Incremental Partitioning-Based Vectorless Power Grid Verification *

Dionysios Kouroussis
Department of ECE
University of Toronto
Toronto, Ontario, Canada
diony@eecg.utoronto.ca

Imad A. Ferzli
Department of ECE
University of Toronto
Toronto, Ontario, Canada
ferzli@eecg.utoronto.ca

Farid N. Najm
Department of ECE
University of Toronto
Toronto, Ontario, Canada
f.najm@utoronto.ca

ABSTRACT

To ensure reliable performance of a chip, design verification of the power grid is of critical importance. This paper builds on previous work that models the working behavior of the circuit in terms of abstracted current constraints and solves for worst-case voltage drop on the grid as a linear program. The main motivation is to allow the efficient verification of local power grid sections or blocks, enabling incremental design analysis of the grid. This approach substantially improves the computational time by reducing the problem size and the constraint set and replacing them by black box macromodels. This increase the capacity of the solver to handle industrial sized grids.

1. INTRODUCTION

The performance and reliability of integrated circuits in modern deep submicron (DSM) technologies is becoming increasingly sensitive to supply voltage variations. It is not uncommon to experience 15% increase in circuit delay due to supply voltage variations along a critical path or to lose 10% in yield due to supply voltage drops overall [1]. We refer to the on-chip power supply network as the *power grid*, or simply the *grid*. There are many sources of voltage fluctuation within on-chip power grids, such as IR-drop, Ldi/dt drop, and resonance between the grid and the package. Often, especially for simulation of the power grid in the chip's core, at frequencies below 1 GHz or so, inductance is neglected and one is focused on the IR-drop given an RC structure of the grid. With denser chips and higher chip currents and operating frequencies, power grid IR and Ldi/dt drops increase. As a result, ensuring the integrity of the power grid through robust design and efficient verification is a crucial matter in modern integrated circuit design methodologies, allowing for considerations to the particulars of microprocessor or ASIC flows [2, 3].

Some design groups start with over-designed grids with the aim of removing the power grid integrity factor in the overall design robustness. The extent of over-design, however, cannot be determined *a priori*, so that this approach comes at the cost of reduced resources for signal routing; furthermore, even grid over-design does not preclude voltage-induced circuit failures. Clearly, not everyone follows the approach of grid over-design and other groups limit the initial resources available for power routing in order to leave enough resources for signal routing, thus effectively running the risk of grid under-designing. This second approach renders grid verification necessary, a process carried out typically by *simulation*, as the design progresses. The difficulty with grid verification by simulation is two-fold: 1) the very large size of the power grids, with node counts in the tens of millions, makes it hard to simulate them for any realistic vector trace of chip workload, and 2) it is not certain what currents should be used to load the grid in these simulations, because at the time the grid is being designed, the circuit that would be loading the grid may not have been fully specified. A standard practice is to simulate the power grid loaded with current sources at their maximum values. While possibly successful in identifying problem spots on the grid and trends in voltage drop distributions, this approach leads to overly pessimistic results since different circuit gates or blocks are not expected to draw maximum currents simultaneously [4]. In consequence, power grids represent an important element of *uncertainty* in the design performance which needs to be checked [2].

The need is clear, therefore, for a *vectorless* power grid verification methodology that is to grid verification what static timing analysis is to timing verification. This paper builds on the work in [5] and develops such a methodology with the following key points: 1) it is useful pre-placement yet captures realistic circuit behavior and block interdependencies, 2) it is easy to integrate in existing flows by requiring reasonably simple inputs from designers, and 3) it enables the designer to focus on only one section of the grid at a time, thus allowing the repeated and iterative modification/verification of a known problematic grid section or block within a very large on-chip power grid, without the need to verify the entire power grid. We refer to this process as *incremental grid verification*. In order to meet this third point above, the grid will be partitioned to smaller sub-grids, which, from the standpoint of runtimes, opens the door to the parallelization of the computational tasks to achieve large speedup on clusters of machines.

2. PROPOSED APPROACH

A framework for dealing with the problem of grid verification with only partial information on the power grid current loads was laid out in [5]. It capitalizes on the fact that it is easy to specify limits on currents in the form of *current constraints* early on in the design process, and the power grid verification problem was reduced to one of maximizing voltage drops on any/every node of the grid given these current constraints. If the worst-case voltage met all voltage drop requirements, the power grid would be deemed *robust*.

Current constraints provide a very flexible framework: they can be in the form of bounds on individual current sources (*local constraints*), and on the *joint* and *simultaneous* operation of several current sources (*global constraints*). Local current constraints can be derived along the lines of [6], and global constraints typically

*This research was supported in part by Micronet, with funding from ATI and from Altera, and by the SRC under contract 2003-TJ-1070.

.

follow from power considerations. An example of current constraints may be: each current source has 0.1 mA of maximum current (local constraint) *and* the group of current sources constituting an ALU draws no more than 50 mW *collectively* at any time (global constraint), *and* the whole chip consumes no more than 5 W of power (global constraint). Global constraints are key, allowing us to capture true circuit operation in a realistic manner, much more so than verifying the grid when all current loads are set to their maximum. Contrast a vector-based power grid verification approach with our vectorless approach: the need to simulate the power grid for all current loads (vector-based) is abstracted by the constrained current space (vectorless). The obtained worst-case voltages do not arise from overly pessimistic currents (vector-based) but from ones which satisfy local and global current constraints and can therefore reflect more realistic circuit operation (vectorless).

The power grid verification in [5] is done by means of a linear program (LP), and current constraints are presented as upper bounds. The approach in [5], however, was limited in the size of the grid that could be analyzed mostly due to the (LP) solver that was used, which relied on the Simplex algorithm. The constraint-driven approach for power grid verification was later used in [7], where a heuristic was suggested to find worst-case voltage drops, relying on a random walk technique.

The present work offers significant advance over prior art by allowing early power grid verification of a *localized* area of the power grid. We refer to this localized block under verification as the *internal grid* and the power grid area outside the internal grid as the *external grid*. We will use macromodeling, in the style of [8], as a means to "divide-and-conquer" the power grid, which in our case involves reducing the number of constraints and "mapping them" to the internal grid. To grid partitioning, we combine notions of worst-case voltage drops [5] and certain locality properties of the grid, stemming from structural and electrical considerations [9]. From a computational standpoint, we have implemented an interior point method (IPM) as the backbone of our solvers, which lead to substantial runtime improvements over previous work. The result is an incremental grid verification of blocks on grids that are much larger than was previously possible, along with runtime savings.

3. POWER GRID ROBUSTNESS

3.1 Power Grid Model

For completeness of presentation, we derive a modified power grid system of equations, which will be useful to our problem solution below, following the derivation in [5]. We consider an RC model of the power grid, where each branch of the grid is represented by a resistor and where there exists a capacitor from every grid node to ground. In addition, some nodes have ideal current sources (to ground) representing the current drawn by the circuit tied to the grid at that point, and some nodes have ideal voltage sources (to ground) representing the connections to the external voltage supply. Let the power grid consist of $n+p$ nodes, where nodes $1, 2, \ldots, n$ have no voltage sources attached, and nodes $(n+1), (n+2), \ldots, (n+p)$ are the nodes with the voltage sources. Let c_k be the capacitance from every node k to ground. Let $i_k(t)$ be the current source connected to node k, where the direction of positive current is from the node to ground. We assume that $i_k(t) \geq 0$ and that $i_k(t)$ is defined for every node $k = 1, \ldots, n$ so that nodes with no current source attached have $i_k(t) = 0, \forall t$. Let $\mathbf{i}(t)$ be the vector of all $i_k(t)$ sources, $u_k(t)$ be the voltage at node k, and $\mathbf{u}(t)$ be the vector of all $u_k(t)$ signals. Applying Modified Nodal Analysis (MNA) leads to:

$$\mathbf{G}\mathbf{u}(t) + \mathbf{C}\dot{\mathbf{u}}(t) = -\mathbf{i}(t) + \mathbf{G}\mathbf{V_{dd}} \tag{1}$$

where $\mathbf{G}$ is an $n \times n$ conductance matrix, $\mathbf{C}$ is an $n \times n$ diagonal matrix of node capacitances, and $\mathbf{V_{dd}}$ is a constant vector each entry of which is equal to V_{dd}. Let $v_k(t) = V_{dd} - u_k(t)$ be the voltage drop at node k, and let $\mathbf{v}(t)$ be the vector of voltage drops, then (1) can be written as:

$$\mathbf{G}\mathbf{v}(t) + \mathbf{C}\dot{\mathbf{v}}(t) = \mathbf{i}(t) \tag{2}$$

This is a *revised* system equation which one can solve directly for the voltage drop values. Notice that the circuit described by (2) consists of the original power grid, but with all the voltage sources set to zero and all the current source directions reversed. In the following, we will mainly be concerned with this *modified power grid* and the revised system of equations (2). The DC analogue of (2) is readily seen as:

$$\mathbf{G}\mathbf{V} = \mathbf{I} \tag{3}$$

3.2 Local and Global Constraints

A local constraint relates to a single current source. For instance, one may specify that current $i_k(t)$ never exceeds a certain fixed level $I_{L,k}$, i.e., $i_k(t) \leq I_{L,k}, \forall t \geq 0$. This upper bound may be simply known from prior simulation, if the cell or block is already available, or it may be a best-guess based on the area of the cell or block and on perhaps the *power density* of the design (total power divided by total area). If further information is available on the circuit behavior over time, then the user may be able to specify an upper bound *waveform*, as a time function, so that $i_k(t) \leq i_{L,k}(t), \forall t \geq 0$. For brevity, we discuss upper bound constraints only, with the understanding that this work can be extended to handle lower bound/interval constraints. We assume that every current source tied to the power grid has an upper bound associated with it. If a grid node does not have a current source attached to it, i.e., $i_k(t) = 0, \forall t \geq 0$, then we specify a fixed zero-current upper bound for that node, $I_{L,k} = 0$. In this way, we have a local constraint associated with every node of the power grid. We express these constraints in vector form as:

$$\mathbf{0} \leq \mathbf{i}(t) \leq \mathbf{I_L}, \forall t \geq 0 \quad \text{or} \quad \mathbf{0} \leq \mathbf{i}(t) \leq \mathbf{i_L}(\mathbf{t}), \forall t \geq 0. \tag{4}$$

A global constraint corresponds to the case when the sum of the currents for a group of current sources is specified to have an upper bound. The upper bound for example, corresponding to the jth global constraint, may be a fixed bound $I_{G,j}$, or a waveform bound $i_{G,j}(t)$. If m is the number of available global constraints, then we express all the global constraints in matrix form as:

$$\mathbf{0} \leq \mathbf{U}\mathbf{i}(t) \leq \mathbf{I_G} \qquad \text{or} \qquad \mathbf{0} \leq \mathbf{U}\mathbf{i}(t) \leq \mathbf{i_G}(t), \tag{5}$$

where $\mathbf{U}$ is a $m \times n$ matrix that contains only 0s and 1s.

3.3 DC Robustness

Checking if the grid is robust entails checking if the voltage drop for any node exceeds some threshold, over all circuit currents that satisfy the constraints (4) and (5). This is a difficult problem, due to the size of the grid, not to mention the infinite number of possible current waveforms. A solution was presented in [5] to tackle the DC version of this grid robustness problem. In the DC problem, we are interested to check if the grid is safe under all possible DC currents that satisfy the constraints. That affords considerable simplification, and yields a problem that can be formulated as an LP. It was also shown in [5] that, due to the monotonicity property of the grid [6, 5] (roughly stated, the higher the currents loading the grids, the greater the voltage drops at the grid nodes), DC robustness guarantees that the grid is safe under "some types" of transient currents as well, but not all. So, there is merit in the DC solution, not least of which are its usefulness early in the design cycle and its ability to quickly discover problems in the grid which lead to errors under both DC and transient currents. Generalizing this approach to the case of transient current waveforms is part of our ongoing research and is outside the scope of this paper.

Focusing on the DC case, and using (3), we can express the DC constraints in terms of currents as:

$$\mathbf{0} \leq \mathbf{I}(t) \leq \mathbf{I_L}. \qquad \mathbf{0} \leq \mathbf{U}\mathbf{I} \leq \mathbf{I_G}. \tag{6}$$

In terms of voltages, this becomes:

$$\mathbf{0} \leq \mathbf{G}\mathbf{V} \leq \mathbf{I_L}, \qquad \mathbf{0} \leq \mathbf{S}\mathbf{V} \leq \mathbf{I_G}, \qquad \mathbf{V} \geq \mathbf{0}, \tag{7}$$

where $\mathbf{S} = \mathbf{U}\mathbf{G}$ is an $m \times n$ matrix mapping global constraints to node voltages. The problem now can be phrased as an LP

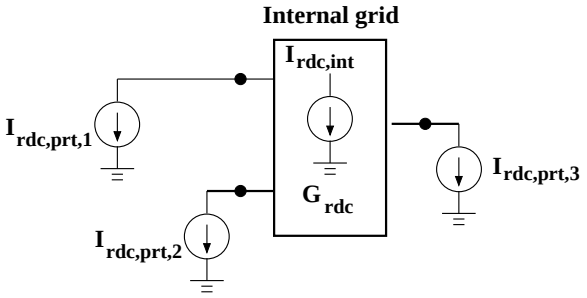

Figure 1: Modified system for the solution of the verification problem for the internal grid using macromodeling.

for finding the maximum voltage drop at the power grid nodes, under the constraints given in (7), with particular interest to the nodes pertaining to the internal power grid. Observe that $\mathbf{S}$ has the following special property, which will become useful below. Recall that every row (and column) of $\mathbf{G}$ corresponds to a node of the grid, excluding nodes corresponding to supply pads (C4 pads). Every row of $\mathbf{S}$ corresponds to a global constraint. If we focus on the i^{th} row of $\mathbf{S}$, let $\mathcal{G}_i$ be the set of grid nodes that belong to that global constraint. Then, the $(i,j)^{\text{th}}$ entry of $\mathbf{S}$ is the sum of all entries in the jth column of $\mathbf{G}$ that are in rows corresponding to nodes in $\mathcal{G}_i$.

3.4 Power Grid Macromodeling

Macromodeling is a "divide-and-conquer" technique applied for power grids in [8], which operates by partitioning the power grid into small blocks, where the behavior of each block is abstracted away at its ports, or nodes that connect this block to the outside. This approach first results in a small "global grid" for initial solution, then the this global solution is mapped back into individual blocks. For a detailed description, the reader is referred to [8].

Our approach, to be developed below, borrows from the ideas of macromodeling. Simply put, we will reduce our system to the internal grid and map all other currents to the port nodes of the internal grid. However, the main difficulty is that we will be dealing with equalities and/or local constraints on the currents outside the internal grid, local constraints on the port nodes, as well as global constraints that may include currents inside and outside the internal grid. Our approach will result in a reduced system, local to the internal grid, by eliminating external constraints from the constraint set. This reduced system will have a conductance matrix $\mathbf{G_{rdc}}$, and modified currents $\mathbf{I_{rdc}}$, that can be derived from the conductance and currents of the original system. This is illustrated in Fig. 1. The following sections detail this process.

4. LOCALITY IMPLICATIONS

We now introduce a refinement to the above problem (7), with regard to currents outside the internal grid (which we refer to as *external currents*), in relation to nodes inside the internal grid under verification. Modern power grids feature on-chip decoupling capacitors (decaps) and a dense and uniform array of C4 bumps (V_{dd} sites in flip-chip technology). Decaps are known to act as low-pass filters having an averaging effect on the currents in a surrounding neighborhood, that is, reducing node voltage sensitivity to current changes. It was also shown in [9] that the grid of C4 bumps strongly contributes to localized and averaged current patterns, again within some vicinity. Determining with precision the size of this "neighborhood of influence" of a C4, decap, or current source is still an open question at this time, but we will state for our purposes that, seen from within the internal power grid, external currents have a reduced effect, compared with their effect in their immediate vicinity.

We will therefore assume, based on the presence of decaps and C4s between the internal and external grids, that *some* external current sources can be treated as *fixed* and known DC values, effectively eliminating the local constraints on them when verifying the internal grid, and replacing these local constraints with an average value derived from the nominal switching activity of the external circuit. Other external current sources, depending on the block from which they are derived, the value of the currents they draw, and their proximity to the internal grid, may still be significant enough to the internal grid that, for accuracy of the analysis, one ought to maintain them as uncertain, with a constraint on their maximum values, rather than fix them at some average. To first order, this would be the case for big current offenders such as I/O buffers or clocks that are close enough to the internal grid, but can also include any external currents whose variation is expected to affect the internal grid significantly, or simply currents on which designers wish to maintain a degree of freedom when verifying the internal power grid.

The following two sections deal with mapping the problem (7) to the internal grid, and we present solutions to both cases of fixed and constrained external currents. For clarity of presentation, we treat these two cases separately in the next two sections, yielding a modification of the standard LP problem in each case. If the external grid needs to be modeled with some currents fixed and others constrained, then the maximum voltages on the internal grid can be found by superposition of the maximum voltages obtained considering the fixed external sources and the maximum voltages obtained for the constrained external sources.

5. PROBLEM REDUCTION WITH FIXED EXTERNAL CURRENTS

In this section, we assume that all external currents can be treated as fixed and known DC values. We refer to nodes on the internal grid with connections to the external grid as *port nodes*, and we refer to nodes on the internal grid under verification simply as *internal nodes* and to nodes on the external grid as *external nodes*. Let n_{ext}, n_{prt}, and n_{int} be respectively the number of external, port, and internal nodes, such that $n_{\text{ext}} + n_{\text{prt}} + n_{\text{int}} = n$. Assuming that external nodes connect only to the internal grid through the port nodes, the verification equations (7) can be written as [8]:

$$\mathbf{0} \leq \begin{bmatrix} \mathbf{G_{11}} & \mathbf{G_{12}} & \mathbf{0} \\ \mathbf{G_{12}^T} & \mathbf{G_{22}} & \mathbf{G_{23}} \\ \mathbf{0} & \mathbf{G_{23}^T} & \mathbf{G_{33}} \\ \mathbf{S_1} & \mathbf{S_2} & \mathbf{S_3} \end{bmatrix} \begin{bmatrix} \mathbf{V_{ext}} \\ \mathbf{V_{prt}} \\ \mathbf{V_{int}} \end{bmatrix} \leq \begin{bmatrix} \mathbf{I_{ext}} \\ \mathbf{I_{prt}} \\ \mathbf{I_{int}} \\ \mathbf{I_G} \end{bmatrix}, \qquad (8)$$

where $\mathbf{V_{ext}}$ and $\mathbf{I_{ext}}$ are n_{ext}-subvectors corresponding to external grid nodes and external current sources, $\mathbf{V_{prt}}$ and $\mathbf{I_{prt}}$ are n_{prt}-subvectors corresponding to port nodes and currents, and $\mathbf{V_{int}}$ and $\mathbf{I_{int}}$ are n_{int}-subvectors corresponding to internal grid nodes and and *internal current sources* (current sources inside the internal grid), and the $\mathbf{G}$ and $\mathbf{S}$ matrices are partitioned into submatrices of appropriate dimensions. Observe that in the above partitioning, $\mathbf{G_{11}}(n_{\text{ext}} \times n_{\text{ext}})$, $\mathbf{G_{22}}(n_{\text{prt}} \times n_{\text{prt}})$, and $\mathbf{G_{33}}(n_{\text{int}} \times n_{\text{int}})$ are M-matrices, whereas $\mathbf{G_{12}}(n_{\text{ext}} \times n_{\text{prt}})$ and $\mathbf{G_{23}}(n_{\text{prt}} \times n_{\text{int}})$ are non-positive matrices.

Since we are fixing external currents when verifying a block, we interpret $\mathbf{I_{ext}}$ as a fixed vector not as a vector of upper bound constraints, and the local constraint on the external currents in (8) turns into the following vector equality:

$$\mathbf{G_{11}V_{ext}} + \mathbf{G_{12}V_{prt}} = \mathbf{I_{ext}}. \qquad (9)$$

Further, we can partition the matrix $\mathbf{U}$ introduced in (5) into $\mathbf{U} = [\mathbf{U_{ext}}\ \mathbf{U_{prt}}\ \mathbf{U_{int}}]$, where $\mathbf{U_{ext}}$, $\mathbf{U_{prt}}$, and $\mathbf{U_{int}}$ are three submatrices of dimensions $m \times n_{\text{ext}}$, $m \times n_{\text{prt}}$, and $m \times n_{\text{int}}$ respectively. Fixed external currents imply a considerable simplification to the global constraints: we can replace $\mathbf{I_G}$ with $\mathbf{I_G} - \mathbf{U_{ext}I_{ext}}$ and take $\mathbf{U} = [\mathbf{0}\ \mathbf{U_{prt}}\ \mathbf{U_{int}}]$. This simply means that, although global constraints may encompass both internal and external current sources, fixing external currents allows us to subtract their values from the upper bounds of the global constraints (when applicable) so that we can always assume global constraints involve only internal currents.

We can write from (9):

$$\mathbf{V_{ext}} = \mathbf{G_{11}^{-1}}(\mathbf{I_{ext}} - \mathbf{G_{12}V_{prt}}). \qquad (10)$$

Substituting (10) into the second inequality of (8) leads to:

$$-\mathbf{G_{12}^T G_{11}^{-1} I_{ext}} \leq (\mathbf{G_{22}} - \mathbf{G_{12}^T G_{11}^{-1} G_{12}})\mathbf{V_{prt}} + \mathbf{G_{23} V_{int}} \leq \mathbf{I_{prt}} - \mathbf{G_{12}^T G_{11}^{-1} I_{ext}}, \quad (11)$$

where $\mathbf{G_{prt}} = \mathbf{G_{22}} - \mathbf{G_{12}^T G_{11}^{-1} G_{12}}$ is the port admittance matrix [8]. Since $\mathbf{G_{11}}$ is an M-matrix, $\mathbf{G_{11}^{-1}} \geq \mathbf{0}$, and since $\mathbf{G_{12}} \leq \mathbf{0}$, then $-\mathbf{G_{12}^T G_{11}^{-1} I_{ext}} \geq \mathbf{0}$. Let

$$\mathbf{I_{map}} = -\mathbf{G_{12}^T G_{11}^{-1} I_{ext}}. \quad (12)$$

Then, based on (11), we observe that mapping the external currents to the ports shifts the local constraint on the i^{th} port current by a *positive* amount equal to the i^{th} component of $\mathbf{I_{map}}$. This shift establishes both *upper* and *lower bounds* on the port currents. The local constraints on the internal nodes are unchanged. We have from (8):

$$\mathbf{0} \leq \mathbf{G_{23}^T V_{prt}} + \mathbf{G_{33} V_{int}} \leq \mathbf{I_{int}}. \quad (13)$$

We now examine the global constraints. From (8), we have:

$$\mathbf{0} \leq \mathbf{S_1 V_{ext}} + \mathbf{S_2 V_{prt}} + \mathbf{S_3 V_{int}} \leq \mathbf{I_G}. \quad (14)$$

Using $\mathbf{S} = \mathbf{UG}$ and $\mathbf{U} = [\mathbf{0}\ \mathbf{U_{prt}}\ \mathbf{U_{int}}]$, leads to:

$$\begin{aligned} \mathbf{S_1} &= \mathbf{U_{prt} G_{12}^T} \\ \mathbf{S_2} &= \mathbf{U_{prt} G_{22}} + \mathbf{U_{int} G_{23}^T} \\ \mathbf{S_3} &= \mathbf{U_{prt} G_{23}} + \mathbf{U_{int} G_{33}}. \end{aligned} \quad (15)$$

Substituting (15) and (10) in (14) yields:

$$\mathbf{U_{prt} I_{map}} \leq [\mathbf{U_{prt}}\ \mathbf{U_{int}}] \begin{bmatrix} \mathbf{G_{prt}} & \mathbf{G_{23}} \\ \mathbf{G_{23}^T} & \mathbf{G_{33}} \end{bmatrix} \begin{bmatrix} \mathbf{V_{prt}} \\ \mathbf{V_{int}} \end{bmatrix} \leq \mathbf{I_G} + \mathbf{U_{prt} I_{map}}. \quad (16)$$

We define the *reduced (internal) grid conductance matrix*

$$\mathbf{G_{rdc}} = \begin{bmatrix} \mathbf{G_{prt}} & \mathbf{G_{23}} \\ \mathbf{G_{23}^T} & \mathbf{G_{33}} \end{bmatrix} \quad (17)$$

to be the conductance matrix of the internal grid in the reduced system (see section 6.1). We further define

$$\mathbf{S_{rdc}} = [\mathbf{S_{rdc,1}}\ \mathbf{S_{rdc,2}}] = [\mathbf{U_{prt}}\ \mathbf{U_{int}}] \begin{bmatrix} \mathbf{G_{prt}} & \mathbf{G_{23}} \\ \mathbf{G_{23}^T} & \mathbf{G_{33}} \end{bmatrix} \quad (18)$$

to be the *reduced matrix of global constraints*. Combining (11), (13), (16), and (18) yields:

$$\begin{bmatrix} \mathbf{I_{map}} \\ \mathbf{0} \\ \mathbf{U_{prt} I_{map}} \end{bmatrix} \leq \begin{bmatrix} \mathbf{G_{prt}} & \mathbf{G_{23}} \\ \mathbf{G_{23}^T} & \mathbf{G_{33}} \\ \mathbf{S_{rdc,1}} & \mathbf{S_{rdc,2}} \end{bmatrix} \begin{bmatrix} \mathbf{V_{prt}} \\ \mathbf{V_{int}} \end{bmatrix} \leq \begin{bmatrix} \mathbf{I_{prt}} + \mathbf{I_{map}} \\ \mathbf{I_{int}} \\ \mathbf{I_G} + \mathbf{U_{prt} I_{map}} \end{bmatrix}. \quad (19)$$

It is clear that (19) corresponds to a system which is analogous to (8), but with size reduced to the number of internal nodes (including ports), in addition to the number of global constraints. Therefore, we now have a reduced form of the standard LP problem, stated in (7), that we can use to find the maximum voltage drops at the internal nodes. Effectively, external nodes were eliminated from the system.

All entries in (19) can be computed in a straightforward fashion; however, $\mathbf{I_{map}}$ and $\mathbf{G_{prt}}$ contain $\mathbf{G_{11}^{-1}}$ in their expressions, which is, at first sight, expensive to compute and store. However, using Cholesky factorization, we will circumvent the need to inverse the port admittance matrix altogether [8]. Instead, the cost of $\mathbf{G_{prt}}$ will be essentially one matrix-matrix multiply and that of $\mathbf{I_{map}}$ one forward solve, given the Cholesky factorization of the top left $(n_{\text{ext}} + n_{\text{prt}}) \times (n_{\text{ext}} + n_{\text{prt}})$ submatrix of $\mathbf{G}$. We will discuss computational efficiency in section 6.2.

6. PROBLEM REDUCTION WITH UNCERTAIN EXTERNAL CURRENTS

In this section, we consider external currents to be constrained, and map the external current constraints to the terminals of the internal grid. This mapping will be in the form of shifted upper bounds in the local and global constraints.

6.1 Macromodeling the External Grid With Current Constraints

Recalling the decomposition in (8), we have:

$$\mathbf{G_{11} V_{ext}} + \mathbf{G_{12} V_{prt}} \leq \mathbf{I_{ext}}. \quad (20)$$

Unlike in (9), $\mathbf{I_{ext}}$ in the above equation is a vector of upper bounds representing local constraints on the external current sources, not a fixed vector of external currents. Since $\mathbf{I_{ext}} \geq \mathbf{0}$, $\mathbf{V_{prt}} \geq \mathbf{0}$, and $\mathbf{G_{12}} \leq \mathbf{0}$, then $\mathbf{I_{ext}} - \mathbf{G_{12} V_{prt}} \geq \mathbf{0}$. Since $\mathbf{G_{11}^{-1}} \geq \mathbf{0}$, then (20) yields:

$$\mathbf{V_{ext}} \leq \mathbf{G_{11}^{-1}}(\mathbf{I_{ext}} - \mathbf{G_{12} V_{prt}}). \quad (21)$$

Pre-multiplying the above inequality by $\mathbf{G_{12}^T} \leq \mathbf{0}$ leads to:

$$\mathbf{G_{12}^T V_{ext}} \geq \mathbf{G_{12}^T G_{11}^{-1}}(\mathbf{I_{ext}} - \mathbf{G_{12} V_{prt}}). \quad (22)$$

Also, we can directly write from (8):

$$\mathbf{G_{12}^T V_{ext}} \leq \mathbf{I_{prt}} - \mathbf{G_{22} V_{prt}} - \mathbf{G_{23} V_{int}} \quad (23)$$

By combining (22) and (23), we can "eliminate" the external node voltages from the constraints and obtain:

$$\mathbf{G_{prt} V_{prt}} + \mathbf{G_{23} V_{int}} \leq \mathbf{I_{prt}} + \mathbf{I_{map}}, \quad (24)$$

where the expressions of $\mathbf{G_{prt}}$ and $\mathbf{I_{map}}$ are the same as in section 5. The local constraints on the internal current sources remain unchanged, as in (13).

Next, we "eliminate" the external nodes from the global constraints. Assuming that $\mathbf{S_1} \leq \mathbf{0}$, we pre-multiply both sides of the inequality in (21) with $\mathbf{S_1}$ to obtain:

$$\mathbf{S_1 V_{ext}} \geq \mathbf{S_1 G_{11}^{-1}}(\mathbf{I_{ext}} - \mathbf{G_{12} V_{prt}}). \quad (25)$$

We note that in this case: $\mathbf{U} = [\mathbf{U_{ext}}\ \mathbf{U_{prt}}\ \mathbf{U_{int}}]$, and in particular that, unlike section 5, we may not assume that $\mathbf{U_{ext}} = \mathbf{0}$. Combining the above inequality with (14), and using the fact that $\mathbf{S} = [\mathbf{S_1}\ \mathbf{S_2}\ \mathbf{S_3}] = \mathbf{UG}$, we get the reduced global constraints:

$$[\mathbf{U_{prt}}\ \mathbf{U_{int}}] \begin{bmatrix} \mathbf{G_{prt}} & \mathbf{G_{23}} \\ \mathbf{G_{23}^T} & \mathbf{G_{33}} \end{bmatrix} \begin{bmatrix} \mathbf{V_{prt}} \\ \mathbf{V_{int}} \end{bmatrix} \leq \mathbf{I_G} + \mathbf{U_{prt} I_{map}} - \mathbf{U_{ext} I_{ext}}, \quad (26)$$

or alternatively:

$$(\mathbf{S_2} - \mathbf{S_1 G_{11}^{-1} G_{12}})\mathbf{V_{prt}} + \mathbf{S_3 V_{int}} \leq \mathbf{I_G} - \mathbf{S_1 G_{11}^{-1} I_{ext}}. \quad (27)$$

If the condition $\mathbf{S_1} \leq \mathbf{0}$ is *not* satisfied, then the problem cannot be so reduced. The requirement that $\mathbf{S_1} \leq \mathbf{0}$ means that the partitioning cannot be arbitrary and *must* keep the the global constraints in mind. We will discuss this in more detail in section 6.4 and devise a scheme to find suitable partitioning of the power grid to enable the incremental verification of a specific block or region of the power grid. For now, we will assume that $\mathbf{S_1} \leq \mathbf{0}$ in our constraint set. Combining (24), (13), and (27), we now have the following reduced problem, where local and global constraints on the external currents have been mapped to the terminal nodes:

$$\begin{bmatrix} \mathbf{G_{prt}} & \mathbf{G_{23}} \\ \mathbf{G_{23}^T} & \mathbf{G_{33}} \\ \mathbf{S_2} - \mathbf{S_1 G_{11}^{-1} G_{12}} & \mathbf{S_3} \end{bmatrix} \begin{bmatrix} \mathbf{V_{prt}} \\ \mathbf{V_{int}} \end{bmatrix} \leq \begin{bmatrix} \mathbf{I_{prt}} + \mathbf{I_{map}} \\ \mathbf{I_{int}} \\ \mathbf{I_G} - \mathbf{S_1 G_{11}^{-1} I_{ext}} \end{bmatrix}. \quad (28)$$

Note that all lower bounds in the reduced system are 0. We abbreviate (28) as $\mathbf{KV'} \leq \mathbf{I_m}$, where again we recognize the LP form.

6.2 Efficient Analysis

Carrying out the above partitioning is computationally easy, except for the one problem that $\mathbf{G_{11}^{-1}}$ is required for computing $\mathbf{G_{prt}}$, $\mathbf{I_{map}}$, and $\mathbf{I'_G} = \mathbf{I_G} - \mathbf{S_1 G_{11}^{-1} I_{ext}}$. In general, it is undesirable to require explicit computation of the inverse. In order to simplify the expression for $\mathbf{I'_G}$, it suffices to make one further requirement (and we will provide reasoning for this below) that $\mathbf{S_1} = \mathbf{0}$, in which case $\mathbf{I'_G} = \mathbf{I_G}$.

In order to resolve the difficulty with $\mathbf{G_{prt}}$ and $\mathbf{I_{map}}$, we take advantage of the result in [8] which allows us to calculate these matrix expressions without the need of inverting $\mathbf{G_{11}}$. This is done using the submatrices of the Cholesky factorization. Looking at the matrix composed of the four sub-matrices $\mathbf{G_{11}}$, $\mathbf{G_{12}}$, $\mathbf{G_{12}^T}$, and $\mathbf{G_{22}}$, let the Cholesky decomposition of that matrix be as follows:

$$\begin{bmatrix} \mathbf{G_{11}} & \mathbf{G_{12}} \\ \mathbf{G_{12}^T} & \mathbf{G_{22}} \end{bmatrix} = \begin{bmatrix} \mathbf{L_{11}} & \mathbf{0} \\ \mathbf{L_{21}} & \mathbf{L_{22}} \end{bmatrix} \begin{bmatrix} \mathbf{L_{11}^T} & \mathbf{L_{21}^T} \\ \mathbf{0} & \mathbf{L_{22}^T} \end{bmatrix}. \quad (29)$$

With this, it was then shown in [8] that $\mathbf{G_{prt}}$ can be simply expressed as $\mathbf{G_{prt}} = \mathbf{L_{22} L_{22}^T}$. Furthermore, it was observed that:

$$\mathbf{G_{12}^T G_{11}^{-1}} = \mathbf{L_{21} L_{11}^T} (\mathbf{L_{11} L_{11}^T})^{-1} = \mathbf{L_{21} L_{11}^{-1}} \quad (30)$$

leading to $\mathbf{I_{map}} = -\mathbf{L_{21} L_{11}^{-1} I_{int}}$. Now $\mathbf{L_{11}^{-1} I_{int}}$ can be computed with one forward solve, so that no inversion is explicit in the process of computing $\mathbf{I_{map}}$. Similarly, we can write

$$\mathbf{G_{12}^T G_{11}^{-1} G_{12}} = \mathbf{L_{21} L_{21}^T}, \quad (31)$$

which means that the bulk of the computation for $\mathbf{G_{prt}}$ lies in a square matrix-matrix multiply, of very small size, equal to the number of port nodes.

This approach of harnessing the Cholesky factors substantially reduces the computation cost associated with the calculation of the reduced system [8]. The most expensive operation in this procedure remains the Cholesky factorization itself. The remaining operations can be done effortlessly since all the matrix factors involved are for the most part triangular.

6.3 Topological Characteristics

We reiterate the point made earlier, that care must be taken in partitioning the grid (in the case of uncertain external currents only). The block should be chosen in a way that guarantees $\mathbf{S_1} \leq \mathbf{0}$. For computational efficiency, it is further desired to achieve $\mathbf{S_1} = \mathbf{0}$. The question becomes: how does one topologically partition the grid graph in order to end up with $\mathbf{S_1} = \mathbf{0}$? We will show in this section that this is very easy to do, in fact, due to the special properties of the matrix $\mathbf{S}$, which were pointed out at the end of section 3.3. If S_{ij} is the (i, j) entry of $\mathbf{S}$ and G_{kj} is the (k, j) entry of $\mathbf{G}$, then it follows from our previous discussion of $\mathbf{S}$ that:

$$S_{ij} = \sum_{k \in \mathcal{G}_i} G_{kj} \quad (32)$$

where, recall, $\mathcal{G}_i$ is the set of grid nodes that belong to the ith global constraints, i.e., nodes where current sources are tied that are included in the ith global constraint. This intimate relationship between $\mathbf{S}$ and $\mathbf{G}$ is key, and is shown in Fig. 2.

Nodes that may be removed (nodes in $\mathbf{V_{ext}}$) are nodes that correspond to columns of $\mathbf{S}$ whose entries are all 0. To identify these nodes, and due to (32), we start by recalling the well-known properties of the conductance matrix $\mathbf{G}$. That matrix is symmetrical, and every row (and column) of $\mathbf{G}$ uniquely corresponds to a node of the grid that is not directly tied to an ideal voltage source (C4 pad). Every diagonal entry of $\mathbf{G}$ is positive, $G_{ii} > 0$, and equals the sum of all conductances tied to that node. Every off-diagonal entry of $\mathbf{G}$ is negative or zero, $G_{ij} \leq 0, i \neq j$, and equals the negative of the conductance (if any) that connects node i to node j. If none of the immediate neighbors of node i is a C4 pad, then the sum of all entries in that row (or in that column) of $\mathbf{G}$ is zero, $\sum_i G_{ij} = 0$. If at least one immediate neighbor of node i is a C4 pad, then $\sum_i G_{ij} > 0$.

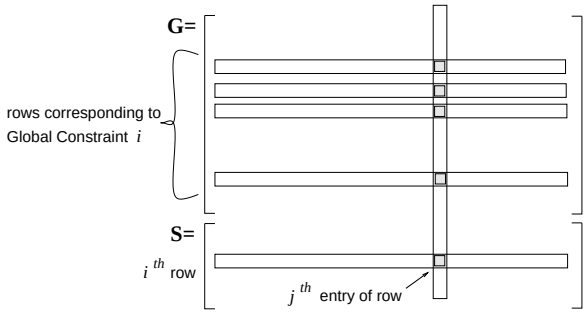

Figure 2: Special properties of the S matrix.

Given (32), the following facts become immediately clear. If node j is not included in $\mathcal{G}_i$, then the diagonal term G_{jj} is not part of the summation (32), and therefore, $S_{ij} \leq 0$. Otherwise, when node j *is* included in $\mathcal{G}_i$, G_{jj} is part of the summation (32), and $S_{ij} \geq 0$. Finally, there are two cases that lead to $S_{ij} = 0$: 1) if neither node j nor any of its immediate neighbors are part of $\mathcal{G}_i$, or 2) if node j and *all* its immediate neighbors are part of $\mathcal{G}_i$, provided that none of those neighbors is a C4 pad.

Therefore, a node that *may* be removed (nodes in V_{ext}) bears the following relationship to *every* global constraint: either that node and all its neighbors are outside that global constraint, or that node and all its neighbors (none of which is a C4 pad) are inside that global constraint.

6.4 Partitioning Approach

It is clear from the concluding remark in the previous section that partitioning may be naturally done by looking at the global constraints and considering which nodes belong in them.

We make some reasonable engineering assumptions about the global constraints. Firstly, we assume that there is one global constraint that covers *all* the nodes of the grid, which we refer to as the "full chip constraint" (FCC). This constraint might stem, typically, from knowledge of the total power dissipation of this chip. Further, we assume that the other global constraints (other than the FCC, that is) correspond to specific design blocks, so that, typically, each constraint may relate to the power dissipation of a specific block. In other words, and since design blocks are distinct objects, these non-FCC constraints do not overlap. Thus, every node is covered by the FCC, and may be covered by at most one other constraint.

With this, our partitioning algorithm becomes as simple as the following. We assume that the user specifies what parts of the grid are definitely to be preserved (these are the parts that one is interested in verifying). All nodes in these parts are marked `keep`, meaning that they will not be eliminated as part of the partitioning. Likewise, nodes whose immediate neighbors include a C4 pad are also marked `keep`. Finally, for every non-FCC global constraint, we mark as `keep` any nodes that have at least one immediate neighbor outside that constraint. With this, any nodes that are not marked `keep` can now be eliminated - they constitute $\mathbf{V_{ext}}$. Nodes within a non-FCC global constraint that are immediate neighbors of nodes outside of it become the "terminal nodes" - they constitute $\mathbf{V_{prt}}$. All other nodes are part of $\mathbf{V_{ext}}$.

As an example, applying this partitioning approach to the small grid shown in Fig. 3 leads to the reduced grid shown in Fig. 4.

6.5 Linear Programming Solution

Once a region of interest on the power grid has been chosen for analysis we may again as in [5] reformulate the problem as a linear program around the reduced constraint set and solve for worst case voltage drop sequentially on every node in the region. Thus having a set of nodes that we are interested in verifying $(v_i, v_j ... v_m)$ we may formulate the sequential linear program:

$$\begin{aligned} \text{maximize :} \quad & v_i, v_j, ... v_m \\ \text{subject to :} \quad & \mathbf{K V'} \leq \mathbf{I_m} \\ & \mathbf{V'} \geq \mathbf{0} \end{aligned} \quad (33)$$

An important note that needs to be mentioned, for readers that

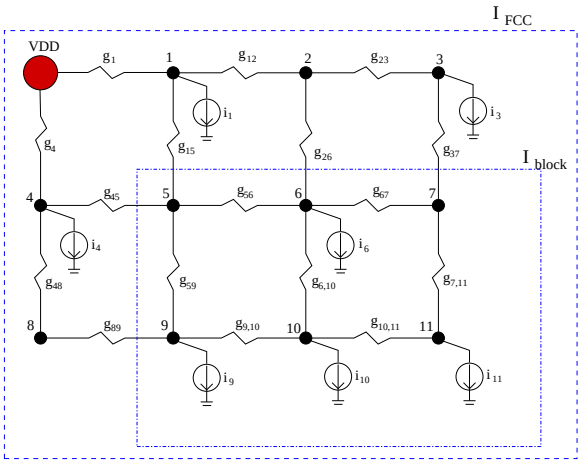

Figure 3: A small grid, showing two global constraints.

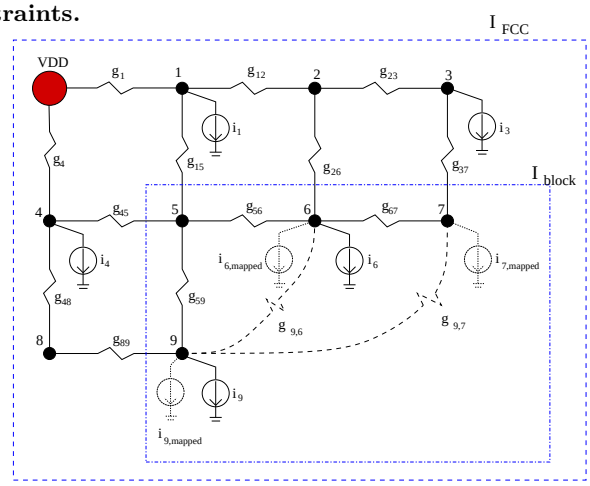

Figure 4: Reduced grid.

are familiar with hierarchical analysis of power grids, is that once the system has been reduced, optimizing for a worst-case voltage drop may only be run on the *internal* nodes. Due to the nature of the mathematics involved (mainly the fact that our system is modeled as constraints) the solution of the worst-case voltages cannot be used to solve for a worst-case voltage drop for *external* nodes of a partition.

7. EXPERIMENTAL RESULTS

For the testing of our technique we have used a power grid synthesizer to generate representative two-layer grids. The pitch and process metalization conductivity is specified by the user as well as a non-uniformity percentage. The grid is generated by first laying down a Manhattan structure and then randomly deleting by the given percentage, nodes of the grid and their adjacent metal branches thus yielding a non-uniform structure. When a node and its branches are deleted the surrounding remaining branches are increased by a random amount around a user specified percentage of their initial values. The rationale behind this is to allow the non-uniform grid to be loaded with currents comparable to its uniform predecessor while exhibiting comparable IR-drops. The number of V_{dd} sites are supplied by the user; and are then distributed at random over the grid nodes. Further, the global constraint geometries and current sources, both bounded and fixed, are defined in a topology layout file. The topology layout is checked and verified to insure that the global constraints do not violate or overlap each other based on the engineering assumptions made in section 6.4.

All experiments were run on a 1 GHz Sun machine with 4 GB memory. In [5], it was not possible to verify the voltage drop of grids much larger than a few thousand nodes with an LP problem formulation. In order to increase speed and the ability to solve much larger grids with our LP formulation we opted to implement an interior point method. Table 1 shows a comparison of analysis times between simplex and IPM. We notice that for the small grid case of 500 nodes, simplex with objective function switching [5] is in fact faster than the IPM, however the performance gains of IPM can be viewed with the much larger grids.

We present computation times of our reduction method with fixed external currents in Table 2. The results shown are for a locale of 50 nodes which are internal to a block. Column three of the table is the block size that the entire grid is reduced to. Column four is the size of the partition that is used to iteratively reduce the grid. Since the partitioning technique used to eliminate the external grid of the block requires a Cholesky factorization, as the size of the partition grows so does the computational complexity. It was understood early on in the experimentation stage that trying to macromodel a section of the grid that is too large could possibly negate the computational benefits of the approach. Due to the fact that the fixed currents are equalities and all exist outside of the block, we decided to partition the grid iteratively by a reduction partition that is chosen to be relatively small so that computation times are not excessive. Using this set size of the reduction partition column five of the table shows the time required to reduce the entire grid to the internal grid. The last column of the table is the total time required to reduce the system and solve for the maximum voltage drop on all 50 nodes.

Table 3 shows the associated computation costs of partitioning grids and solving for the worst-case voltage drop for a localized area containing 50 nodes using current constraints. In Table 3 the last two columns show a comparison in run-time (for the verification of the 50 node locale) between using strictly the interior point method and using the interior point but with partitioning. The partitioning approach allows for faster run time with about $10X$ - $20X$ speed up. It is significant to observe from our results the simulation times of the last two grids. The time required to simulate the internal grid of the 1M node grid is less than that of the 500k grid. This is due to the computational cost associated with the partitioning method as was discussed in the previous paragraph. In the 1M grid case the global constraints which influence our partitioning approach as discussed in section 6.4 happen to be smaller than those of the 500k grid. Table 4 shows the average difference of voltage drop values between solving the grid with constraints using IPM and solving using IPM with partitioning for three grids for a locality of 100 nodes in each run. The results indicate good accuracy as the difference ranges from about 0.8% to 3%.

8. CONCLUSION

In today's integrated circuit designs, voltage fluctuations on the power grid are a key concern. Due to the large size of grids and the difficulty in obtaining proper stimulus for all possible circuit operations, simulation of the voltage drops on the grid are extremely difficult. In this work we have taken the power grid verification approach which uses current constraints to abstract the behavior of the circuit, and improved it in terms of performance and its capacity to handle much larger grids, while focusing on a local area within the grid. By implementing an approach that partitions the grid around the current constraints and has the capacity to handle fixed current constraints we are able to eliminate portions of the grid outside the area under verification and then solve the reduced problem as a linear program. We believe this approach will be very useful going forward in incremental design verification of the power grid, given the ability to simulate grids of industrial size and within practical computation times.

9. REFERENCES

[1] Power grid verification. White Paper, Cadence Design Systems, San Jose, CA, 2002. [Online]. Available: `www.cadence.com/whitepapers/4101_PowerGridVerif_WP.pdf`.

[2] A. Dharchoudhury, R. Panda, D. Blaauw, R. Vaidyanathan, B. Tutuianu, and D. Bearden. Design and analysis of power

Table 1: Comparison of LP solvers

Grid Size	Complexity Analysis			
	Simplex		Interior Point Method	
	Avg. time/node (sec.)	Total time	Avg. time/node (sec.)	Total time
500	0.06	31.4 sec.	0.14	74.1 sec.
1k	0.87	894 sec.	0.2	198 sec.
2k	10.5	5.9 hr.	0.57	19 min.
3k	17.5	14.7 hr.	0.84	41 min.
5k	-	-	2.5	3.5 hr.
10k	-	-	8.4	23.5 hr.
27k	-	-	30.5	-
43k	-	-	96.8	-

Table 2: Run Time Results for a Localized Area of 50 nodes with fixed current sources using IPM

Size of Grid (# nodes)	# C4s	# nodes of internal	Size of reducing partition	Time to reduce system	Total Time
10k	50	3k	7k	7.6 sec.	49.6 sec.
40k	200	3k	6k	24 sec.	66 sec.
80k	200	5k	8.5k	70 sec.	3.3 min
160k	300	5k	12k	3.7 min.	5.8 min.
500k	400	10k	21k	26 min.	32 min.
1M	400	12	30k	71 min.	99 min.

Table 3: Run Time Results for a Localized Area of 50 nodes with upper bound current constraints using IPM

Size of Grid (# nodes)	# C4s	# Global Constraints	# of partitions	max. Size of partitions	Time to generate max. partition	Analysis Time without partition	Analysis Time with partition
10k	50	2	1	8k	9 sec.	12.5 min.	25 sec.
40k	200	2	1	20k	1 min.	118 min.	12 min.
80k	200	5	3	40k	4 min.	350 min.	16 min.
160k	300	5	4	80k	17 min.	-	37 min.
500k	400	5	3	150k	70 min.	-	256 min.
1M	400	14	12	80k	17 min.	-	245 min.

Table 4: Voltage Drop Difference between IPM flat power grid verification and IPM using partitioning

Grid Size	Local size	Avg. % Difference	% STD
500	100	1.20	0.34
1k	200	3.05	0.48
10k	2000	0.86	0.21

distribution networks in PowerPC^{TM} microprocessors. In *ACM/IEEE Design Automation Conference*, pages 738–743, San Francisco, CA, June 15-19 1998.

[3] R. Singh, editor. *Signal Integrity Effects in Custom IC and ASIC Designs*. IEEE Press, New York, NY, 2002.

[4] G. Steele, D. Overhauser, S. Rochel, and S. Z. Hussain. Full-chip verification methods for DSM power distribution systems. In *ACM/IEEE Design Automation Conference*, pages 744–749, San Francisco, CA, June 15-19 1998.

[5] D. Kouroussis and F. N. Najm. A static pattern-independent technique for power grid voltage integrity verification. In *ACM/IEEE Design Automation Conference*, pages 99–104, Anaheim, CA, June 2-6 2003.

[6] H. Kriplani, F. N. Najm, and I. Hajj. Pattern independent maximum current estimation in power and ground buses of CMOS VLSI circuits: algorithms, signal correlations, and their resolution. *IEEE Transactions on Computer-Aided Design*, 14(8):998–1012, August 1995.

[7] H. Qian, S. R. Nassif, and S. S. Sapatnekar. Early-stage power grid analysis for uncertain working modes. In *ACM International Symposium on Physical Design*, pages 132–137, Phoenix, AZ, April 18-21 2004.

[8] M. Zhao, R. V. Panda, S.S. Sapatnekar, and D. Blaauw. Hierarchical analysis of power distribution networks. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 21(2):159–168, February 2002.

[9] E. Chiprout. Fast flip-chip power grid analysis via locality and grid shells. In *IEEE/ACM International Conference on Computer-Aided Design*, pages 485–488, San Jose, CA, November 7-11 2004.