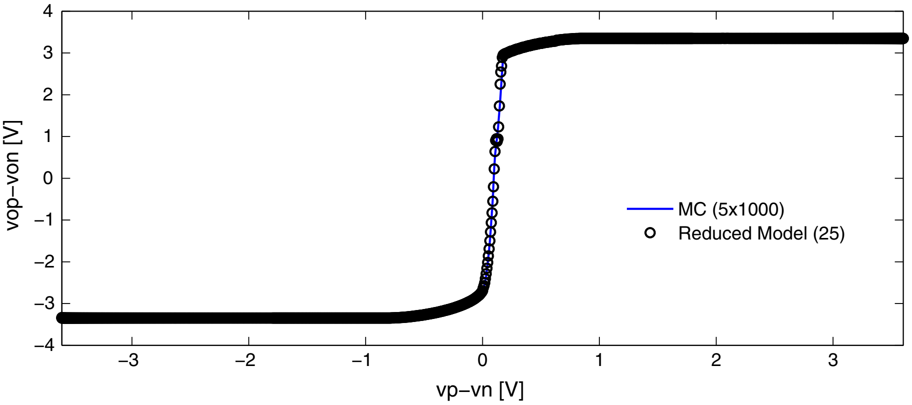


Fig. 9 OTA transient simulation sample

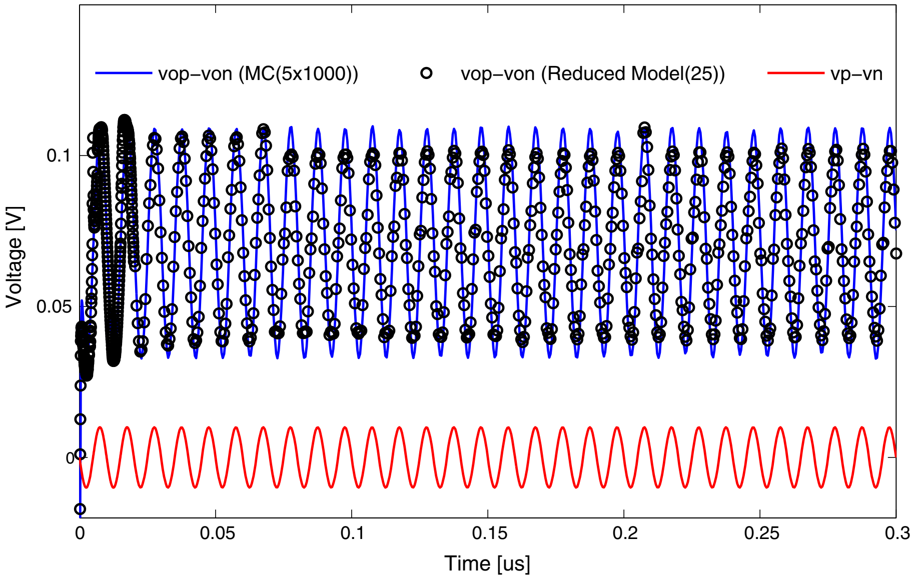
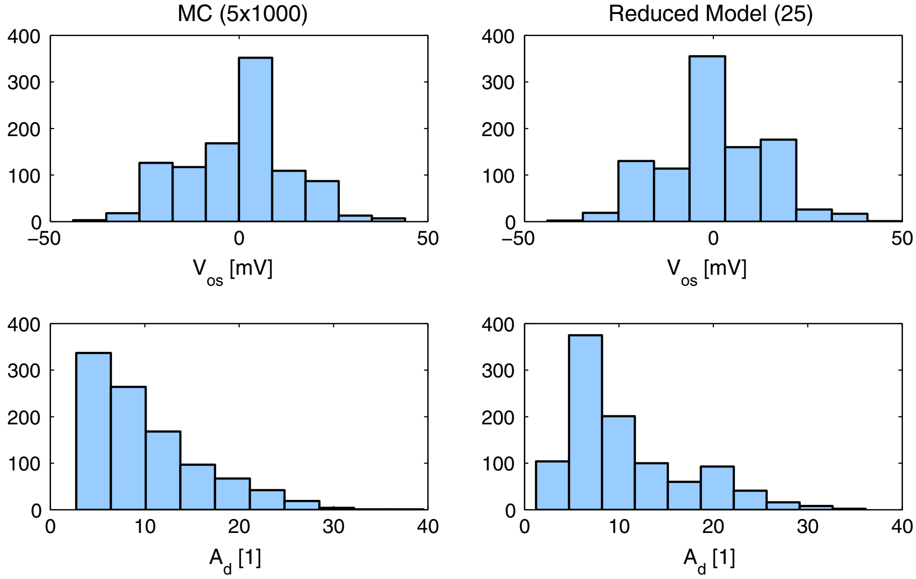


Fig. 10 OTA differential gain and offset voltage distributions



generated by solving only 25 state differential equations and reconstructing the original state space transient behavior that corresponds to the same parameter sample s. Figure 10 shows the OTA differential gain A_d and offset voltage V_{os} statistical distributions for 1000 points MC simulation of the 5 state original OTA model in the left column and the simulation of the 25 state reduced OTA model in the right column, obtained through the proposed method, are very close.

Table 3 compares the numerical values of the mean and the standard deviation of the offset voltage V_{os} and the differential gain A_d using a different number of clusters (columns 3, 4, and 5) and shows that they almost have the same characteristics. The relative errors of the state space vectors $\frac{\|x-\hat{x}\|}{\|x\|}$ and the output vectors $\frac{\|y-\hat{y}\|}{\|y\|}$ are also shown in rows 7 and 8 of Table 3, respectively. The state space vector x corresponds to the MC simulation traces and the state vector $\hat{x}$ is obtained by backward projection of the 25 state reduced OTA model simulation traces. The last row of Table 3 shows that the simulation speedup when using the reduced model ranges from 89 to 220 compared with the MC simulation . The reduced OTA model runs 220 times faster when using 10 clusters than MC simulations while providing very close statistical behavior to the MC simulation.

Table 3 Comparison of the simulated OTA Performance using the MC method ($n \times N = 5 \times 1000$) and the reduced model ($q = 25$)

Clusters	*MC*	Reduced model		
	–	30	20	10
$\mu(V_{os})\,(mv)$	-1.00	-1.00	-1.01	-0.98
$\sigma((V_{os})\,(mv)$	13.80	13.80	13.90	13.80
$\mu(A_d)[-1]$	10.20	10.32	10.67	11.39
$\sigma(A_d)[-1]$	5.88	6.28	6.47	6.03
$\frac{\|x-\hat{x}\|}{\|x\|}\,\%$	–	0.71	0.72	0.72
$\frac{\|y-\hat{y}\|}{\|y\|}\,\%$	–	0.09	0.09	0.34
Speedup	–	89	128	220

5.3 Three inverter chain

In this application, we consider a three inverter chain composed of three CMOS inverters, as shown in Fig. 11 which is a basic cell of many integrated circuits such as oscillators, and transmission lines, etc. We apply our method to analyze the effect of threshold voltage mismatch and current factor mismatches due to PV on the three inverter chain gain (G) and rise time (t_r). The gain is computed as the steepest slope of input output DC transfer curve when the input is swept from 0 to 1.8 V. The rise time is computed as the time required for the output x_3 to increase its value from 10 to 90 % of its maximal value when the input u is a sharp step input $u(t) = 1.8 \cdot H(t - 2.5 \times 10^{-9})$. The capacitance at each output node is C = 100 fF. The size of the original problem is $n \times N = 3 \times 1000$. It corresponds to three state variables x_1, x_2, x_3 and 1000 sample points Gaussian threshold voltage v_t and current factor β distributions. The state variables are divided into two systems: the first system has 2000 equations and corresponds to the states x_1 and x_3 as they have the same derivative sign. The second system has 1000 equations and corresponds to the state x_2. The two systems of equations are reduced using two projection bases of size 10×2000 and 5×1000, respectively, which makes the total reduction size $q = 15$.

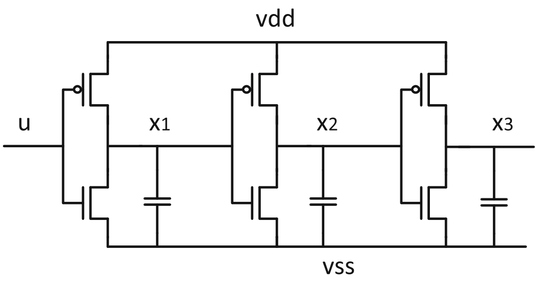


Fig. 11 Three inverter chain circuit

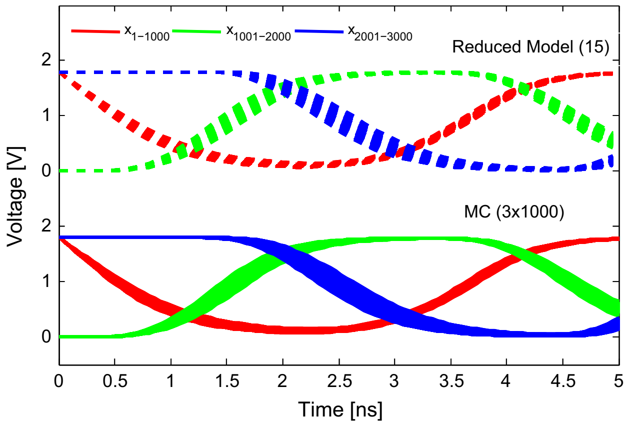


Fig. 12 Inverter chain transient behavior

Figure 12 compares the transient behavior of the three inverter chain states using the 1000 points MC simulation (bottom curves) and the backward projection of the reduced model generated by our method (top curves). The variation of the transient behavior due to process variation is the same in both graphs.

Figure 13 compares the statistical distributions of the gain G and the rise time t_r for 1000 points MC simulation of the 3 state original inverter chain model, shown in the left column, and the simulation of the 15 state reduced inverter chain model, shown in the right column, obtained through the proposed method.

Table 4 compares the numerical values of the mean and the standard deviation of the gain G and the rise time t_r using a different number of clusters (columns 3, 4, and 5) and shows that they almost have the same characteristics. The relative errors of the state space vectors $\frac{\|x-\hat{x}\|}{\|x\|}$ and the output vectors $\frac{\|y-\hat{y}\|}{\|y\|}$ are also shown in rows 7 and 8 of Table 4, respectively. The state space vector x corresponds to the MC simulation traces and the state vector $\hat{x}$ is obtained by backward projection of the 15 state reduced three inverter circuit model simulation traces. The largest values of the relative error $\frac{\|x-\hat{x}\|}{\|x\|} = 3.27\,\%$ and $\frac{\|y-\hat{y}\|}{\|y\|} = 3.98\,\%$, which are only observed when five linearization points are used to build the reduced model, are still in the acceptable range (≤ 5 %). The last row of Table 4 shows that the simulation speedup when using the reduced model ranges from 97 to 215 compared with the MC simulation.

5.4 Discussion

We have proved the efficiency of our proposed fast statistical simulation method for three nonlinear circuits: a current mirror, an operational transconductance amplifier and a three inverter chain. The main challenge was to get

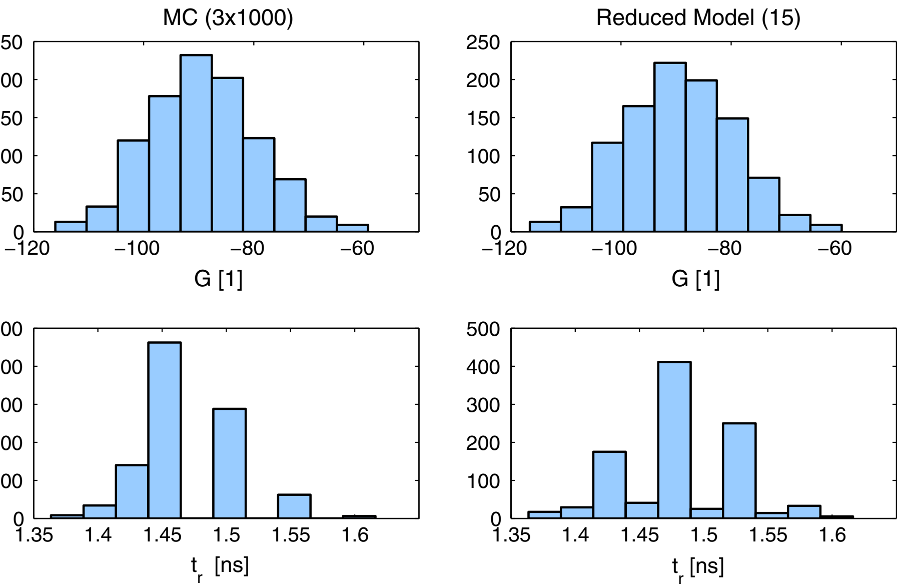


Fig. 13 Three inverter chain gain and rise time distributions

Table 4 Comparison of the simulated three inverter chain performance using the MC method ($n \times N = 3 \times 1000$) and the reduced model ($q = 15$)

Clusters	*MC*	Reduced model		
	–	20	10	5
$\mu(G)[-1]$	89.71	90.06	89.31	82.20
$\sigma(G)[1]$	9.76	9.79	9.38	10.72
$\mu(t_r)\,(ns)$	1.49	1.47	1.43	1.45
$\sigma(t_r)\,(ns)$	0.05	0.05	0.04	0.05
$\frac{\|x-\hat{x}\|}{\|x\|}\%$	–	1.03	2.56	3.27
$\frac{\|y-\hat{y}\|}{\|y\|}\%$	–	1.34	2.13	3.98
Speedup	–	97	153	215

the same statistical behavior as the one obtained by using the MC method but in a smaller amount of time. As shown by numerical experiments provided in this section, the reduced models of the considered applications are capable of reproducing the transistor mismatch effect that was also simulated with the MC method. They yield speedup values in the range 100–500, accurate statistical properties, and small relative errors (less than 5 % relative error) when compared to 1000 samples MC simulations of the original circuit model. We expect that the method can yield much higher speedup values and accurate results if it is applied to larger circuits such as a oscillators, analog filters, mixers, and voltage references, etc. The distributions of the circuits performances shown in Figs. 6, 10, and 13 have almost the same mean and standard deviation values but they display slight variations in the histogram bins compared with the MC method results. These slight variations are related to the accuracy of the reduced model and can be reduced by increasing the number of clusters used by the reduced model at the cost of smaller speedup values. For example, by using a 30 clusters for all three applications, we can achieve a speedup very close to 100 with the same behavior as the MC method. These clusters are expanded using the small perturbation method in order to cover the behavior of the multiple circuit instances. Therefore, we have assumed that the linearization points required for the accurate simulation of the perturbed parameter are either in the same cluster or can be obtained from the remaining clusters. However, this assumption might be unsafe for strongly nonlinear circuits where the DC behavior can completely change under PVT variation. In this case, it might be interesting to use local polynomial models instead of linearized models for better accuracy [18]. Also, the use of numerical continuation methods for dynamical systems [46] instead of the cluster perturbation model might be helpful as well in terms of accuracy.

6 Conclusion

In this paper, we described a new method for increasing the computational efficiency of sampling based statistical methods, such as the MC method. The proposed method is based on nonlinear model order reduction techniques. We first transformed a dynamical system model of n state variables that needs to be solved N times to a larger model of $n \times N$ state variables. We then utilized state space clustering, linearization and proper orthogonal decompositions techniques to reduce the order of the model. The reduced model size q tends to be much smaller than $n \times N$

and can be solved in a more reasonable simulation time. The behavior of the N original models can be retrieved via a simple state space projection. Finally, we did an in-depth and detailed examination of the method and demonstrated the validity of our approach using three case studies: a current mirror ($n \times N = 4000$), an operational transconductance amplifier ($n \times N = 5000$), and a three-stage inverter chain ($n \times N = 3000$) subject to threshold voltage and current factor process variations. The experimental results showed that the reduced models are accurate and provide the same statistics as the MC simulation of the original models while leading to high simulation speedup values.

Even though, one of the main advantages of the presented method lies in the quick estimation of statistical variations of dynamic system performances, it does have some limitations inherited from the used MOR method. The first limitation is that the obtained statistical simulation results cannot be used directly for other MC parameter samples. The process of generating and simulating reduced models for new parameter samples has to be repeated again. The second limitation is related to the large dimensions of the matrices required while generating reduced models. Therefore, an algorithm, which can transform them into sparse matrices and can delete the no longer needed ones on the fly, can be implemented to reduce the memory usage. The third limitation results from the clusters perturbation step, in the case where the circuit behavior changes completely under the effect of process variation. Therefore, a different method for the clusters expansion such as the numerical continuation method for dynamical systems [46] may be employed. Then, the presented method has to be applied in the presence of more severe process variation to make sure that it always yields accurate reduced models.

As a future work, we plan to further develop our approach to include different MOR methods which approximate the nonlinear behavior using polynomial representations or employ symbolic transformation of nonlinear representation and then compare them to the current piecewise linear representation based MOR method. For example, the general purpose nonlinear model order reduction using piecewise polynomial representations [19] or the projection based nonlinear model order reduction approach using quadratic-linear representation of nonlinear systems [40] deserve further investigation in order to be applied in this context. The presented method can also serve as a basis for the analysis of circuit behavior in the presence of noise and the analysis of biologic systems which can be similarly formulated as nonlinear circuit analysis problems. Another potential application of MOR methods and specifically projection based methods is the reduction of the infinite circuit parameter space dimension which represents a bottleneck for circuit verification, optimization, sizing and synthesis approaches.

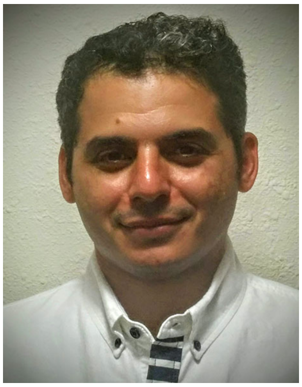
References

1. Kinget, P. R. (2005). Device mismatch and tradeoffs in the design of analog circuits. *IEEE Journal of Solid-State Circuits*, *40*(6), 1212–1224. doi:10.1109/JSSC.2005.848021.
2. Zhigang, H., Tan, S. X. D., Shen, R., & Shi, G. (2011). Performance bound analysis of analog circuits considering process variations. In *ACM/IEEE design automation conference* (pp. 310–315). doi:10.1145/2024724.2024799.
3. Kuo, P. Y., Saibua, S., Huang, G., & Zhou, D. (2012). An efficient method for evaluating analog circuit performance bounds under process variations. *IEEE Transactions on Circuits and Systems II*, *59*(6), 351–355. doi:10.1109/TCSII.2012.2190872.
4. Lahiouel, O., Aridhi, H., Zaki, M. H., & Tahar, S. (2014). Enabling the DC solutions characterization using a fuzzy approach. *IEEE international conference on new circuits and systems* (pp. 161–164). doi:10.1109/NEWCAS.2014.6934008.
5. Jacoboni, C., & Lugli, P. (1989). *The Monte Carlo method for semiconductor device simulation*. Vienna: Springer. doi:10.1007/978-3-7091-6963-6.
6. Bayrakci, A. A., Demir, A., & Tasiran, S. (2010). Fast Monte Carlo estimation of timing yield with importance sampling and transistor-level circuit simulation. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, *29*(9), 1328–1341. doi:10.1109/TCAD.2010.2049042.
7. Keramat, M., & Kielbasa, R. (1997). Latin hypercube sampling Monte Carlo estimation of average quality index for integrated circuits. In J. J. Becerra & E. G. Friedman (Eds.), *Analog design issues in digital VLSI circuits and systems* (pp. 131–142). New York: Springer. doi:10.1007/978-1-4615-6101-9_11.
8. Keramat, M., & Kielbasa, R. (1999). Modified latin hypercube sampling Monte Carlo estimation for average quality index. *Analog Integrated Circuits and Signal Processing*, *19*(1), 87–98. doi:10.1023/A:1008386501079.
9. Singhee, A., & Rutenbar, R. A. (2010). Why quasi-Monte Carlo is better than Monte Carlo or latin hypercube sampling for statistical circuit analysis. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, *29*(11), 1763–1776. doi:10.1109/TCAD.2010.2062750.
10. Wang, J., Ghanta, P., & Vrudhula, S. (2004). Stochastic analysis of interconnect performance in the presence of process variations. In *ACM/IEEE international conference on computer aided design* (pp. 880–886). doi:10.1109/ICCAD.2004.1382698.
11. Strunz, K., & Su, Q. (2008). Stochastic formulation of SPICE-type electronic circuit simulation with polynomial chaos. *ACM Transactions on Modeling and Computer Simulation*, *18*(4), 1–23. doi:10.1145/1391978.1391981.
12. Yu, H., Liu, X., Wang, H., & Tan, S. X. (2010). A fast analog mismatch analysis by an incremental and stochastic trajectory piecewise linear macromodel. In *IEEE Asia and South Pacific design automation conference* (pp. 211–216). doi:10.1109/ASPDAC.2010.5419894.
13. Antoulas, A. C. (2005). *Approximation of large-scale dynamical systems. Advances in design and control*. Philadelphia: Society for Industrial and Applied Mathematics. doi:10.1137/1.9780898718713.
14. Hinze, M., Kunkel, M., & Vierling, M. (2011). POD model order reduction of drift diffusion equations in electrical networks. In *Model reduction for circuit simulation, Lecture Notes in Electrical Engineering* (Vol. 74, pp. 177–192). New York: Springer. doi:10.1007/978-94-007-0089-5_10.

15. Mi, N., Tan, S. X., Liu, P., Cui, J., Cai, J., & Hong, X. (2007). Stochastic extended Krylov subspace method for variational analysis of on-chip power grid networks. In *ACM/IEEE international conference on computer aided design* (pp. 48–53). doi:10.1109/ICCAD.2007.4397242.
16. El-Moselhy, T., & Daniel, L. (2010). Variation aware interconnect extraction using statistical moment preserving model order reduction. In *ACM/IEEE design, automation and test in Europe* (pp. 453–458). doi:10.1109/DATE.2010.5457161.
17. Aridhi, H., Zaki, M. H., & Tahar, S. (2012). Towards improving simulation of analog circuits using model order reduction. In *ACM/IEEE design, automation and test in Europe* (pp. 1337–1342). doi:10.1109/DATE.2012.6176699.
18. Rewienski, M., & White, J. (2006). A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, *22*(2), 155–170. doi:10.1109/TCAD.2002.806601.
19. Dong, N., & Roychowdhury, J. (2008). General-purpose nonlinear model-order reduction using piecewise-polynomial representations. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, *27*(2), 649–654. doi:10.1109/TCAD.2007.907272.
20. Zhang, Z., El-Moselhy, T. A., Elfadel, I. M., & Daniel, L. (2013). Stochastic testing method for transistor-level uncertainty quantification based on generalized polynomial chaos. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, *32*(10), 1533–1545. doi:10.1109/TCAD.2013.2263039.
21. Zhang, Z., Elfadel, I. M., & Daniel, L. (2013b). Uncertainty quantification for integrated circuits: Stochastic spectral methods. In *ACM/IEEE international conference on computer aided design* (pp. 803–810). doi:10.1109/ICCAD.2013.6691205.
22. Zhang, Z., Yang, X., Oseledets, I. V., Karniadakis, G. E., & Daniel, L. (2015). Enabling high-dimensional hierarchical uncertainty quantification by ANOVA and tensor-train decomposition. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, *34*(1), 63–76. doi:10.1109/TCAD.2014.2369505.
23. Zhang, Z., Yang, X., Marucci, G., Maffezzoni, P., Elfadel, I. M., Karniadakis, G., & Daniel, L. (2014). Stochastic testing simulator for integrated circuits and MEMS: Hierarchical and sparse techniques. In *IEEE custom integrated circuits conference* (pp. 1–8). doi:10.1109/CICC.2014.6946009.
24. Ma, X., & Zabaras, N. (2010). An adaptive high-dimensional stochastic model representation technique for the solution of stochastic partial differential equations. *Elsevier Journal of Computational Physics*, *229*(10), 3884–3915. doi:10.1016/j.jcp.2010.01.033.
25. Oseledets, I. V. (2011). Tensor-train decomposition. *SIAM Journal on Scientific Computing*, *33*(5), 2295–2317. doi:10.1137/090752286.
26. Kim, J., Jones, K. D., & Horowitz, M. A. (2010). Fast, non-Monte-Carlo estimation of transient performance variation due to device mismatch. *IEEE Transactions on Circuits and Systems I*, *57*(7), 1746–1755. doi:10.1109/TCSI.2009.2035418.
27. Oehm, J., & Schumacher, K. (1993). Quality assurance and upgrade of analog characteristics by fast mismatch analysis option in network analysis environment. *IEEE Journal of Solid-State Circuits*, *28*(7), 865–871. doi:10.1109/4.222191.
28. Cadence Design Systems. (2004). Virtuoso spectre circuit simulator reference product version 5.1.41.
29. Ma, J. D., & Rutenbar, R. A. (2007). Interval-valued reduced-order statistical interconnect modeling. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, *26*(9), 1602–1613. doi:10.1109/ICCAD.2004.1382621.
30. Pillage, L. T., & Rohrer, R. A. (1990). Asymptotic waveform evaluation for timing analysis. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, *9*(4), 352–366. doi:10.1109/43.45867.
31. Odabasioglu, A., Celik, M., & Pileggi, L. T. (1998). PRIMA: Passive reduced-order interconnect macromodeling algorithm. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, *17*(8), 645–654. doi:10.1109/43.712097.
32. Song, Y., Sai, M. P. D., & Yu, H. (2014). Zonotope-based nonlinear model order reduction for fast performance bound analysis of analog circuits with multiple-interval-valued parameter variations. In *ACM/IEEE design, automation and test in Europe* (pp. 1–6). doi:10.7873/DATE.2014.024.
33. Eppstein, D. (1995). Zonohedra and zonotopes. Technical report 95–53, University of California, Irvine, CA. Accessed December, 1995 from http://www.ics.uci.edu/eppstein/pubs/Epp-TR-95-53.
34. Gu, C. (2012). Model order reduction of nonlinear dynamical systems. Ph.D. thesis, University of California, Berkeley, CA.
35. Lahiouel, O., Aridhi, H., Zaki, M. H., & Tahar, S. (2011). A tool for modeling and analysis of electronic circuits and systems. Technical Report, Concordia University, Montreal, QC, Canada. Accessed October, 2011 from http://www.hvg.ece.concordia.ca/Publications/TECH_REP/TMAAC_TR12/.
36. Buhler, M., Koehl, J., Bickford, J., Hibbeler, J., Schlichtmann, U., Sommer, R., Pronath, M., & Ripp, A. (2006). DFM/DFY design for manufacturability and yield: Influence of process variations in digital, analog and mixed-signal circuit design. In *ACM/IEEE design, automation and test in Europe* (pp. 1–6). doi:10.1109/DATE.2006.243763.
37. Pelgrom, M. J. M., Duinmaijer, A. C. J., & Welbers, A. P. G. (1989). Matching properties of MOS transistors. *IEEE Journal of Solid-State Circuits*, *24*(5), 1433–1439. doi:10.1109/JSSC.1989.572629.
38. Mcandrew, C. C., Bates, J., Ida, R. T., & Drennan, P. (1997). Efficient statistical BJT modeling, why β is more than $\frac{I_c}{I_b}$. In *IEEE bipolar/BiCMOS circuits and technology meeting* (pp. 28–31). doi: 10.1109/BIPOL.1997.647349.
39. Gerstnern, T., & Griebel, M. (1998). Numerical integration using sparse grids. *Numerical Algorithms*, *18*(3—-4), 209–232. doi:10.1023/A:1019129717644.
40. Gu, C. (2011). QLMOR: A projection-based nonlinear model order reduction approach using quadratic-linear representation of nonlinear systems. *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, *30*(9), 1307–1320. doi:10.1109/TCAD.2011.2142184.
41. Dejonghe, D., Deschrijver, D., Dhaene, T., & Gielen, G. (2013). Extracting analytical nonlinear models from analog circuits by recursive vector fitting of transfer function trajectories. In *ACM/IEEE design, automation and test in Europe* (pp. 1448–1453). doi:10.7873/DATE.2013.295.
42. Hartigan, J. A., & Wong, M. A. (1979). Algorithm AS 136: A K-means clustering algorithm. *Journal of the Royal Statistical Society Series C (Applied Statistics)*, *28*(1), 100–108. doi:10.2307/2346830.
43. MATLAB. (2014). Documentation center. http://www.mathworks.com/.
44. Yeh, T. H., Lin, J. C. H., Wong, S. C., Huang, H., & Sun, J. Y. C. (2001). Mismatch characterization of 1.8v and 3.3v devices in 0.18μm mixed signal CMOS technology. In *IEEE international conference on microelectronic test structures* (pp. 77–82). doi:10.1109/ICMTS.2001.928641.
45. Baker, R. J. (2010). *CMOS circuit design, layout, and simulation*. Hoboken: Wiley-IEEE Press. doi:10.1002/9780470891179.
46. Doedel, E. J. (2007). *Numerical continuation methods for dynamical systems* (pp. 1–49), Lecture notes on numerical

analysis of nonlinear equations. Dordrecht: Springer. doi:10.1007/978-1-4020-6356-5_1.

Henda Aridhi received the Engineering Diploma and Master's degrees in electrical engineering from the Tunisia Polytechnic School, Tunisia, in 2005 and 2008, respectively. She worked for CISCO systems as a SW/HW verification engineer from 2005 to 2008. She continued her graduate studies in the field of analog and mixed signal (AMS) systems, under the supervision of Prof. Sofiene Tahar, at the Hardware Verification Group, Concordia University, Montreal, Canada, where she received her Ph.D. degree in electrical engineering in 2015. Her research interests include design and verification of digital and analog systems, model order reduction for nonlinear systems, and formal hardware verification.



Mohamed H. Zaki did his Bachelor's degree in Electrical Engineering from Ain Shams University, Egypt, in 2000. He continued his graduate studies at the Hardware Verification Group of Concordia University, Montreal, Canada, under the supervision of Prof. Sofiène Tahar, where he obtained his Master's and Ph.D. degrees in 2002 and 2008, respectively. He then conducted a post-doctoral training at the Computer Science department of the University of British Columbia, Vancouver. Since 2010, he is working as a consultant at the start-up company Innovia Technologies, and is a visiting research associate at Concordia University in the field of AMS design verification.

Sofiene Tahar received the Diploma degree in Computer Engineering from the University of Darmstadt, Germany, in 1990, and the Ph.D. degree with distinction in Computer Science from the University of Karlsruhe, Germany, in 1994. Currently, he is a Professor and Research Chair in formal verification of systems-on-chip with the Department of Electrical and Computer Engineering, Concordia University, Montreal. He has made contributions and published papers in the areas of formal hardware verification, system-on-chip verification, AMS circuit verification, and probabilistic, statistical and reliability analysis of systems. Dr. Tahar is the Founder and Director of the Hardware Verification Group at Concordia University. In 2007, he was named University Research Fellow upon receiving Concordia University's Senior Research Award. He is a Senior member of ACM and a Professional Engineer in the Province of Quebec.