# Variability and Statistical Analysis Flow for Dynamic Linear Systems with Large Number of Inputs

A. Lucas Martins[1] Jorge Fernández Villena[2] L. Miguel Silveira[1]
[1]INESC-ID, Instituto Superior Técnico Universidade de Lisboa, Lisbon, Portugal
[2]Cadence Design Systems Munich, Germany
lucas@algos.inesc-id.pt jvillena@cadence.com lms@inesc-id.pt

**Abstract— Fast analysis of the dynamics of large linear systems with large number of inputs, such as power grid (PG) nets, is a required component of system verification platforms. Such analysis, exhibiting a considerable memory footprint and requiring intensive computations and advanced numerical techniques, has been the framework of recent approaches. However analyzing the effect of design variability, which can have a critical impact on the power distribution across the chip, especially when considering its dynamic performance, poses a unmet challenge. Existing approaches collect information about the voltage and current fluctuations in key nodes that may lead to erroneous behavior or relevant performance changes. This is achieved through repetitive extraction and/or simulation of the large linear RC network for a very broad number of parameter settings. Unfortunately network size and the plethora of different settings that requires investigation implies that such an approach can be exceedingly time consuming, even if parallel architectures are used. In order to address such a challenge, this paper introduces an alternative analysis flow that builds a parameterized model of the time domain node voltages on the fly, using the nominal time domain simulation as starting point. Once such model is generated, the effect of variability in the time response can be efficiently evaluated for multiple settings, allowing collection of relevant variation and statistic information of the impact of a large number of parameters in the current design. The performance of the methodology is evaluated on an set of standard PG extracted netlists, showing large improvements in terms of speed with modest memory requirements while maintaining an acceptable degree of accuracy.**

## I. Introduction

Current nanometer range printed feature sizes are a small fraction of the wavelength of light used in lithographic processes. Such subwavelength printing makes features highly susceptible to any variations in the lithographic process conditions, which in turn leads to printed designs exhibiting increased variability. A victim of these variations is the critical area of power grid (PG) design. Todays VLSI designs feature huge power distribution networks that span across the entire chip area and through several layers to provide circuit bias. PG analysis requires the detailed interconnect and parasitic extraction of the layout, which usually leads to large resistor and capacitor networks with several million elements whose simulation is computationally very challenging. Variations in wire dimensions resulting from lithographic errors have a strong impact on the PG behavior in multiple ways, potentially causing voltage fluctuations and considerable current variations, that can lead, for example, to exacerbated electromigration problems. Analyzing the PG network in such scenarios entails extracting the interconnect and parasitic model of the power grid accounting for litho-induced errors, and then performing a full time-domain simulation for every possible variation setting to verify the currents on all wires. Given the number of wires to consider and parameters to account for, and even discarding systematic errors, the amount of potential random fluctuations alone would lead to an intractable number of settings to analyze in order to extract some meaningful statistical information about the impact of the variations.

In this paper, we propose an alternative, more flexible, approach for efficiently estimating the effects of variability in dynamic PG and interconnect analysis. The method is based on a linear approximation of the voltage at the grid nodes, and is divided into two main steps. The first step is the model generation, in which a standard time domain simulation of the PG is performed, combined with the storage of the nominal (no variation) voltages at every node in the model. Each time step of the analysis is combined with a variational analysis, in which a low order approximation of the node voltages with respect to the parameters is obtained. This representation with respect to the number of parameters can be efficiently generated from the nominal voltages without the need for new factorizations or expensive solves. To minimize the required memory storage, the nominal voltages and parameter coefficients are stored in a compressed form, using an on-the-fly incremental principal component analysis. The final result is a parameterized low rank approximation of the node voltages, where the full set of voltages can be represented as a linear combination of a comparatively small set of vectors. The second step, or model evaluation, only involves a set of dense matrix operations for a given parameter setting. Thus the extraction and analysis of statistical information of the variability impact on the PG design can be efficiently obtained by evaluating a large number of parameter setting in reasonable runtime, without the need for any additional time domain simulations.

This scheme can be easily combined with Litho/CMP simulators and embedded in enhanced design cycles [16]. The variations resulting from lithography deviations and corrections can be estimated and their effects taken into account by the proposed approach to determine their impact on circuit performance, e.g. to estimate the distribution of the maximum resistor current (relevant for electromigration analysis), the maximum voltage drop (relevant for grid integrity analysis), or to obtain statistical information about node variations.

The manuscript is structured as follows. Section II briefly reviews the PG problem and the basics of the variability modeling and analysis. Section III introduces the proposed methodology, and discusses computational implementations and practical considerations. Finally results for a set of industrial PG benchmarks are presented in Section IV, and conclusions drawn in Section V.

## II. Background and Problem Definition

### A. Variational Power Grid Model

A power grid model can be obtained through extraction and assumes that VDD and GND strips, as well as vias, are modeled resistively. The coupling resulting from the overlapping between metal strips in different levels is modeled through a coupling capacitance or a set of capacitances to ground. While simplified, this type of model

is representative of what is used in commercial tools [17] and is sufficient for most analysis, including voltage and IR drop computation or electromigration. For analysis of the dynamic behavior of the grid, a set of equations can be obtained:

$$\mathbf{G}\ \mathbf{v}(\mathbf{t}) + \mathbf{C}\ \frac{d}{dt}\mathbf{v}(\mathbf{t}) = \mathbf{i}(\mathbf{t}) \tag{1}$$

where $\mathbf{G}, \mathbf{C} \in \mathbb{R}^{n\times n}$ are respectively the conductance and capacitance matrices of the system, $\mathbf{v}(\mathbf{t}) \in \mathbb{R}^n$ is the vector of grid node voltages through time, and $\mathbf{i}(\mathbf{t}) \in \mathbb{R}^n$ the vector of bias currents at some of the grid nodes (to simplify, grid bias is assumed to have been converted to current sources through Norton equivalents). Pin inductance can be easily decouple from the internal RC netlist and be accounted for as part of the external driver. Although an RLC equivalent model accounting for inductive packaging connections can also be extracted and analyzed with our approach, for the sake of simplicity we will limit ourselves to on-chip PG analysis for which inductance effects are usually negligible, and assume an RC model.

Solution of (1) can be computationally expensive for very large matrices. However, the high sparsity and regular structure of the matrices also allows efficient direct and iterative methods to be applied, albeit at some memory and computational cost, given the large matrix sizes [1–4]. Alternative approaches to solving the system exploit the connection with Random Walks [5], whereas others try to apply model order reduction [6–8], to compress the system matrix and speed up the solve, although the level of compression is limited by the large number of independent inputs that usually drive this systems.

For variability analysis, we consider possible variations on the network elements, resistors and capacitors. The value of the resistance depends on the resistivity $\rho$, length $L$ and section $S$ of the wires, which for most planar structures is represented as the product of the width $W$ and thickness $T$, whereas the capacitance depends on the medium permittivity, $\epsilon$, the area of the plates, $A$, and its distance, $d$. The values of these electrical and geometrical parameters are subject to the effect of fluctuations, random and systematic deviations inherent to the fabrication process. We can represent $R$ and $C$ as functions of the parameters in the set $\mathbf{p} = [\Delta\rho, \Delta L, \Delta W, \Delta T, \Delta\epsilon, \Delta A, \Delta d]$,

$$R(\mathbf{p}) = \tfrac{(\rho+\Delta\rho)(L+\Delta L)}{(W+\Delta W)(T+\Delta T)} \qquad C(\mathbf{p}) = (\epsilon+\Delta\epsilon)\tfrac{A+\Delta A}{d+\Delta d} \tag{2}$$

Different variability models can be applied, leading to different approximations. For instance, if all variations are considered independent and a low order approximation is deemed sufficient (which is reasonable for small parameter variations around the nominal value),

$$\begin{aligned} R(\mathbf{p}) &\approx R_0\left(1 + \Delta\rho + \Delta L + \textstyle\sum_{k=1}^{O}(-\Delta W)^k + \sum_{k=1}^{O}(-\Delta T)^k\right) \\ C(\mathbf{p}) &\approx C_0\left(1 + \Delta\epsilon + \Delta A + \textstyle\sum_{k=1}^{O}(-\Delta d)^k\right) \end{aligned} \tag{3}$$

where $R_0$ and $C_0$ are the nominal values (no variation in the parameter set) and $O$ is the truncated order of the approximation (usually a small order will suffice for near perfect approximation). Alternatively for inter-chip variability analysis, these variations can be assumed as localized in certain chip areas or regions and thus a different set of parameters can be considered for different areas within the chip. In any case, potentially hundreds of parameters must be handled during the simulation and design stages. If the parameter dependence for each resistor (capacitor) is represented as an arbitrary order Taylor series such as in (3), the parameter dependent admittance (capacitance) matrix $\mathbf{G}(\mathbf{p})$ ($\mathbf{C}(\mathbf{p})$) can be efficiently formed using the incidence of each element in the circuit. Alternatively, a representation such as (3) might be available for the admittance directly, from which $\mathbf{G}(\mathbf{p})$ can be computed through stamping (similarly for $\mathbf{C}(\mathbf{p})$).

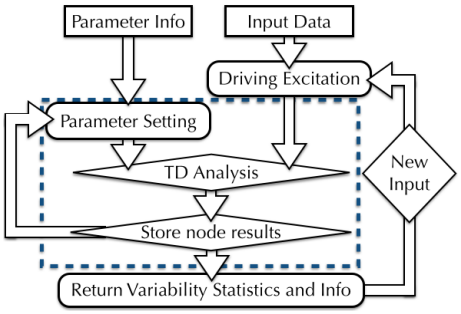


Fig. 1. Traditional analysis flow. For each time-domain (TD) stimuli, a parameter setting is selected, and the full TD simulation is done. New settings modify the system matrices, and imply a completely new TD analysis.

A common representation is to generate a set of sensitivities of the admittance and capacitance matrices, $\mathbf{G}(\mathbf{p})$ and $\mathbf{C}(\mathbf{p})$, to an arbitrary order. For instance, the first order Taylor series can be written as a function of the nominal, $\mathbf{G_0}$ and $\mathbf{C_0}$ , and the sensitivities, $\mathbf{G}_k$ and $\mathbf{C}_k$, to a set of $P$ parameters, $p_k$, as

$$\mathbf{G}(\mathbf{p}) = \mathbf{G}_0 + \textstyle\sum_{k=1}^{P} \mathbf{G}_k\, p_k \qquad \mathbf{C}(\mathbf{p}) = \mathbf{C}_0 + \sum_{k=1}^{P} \mathbf{C}_k\, p_k \tag{4}$$

This representation can be trivially extended to higher order.

### *B. Estimating the Effect of Variations*

To evaluate the time-response under a certain parameter setting, $\mathbf{p}$, we can discretize (1) in the time-domain, for instance using the backward-Euler method with a fixed timestep $h$

$$\mathbf{Y}\ (\mathbf{p})\mathbf{v}(t,\ \mathbf{p}) = \mathbf{i}(t) + \frac{1}{h}\ \mathbf{C}\ (\mathbf{p})\mathbf{v}(t-h,\ \mathbf{p}) \tag{5}$$

where $\mathbf{Y}(\mathbf{p}) = \mathbf{G}(\mathbf{p}) + \mathbf{C}(\mathbf{p})/h$, and we assumed that the parameter set values are time independent and the external stimuli does not depend upon the parameters. The generation of the time-evolution of the grid voltages on an interval of interest $[0, T]$, due to a specific set of stimuli requires two steps: a) evaluate the system matrices, i.e. obtain $\mathbf{G}(\mathbf{p})$ and $\mathbf{C}(\mathbf{p})$ for the given parameter setting $\bar{\mathbf{p}}$, and b) solve the system in sequence for $t = 0, \cdots, Mh = T$.

The evaluation of $\mathbf{G}(\mathbf{p})$ and $\mathbf{C}(\mathbf{p})$ can be easily done, either by stamping the values of (3) for the given parameter, or by evaluating the low order approximation in (4). This generates a new system valid for the current parameter set which can be solved, the solution recorded and the process repeated for the next parameter setting. For an illustration of the process, see Figure 1.

Herein lies one of the basic difficulties with estimating the impact of parameter variability on grid behavior: for each parameter setting $\bar{\mathbf{p}}$ the matrices in (5) change; to generate the time-evolution of the voltage solution $\mathbf{v}(t,\ \bar{\mathbf{p}})$ , implies a re-factorization of the matrix $\mathbf{Y}(\bar{\mathbf{p}})$ or a new solution process if an iterative method is used. This is very costly and in fact to compute the impact of multiple parameter settings is quickly overwhelming.

Some work has been devoted to the analysis of such networks under parameter variations [9], including combination with MOR methods [10], the extension of Random Walks for variational analysis [11, 12], incremental sparse methods [13], bounded effects and estimates for voltage drop variations for statistical methods [14], and the use of Hermite Polynomials for the generation of a variational model that allows a stochastic analysis [15]. Most of these existing methodologies for the variability analysis of PGs focus on the idea of accelerating the solution of this system for different parameter settings, either

by incremental analysis and approximations, system parametrization and model reduction, or localized updates of the solution. For large parameter settings, for instance, to estimate the distribution of peak resistor current (relevant for electromigration analysis), peak voltage drop (relevant for grid integrity analysis) or any other required metric, more efficient methods are required.

## III. Proposed Variability Analysis

Parameter variations are responsible for the time evolution of the node voltages drifting from the nominal, whose estimation is critical for safe and reliable PG design. Our analysis relies on two basic assumptions. We conjecture that the node voltages are smoothly time-varying and can be accurately recovered with low order approximations. Furthermore, the voltages for different time steps and parameter settings are correlated, both from a spatial as well as a time viewpoint. This correlation translates into the numerical concept of rank deficiency, which means that we can represent the voltages in the nodes for different time points and parameter settings as a linear combination of a small basis vectors:

$$\mathbf{v}(mh, \mathbf{p}) \approx \mathbf{Q}\, \alpha(mh, \mathbf{p}) \tag{6}$$

where $m$ corresponds to a discretized time step, $t = mh$, $\mathbf{p}$ is the parameter vector, and $\mathbf{Q} \in \mathbb{R}^{n \times q}$ is the compressed basis, with $q << n$ a relatively small number and $\alpha(m, \mathbf{p})$ a set of coefficients (an appropriate tolerance, $\mu$ is used at each timepoint to determine whether a new vector should be added to the basis). If no compression were used, the above expression would still be valid but now $Q \in \mathbb{R}^{n \times (M(P+1))}$ which is huge, rendering the method unusable.

Our approach involves two steps. The first step is the proposed model generation, where the terms for the representation in (6) are generated. The second is the analysis, in which the representation in (6) is used to evaluate the node voltages at different time points for as many different parameter settings as desired.

### A. Model Generation – SET UP

The model generation further requires the separation of the time space and the parameter space. We will use a standard time-domain simulation of the nominal model (zero variation), to generate the required basis. Following the approach outlined in (5), we can fix a time step $h$, and establish the backward-Euler regression formula to compute the voltage vector at the next time point,

$$\mathbf{Y}_0 \mathbf{v}_0^{(m)} = \mathbf{i}^{(m)} + \frac{1}{h}\mathbf{C}_0 \mathbf{v}_0^{(m-1)} \tag{7}$$

where to simplify notation we now write $\mathbf{v}_0^m$ to represent the nominal voltage vector at the $m$-th time point ($t = mh$) (similarly for $\mathbf{i}^{(m)}$). In order to determine the effect of perturbations on the parameters, we handle each time point as a static model, and pursue a similar approach to the one proposed in [18] for the static case. We aim to generate a set of linear terms that approximate the effect of the parameter variations at such time step

$$\mathbf{v}^{(m)}(\mathbf{p}) = \mathbf{v}_0^{(m)} + \textstyle\sum_k^P p_k\, \mathbf{v}_k^{(m)} \tag{8}$$

Such an approximation is presumably sufficiently accurate if the range of variation of the parameter set $\mathbf{p}$ is small compared to the nominal value, which is often the norm. Otherwise higher order terms can be added for the sake of accuracy, at the cost of rapidly increasing the computing requirements, specially when multiple parameters are involved. Similarly to [18], the coefficients $\mathbf{v}_k^{(m)}$ can be efficiently computed by either expanding the voltage in terms of a low order Taylor series with respect to the parameters (an approach previously exploited in [19] for parametric Model Order Reduction), or by matching the perturbed response of the voltage vector at the maximum (or any other given) variation for each parameter. Both cases establish a recursion for each parameter that avoids parameter settings. We illustrate here how to compute these for a single parameter.

For a single parameter $p_1$, with maximum value $\hat{p}_1$, and a single term $\mathbf{v}_1^{(m)}$, matching the response at the maximum variation leads to

$$\mathbf{v}_0^{(m)} + \hat{p}_1\, \mathbf{v}_1^{(m)} = \mathbf{v}(\hat{p}_1)^{(m)} \tag{9}$$

We can compute the value of $\mathbf{v}(\hat{p}_1)^{(m)}$ by applying backward-euler,

$$\mathbf{Y}(\hat{p}_1)\mathbf{v}(\hat{p}_1)^{(m)} = \mathbf{i}^{(m)} + \frac{1}{h}\mathbf{C}(\hat{p}_1)\mathbf{v}(\hat{p}_1)^{(m-1)} \tag{10}$$

We can approximate the matrices by their first order parametric representation $\mathbf{Y}(\hat{p}_1) = \mathbf{Y}_0 + \hat{p}_1\, \mathbf{Y}_1$, $\mathbf{C}(\hat{p}_1) = \mathbf{C}_0 + \hat{p}_1\, \mathbf{C}_1$, and do the same for the previous time point voltage, $\mathbf{v}(\hat{p}_1)^{(m-1)} = \mathbf{v}_0^{(m-1)} + \hat{p}_1 \mathbf{v}_1^{(m-1)}$. After some linear algebra, we get

$$\mathbf{v}(\hat{p}_1)^{(m)} = (\mathbf{Y}_0 + \hat{p}_1\, \mathbf{Y}_1)^{-1} \left( \mathbf{i}^{(m)} + \frac{1}{h}\mathbf{C}_0 \mathbf{v}_0^{(m-1)} + \hat{p}_1\, \mathbf{b} \right) \tag{11}$$

where $\mathbf{b} = 1/h \left( \mathbf{C}_1 \mathbf{v}_0^{(m-1)} + \mathbf{C}_0 \mathbf{v}_1^{(m-1)} + \hat{p}_1\, \mathbf{C}_1\, \mathbf{v}_1^{(m-1)} \right)$.

We can approximate the inverse term in (11) by a truncated series

$$(\mathbf{Y}_0 + \hat{p}_1\, \mathbf{Y}_1)^{-1} = \left( \mathbf{I} - \mathbf{\Delta}^1 + \mathbf{\Delta}^2 - \ldots \right) \mathbf{Y}_0^{-1} \tag{12}$$

where $\mathbf{\Delta} = \hat{p}_1\, \mathbf{Y}_0^{-1} \mathbf{Y}_1$. By substituting the previous results in (9), reordering the terms, and using the backward-euler equivalence in (7), we finally arrive at the following expression

$$\mathbf{v}_1^{(m)} = \left( \mathbf{I} - \mathbf{\Delta} + \mathbf{\Delta}^2 - \ldots \right) \mathbf{Y}_0^{-1} \left( \mathbf{b} - \mathbf{Y}_1 \mathbf{v}_0^{(m)} \right) \tag{13}$$

A simplified approach is to expand the voltage $\mathbf{v}(p)^{(m)}$ and matrices $\mathbf{G}(p)$ and $\mathbf{C}(p)$ in (5) by their corresponding first order expansion (8) and (4), and to match the coefficients of the same order. This is equivalent to obtain the derivative of (5) with respect to the given parameter $p_1$ around its nominal value, which, neglecting high order terms, eventually give us the following approximation

$$\mathbf{v}_1^{(m)} = \mathbf{Y}_0^{-1} \left( \frac{1}{h}(\mathbf{C}_1 \mathbf{v}_0^{(m-1)} + \mathbf{C}_0 \mathbf{v}_1^{(m-1)}) - \mathbf{Y}_1 \mathbf{v}_0^{(m)} \right) \tag{14}$$

Note that the computation of the terms in the series in (13) and in (14) can be done in a recursive fashion, where the solves only involve $\mathbf{Y}_0$, and a series of matrix vector products that only depend on previously available data, namely the nominal voltage at the previous and current time points ($\mathbf{v}_0^m$ and $\mathbf{v}_0^{m-1}$), and the perturbation of the voltage at the previous time point ($\mathbf{v}_1^{m-1}$). The same procedure can be followed for all the other parameters.

These computations are independent for each parameter, and thus embarrassingly parallelizable. Furthermore, if $\mathbf{Y}_0$ has been factorized (and stored), we only need to re-use the factors to solve different right hand sides to generate all the $\mathbf{v}_k^{(m)}$ terms.

Once the vectors for the current time point $(m)$ have been generated, we use them to update the basis $\mathbf{Q}$ with the dominant vectors. In order to do so, we need to orthogonalize the new vectors with respect to the vectors already in the basis. Then we can apply a rank-revealing QR (**rrqr**) factorization and truncation based on the associated vector norms, to keep the dominant vectors in the basis, and use them to update the $\mathbf{Q}$ matrix, and the $\mathbf{R}$ matrix, that we will need to keep track of the relation of the vectors to the time points and parameters.

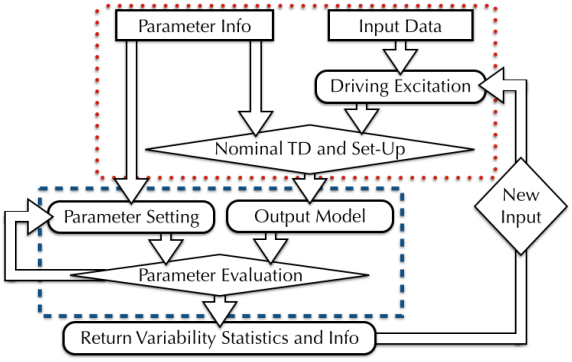


Fig. 2. Proposed variability analysis. A single TD analysis is required for each stimuli, which is combined with the model generation. The model evaluation is then independently applied to analyze the effect of the parameter variations in the corresponding TD run.

### B. Model Use - EVALUATION

The set up phase provides two matrices. $\mathbf{Q} \in \mathbb{R}^{n \times q}$ is the compressed basis that can be used to reconstruct the voltage on the $n$ circuit nodes for any of the $M$ time points and for any parameter setting. $\mathbf{R} \in \mathbb{R}^{q \times M\,(P+1)}$ is the matrix that allows us to select and weight the vectors in $\mathbf{Q}$ to actually generate the voltage vector.

The columns of $\mathbf{R}$ can be subdivided in subsets of $P+1$ columns. Each of these subsets is related to a time point $m$, with the first column the one related to the nominal response, and each of the remaining $P$ columns related to a parameter in the parameter set $\mathbf{p}$. If we want to evaluate the voltages for the time point $m$, for a given parameter setting $\bar{\mathbf{p}}$, we just need to select the corresponding columns in $\mathbf{R}$, and apply a set of matrix vector products

$$\mathbf{v}^{(m)}(\mathbf{p} = \bar{\mathbf{p}}) = \mathbf{Q}\left(\mathbf{R}(m)\,[1\ \ \bar{\mathbf{p}}]^T\right) \tag{15}$$

where $\mathbf{R}$ is the sub-block of $\mathbf{R}$ that contains the columns that go from $m\,(P+1)$ to $(m+1)\,(P+1)$, and $[1\ \ \bar{\mathbf{p}}]$ is a row vector of size $P+1$ with value 1 for the nominal position, and the value of the parameters for the given setting in the remaining positions.

An interesting advantage of this approach is that time-dependent parameter variations (e.g. to determine the effect of temperature variations whose value is known to vary with time) can be easily accommodated since the computation in the evaluate stage is done one timepoint at a time using the values of the parameters at each timepoint.

### C. Computational Issues

Figure 2 presents a depiction of the proposed flow, in which the SET UP (dotted red box) and EVALUATION (dashed blue box) stages are clearly separated. The SET UP stage, akin to model generation, is done only once, and computes the terms $\mathbf{v}_k(k = 1, \cdots, P)$ required for generating the approximation. The selection of the value $\hat{p}_k$ will determine the region of best accuracy. Zero value will give a series approximation with better accuracy at small perturbations. Fixing $\hat{p}_k$ at a given variation will give better accuracy around nominal and that given variation, but its accuracy may degrade for other values if the voltage response is not close to linear. We assume that the sensitivities of the conductance and capacitance matrices, $\mathbf{G}_k$ and $\mathbf{C}_k$, are given, or in cases where they are not available, we have access to an extractor and can generate them through a simple method such as differentiation, i.e. $\mathbf{G}_k = (\mathbf{G}(\hat{p_k}) - \mathbf{G}_0)/(\hat{p_k} - p_0)$.

The complete flow requires factorizing the nominal pencil term, $\mathbf{Y}_0 = \mathbf{G}_0 + \mathbf{C}_0/h$ (e.g. using Cholesky factorization) only once, and storing the factors. These factors are used to solve (i.e. back-solve using the factors) for the nominal voltage $\mathbf{v}_0$, and then for the sensitivities of the linear approximation $\mathbf{v}_k$ at each time step. For a linear approximation the total number of (back-)solves per time point is directly proportional to the total number of parameters, $P$. The generation of the different linear terms $\mathbf{v}_k$ for each time step in the SET UP phase is perfectly independent once the factorization of the nominal matrix is done. Multiple cores or machines can therefore be used in any type of architecture with speedup limited only by the number of parameters of the problem. The factorizations can be done either as a block for each time step, or as rank-one update for each parameter. In any case they can be efficiently applied by using implicit Householder-based rank-revealing QR methods [20]. A relative threshold $\mu$ can be used to drop the columns of $\mathbf{Q}$ and rows of $\mathbf{R}$ associated with rank deficient vectors, and thus reduce memory requirements for the storage. Also, the analysis can be focused on a limited number ($r < n$) of relevant nodes (e.g. critical nodes or outputs) which could help to minimize memory requirements.

The EVALUATION stage, akin to model evaluation, computes the results of the variational analysis. Since the "coefficients" $\mathbf{v}_k$ of the linear approximation have been pre-computed, the actual variational analysis is very fast, as it only requires a set of dense matrix vector products for each parameter setting. The values of each parameter setting are stored in a column vector, and left-multiplied by the columns of matrix $\mathbf{R}$ corresponding to the desired time point, to generate the coefficient vector $\alpha(m, \mathbf{p})$ in (6) for the given parameter setting. This vector is used as a weighting for the basis $\mathbf{Q}$ to reconstruct the voltage in the nodes. Efficient level 2 BLAS routines can be applied in this stage. Furthermore, different parameter settings can be divided between an arbitrary number of machines and computed concurrently both in shared and/or distributed memory architectures. The computation is fully/embarrassingly parallel. Additionally, the evaluation of the dense matrix vector products can be accelerated by using GPUs, since the basis remains constant for any time point and parameter settings: multiple parameter settings can be loaded to the GPU and computed in a vectorial fashion.

Interestingly, the proposed approach can benefit from complementary acceleration techniques to solve the time domain analysis of the nominal system with constant matrix $\mathbf{Y}_0$, including $\mathcal{H}$-matrix representations and fast solvers, Model Order Reduction (MOR), or multigrid and iterative methods [1, 2, 4] to speed up the SET UP phase.

## IV. Experiments and Results

In this section we present results from application of the proposed approach to a set of realistic designs taken from the IBM Power Grid Benchmarks for Transient Analysis [17]. Table I shows the characteristics of the examples used where n stands for the effective number of nodes (and therefore of capacitors) and r for the number of resistors, after processing the data and removing shorts. We have further divided the PG into R local regions, using topological and net information. This division tries to emulate locality of the perturbations. For each region we used 6 independent parameters from the set of parameters (resistivity, length, width and thickness of resistor wires and permittivity, plate width and plate distance of capacitors). P stands for the total number of independent parameters (6 times the number of regions). The parameters affect the resistors in the corresponding region, with a $3\sigma$ effect of $\pm 10\%$ for the resistivity, $\pm 1\%$ for the length, $\pm 30\%$ for the thickness and $\pm 30\%$ for the width. Capacitor values in each region are affected with a $3\sigma$ effect of $\pm 10\%$ for the permittivity, $\pm 30\%$ for each dimension in plate area and $\pm 10\%$ for plate separation). We use the original perturbed system, i.e. the stamp of the per-

TABLE I
BENCHMARKS CHARACTERISTICS

| PG | #nodes | #inp | #res | #cap | R | P |
|---|---|---|---|---|---|---|
| ibmpg1t | 25095 | 8868 | 40801 | 25095 | 8 | 48 |
| ibmpg2t | 163577 | 36792 | 245163 | 163577 | 32 | 192 |
| ibmpg3t | 1039624 | 189492 | 1602626 | 1039624 | 16 | 96 |

TABLE II
EVALUATING COMPRESSIBILITY IN THE TIME DOMAIN

| $\mu$ | q | $E_{abs}$ | $A_{abs}$ | $A_{rel}(\%)$ | Mem Savings |
|---|---|---|---|---|---|
| 1.0e-06 | 363 | 0.186 | 0.003 | 0.71% | 92.66% |
| 1.0e-04 | 262 | 0.189 | 0.003 | 0.71% | 94.71% |
| 1.0e-02 | 83 | 0.194 | 0.003 | 3.33% | 98.32% |
| 1.0 | 50 | 2.220 | 0.022 | 18.45% | 98.99% |

turbed resistors and capacitors under the effect of some parameters, using a $3^{rd}$ order approximation in (8), as the golden solution against which the accuracy of the methods is presented. We compare standard Taylor series approximation (10), with the proposed approach, using the same sensitivities as in (10), obtained by direct differentiation at the maximum perturbation for each parameter.

All our examples were run in a MATLAB environment, which means times are merely indicative, since it is a non-compiled language. Nonetheless, since all experiments were conducted in the same environment, the relative comparisons are a relatively fair indication. To quantify the results, we show the maximum absolute error ($E_{abs}$), the average absolute error ($A_{abs}$), and the average relative error ($A_{rel}$), obtained from the simulation of a large number of parameter settings and considering all timepoints in the simulation:

$$\begin{aligned} E_{abs} &= \max |\mathbf{v}^m(\bar{\mathbf{p}}) - \mathbf{v_r^m}|\,,\ \forall m = 1, \cdots, M \\ A_{abs} &= \text{mean}\left(|\mathbf{v}^m(\bar{\mathbf{p}}) - \mathbf{v_r^m}|\right),\ \forall m = 1, \cdots, M \\ A_{rel1} &= \text{mean}\left(|\mathbf{v}^m(\bar{\mathbf{p}}) - \mathbf{v_r^m}|\,./\,|\mathbf{v_r^m}|\right),\ \forall m = 1, \cdots, M \end{aligned} \tag{16}$$

where $\mathbf{v}^m(\bar{\mathbf{p}})$ is the system solution with the perturbed response at timepoint $t = mh$, and $\mathbf{v_r^m}$ is the reference solution at the same timepoint. These errors are obtained from the MC analysis of a number of parameter settings generated using a normal distribution for the $3\sigma$ variation as indicated above.

To start with, we ascertain the validity of our assumption that the node voltages are correlated and their representation can be compressed, thus saving memory as well as computation time during the evaluation of the impact of variations. Table II shows, for example `ibmpg1t`, the effect of decreasing the tolerance $\mu$ (see the discussion around (6)) on our ability to accurately recover the time-domain representation of the network (errors shown are the maximum seen for a set of 1000 random parameter settings). The data shows that the representation is in fact quite compressible. For values of $\mu$ of $10^{-2}$ and comparing our approach with a method based on an uncompressed basis (see the discussion around Eqn. (6)) we achieve savings of over $98\%$ in memory while incurring on a very reasonable absolute and relative error. This proves our assumption that the time-domain waveforms are indeed highly correlated and the data is compressible along the time dimension.

Next, in Table III, we show, for examples `ibmpg1t` and `ibmpg2t`, the effect of an increasing number of parameters in our ability to compress the representation. The data shows that for increasing number of parameters (P), more basis vectors (q) are required, but the rate of increasing is slower than the increase in the number of parameters, again showing considerable compressibility and the potential for considerable memory savings which will later translate into computational savings in the EVALUATION phase.

TABLE III
EVALUATING COMPRESSIBILITY IN THE PARAMETER SPACE

| PG | P | 24 | 48 | 96 | 192 |
|---|---|---|---|---|---|
| ibmpg1t | q | 62 | 83 | 111 | 147 |
| | Mem Savings | 97.54% | 98.32% | 98.87% | 99.246% |
| ibmpg2t | q | 47 | 64 | 89 | 112 |
| | Mem Savings | 98.14% | 98.71% | 99.09% | 99.425% |

There are remarkable memory savings in compressing the representation even for large number of parameters. This again is in line with our assumptions of considerable correlation in the parameter space.

A third dimension where there is also potential for large compression is the space dimension. To achieve such compression would require employing model order reduction schemes which, as discussed, can be readily applied on the current problem in conjunction with time and parameter space compression. Such effort has in fact been heavily researched [6–8] but is not pursued here at this time.

Table IV presents the main results for the benchmarks used. The tables have the same structure, presenting the errors according to the criteria in (16), using the golden solution as reference $\mathbf{v_r}$. They also show the times required for both the Taylor Series approximation as well as the proposed methodology for computing the response for all the parameter settings considered. The computation time for the proposed solution is split between the set up time (required for generating the different approximation terms, and thus computed only once independently of the number of settings) and the evaluation time, which depends on the number of settings under investigation. For the sake of space, the times required for the golden solution are omitted, since they are the same as the Taylor Series (once the matrix is evaluated for a parameter setting, both approaches require a full TD simulation). The results also show the number of basis vectors and the memory required for the compressed approach.

The proposed approach requires more time in the set up stage, but the extra effort is rewarded during the evaluation stage, where even for a relatively small number of parameter settings (1000 in this case), we already have a large speed up: $9.4\times$, $1.7\times$, and $2.5\times$ for `ibmpg1t`, `ibmpg2t` and `ibmpg3t` respectively. The speed up per iteration (without taking into account the Set Up time, and thus an indicator of the actual speed up for very large parameter settings) can be larger than 20 fold for the larger grid. This speed up is expected to increase for larger examples, since the cost of the solve required by TS is $O(n^\beta)$, with $\beta > 1$, whereas the cost of evaluating the proposed model, which can be done using highly efficient BLAS routines, is $O(nq)$, with $q$ the size of the basis.

In terms of accuracy, both approaches exhibit acceptable levels for the problem at hand, leading to sample distributions that present similar mean and standard deviation as the golden solution used. Furthermore the worst case error for all parameter settings and timepoints ($E_{abs}$) is similar in all cases which implies that there are no large outliers in the approximated methods. In all cases the accuracy is quite reasonable, considering the simplicity of the approximation used, with very small relative errors shown in all cases.

The proposed flow can be efficiently used to extract and monitor statistical information of the performance of the system while subject to parameter variations. By performing a large number of parameter evaluations, the average and standard deviation values of a time domain simulation for multiple nodes can be analyzed. For example, Figure 3 shows the nominal (no variation) time domain evolution of the voltage of a node in `ibmpg2t`, along with the average and the $3\sigma$ limits of the information extracted from 1000 MC settings. It is clear that the impact of the variation, even on a relatively simple case,

TABLE IV
RESULTS OF TIME DOMAIN VARIABILITY ANALYSIS

**ibmpg1t** (8 regions, 48 parameters), 1000 settings

| | Basis (Mem.) | Set Up | Evaluation | $E_{abs}$ | $A_{abs}$ | $A_{rel}$ |
|---|---|---|---|---|---|---|
| TS | – (19MB) | 0.7 | 2108 | 0.158 | 0.002 | 0.44% |
| Prop. | 83 (38MB) | 81 | 144 | 0.194 | 0.003 | 3.33% |

**ibmpg2** (32 regions, 192 parameters), 1000 Parameter settings

| | Basis (Mem.) | Set Up | Evaluation | $E_{abs}$ | $A_{abs}$ | $A_{rel}$ |
|---|---|---|---|---|---|---|
| TS | – (415MB) | 632 | 18058 | 0.208 | 0.004 | 0.70% |
| Prop. | 112 (571MB) | 9922 | 996 | 0.208 | 0.005 | 3.50% |

**ibmpg3** (16 regions, 96 parameters), 1000 Parameter settings

| | Basis (Mem.) | Set Up | Evaluation | $E_{abs}$ | $A_{abs}$ | $A_{rel}$ |
|---|---|---|---|---|---|---|
| TS | – (1.33GB) | 16 | 229431 | 0.210 | 0.006 | 0.95% |
| Prop. | 107 (2.19GB) | 81453 | 8691 | 0.146 | 0.006 | 5.32% |

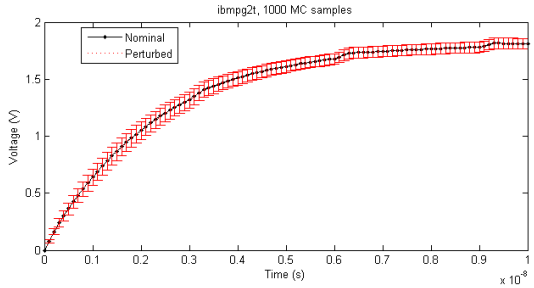


Fig. 3. TD evolution of the nominal voltage of one node in `ibmpg2t`, along with the average and $3\sigma$ limits obtained by the variability analysis.

can be quite relevant. This is a valuable information for designers, that can be exploited in the design cycle to minimize failures due to process and parameter variation.

## V. CONCLUSIONS

We presented an efficient methodology to analyze the effect of parameter variations on the dynamic behavior of power grids and large RC interconnect networks. The approach relies on two main assumptions: the compressibility of the node responses and the smoothness of the variational dependence. Considerable compressibility has been shown in both the time domain and the parameter space with space compression also trivially within reach. For moderate parameter variations often a linear or low order approximation is sufficiently accurate, but the approach is quite general and can be computed to any order. Model generation is very efficient, as it only requires one matrix factorization of the nominal system matrix which is then reused to compute the different terms of the approximation. Once this model is generated, it allows a very fast evaluation of the effect of different parameter settings, requiring only dense matrix vector products at each timepoint for each parameter setting. The performance of the method has been proved using a set of simple power grids. Improved results are expected in larger examples due to increased compressibility.

## ACKNOWLEDGMENTS

This work was partially supported by national funds from Portugal's FCT - Fundação para a Ciência e a Tecnologia, under projects PTDC/EEI-ELC/3002/2012 and UID/CEC/50021/2013.

## REFERENCES

[1] S. R. Nassif and J. N. Kozhaya, "Fast power grid simulation," in *Proc. ACM/IEEE Design Automation Conference (DAC)*, Las Vegas, Nevada, U.S.A., June 2000, pp. 156–161.

[2] J. N. Kozhaya, S. R. Nassif, and F. N. Najm, "A multigrid-like technique for power grid analysis," *IEEE Trans. on Computer-Aided Design of Integrated Circuits*, vol. 21, pp. 1148–1160, October 2002.

[3] Y. Zhong and M. D. F. Wong, "Fast algorithms for ir drop analysis in large power grid," in *Proc. IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2005, pp. 351–357.

[4] J. M. Silva, J. R. Phillips, and L. M. Silveira, "Efficient simulation of power grids," *IEEE Trans. on Computer-Aided Design of Integrated Circuits*, vol. 29, no. 10, pp. 1523–1532, October 2010.

[5] H. Qian, S. R. Nassif, and S. S. Sapatnekar, "Power grid analysis using random walks," *IEEE Trans. on Computer-Aided Design of Integrated Circuits*, vol. 24, no. 8, pp. 1204–1224, August 2005.

[6] J. R. Phillips and L. M. Silveira, "Poor Man's TBR: A simple model reduction scheme," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 1, pp. 43–55, Jan. 2005.

[7] J. Rommes and W. Schilders, "Efficient methods for large resistor networks," *IEEE Trans. Computer-Aided Design of Circuits and Systems*, vol. 29, no. 1, pp. 28 – 39, 2010.

[8] R. Ionutiu, J. Rommes, and W. Schilders, "SparseRC: Sparsity preserving model reduction for rc circuits with many terminals," *IEEE Trans. Computer-Aided Design of Circuits and Systems*, vol. 30, no. 12, pp. 1828 – 1841, 2011.

[9] P. Ghanta, S. Vrudhula, S. Bhardwaj, and R. Panda, "Stochastic variational analysis of large power grids considering intra-die correlations," in *Proc. Design Automation Conference (DAC)*, 2006, pp. 211–216.

[10] N. Mi, S.-D. Tan, Y. Cai, and X. Hong, "Fast variational analysis of on-chip power grids by stochastic extended krylov subspace method," *IEEE Trans. on Computer-Aided Design of Integrated Circuits*, vol. 27, no. 11, pp. 1996–2006, 2008.

[11] P. Li, "Statistical sampling-based parametric analysis of power grids," *IEEE Trans. on Computer-Aided Design of Integrated Circuits*, vol. 25, no. 12, pp. 2852–2867, 2006.

[12] B. Boghrati and S. Sapatnekar., "Incremental solution of power grids using random walks," in *Proc. Asian and South-Pacific Design Automation Conference (ASP-DAC)*, 2010, pp. 757–762.

[13] P. Sun, X. Li, and M. Ting., "Efficient incremental analysis of on-chip power grid via sparse approximation," in *Proc. ACM/IEEE Design Automation Conference (DAC)*, 2011, pp. 676–681.

[14] I. A. Ferzli and F. N. Najm, "Analysis and verification of power grids considering process-induced leakage-current variations," *IEEE Trans. on Computer-Aided Design of Integrated Circuits*, vol. 25, no. 1, pp. 126–143, 2006.

[15] P. Ghanta, S. Vrudhula, R. Panda, and J. Wang, "Stochastic power grid analysis considering process variations," in *Proc. Design, Automation and Test in Europe conference (DATE)*, Germany, 2005, pp. 964–969.

[16] S. Banerjee, K. Agarwal, and S. Nassif, "Design-aware lithography," in *Proc. ACM International Symposium on Physical Design*, 2012, pp. 3–8.

[17] S. Nassif, "Power grid analysis benchmarks," in *Proc. of the 2008 Asia and South Pacific Design Automation Conference*, 2008, pp. 376–381.

[18] J. F. Villena and L. M. Silveira, "Efficient analysis of variability impact on interconnect lines and resistor networks," in *DATE'2014 - Design, Automation and Test in Europe, Exhibition and Conference*, Dresden, Germany, March 2014.

[19] ——, "SPARE - a scalable algorithm for passive, structure preserving, parameter-aware model order reduction," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 29, no. 6, pp. 925–938, June 2010.

[20] C. Bischof and G. Quintana-Ort, "Computing rank-revealing qr factorization of dense matrices," *ACM Trans. On Mathematical Software*, vol. 24, no. 2, pp. 226–253, June 1998.