

# Efficient Analysis of Variability Impact on Interconnect Lines and Resistor Networks

Jorge Fernández Villena, and L. Miguel Silveira

INESC ID / Técnico - U. Lisboa

Rua Alves Redol 9, 1000-029 Lisbon, Portugal

email: {jorge.fernandez,lms}@inesc-id.pt

**Abstract**—Continued technology scaling coupled with limited lithographic capabilities is a leading cause of increased design variability. In the nanometer regime lithography tools have failed to keep pace with Moore’s Law and printed feature sizes are a small fraction of the wavelength of light used in current processes. Such sub-wavelength printing makes features highly susceptible to perturbations in the lithographic process conditions which leads to printed designs exhibiting increased variability. Such variability directly affects design behavior and performance in multiple ways. One of the areas of concern is power grid (PG) design, where lithographic errors may locally modify the wire widths. These variations, that may affect any and all wires in the grid, have a critical impact on the power distribution across the chip, introducing considerable current fluctuations which are a potential cause for electromigration effects. To analyze and account for the impact of these errors requires a complete extraction of the PG, which generates a large resistive network, potentially with several million elements, whose simulation is computationally challenging. This paper proposes a fast and accurate variability analysis of very large resistor networks, such as PG extracted netlists, that allows estimating the effects of multiple parameter settings in reasonable time. The proposed model can be easily combined with Litho/CMP simulators in order to boost much needed design-aware lithography.

**Keywords**—Litho-induced Variability Analysis, Electromigration, Power Grid Analysis

## I. INTRODUCTION

Producing layouts consisting of a set of polygons on multiple levels designed in adherence to strict foundry established rules used to be sufficient to guarantee manufacturability and reliable behavior. However, the nanometer regime has introduced new challenges, namely in lithography where tools have failed to keep pace with Moore’s Law and printed feature sizes are a small fraction of the wavelength of light used in lithographic processes. Such sub-wavelength printing makes features highly susceptible to any variations in the lithographic process conditions which in turn leads to printed designs exhibiting increased variability, directly affecting design behavior and performance. An area of current concern is power grid design, an important step in IC design. Today’s VLSI designs feature huge power distribution (PG) networks, carrying large currents, spanning across the entire chip area through several layers are used to provide circuit bias.

Variations in wire width resulting from lithographic errors impact grid behavior in multiple ways, potentially causing voltage fluctuations and considerable current variations. Such

variations in current can lead to electromigration problems and considerable effort is devoted nowadays to analyze the network in such scenarios. This entails extracting a model of the power grid accounting for litho-induced errors and simulating to check currents on all wires. Given the number of wires to consider and even discounting systematic errors, random fluctuations alone may lead to an intractable number of settings to analyze. Furthermore an analysis for each possible variation setting is unfeasible as extraction of interconnect and power grid (PG) networks leads to large resistor networks, potentially with several million elements, whose simulation is computationally challenging. Different solutions have been proposed for the analysis of generic PG including acceleration techniques such as multi-grid approaches [1], [2], iterative methods [3], Random Walk [4], Hierarchical representations [5], and Model Order Reduction LMS[6], [7], [8], with mixed results. Some work has also 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]. All such techniques lead to powerful but costly variational models that can be evaluated in order to estimate variability effects. In this paper, we propose an alternative more flexible approach for efficiently estimating the effects of variability in static PG and interconnect analysis. The method is based on a linear approximation of the voltage at the grid nodes. The contributions of the proposed work are:

- Two different techniques for generating the linear approximation of the parameter dependent voltages, which is computed for all nodes, and can be efficiently generated for very large number of parameters, enabling the estimation of the effect of intra-die variations.
- A variability analysis flow, divided into a Set Up phase, which generates the model, and a Solve phase which uses the model to compute the effect of the variability.
- The Set Up phase is performed once, and only requires a single sparse matrix factorization and multiple sparse matrix vector products and solves.
- The Solve phase involves a dense matrix vector product for each parameter evaluation, which allows to efficiently analyze a large number of parameter settings.
- The proposed flow can be trivially extended to handle input variations, and be combined with any other acceleration technique, such as Model Order Reduction (MOR),  $\mathcal{H}$ -matrix representations, multi-grid methods, etc.

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 the impact of those variations 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.

The manuscript is structured as follows. Section II briefly reviews the PG problem, and presents the basics of the variability model and analysis to be proposed. Section III introduces two variants for the proposed methodology, and discusses computational implementations and practical considerations. Section IV discusses additional efficiency-improving techniques to extend the scope of the approach. Finally results for industrial PG benchmarks are presented in Section V, and conclusions drawn in Section VI.

## II. BACKGROUND AND PROBLEM DEFINITION

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 drop computation or electromigration. While a dynamic formulation is equally trivial to formulate, in this paper we are interested in the DC behavior of the grid for which a set of equations can be obtained:

$$\mathbf{G}\mathbf{v} = \mathbf{i} \quad (1)$$

where  $\mathbf{G} \in \mathbb{R}^{n \times n}$  is the  $\mathbf{v} \in \mathbb{R}^n$  is the vector of grid node voltages, and  $\mathbf{i} \in \mathbb{R}^n$  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). The trivial solution to (1) is  $\mathbf{v} = \mathbf{G}^{-1}\mathbf{i}$ . However, although  $\mathbf{G}$  is very sparse, it is also very large. Fortunately, efficient direct and iterative methods can be used to solve (1), albeit at some memory and computational cost, given the large matrix sizes [1], [2], [3], [5]. Once the grid node voltages are known, computing resistor currents is a trivial task. If solution of (1) is within reach, then simplified dynamic solution and other types of analysis can be similarly conducted. As long as  $\mathbf{G}$  remains the same, further solutions of (1) even with different right-hand side vectors can be obtained efficiently (factoring the matrix and reusing the factors).

For static analysis, we focus on resistive networks where 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  $T$ .

$$R = \rho L S^{-1} = \rho L (WT)^{-1} \quad (2)$$

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

the variation of the resistance value as depending on the variation of the parameters in the set  $\mathbf{p} = [\Delta\rho, \Delta L, \Delta W, \Delta T]$ ,

$$R(\Delta\rho, \Delta L, \Delta W, \Delta T) = \frac{(\rho + \Delta\rho)(L + \Delta L)}{(W + \Delta W)(T + \Delta T)} \quad (3)$$

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

$$R(\mathbf{p}) \approx R_0 \left( 1 + \Delta\rho + \Delta L + \sum_{k=1}^O (-\Delta W)^k + \sum_{k=1}^O (-\Delta T)^k \right) \quad (4)$$

where  $R_0$  is the nominal value when there is zero 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 area or region 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 is represented as an arbitrary order Taylor series such as in (4),  $\mathbf{R}(\mathbf{p})$ , the parameter dependent admittance matrix  $\mathbf{G}(\mathbf{p})$  can be efficiently formed using the incidence of the resistors in the circuit. Alternatively, a representation such as (4) might be available for the admittance directly, from which  $\mathbf{G}(\mathbf{p})$  can be directly computed through stamping.

From the above, it is easy to generate a set of sensitivities of the admittance matrix  $\mathbf{G}(\mathbf{p})$  to an arbitrary order, that can include cross-terms. For instance, the first order Taylor series can be written as a function of the nominal,  $\mathbf{G}_0$ , and the sensitivities,  $\mathbf{G}_k$ , to a set of  $P$  parameters,  $p_k$ , as

$$\mathbf{G}(\mathbf{p}) = \mathbf{G}_0 + \sum_{k=1}^P \mathbf{G}_k p_k. \quad (5)$$

Higher order approximations can be trivially generated. In order to generate the voltages due to a certain variation, the use of such formulations requires two steps: a) evaluate the matrix, i.e. obtain  $\mathbf{G}(\mathbf{p})$  for the given parameter setting  $\bar{\mathbf{p}}$ , and b) solve the system with the evaluated matrix. Herein lies the basic difficulty with estimating the effects of parameter variability: as the matrix changes with each parameter setting, to generate the voltage solution  $\mathbf{v}(\bar{\mathbf{p}}) = \mathbf{G}(\bar{\mathbf{p}})^{-1}\mathbf{i}$  for each parameter setting  $\bar{\mathbf{p}}$ , implies a re-factorization of the matrix or a new solution process if an iterative method is used. This is very costly and in fact to compute the effects at multiple parameter settings is quickly overwhelming.

Most of the 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 a comprehensive overview of existing approaches the reader is referred to [9], [10], [11], [12], [13], [14], [15]. For large parameter settings, for instance to estimate the distribution of the maximum resistor current (relevant for electromigration analysis), the maximum voltage drop (relevant for grid integrity analysis) or any other required metric, more efficient methods are required.

### III. PROPOSED LINEAR APPROXIMATION

Instead of using the traditional Taylor series approaches, we propose a simple yet efficient approximation of the voltage vector. Node voltages are expected to have a close-to-linear response to parameter changes, since their value is proportional to the resistance path. Therefore we propose to reformulate the system by expanding the voltage in a Taylor series, presumably sufficiently accurate in the range of variation of the parameter set  $\mathbf{p}$ , such as, (for a first order approximation)

$$\mathbf{v}(\mathbf{p}) = \mathbf{v}_0 + \sum_k^P \mathbf{v}_k p_k \quad (6)$$

Note that this representation can be extended to any desired or required order. However, doing so would rapidly increase the cost of computing the approximation, specially when multiple parameters are involved. Next, we present two different approaches to efficiently compute the coefficients  $\mathbf{v}_k$ .

#### A. SPARE approximation

The first approach (SPARE) is based on [18] and akin to adjoint methods. It uses the Taylor Series representation of the matrix  $\mathbf{G}(\mathbf{p})$  as in (5), and the expansion of the voltage in (6). Replacing both  $\mathbf{G}(\mathbf{p})$  and  $\mathbf{v}(\mathbf{p})$  by their series approximation in (1) and following the procedure in [18], we can match coefficients of the same powers of  $p_k$ , and obtain a representation for each of the  $\mathbf{v}_k$  terms. As an illustration, let us suppose two parameters,  $p_1$  and  $p_2$ , with  $2^{nd}$  order expansion, and one cross term. Adopting a double subindex notation, where the first subindex is the order w.r.t. the  $1^{st}$  parameter, and the second the order w.r.t. the  $2^{nd}$  parameters:

$$\begin{aligned} \mathbf{v}_{00} &= \mathbf{G}_{00}^{-1} \mathbf{i} \\ \mathbf{v}_{10} &= -(\mathbf{G}_{00})^{-1} \mathbf{G}_{10} \mathbf{v}_{00} \\ \mathbf{v}_{01} &= -(\mathbf{G}_{00})^{-1} \mathbf{G}_{01} \mathbf{v}_{00} \\ \mathbf{v}_{20} &= -(\mathbf{G}_{00})^{-1} (\mathbf{G}_{10} \mathbf{v}_{10} + \mathbf{G}_{20} \mathbf{v}_{00}) \\ \mathbf{v}_{02} &= -(\mathbf{G}_{00})^{-1} (\mathbf{G}_{01} \mathbf{v}_{01} + \mathbf{G}_{02} \mathbf{v}_{00}) \\ \mathbf{v}_{11} &= -(\mathbf{G}_{00})^{-1} (\mathbf{G}_{11} \mathbf{v}_{00} + \mathbf{G}_{10} \mathbf{v}_{01} + \mathbf{G}_{01} \mathbf{v}_{10}) \end{aligned} \quad (7)$$

Note that, although for most cases only first order sensitivities for the admittance matrix are provided, an arbitrary order can be generated, with different order approximations with respect to different parameters depending on the accuracy required. However, complexity and cost increase slowly with the order.

For the proposed linear approximation of node voltages chosen, which, as we shall see in the examples, is sufficient for the cases tested, we use only first order, thus obtaining the following approximation for the node voltages (using the original single subscript notation)

$$\mathbf{v}(\mathbf{p}) = \mathbf{v}_0 - \sum_k^P \mathbf{G}_0^{-1} \mathbf{G}_k \mathbf{v}_0 p_k \quad (8)$$

where  $\mathbf{v}_0 = \mathbf{G}_0^{-1} \mathbf{i}$  is the nominal voltage on the nodes. All terms can be computed very efficiently, as the method only requires to store the factorization of the nominal matrix  $\mathbf{G}_0$ , and reuse these factors on the result of some sparse matrix vector products.

#### B. Output Linear Approximation

An alternative approach (OLA) to generate the coefficients is based on matching the perturbed response of the voltage vector at the maximum variation for each parameter. This

approach is restricted to a first order approximation of the voltage vector, which is sufficient to meet most accuracy requirements. For a single parameter  $p_1$ , with maximum value  $\hat{p}_1$ , and a single term  $\mathbf{v}_1$ , matching the response at the maximum variation leads to

$$\begin{aligned} \mathbf{v}_0 + \mathbf{v}_1 \hat{p}_1 &= \mathbf{v}(\hat{p}_1) \\ \mathbf{G}_0^{-1} \mathbf{i} + \mathbf{v}_1 \hat{p}_1 &= (\mathbf{G}(\hat{p}_1))^{-1} \mathbf{i} \end{aligned} \quad (9)$$

After some algebra, and substituting the perturbed admittance by its first order approximation, we get

$$\begin{aligned} \mathbf{v}_1 \hat{p}_1 &= ((\mathbf{G}_0 + \mathbf{G}_1 \hat{p}_1)^{-1} - \mathbf{G}_0^{-1}) \mathbf{i} \\ &= ((\mathbf{I} + \mathbf{G}_0^{-1} \mathbf{G}_1 \hat{p}_1)^{-1} - \mathbf{I}) \mathbf{G}_0^{-1} \mathbf{i}. \end{aligned} \quad (10)$$

As  $\mathbf{v}_0 = \mathbf{G}_0^{-1} \mathbf{i}$ , defining  $\Delta = \mathbf{G}_0^{-1} \mathbf{G}_1 \hat{p}_1$ , we get

$$\mathbf{v}_1 = \frac{1}{\hat{p}_1} (-\Delta \mathbf{v}_0 + \Delta^2 \mathbf{v}_0 - \Delta^3 \mathbf{v}_0 + \dots) \quad (11)$$

where again  $\hat{p}_1$  is the maximum variation of parameter  $p_1$ . We can apply the same reasoning on all the parameters independently, and truncate the series of each one to the required order (that can be different for each term), to generate the terms  $\mathbf{v}_k$  in (6). It is important to note that the terms  $\Delta^j \mathbf{v}_0$  can be computed recursively  $\Delta^j = \hat{p}_1 \mathbf{G}_0^{-1} \mathbf{G}_1 \Delta^{j-1} \mathbf{v}_0$ . Such computation only requires to generate the factorization of the nominal matrix  $\mathbf{G}_0$ , and use the factors to solve different rhs to generate all the  $\mathbf{v}_k$  terms. For the truncation of the approximation in (11), a simple approach is to truncate when the maximum absolute value of the vector  $\Delta^j \mathbf{v}_0$  is smaller than a certain threshold (usually a small fraction of the maximum value of  $\mathbf{v}_0$ ).

#### C. Computational Issues

The proposed approaches are highly efficient, since they compute the voltage on the nodes for each parameter setting as the result of the sum of multiple vectors. The proposed flow for the variability analysis of large resistor networks is detailed in Algorithm 1, and is composed of two main parts.

The SET UP stage, akin to model computation, is done only once, and computes the terms  $\mathbf{v}_k$  required for the approximation. We assume that the sensitivities of the admittance,  $\mathbf{G}_k$  are given, or in cases where they are not available we have access to an extractor and can generate them by simple differentiation, i.e.  $\mathbf{G}_j = (\mathbf{G}(\hat{p}_j) - \mathbf{G}_0) / (\hat{p}_j - p_0)$ . The SET UP requires factorizing the nominal term,  $\mathbf{G}_0$  (e.g. using Cholesky factorization), and storing the factors. These factors are used to solve for the nominal voltage  $\mathbf{v}_0$ , and then for the sensitivities of the linear approximation  $\mathbf{v}_k$ , using either the SPARE (steps 5-7) or the OLA (steps 8-13) approach. These sensitivities are stored columnwise in a matrix  $\mathbf{X}$ .

The SOLVE stage, akin to model evaluation, computes the results of the variational analysis. The values of each parameter setting are stored in a column vector, and left-multiplied by matrix  $\mathbf{X}$  to generate the perturbation in the voltages, that is added to the nominal value  $\mathbf{v}_0$  (steps 16-18).

It is important to note that the complete flow requires to factorize the nominal  $\mathbf{G}_0$  matrix only once, and then reuse the factors to perform a small number of solves (2 or 3 times the number of parameters for the OLA approach). For a linear approximation the total number of solves is directly proportional to the number of parameters. The actual

---

**Algorithm 1** Variability Analysis for Resistor Networks
 

---

```

1: SET UP STAGE
2:   given  $P + 1$  sensitivities  $\mathbf{G}_k \in \mathbb{R}^{n \times n}$  and input  $\mathbf{i} \in \mathbb{R}^n$ 
3:   given maximum variation vector  $\hat{\mathbf{p}} \in \mathbb{R}^P$ , define  $\mathbf{X} = \mathbf{zeros}(n, P)$ 
4:    $LL^T = \mathbf{chol}(\mathbf{G}_0)$ ;  $\mathbf{v}_0 = L^{-T}L^{-1}\mathbf{i}$ ;
5:   IF (SPARE)
6:     FOR  $j=1 : P$ 
7:        $\mathbf{X}(:, j) = -L^{-T}L^{-1}\mathbf{G}_j\mathbf{v}_0$ ;
8:   IF (OLA)
9:     FOR  $j=1 : P$ 
10:       $\mathbf{x}_c = -\hat{\mathbf{p}}(j)L^{-T}L^{-1}\mathbf{G}_j\mathbf{v}_0$ ;  $\mathbf{X}(:, j) = \mathbf{x}_c/\hat{\mathbf{p}}(j)$ ;
11:      WHILE  $\max|\mathbf{x}_c| > tol * \max|\mathbf{v}_0|$ 
12:         $\mathbf{x}_c = -\hat{\mathbf{p}}(j)L^{-T}L^{-1}\mathbf{G}_j\mathbf{x}_c$ ;
13:         $\mathbf{X}(:, j) = \mathbf{X}(:, j) + \mathbf{x}_c/\hat{\mathbf{p}}(j)$ ;
14:   STORE  $\mathbf{v}_0$  and  $\mathbf{X}$ 
15: SOLVE STAGE
16:   given  $M$  parameter settings  $\bar{\mathbf{p}}_j \in \mathbb{R}^P$ 
17:   FOR  $j=1 : M$ 
18:      $\mathbf{V}(:, j) = \mathbf{v}_0 + \mathbf{X}\bar{\mathbf{p}}_j$ ;
19:   RETURN the solutions stored columnwise in  $\mathbf{V}$ 

```

---

variational analysis is very fast, as it only requires a dense matrix vector product for each parameter setting.

We should point out that the computations above can be efficiently parallelizable for both shared and distributed memory architectures as well as GPU enhanced machines. The generation of the different linear terms  $\mathbf{v}_k$  in the SET UP phase is perfectly independent once the factorization of the nominal matrix is done. Multiple cores can therefore be used in any type of architecture with speedup limited only by the number of parameters of the problem. For the SOLVE phase, it is clear that the different parameter settings can be divided between an arbitrary number of machines and computed concurrently. Additionally, the evaluation of the proposed linear approach which only requires a dense matrix vector product, can be accelerated with the use of a GPU since the matrix is the same for different parameter settings. Therefore multiple parameter settings can be loaded to the GPU and computed at a time.

#### IV. ADDITIONAL CONSIDERATIONS

This section presents some considerations, that although beyond the scope of the manuscript, can be easily taken into account and incorporated with the proposed approaches.

##### A. Input Variability

An interesting case that has not been taken into account is how to address changes in the input vector. If the variation of the input does not depend on the parameters, this is straightforward, as the new input vector can be represented as  $\mathbf{i} + \Delta\mathbf{i}$ , and the term  $\Delta\mathbf{i}$  induces a perturbation on the linear terms in (6). On the other hand, if the input depends on the parameter set  $\mathbf{p}$

$$\mathbf{i}(\mathbf{p}) = \mathbf{i}_0 + \sum_k^P \mathbf{i}_k p_k \quad (12)$$

such representation must be included when computing the terms for (6). For the SPARE approach, the parameter dependent input terms must be included when matching the coefficients of the same power, in order to generate the expressions for the terms of the voltage approximation. For the OLA approach, the perturbation can be taken into account

when computing the linear terms, and thus the expression in (9) becomes

$$\mathbf{G}_0^{-1}\mathbf{i}(\hat{\mathbf{p}}_1) + \mathbf{v}_1 \hat{\mathbf{p}}_1 = (\mathbf{G}_0(\hat{\mathbf{p}}_1))^{-1} \mathbf{i}(\hat{\mathbf{p}}_1) \quad (13)$$

where we can approximate  $\mathbf{i}(\hat{\mathbf{p}}_1) = \mathbf{i}_0 + \mathbf{i}_1 \hat{\mathbf{p}}_1$ . The rest of the reasoning remains the same, including the model evaluation.

##### B. System Reduction and Solve Acceleration

An interesting feature of the proposed flow is that it can be easily combined with existing methods for accelerating the solution of the system, and thus speed up the SET UP phase in which the proposed model is generated.

The proposed method shifts the parameter dependence to the output, and thus allows to apply standard non-parameterized Model Order Reduction (MOR) approaches to the system sensitivities  $\mathbf{G}_k$  to reduce the dimension of the problem and further accelerate solution. For example, following [7], [8] let us assume we have a transformation  $\mathbf{T}$  that generates a suitable reduction of the nominal matrix  $\mathbf{G}_0$  while maintaining the relevant (defined as external) subset of nodes in  $\mathbf{v}$ , so that

$$\mathbf{T}^T \mathbf{G}_0 \mathbf{T} \tilde{\mathbf{v}}_0 = \mathbf{T}^T \mathbf{i} \quad (14)$$

where  $\tilde{\mathbf{v}}_0 \in \mathbb{R}^{q \times 1}$  with  $q \ll n$  is the reduced set of voltages that approximate the original ones at the external nodes. We can use this transformation and the reduced matrix  $\tilde{\mathbf{G}}_0 = \mathbf{T}^T \mathbf{G}_0 \mathbf{T}$  to compute the reduced terms  $\tilde{\mathbf{v}}_k \in \mathbb{R}^{q \times 1}$  that approximate the parameterized response of the voltages at the  $q$  external nodes,

$$\mathbf{v}(\mathbf{p})|_q \approx \tilde{\mathbf{v}}_0 + \sum_k^P \tilde{\mathbf{v}}_k p_k \quad (15)$$

where, for the case of the OLA approach (similar reasoning can be followed for the SPARE case), we can compute the terms  $\tilde{\mathbf{v}}_k$  as

$$\tilde{\mathbf{v}}_j = \frac{1}{\hat{p}_1} (-\tilde{\Delta} + \tilde{\Delta}^2 - \tilde{\Delta}^3 + \dots) \tilde{\mathbf{v}}_0 \quad (16)$$

where  $\tilde{\Delta} = \hat{p}_1 \tilde{\mathbf{G}}_0^{-1} \mathbf{T}^T \mathbf{G}_j \mathbf{T} \tilde{\Delta}^{j-1}$ . This means that we can apply the same transformation to the sensitivities to generate a set of reduced (transformed) matrices  $\tilde{\mathbf{G}}_j = \mathbf{T}^T \mathbf{G}_j \mathbf{T}$  that can be used to compute the terms of the linear approximation.

Note that this is possible as we are working on the static case, and the reduction [7], [8] preserves the external nodes, so **the approximation of (9) by the reduced matrices is exact**. This has two advantages. On one hand, the computation of the linear terms in the SET UP phase is faster, since we are working with smaller matrices. On the other hand, we are only storing the values for a reduced set of entries (the predefined external nodes), and thus we are reducing the storage required for the  $v_k$  terms. We also speed up the computations in the SOLVE stage, as we will use a matrix with a reduced number of rows in the dense matrix vector product.

A different improvement that can be applied to the proposed method is to use other compression and acceleration techniques to solve the nominal system with the matrix  $\mathbf{G}_0$ , and thus speed up the SET UP phase. Any compression and fast solve, such as [1], [2], [5], can be easily combined with the proposed flow. Traditional parameterized system representation, e.g. Taylor series, in which the system matrix depends on a large set of parameters, does not allow efficient use of

TABLE I. BENCHMARKS CHARACTERISTICS

	ibmpg1	ibmpg2	ibmpg6	ibmpgnew2
#n	16,327	126,905	834,252	1,460,083
#r	29,750	207,995	1,645,626	2,351,136
#R	8	32	16	68
#P	32	128	64	272
$E_{\text{abs}}$	0.7157	0.6665	0.7944	0.7283
$E_{\text{rel}}$	89.66%	78.49%	67.00%	67.21%
$A_{\text{abs}}$	0.0543	0.0641	0.0748	0.0648
$A_{\text{rel}}$	8.14%	7.33%	8.41%	8.43%

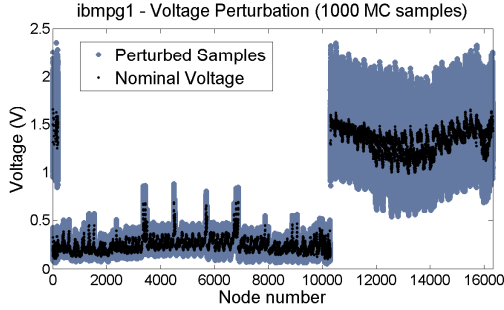


Fig. 1. (**ibmpg1** example): black points indicate nominal voltage of each node in the grid, whereas gray points indicate perturbed voltage values at the same nodes obtained for 1000 MC parameter samples.

most of these techniques, as the matrix values change for each parameter setting, and thus the factorizations or matrix compression must be recomputed. In our approach, however, full advantage is taken from these techniques.

## V. EXPERIMENTS AND RESULTS

We compare the performance of the proposed methodology for a set of realistic examples. We use the original perturbed system, i.e. the stamp of the perturbed resistors under the effect of some parameters, using a 3<sup>rd</sup> order approximation in (4), as the golden solution against which the accuracy of the methods is presented. We compare the standard Taylor series approximation (5), with the proposed SPARE and OLA approaches, for which we used the same sensitivities as in (5), that were obtained by direct differentiation at the maximum perturbation for each parameter. A tolerance of  $10^{-3}$  was assumed. All routines were coded in MATLAB, which means times are merely indicative, since it is a non-compiled language.

Table I shows the characteristics of the realistic examples, taken from the IBM Power Grid Benchmarks for DC Analysis [17], where  $n$  stands for the effective number of nodes and  $r$  for the number of resistors, after processing the data and removing shorts. Therefore, the number of nodes equals the size of the matrix. We have further divided the PG into  $R$  regions, following the nets to which the resistors belong, and further dividing these nets into 4 local regions. This last division tries to emulate locality of the perturbations. For each region we used 4 independent parameters (resistivity, length, width and thickness).  $P$  stands for the total number of independent parameters. 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. This means that every resistor in the net can vary up to  $\pm 94.4\%$  of its nominal value, which is in

fact a very large variation. To quantify the results, we show the maximum absolute error ( $E_{\text{abs}}$ ), the maximum relative error ( $E_{\text{rel}}$ ), the average absolute error ( $A_{\text{abs}}$ ), and the average relative error ( $A_{\text{rel}}$ ), obtained from the simulation of a large number of parameter settings. The formulas for these values are, for each parameter setting  $\bar{\mathbf{p}}$

$$\begin{aligned} E_{\text{abs}} &= \max |\mathbf{v}(\bar{\mathbf{p}}) - \mathbf{v}_{\mathbf{r}}| \\ E_{\text{rel}} &= \max (|\mathbf{v}(\bar{\mathbf{p}}) - \mathbf{v}_{\mathbf{r}}| ./ |\mathbf{v}_{\mathbf{r}}|) \\ A_{\text{abs}} &= \text{mean} (|\mathbf{v}(\bar{\mathbf{p}}) - \mathbf{v}_{\mathbf{r}}|) \\ A_{\text{rel}} &= \text{mean} (|\mathbf{v}(\bar{\mathbf{p}}) - \mathbf{v}_{\mathbf{r}}| ./ |\mathbf{v}_{\mathbf{r}}|) \end{aligned} \quad (17)$$

where  $\mathbf{v}(\bar{\mathbf{p}})$  is the system solution with the perturbed response, and  $\mathbf{v}_{\mathbf{r}}$  is the reference solution. These errors are obtained from the MC analysis of a number of parameter settings generated using a normal distribution for the  $3\sigma$  variation, where  $\Delta\rho = \pm 0.1$ ,  $\Delta L = \pm 0.01$ ,  $\Delta W = \pm 0.3$ , and  $\Delta T = \pm 0.3$  in (4). In Table I, these errors are obtained using the nominal response (zero variation) as reference  $\mathbf{v}_{\mathbf{r}}$ , and our golden solution for the generation of  $\mathbf{v}(\bar{\mathbf{p}})$ . Thus these results quantify the actual variation in the voltage nodes that we are trying to model, which, as presented in Table I, can be as large as 89%. To illustrate the effect the parameters have on the voltage on the nodes, Figure 1 shows the nominal values for the voltage at the nodes of example **ibmpg1**, as well as the voltages computed for 1000 Monte Carlo samples of the parameters.

Table II presents the results for each benchmark. All the tables have the same structure, presenting the errors according to the criteria in (17), using in this case the golden solution as reference  $\mathbf{v}_{\mathbf{r}}$ . The table also shows the times required for the Taylor Series approximation and the proposed methodology for computing the solution of the total number of parameter settings, divided between the set up time (required for generating the different approximation terms, and thus computed once independently of the number of settings) and the solve time, which depends on the number of settings.

It is clear that the proposed approaches are much faster than the Taylor Series methodology (which takes the same amount of time for both the admittance and resistance Taylor Series), since no matrix factorization or solve is required during the SOLVE stage. The speed up per iteration (without taking into account the Set Up time) ranges from  $23\times$  for the smaller examples (**ibmpg1** and **ibmpg2**), to  $85\times$  for the **ibmpg6** and  $74\times$  for the **ibmpgnew2**. A relevant feature is that this speed up will increase with the complexity and time required to solve the system, since the cost of the solve operation is  $O(n^\beta)$ , with  $\beta > 1$  and  $n$  the number of nodes, whereas the cost of evaluating the proposed model is the cost of a dense matrix vector product,  $O(n(P+1))$ , with  $P$  the number of parameters, which can be done using highly efficient BLAS routines. In terms of accuracy, both proposed approaches outperform the admittance Taylor Series approximation (using exactly the same sensitivities). The OLA approach performs slightly better than the SPARE approach in terms of accuracy, at the cost of a more expensive Set Up phase. This is due to the use of a higher order approximation in the generation of the linear  $\mathbf{v}_k$  terms, and matches the maximum perturbation. The maximum errors remain under 20% for the proposed SPARE approach, and under 12% for the proposed OLA approach for all cases, whereas the Taylor Series error remains over 24%, and the deviation from the nominal value of the voltages can reach up to 85%. The average errors remain under 2% for the Taylor

TABLE II. RESULTS OF MC ANALYSIS

**ibmpg1** (8 regions, 32 parameters), 1000 settings

	Set Up	Solve	$E_{abs}$	$E_{rel}$	$A_{abs}$	$A_{rel}$
TS	0.10	44.24	0.2950	36.44%	0.0104	1.72%
SPARE	0.61	1.47	0.1654	18.21%	0.0102	1.55%
OLA	1.10	1.94	0.1115	11.15%	0.0020	0.27%

**ibmpg2** (32 regions, 128 parameters), 1000 settings

	Set Up	Solve	$E_{abs}$	$E_{rel}$	$A_{abs}$	$A_{rel}$
TS	3.99	888.16	0.2186	29.74%	0.0122	1.58%
SPARE	42.64	36.43	0.1453	14.84%	0.0123	1.44%
OLA	73.77	37.29	0.1089	10.81%	0.0036	0.49%

**ibmpg6** (16 regions, 64 parameters), 1000 settings

	Set Up	Solve	$E_{abs}$	$E_{rel}$	$A_{abs}$	$A_{rel}$
TS	14.69	8,743.10	0.2521	24.77%	0.0135	1.68%
SPARE	197.56	102.23	0.1502	12.55%	0.0140	1.58%
OLA	332.82	101.96	0.1237	7.10%	0.0012	0.16%

**ibmpgnew2** (68 regions, 272 parameters), 500 settings

	Set Up	Solve	$E_{abs}$	$E_{rel}$	$A_{abs}$	$A_{rel}$
TS	100.77	35,180.80	0.2688	25.46%	0.0129	1.76%
SPARE	2,609.55	473.57	0.1755	14.93%	0.0124	1.60%
OLA	4,900.85	474.68	0.1419	11.16%	0.0037	0.32%

Series and SPARE approaches, with the best accuracy obtained by the OLA approach, whose average error remains under 1%. Next we performed a Monte-Carlo simulation using 1000 random parameter settings on **ibmpg2** example, split into 32 regions, with 4 parameters per region (128 parameters total), and compared maximum relative errors versus the nominal voltage values. The results, show a significant reduction in both mean value (almost an order of magnitude) as well as standard deviation when using either the SPARE or OLA approaches compared with using a Taylor series.

## VI. CONCLUSIONS

We presented an efficient methodology to analyze the effect of parameter variations on static systems, such as large resistive networks arising from the simulation of PGs or DC analysis of general interconnects. The approach relies on the approximation of the node voltages w.r.t. large sets of parameters. 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 a dense matrix vector product for each setting. The performance of the method has been proved using a linear approximation in large realistic examples, with up to  $1.4M$  nodes and  $2.3M$  resistors, depending on 272 different parameters.

## VII. ACKNOWLEDGMENTS

This work was partially supported by national funds through FCT, Fundação para a Ciência e a Tecnologia, under projects PTDC/EEI-ELC/3002/2012 and PEst-OE/EEI/LA0021/2013.

## 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] 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.
- [5] 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.
- [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, "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.