Handling Inductance in Early Power Grid Verification*

Nahi H. Abdul Ghani
Department of ECE
University of Toronto
Toronto, Ontario, Canada
nahi@eecg.utoronto.ca

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

ABSTRACT

As part of integrated circuit design verification, one should check if the voltage drop on the power grid exceeds some critical threshold. One way to do this is by simulation, but that is computationally expensive and gets prohibitive for large circuits with a large variety of possible operational modes. Another limitation of a simulation-based approach is that it requires complete knowledge of the logic circuitry drawing current from the grid, thus precluding grid verification early in the design process. In this paper, we model the grid as an RLC circuit and we propose three verification techniques that can be applied in the early stages of the design process. These techniques do not require exact knowledge of the circuit currents. Instead, the currents drawn by the logic beneath the power grid are described by means of *current constraints* that capture the uncertainty about circuit details and activity. The first verification approach gives the exact worst-case voltage drop at every node of the grid, but it is slow. A second faster approach gives conservative bounds on the worst-case voltage drop at every node of the grid. The third approach is much faster; it is a conservative approach which simply checks if the grid voltage drop exceeds some pre-defined thresholds, without actually computing the worst-case voltage drop at every node.

1. INTRODUCTION

A well-designed power grid in integrated circuits (ICs) should guarantee the proper logic functionality at the intended design speed. In deep sub-micron (DSM) technologies, with reduced supply voltage levels, modern IC designs are becoming increasingly susceptible to supply voltage problems. A key concern is the fact that logic circuits slow down under reduced supply voltage and that overall circuit timing performance is thus at risk [1]. It is clear that proper analysis and verification of the power grid are required for reliable high-speed chip design. There are two primary causes of supply voltage violations on the power grid: IR drop, which is a result of the resistivity of metal rails of the grid and Ldi/dt drop, due to the inductances of the rails and the interaction between the grid and the package. Inductive effects on the power grid must be included when verifying circuits operating at high frequencies [5, 6, 9].

An RLC model of the power grid manifests significant differences from its RC counterpart, which become consequential to any power grid verification methodology. We note in particular that the presence of inductances can cause the voltage on a given node of the power grid to fluctuate in both directions, either increasing or decreasing (for an RC grid model, voltage levels can only be inferior to V_{dd}, under the assumption that the circuit does not inject current into the grid). Therefore, one must account for *overshoots*, where the voltage level at a given node may exceed the nominal supply voltage level (V_{dd}), as well as *undershoots*, where the voltage drops below supply level. Consequently, grid safety must be stated in terms of *safety intervals* (*e.g.*, $\pm 5\% V_{dd}$) at each node, instead of simply a threshold value (*e.g.*, $10\% V_{dd}$ drop) under which the voltage at a given node must not fall.

From a methodology standpoint, one would like to verify grid safety without the need to *simulate* the grid for exhaustive current traces. For one thing, the number of traces needed to cover the space of voltage drops exhibited on the grid is intractable for modern industrial designs where grids can consist of several million nodes. Another major drawback of a simulation-based approach is its inability to handle grid verification early in the design process, when grid modifications can be most easily incorporated. Instead, and in order to enable an early and simulation-free verification methodology, we will verify the grid under partial current specifications, in the form of *current constraints* [7], the form of which will be detailed in section 4. The problem of power grid verification under these constraints is formulated and solved as a linear program (LP) involving the maximization/minimization of voltage drop at a given node on the grid. These constraints capture designers' knowledge of circuit characteristics, or they may be viewed as a way to incorporate design specifications into the grid verification early in the design cycle, enabling spec-based grid design (having verified the grid with certain current constraints, these constraints become design constraints, to be met by all subsequent design activity). In [4], a conductance-only model of the grid is adopted to perform vectorless grid verification under DC currents. In [7], an RC grid model is used and the grid is verified under transient currents. In this work, we extend the applicability of this constraint-based approach to allow for inductance.

We model the grid as an RLC network and we consider IR and Ldi/dt drops. The latter includes the inductance of the metal lines and the inductance of interconnections between the grid and the package, which we model as a tank circuit with lumped parasitic inductances and resistances [10]. In section 5.3, we experimentally give evidence to the fact that performing constraint-based verification using an RC model alone can underestimate worst-case drops and totally neglects overshoots, thus establishing the need for an inductance-aware grid verification under current constraints. Further, under the RLC model, we show that we are able to maintain the formulation of grid verification in an LP form, with the important distinction over prior art that both a minimization and a maximization need to be performed to account for overshoots and undershoots, that is, to verify whether a given node voltage falls within its safety interval under the specified current constraints. In the following, we say that a node is safe if all its feasible voltages under the current constraints are within its safety interval, and unsafe otherwise. We say a grid is safe if all its nodes are safe. Our approach for inductance-

*This work has been supported by Intel Corp., by Altera Corp., by the SRC (www.src.org) and by NSERC (www.nserc.ca).

ICCAD'06, November 5-9, 2006, San Jose, CA

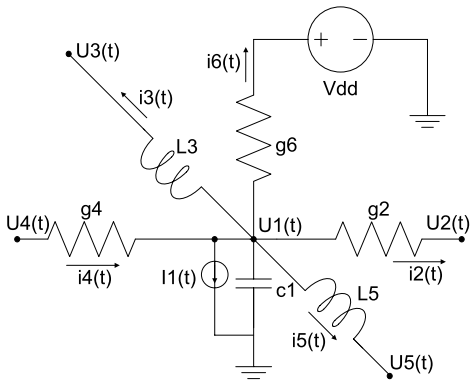

Figure 1: Typical power grid node and neighbors.

aware, constraint-based grid verification is three-fold: first, we show a method to find the "exact" worst-case voltage drop at every node on the grid , then we present an algorithm that computes "bounds" on the maximum and minimum worst-case drops on every node on the grid (section 5). Finally, we present a conservative criterion, which we label the "safety check", to efficiently check grid safety (section 6).

The paper is organized as follows. In section 2, an RLC model of the power grid is adopted to model the power grid and package connection, and the system equations are derived from the equations of general RLC circuits. In section 3, we show the time-discretized versions of these system equations. In section 4, we briefly describe the notion of current constraints, and our first two solution approaches that find the worst-case voltage drop are discussed and experimentally verified in section 5. Section 6 then discusses the grid safety check method and shows relevant experimental results. Section 7 provides a discussion of the impact of this approach on the ability to verify specific areas of concern, and section 8 gives some concluding remarks.

2. THE POWER GRID RLC MODEL

In this paper, we consider an RLC model of the power grid, where each branch is represented either by a resistor or by an inductor. There exists also a capacitor from every node to ground, and some nodes have ideal current sources (to ground) representing the currents drawn by the underlying logic circuitry. Other nodes may be connected to ideal voltage sources (assuming flip-chip technology, we will refer to an ideal supply voltage source as a C4, with the understanding that any inductance that is part of a true C4 pad structure has already been modeled and included in the grid description). In the derivation of the system equations, we will use the state-variable approach to network analysis [2]. This approach provides the necessary set of equations to solve a general RLC circuit, using capacitor voltages and branch currents as the set of independent state variables.

Let the power grid consist of $n+p$ nodes, where nodes $1, 2, \ldots, n$ have no voltage sources attached, and the remaining nodes $(n+1), (n+2), \ldots, (n+p)$ are the nodes where the p voltage sources are connected. 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 also assume 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. Let $u_k(t)$ be the voltage at every node k, $k = 1, \ldots, n$ and let $\mathbf{u}$(t) be the vector of all $u_k(t)$ voltage signals. Moreover, let $i_l(t)$ represent the branch currents where $l = 1, \ldots, m$, and let $\mathbf{i}$(t) be the vector of all branch currents. Notice that m represents the number of branches in the grid. The time-domain equations that describe the circuit can be derived by applying Kirchoff's Current Law (KCL) at every node k, where $k = 1, 2, \ldots, n$. For example, applying KCL at node 1 in Fig. 1 and rearranging the variables leads to:

$$i_2(t) + i_3(t) - i_4(t) + i_5(t) + i_6(t) + c_1 \frac{du_1(t)}{dt} = -I_1(t) \quad (1)$$

In general, applying KCL at every non-C4 node of the grid, we get:

$$\mathbf{C}\mathbf{u}'(t) + \mathbf{M}\mathbf{i}(t) = -\mathbf{I}(t) \quad (2)$$

where $\mathbf{C}$ is an $n \times n$ diagonal matrix of node capacitances, $\mathbf{M}$ is an $n \times m$ incidence matrix whose elements are either ± 1 or 0, as in [2]. The term ± 1 occurs in location m_{ij} of the matrix when node i is connected to the j^{th} branch, else a 0 occurs. The sign of the non-zero terms depends on the positive direction of current in the branch and on the node under consideration. If the current assignment is away from the node, then the sign is positive, else it is negative.

If we define $v_k(t) = V_{dd} - u_k(t)$ to be the voltage drop at node k and $\mathbf{v}(t)$ as the vector of all voltage drops. This gives our first system equation as:

$$\boxed{\mathbf{C}\mathbf{v}'(t) - \mathbf{M}\mathbf{i}(t) = \mathbf{I}(t)} \quad (3)$$

Notice that in (3) we do not take into account the relationship between the branch current and the branch voltage. It remains to model this relationship. The necessary equations were derived in [2] and are simply stated here. Notice that these equations will be applied to all branches. So, the number of equations will be m. Relating all branch currents to the voltage drop across the respective branches we get the second system equation:

$$\boxed{\mathbf{M^T}\mathbf{v}(t) + \mathbf{R}\mathbf{i}(t) + \mathbf{L}\mathbf{i}'(t) = \mathbf{0}} \quad (4)$$

where $\mathbf{R}$ is an $m \times m$ diagonal matrix with resistance values in diagonal entries corresponding to resistive branches and zeros elsewhere, and $\mathbf{L}$ is an $m \times m$ diagonal matrix with inductance values in diagonal entries corresponding to inductive branches and zeros elsewhere. Notice that the matrix multiplying the voltage vector is the transpose of the incidence matrix [2]. The pair of equations (3) and (4) represent the complete behavior of the power grid, as a dynamical system.

3. TIME DISCRETIZATION

We can write a discrete-time version of the system equations, by considering that, for small $\Delta t > 0$, the derivative of a function $x(t)$ can be approximated by:

$$x'(t) \approx \frac{x(t) - x(t - \Delta t)}{\Delta t}$$

Applying this to $\mathbf{v}'(t)$ in (3) leads to:

$$\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t) - \mathbf{M}\mathbf{i}(t) = \mathbf{I}(t) + \frac{\mathbf{C}}{\Delta t}\mathbf{v}(t - \Delta t) \quad (5)$$

and applying the same to $\mathbf{i}'(t)$ in (4) leads to:

$$\mathbf{M^T}\mathbf{v}(t) + \left(\mathbf{R} + \frac{\mathbf{L}}{\Delta t}\right)\mathbf{i}(t) = \frac{\mathbf{L}}{\Delta t}\mathbf{i}(t - \Delta t) \quad (6)$$

Now, (5) and (6) represent the system of equations which will be used to verify the voltage drop across the grid. The equation variables are the node voltages and the branch currents. These equations relate the time t voltage and current variables to those at time $t - \Delta t$.

4. CURRENT CONSTRAINTS

As in [7], we will adopt the notion of an incomplete current specification which is refered to as *current constraints*. Two types of constraints are defined: *local constraints* and *global constraints*. For both constraints, we will assume that $\mathbf{I}(t) \geq 0$ so that currents flow from the grid into the underlying circuit. The first type of

constraints are upper bounds on the individual current sources. They can be expressed as:

$$\mathbf{0} \leq \mathbf{I}(t) \leq \mathbf{I_L}, \forall t \geq 0 \quad (7)$$

These constraints are defined for every node of the grid, so that a node with no current source connected would have a zero $\mathbf{I_L}$ component. Note that $\mathbf{I_L}$ is a vector of constant current values, and $\mathbf{I}(t)$ is the vector of unknown current waveforms that are drawn by the underlying logic. Using local constraints alone is pessimistic because it is very unlikely that various circuit blocks would draw their maximum currents at the same time. So, it is useful to express constraints related to sub-groups of current sources. This type of constraints are called global constraints. They represent the total power dissipation of a group of current sources. Assuming we have a total of κ global constraints, they can be expressed in matrix form as:

$$\mathbf{0} \leq \mathbf{SI}(t) \leq \mathbf{I_G}, \forall t \geq 0 \quad (8)$$

where $\mathbf{S}$ is an $\kappa \times n$ matrix that consists only of 0s and 1s which indicate which current sources are in each constraint. Note here that $\mathbf{I_G}$ is a vector of constant current values, and $\mathbf{SI}(t)$ is a vector of sums of current waveforms that are drawn by the underlying logic.

To reiterate, current constraints capture the uncertainty about the circuit currents arising from both unknown circuit behaviors and the fact that one is uncertain about circuit details early in the design flow. Our intention is to verify that the grid is safe (i.e., its voltages remain within bounds), under all unknown transient current waveforms which satisfy these constraints.

An important question is how the constraints are to be obtained in practice. If the block, represented by a certain current source, is pre-existing and small, one may simply simulate the block with a circuit simulator and find its peak current. If the block is pre-existing but large, one may be only able to learn about some of its behaviors by simulation and would need to supplement that with previous design expertise about that design block. For example, if its power dissipation is known to be below a certain peak, then that leads to a bound on its current. If the block is not pre-existing, but its functionality is known, one would need to rely on engineering judgement and designer expertise to generate the constraints. If early in the design flow nothing is known about this block, not even its detailed functionality, one typically is able to come up with an area budget for it. From that, and from the projected power density (Watts per unit area) for the target process technology, one can generate a constraint. The bottom line: *something* is typically known about that block, which with good engineering judgement can be formulated into constraints. Many design groups today design the power grid using simple spreadsheet applications to capture engineering judgement and budgets. The proposed constraint-based approach is a more scientific and reliable approach for doing the same, but it must also depend on good designer expertise and judgement. After all, if truly nothing is known about the currents, then the grid simply cannot be verified.

Combining local and global constraints we get:

$$\mathbf{0} \leq \mathbf{UI}(t) \leq \mathbf{I_m}, \forall t \geq 0 \quad (9)$$

where $\mathbf{U}$ is an $(n+\kappa) \times n$ matrix. The upper $n \times n$ part of $\mathbf{U}$ is an identity matrix that captures the local constraints. The second $\kappa \times n$ part of $\mathbf{U}$ captures the global constraints.

5. WORST-CASE VOLTAGE DROP

We now present the approach for finding the worst-case voltage drop at every node given the set of current constraints, using the time-discretized set of equations (5) and (6). Recall that inductive effects induce voltage overshoots (voltage increases over V_{dd}) and undershoots (voltage droops below V_{dd}). Therefore, checking for voltage integrity violations on any given node (*i.e.*, checking whether its voltage remains within its safety interval), involves maximizing the voltage drop to check for maximum undershoot and minimizing the voltage drop (where the minima will be negative values) to check for maximum overshoot. Section 5.1 formulates and solves the exact worst-case problem considering overshoots and undershoots. This approach will be seen to be too expensive; therefore, in section 5.2 we present an efficient technique for finding an upper bound on the maximum worst-case voltage drop and a lower bound on the minimum worst-case voltage drop on every node. Finally, section 5.3 shows experimental results to corroborate these approaches.

A starting point for the whole analysis is to assume that the current sources are zero for all $t \leq 0$, so that the grid is safe and has zero voltage drop at time 0, and to then examine how the system evolves over time for $t \geq 0$. As will be seen, this assumption does not affect the generality of the solution. It is only a way to start the algorithm, but then the algorithm seeks the steady state behavior, once the effect of this initial condition has died down.

5.1 Exact worst-case

To find the exact worst case voltage drop (minimum and maximum) given our local and global constraints defined in (9), we combine (5) and (9) to get:

$$0 \leq \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t) - \mathbf{UMi}(t) - \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t-\Delta t) \leq \mathbf{I_m} \quad (10)$$

This step is possible only because the matrix $\mathbf{U}$ consists of only non-negative (0, 1) entries. Consider a sequence of k time steps Δt starting at time 0, then we can write (6) and (10) at any time step $t - q\Delta t$, $q = 0, \ldots (k-1)$. For $q = 1$, we get:

$$\mathbf{M^T}\mathbf{v}(t-\Delta t) + \left(\mathbf{R} + \frac{\mathbf{L}}{\Delta t}\right)\mathbf{i}(t-\Delta t) = \frac{\mathbf{L}}{\Delta t}\mathbf{i}(t-2\Delta t) \quad (11)$$

$$0 \leq \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t-\Delta t) - \mathbf{UMi}(t-\Delta t) - \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t-2\Delta t) \leq \mathbf{I_m} \quad (12)$$

We can write a similar pair of equations for $q = 2, 3, \ldots$ and, for $q = (k-1)$, we get:

$$\mathbf{M^T}\mathbf{v}(t-(k-1)\Delta t) + \left(\mathbf{R} + \frac{\mathbf{L}}{\Delta t}\right)\mathbf{i}(t-(k-1)\Delta t) = 0 \quad (13)$$

$$0 \leq \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t-(k-1)\Delta t) - \mathbf{UMi}(t-(k-1)\Delta t) \leq \mathbf{I_m} \quad (14)$$

since $\mathbf{v}(t-k\Delta t) = \mathbf{v}(0) = 0$ and $\mathbf{i}(t-k\Delta t) = \mathbf{i}(0) = 0$.

The above set of equations captures the set of all feasible voltage waveforms over the interval $[0, k\Delta t]$. Therefore, in order to find the worst-case voltage drop at a node at time $t = k\Delta t$, we need to consider all the above equation pairs from time $t = 0$ to time $t = k\Delta t$. In order to find the maximum/minimum voltage drop at some node i on the grid, we have to perform the following optimization problems:

Maximize/Minimize: $v_i(t)$
Subject to: for $q = 0, \ldots, (k-1)$

$$\mathbf{M^T}\mathbf{v}(t-q\Delta t) + \left(\mathbf{R} + \frac{\mathbf{L}}{\Delta t}\right)\mathbf{i}(t-q\Delta t) = \frac{\mathbf{L}}{\Delta t}\mathbf{i}(t-(q+1)\Delta t) \quad (15)$$

$$0 \leq \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t-q\Delta t) - \mathbf{UMi}(t-q\Delta t) - \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t-(q+1)\Delta t) \leq \mathbf{I_m}$$

Clearly, the two optimization problems above are linear programs (LP). Notice that the number of constraints is multiplied by the number of time steps, which can lead to a potentially very large problem.

Recall, the grid is considered initially safe for all $t \leq 0$. Now, we examine how the system evolves over time for $t > 0$. Since the optimization problem at time step $t - \Delta t$ is a subset of the

Table 1: Test-Case power grids and results of the exact worst-case optimization problems

Power Grid				Drop Statistics (% *of* V_{dd})		Execution Cost
Circuit	Nodes	C4s	Sources	Min	Max	Time
G1	95	15	38	-12.03	24.67	5.29 min.
G2	158	11	48	-1.49	20.74	15.43 min.
G3	275	41	74	-15.23	31.74	59.01 min.
G4	460	24	150	-1.33	36.96	3.56 h.
G5	492	41	180	-0.86	12.89	3.49 h.
G6	692	37	176	-3.18	7.78	11.06 h.
G8	913	48	383	-1.66	40.41	21.33 h.
G7	973	166	215	-9.13	17.58	21.70 h.
G9	1163	62	190	-0.56	6.92	26.68 h.

problem at time step t, it is easy to see that the exact maximum worst-case voltage drop at any node is a non-decreasing function of time over $t \geq 0$. Similarly, the exact minimum worst-case voltage drop is a non-increasing function of time over $t \geq 0$. Since the grid is a bounded-input bounded-output (BIBO) system, the exact maximum and minimum worst-case voltage drops are bounded and, being monotone, they must converge to certain "steady state" values.

Therefore, a procedure to compute the exact worst-case voltage drop at any node can be described as follows: we start with $k = 1$, and gradually increase k while solving the optimization problem (15) for each k, until we get the same maximal/minimal voltage drop value at a given node for two consecutive time steps, within some user-specified tolerance ϵ. This would mean that the steady state is reached and the solution represents the maximum/minimum worst-case voltage drop at node i under the current constraints.

As indicated previously, it is expected that this exact worst-case approach would be expensive. Indeed, the analysis for a grid with only 95 nodes can take over 10 hours to complete. Nevertheless, the exact worst-case approach is useful as an accuracy benchmark, against which the following more efficient approaches can be tested.

5.2 Upper and Lower Bounds

We define a voltage vector $\mathbf{v}$(t) to be *feasible* if there exist current source waveforms (with $\mathbf{I}(t) = 0, \forall t \leq 0$) that satisfy the current constraints and which can cause the grid to realize the voltage values in $\mathbf{v}(t)$ at time t (starting as always with $\mathbf{v}(t) = 0, \forall t \leq 0$). We define $\mathcal{V}_f(t)$ to be the set of all feasible voltage vectors at time t. We also introduce two voltage vectors $\mathbf{v}_{max}(t)$ and $\mathbf{v}_{min}(t)$, defined by:

$$\forall i, \quad v_{max,i}(t) = \max_{\mathbf{v}(t) \in \mathcal{V}_f(t)} v_i(t) \tag{16}$$

$$\forall i, \quad v_{min,i}(t) = \min_{\mathbf{v}(t) \in \mathcal{V}_f(t)} v_i(t) \tag{17}$$

Notice that $v_{max,i}(t)$ and $v_{min,i}(t)$ are in fact the solutions of the exact worst-case problem (maximization and minimization) described in the preceding section. Notice also that $\mathbf{v}_{max}(t)$ and $\mathbf{v}_{min}(t)$ may not be feasible voltage vectors, and that finding them for every t, especially for $t \to \infty$ can be very expensive. Therefore, we will propose an algorithm by which two vectors $\mathbf{v}_{up}(t)$ and $\mathbf{v}_{dn}(t)$ are computed such that $\mathbf{v}_{up}(t) \geq \mathbf{v}_{max}(t)$ and $\mathbf{v}_{dn}(t) \leq \mathbf{v}_{min}(t)$, $\forall t$.

From (6) and (10), the space of voltages allowed by the current constraints can be expressed as:

$$\mathbf{M^T v}(t) + \left(\mathbf{R} + \frac{\mathbf{L}}{\Delta t}\right)\mathbf{i}(t) = \frac{\mathbf{L}}{\Delta t}\mathbf{i}(t - \Delta t) \tag{18}$$

$$0 \leq \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t) - \mathbf{UMi}(t) - \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t - \Delta t) \leq \mathbf{I_m} \tag{19}$$

Suppose we know that, at a certain time $t - \Delta t$, the voltage drop on the grid satisfies:

$$\mathbf{v}_{dn}(t - \Delta t) \leq \mathbf{v}(t - \Delta t) \leq \mathbf{v}_{up}(t - \Delta t) \tag{20}$$

One would have hoped that, as was done in [7] for the *RC* case, one can develop an algorithm around (18), (19) and (20) that gives new bounds at time t: $\mathbf{v}_{dn}(t)$ and $\mathbf{v}_{up}(t)$. Unfortunately, it is not quite that simple in the *RLC* case. Instead, we have found it useful to extend the analysis to include two time-steps. Specifically, as a first step, one can augment (18) and (19) with the re-application of (19) at $t - \Delta t$, as in (24) below. Secondly, we can write Ohm's law for all resistive branches, at time $t - \Delta t$, in the form of (23) below, where $\mathbf{M_R}$ is a version of the incidence matrix in which only resistive branches have non-zero (± 1) entries. As a result, the space of feasible voltages $\mathbf{v}(t)$ can be expressed as:

$$\mathbf{M^T v}(t) + \left(\mathbf{R} + \frac{\mathbf{L}}{\Delta t}\right)\mathbf{i}(t) = \frac{\mathbf{L}}{\Delta t}\mathbf{i}(t - \Delta t) \tag{21}$$

$$0 \leq \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t) - \mathbf{UMi}(t) - \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t - \Delta t) \leq \mathbf{I_m} \tag{22}$$

$$\mathbf{M_R^T v}(t - \Delta t) + \mathbf{Ri}(t - \Delta t) = 0 \tag{23}$$

$$0 \leq \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t - \Delta t) - \mathbf{UMi}(t - \Delta t) - \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t - 2\Delta t) \leq \mathbf{I_m} \tag{24}$$

The above equations and inequalities, along with the following bounds at two previous time steps:

$$\mathbf{v}_{dn}(t - \Delta t) \leq \mathbf{v}(t - \Delta t) \leq \mathbf{v}_{up}(t - \Delta t) \tag{25}$$

$$\mathbf{v}_{dn}(t - 2\Delta t) \leq \mathbf{v}(t - 2\Delta t) \leq \mathbf{v}_{up}(t - 2\Delta t) \tag{26}$$

now form the basis for a successful algorithm for obtaining the bounds at time t, $\mathbf{v}_{dn}(t)$ and $\mathbf{v}_{up}(t)$, as will be described and proven below. The algorithm is as follows:

Step 1. For $t = -\Delta t, 0:$ $\mathbf{v}_{up}(t) = 0$ and $\mathbf{v}_{dn}(t) = 0$

Step 2. $t = t + \Delta t$

Step 3. For $i = 1, \ldots, n:$

(a) Maximize $v_i(t)$ subject to (21-26) and, if v^* is the resulting (maximum) value, set $v_{up,i}(t) = v^*$.

(b) Minimize $v_i(t)$ subject to (21-26) and, if v^* is the resulting (minimum) value, set $v_{dn,i}(t) = v^*$.

Step 4. If $\mathbf{v}_{up}(t)$ and $\mathbf{v}_{dn}(t)$ have not converged, go to Step 2, else stop.

This algorithm is obviously much faster than the exact approach as it involves optimization across two time steps only. The following claim proves the correctness of the algorithm.

CLAIM 1. *If $\mathbf{v}_{up}(t)$ and $\mathbf{v}_{dn}(t)$ are generated according to the above algorithm, then $\mathbf{v}_{up}(t) \geq \mathbf{v}_{max}(t)$ and $\mathbf{v}_{dn}(t) \leq \mathbf{v}_{min}(t)$, $\forall t \geq 0$.*

PROOF. Since the base cases $\mathbf{v}_{up}(t) \geq \mathbf{v}_{max}(t)$ and $\mathbf{v}_{dn}(t) \leq \mathbf{v}_{min}(t)$ are trivially satisfied (as an equality) at times $t = -\Delta t$

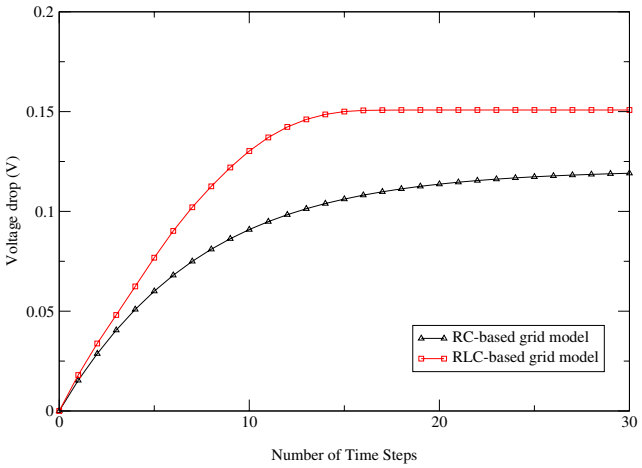

Figure 2: Maximum Voltage drop Comparison for the same node using the *RC* and the *RLC* models

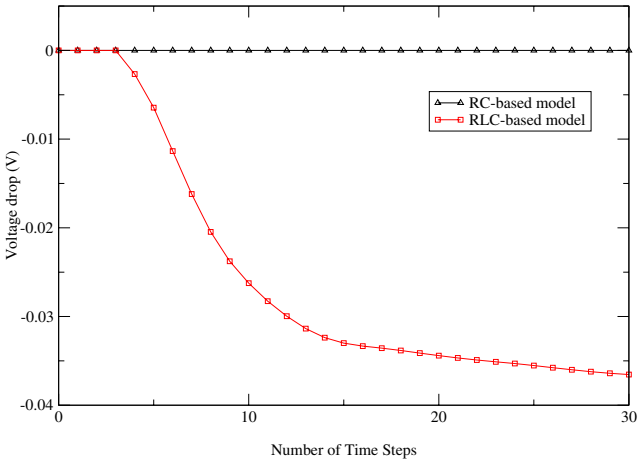

Figure 3: Minimum Voltage drop Comparison for the same node using the *RC* and the *RLC* models

and $t = 0$, then the claim is true by induction if we prove the following:

$$\left.\begin{array}{l}\mathbf{v}_{up}(t-\Delta t) \geq \mathbf{v}_{max}(t-\Delta t)\\ \mathbf{v}_{dn}(t-\Delta t) \leq \mathbf{v}_{min}(t-\Delta t)\\ \mathbf{v}_{up}(t-2\Delta t) \geq \mathbf{v}_{max}(t-2\Delta t)\\ \mathbf{v}_{dn}(t-2\Delta t) \leq \mathbf{v}_{min}(t-2\Delta t)\end{array}\right\} \Rightarrow \begin{array}{l}\mathbf{v}_{up}(t) \geq \mathbf{v}_{max}(t)\\ \mathbf{v}_{dn}(t) \leq \mathbf{v}_{min}(t)\end{array} \quad (27)$$

Define $\mathcal{V}_b(t)$ to be the set of all vectors $\mathbf{v}(t)$ that satisfy equations (21-26). Notice that $\mathcal{V}_b(t)$ is the solution space of the core optimization problems in our algorithm, which may be summarized as:

$$\forall i, \quad v_{up,i}(t) = \max_{\mathbf{v}(t)\in\mathcal{V}_b(t)} v_i(t) \quad (28)$$

$$\forall i, \quad v_{dn,i}(t) = \min_{\mathbf{v}(t)\in\mathcal{V}_b(t)} v_i(t) \quad (29)$$

Consider any $\mathbf{v}(t) \in \mathcal{V}_f(t)$, then there exist other feasible voltage vectors $\mathbf{v}(t-\Delta t)$ and $\mathbf{v}(t-2\Delta t)$ such that $\mathbf{v}(t), \mathbf{v}(t-\Delta t)$, and $\mathbf{v}(t-2\Delta t)$ satisfy equations (21-24). For $k = 1, 2$, and since $\mathbf{v}_{min}(t-k\Delta t) \leq \mathbf{v}(t-k\Delta t) \leq \mathbf{v}_{max}(t-k\Delta t)$ due to (16) and (17), and assuming the left hand side of (27) is true, then (25) and (26) are true. So $\mathbf{v}(t)$ satisfies equations (21-26). Thus, $\mathbf{v}(t) \in \mathcal{V}_b(t)$. Since this is true for *any* $\mathbf{v}(t) \in \mathcal{V}_f(t)$, then it follows that:

$$\mathcal{V}_f(t) \subset \mathcal{V}_b(t) \quad (30)$$

Given (16) and (17) and given (28) and (29), it then follows that $\mathbf{v}_{up}(t) \geq \mathbf{v}_{max}(t)$ because maximizing over a superset of $\mathcal{V}_f(t)$ must give a result that is at least as large as maximizing over $\mathcal{V}_f(t)$. Similarly, it follows that $\mathbf{v}_{dn}(t) \leq \mathbf{v}_{min}(t)$. This completes the proof. □

5.3 Experimental Results

We solved the LPs associated with the above (upper/lower-bound) algorithm using the Primal-Dual Interior Point Method algorithm [8], implemented in the optimization package PCx [3]. A number of tests were conducted on a set of randomly-generated test-case power grids, using a similar approach to what was described in [4], and computations were carried out on a 1 GHz SUN machine with 4 GB of memory. The grids generated included user-defined RLC models for the package-grid interconnections.

To demonstrate the disparity between verification results from an *RC* versus an *RLC* grid model, we show sample results on a 58-node grid, an instance of which is based on the *RLC* model and an analogous instance of which was obtained by setting all inductances to zero, leading to an *RC* model. We optimized the voltage drop for an arbitrary grid node in each case. Fig. 2 shows

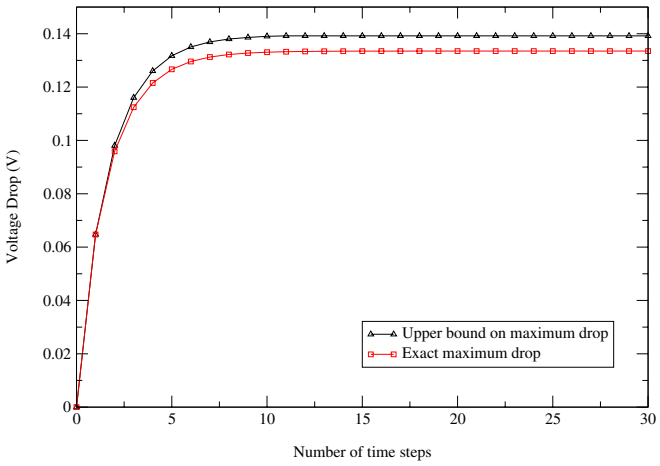

Figure 4: Maximum voltage drop comparisons for a node in G1

the exact maximum voltage drop on the node under both models. It is clear from the figure that the *RC* model underestimates the maximum undershoot occuring on the grid (30% in this case). Thus a grid that is considered safe under the *RC* model might be unsafe under the *RLC* model. Fig. 3 shows the exact minimum voltage drop on the same node under both models, and therefore captures any voltage drop violations due to overshoots. Notice that in the *RC* model is completely oblivious to any overshoot on the grid.

Table 1 shows several test grids with the number of nodes, the number of voltage sources (C4s), and the number of current sources indicated in each case. For each grid, a single value is chosen as the local upper-bound current constraint and a set of global current constraints are specified. The table gives the results of running the algorithm in section 5.2. It shows the minimum and maximum voltage drop for each grid, normalized as a % of V_{dd}. For each grid, the table also indicates the total CPU time required for performing the voltage optimizations (min/max) on all the grid nodes. Notice how increasing the number of voltage suplies (C4s) for grids of comparable size will decrease the maximum voltage drop. Notice also that all the grids being tested witnessed overshoots (negative voltage drop), a phenomena that never occurs in the *RC*-based model. Figs. 4 and 5 show the maximum and minimum voltage drop versus time for a node in G1. These figues shows that the bounds of the algorithm of section 5.2 are fairly tight. The exact solution for G1 took around 10 hours while the upper/lower bound algorithm took around 5

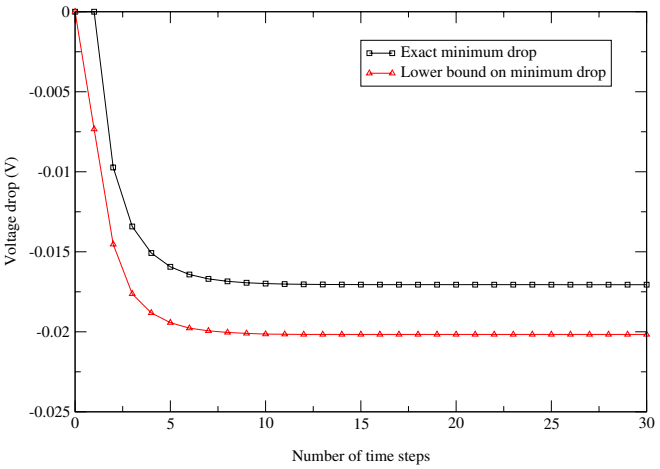

Figure 5: Minimum voltage drop comparisons for a node in G1

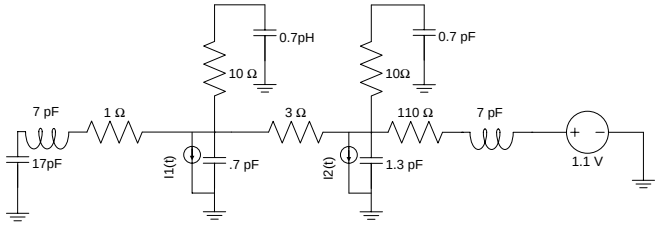

Figure 6: A 5-node grid

minutes. Although faster, this bounds based approach is still too slow for large grids. The next section describes a more efficient conservative approach for checking grid safety.

6. GRID SAFETY CHECK

We now present our third solution approach where we check if the grid is safe, under all transient currents that satisfy the current constraints, without having to actually find the worst-case voltage drop at every node. Therefore, it is cheaper than both methods of section 5 and it provides a fast answer under specific given voltage drop thresholds.

6.1 Conditional Safety

Suppose two thresholds for grid safety are given, so that the grid is considered safe when all its node voltage drops $v_i(t)$ are between $\mathbf{v_{max}}$ and $\mathbf{v_{min}}$, where $\mathbf{v_{max}}$ and $\mathbf{v_{min}}$ are $(n \times 1)$ vectors. Assume also that at time $(t-2\Delta t)$ and $(t-\Delta t)$ the grid *was* safe, so that:

$$\mathbf{v_{min}} \leq \mathbf{v}(t-\Delta t) \leq \mathbf{v_{max}} \quad (31)$$

$$\mathbf{v_{min}} \leq \mathbf{v}(t-2\Delta t) \leq \mathbf{v_{max}} \quad (32)$$

where, as before, $\mathbf{v}(\cdot)$ is the vector of all node voltage drops. We can now formulate the following conditional safety problem:

PROBLEM 1. *If the grid is safe at times $(t-\Delta t)$ and $(t-2\Delta t)$, check if it is safe at time t.*

To solve this "safety check problem", it is sufficient to maximize and minimize every component of $\mathbf{v}(t)$, subject to some constraints to be stated below. Doing both optimizations (maximization and minimization) is necessary to check if either $\mathbf{v_{max}}$ or $\mathbf{v_{min}}$ is violated. Based on (21-24), the optimization algorithm can be written as:

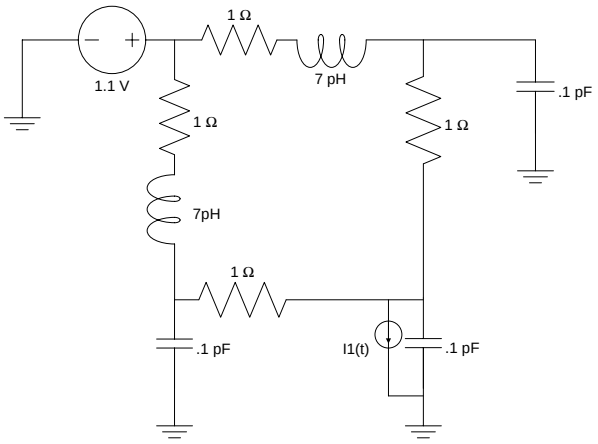

Figure 7: A 3-node grid

Step 1. For $i = 1, \ldots, n$:
While $v_i(t) \in [\mathbf{v_{min}}, \mathbf{v_{max}}]$, Maximize/Minimize $v_i(t)$
Else (if $v_i \notin [\mathbf{v_{min}}, \mathbf{v_{max}}]$) declare grid unsafe & abort
Subject to:

$$\mathbf{v_{min}} \leq \mathbf{v}(t-\Delta t) \leq \mathbf{v_{max}}$$

$$\mathbf{v_{min}} \leq \mathbf{v}(t-2\Delta t) \leq \mathbf{v_{max}}$$

$$\mathbf{M^T v}(t) + \left(\mathbf{R} + \frac{\mathbf{L}}{\Delta t}\right)\mathbf{i}(t) = \frac{\mathbf{L}}{\Delta t}\mathbf{i}(t-\Delta t)$$

$$0 \leq \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t) - \mathbf{UMi}(t) - \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t-\Delta t) \leq \mathbf{I_m} \quad (33)$$

$$\mathbf{M_R^T v}(t-\Delta t) + \mathbf{Ri}(t-\Delta t) = 0$$

$$0 \leq \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t-\Delta t) - \mathbf{UMi}(t-\Delta t) - \mathbf{U}\frac{\mathbf{C}}{\Delta t}\mathbf{v}(t-2\Delta t) \leq \mathbf{I_m}$$

Step 2. Declare grid safe and stop.

Notice that the size of this problem is much smaller than the previous optimization formulations, because it is performed at two time steps only and does not involve stepping repeatedly through time. If the grid is safe at two previous time steps, then we check if it is safe at time t. If yes, then since any grid can be assumed to have been safe at some pair of time points in the distant past (similar to the argument in [7]), then by induction the grid is safe for all future times. If the grid turned out to be unsafe using the above algorithm, then the result is inconclusive; some node voltages may or may not be violated.

6.2 Experimental Results

We generated random test grids and used the same optimization package as before to solve the LPs associated with the optimization problems (33). Table 2 shows, for each of the generated grids, the number of nodes that violate the safety interval in the case of an RC model and when an RLC grid model is considered. This safety check was done using a maximum voltage drop threshold of 10% of V_{dd} and a minimum voltage drop threshold of -5% of V_{dd}. Test results show that the RC-model significantly underestimates the voltage deviations of the grid for all the test cases under consideration.

Notice also how increasing the number of voltage sources on the grid decreases the number of unsafe nodes. Table 2 shows also that the safety check on the RLC model achieves a significant speed-up when compared to the RLC worst-case voltage drop methods of section 5. For instance, finding the worst-case voltage drop bounds for a 1163-node grid takes around 26 hours while

Table 2: Safety check results comparing the RC-based model and the RLC-based model

Power Grid				Voltage Drop Threshold $\%V_{dd}$ $V_{max} = 10\%$ and $V_{min} = -5\%$		Execution Cost	
Circuit	Nodes	C4s	Sources	RC model	RLC model	RC Time	RLC Time
P1	34	15	20	0	6	2.68 s.	1.64 s.
P2	90	10	10	3	26	1.01 s.	1.41 s.
P3	204	21	33	29	76	29.61 s.	56.34 s.
P4	252	37	27	0	61	2.14 min.	57.1 s.
P6	450	34	15	4	180	3.02 min.	36.5 s.
P7	458	26	30	53	214	4.22 min.	1.13 min.
P8	666	63	89	0	158	24.53 min.	7.28 min.
P9	870	154	90	27	225	57.9 min.	32.46 min.
P10	1039	50	80	72	380	1.01 h.	41.29 min.
P11	1218	151	100	20	237	1.21 h.	1.07 h.

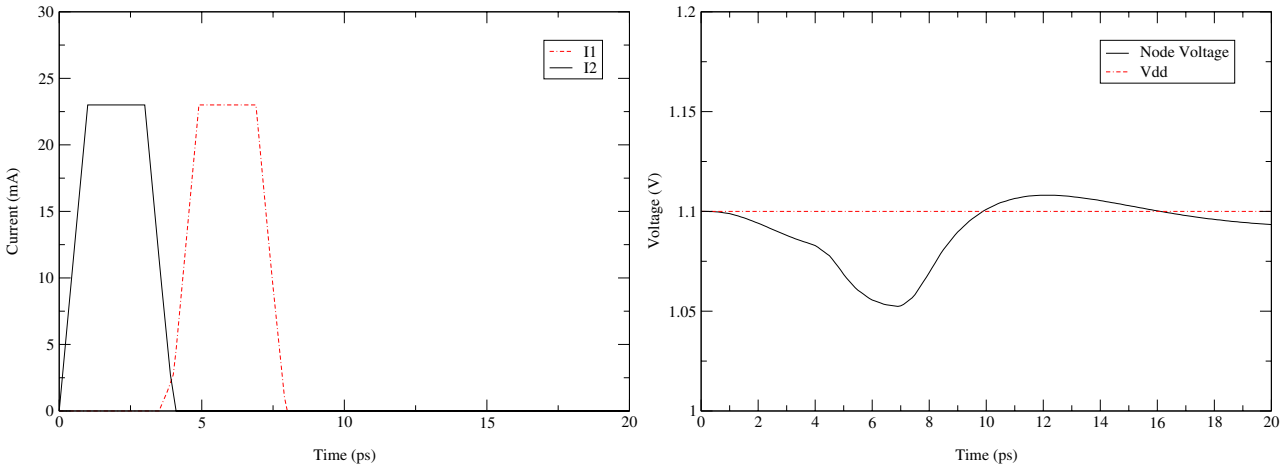

Figure 8: Current Configuration leading to maximum Overshoots

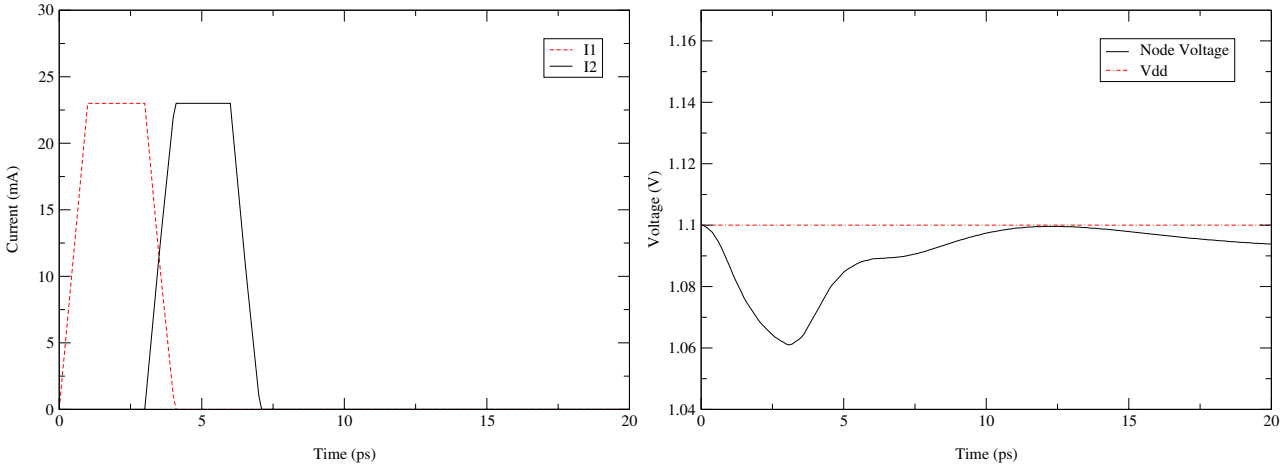

Figure 9: Current Configuration resulting in no Voltage Overshoots

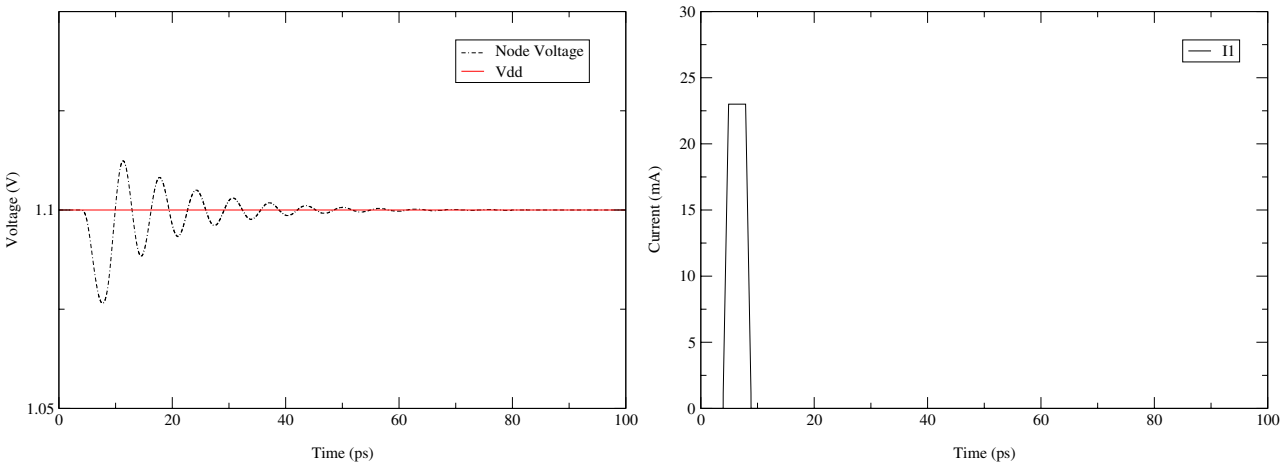

Figure 10: Voltage overshoots on a node, and the current causing it

checking the safety of a 1218-node grid requires about 1 hour. One might argue that such grids are small when compared to full-chip grids containing millions of nodes. Nevertheless, our approach is important for at least two main reasons: 1) it is the *first* approach to rigorously check the safety of an *RLC* grid in a truly vectorless approach - remember that we are checking the grid over *all transient currents* that satisfy the constraints, 2) coupled with a hierarchical modeling approach, our method can be applied either to parts of the grid, or to the top-level main feeder network of the grid. The ability to test the main feeder network early in the design flow is a major advantage of our technique. We believe that these techniques can lead to practical methods for early vectorless grid verification.

7. DISCUSSION

In this section, we will give some discussion to illuminate the impact of this work, and the importance of handling inductance. The discussion will focus on the two small grid circuits shown in Figs. 6 and 7.

One of the difficulties in analyzing portions of the grid around a C4 pad is that only certain temporal arrangements of the circuit currents may lead to resonance with the package inductance, while others do not. An example is shown in Fig. 8, which shows the result of an HSPICE simulation of the circuit in Fig. 6 under a specific pair of current waveforms assigned to the two current sources in the circuit. The simulation shows a distinct overshoot where the node voltage goes above V_{dd}. In contrast to this, Fig. 9 shows a simulation of the same circuit for a slightly different arrangement of the currents. There is no overshoot in this case. It is an expensive proposition in practice to simply use trial and error to debug a situation like this and look for overshoots. It may be feasible to do with just two current sources, but becomes impossible with even a few more sources. Instead, with our approach, a single-shot verification can uncover the fact that *there exists* a combination of feasible current waveforms which can cause an overshoot above some threshold.

The second circuit was shown to be safe using an RC safety check, as in [7]; however, HSPICE simulations of this circuit using a current waveform in the feasible current space showed that all three nodes witnessed overshoots. Fig. 10 shows the voltage on one of these nodes along with the current source. This result agreed with the RLC safety check that indicated all three nodes were unsafe. As a result, adopting an inductance-aware power grid model is essential to capture realistic grid voltage fluctuations.

8. CONCLUSION

Power grid verification has become an integral part of reliable chip design. Modeling the grid as an *RLC* network accurately captures maximum voltage drop as well as overshoots. Such effects were either under-estimated or not accounted for in previously proposed techniques for verification. In this work, we proposed an inductance-aware grid verification approach under current constraints. We modeled the grid as an *RLC* circuit while maintaining the grid formulation in a Linear Program (LP) form. Then we gave two algorithms that find the worst-case voltage drop at every node of the grid. Finally, we offered a conservative safety check algorithm to verify grid robustness.

9. REFERENCES

[1] R. Ahmadi and F. N. Najm. Timing analysis in presence of power supply and ground voltage varia tions. In *IEEE/ACM International Conference on Computer-Aided Design*, pages 176–183, San Jose, CA, November 9-13 2003.

[2] T. R. Bashkow. The A matrix, new network description. *IRE transactions on circuit Theory*, 4(3), September 1957.

[3] J. Czyzyk, S. Mehrotra, M. Wagner, and S. J. Wright. *PCx User Guide (Version 1.1)*. Optimization Technology Center, Northwestern University, November 1997.

[4] D. Kouroussis, I. Ferzli, and F. N. Najm. Incremental partitioning-based vectorless power grid verification. In *International Conference on Computer-Aided Design (ICCAD-05)*, pages 358–364.

[5] W. H. Lee, S. Pant, and D. Blaauw. Analysis and reduction of on-chip inductance effects in power supply grids. In *IEEE International Symposium on Quality Electronic Design*, pages 131–136, San Jose, CA, March 22-24 2004.

[6] A. Muramatsu, M. Hashimoto, and H. Onodera. Effects of on-chip inductance on power distribution grid. In *ACM International Symposium on Physical Design*, pages 63–69, San Francisco, CA, April 3-6 2005.

[7] M. Nizam, F. N. Najm, and A. Devgan. Power grid voltage integrity verification. In *International Symposium on Low Power Electronics and Design (ISLPED'05)*, pages 239–244, San Diego, CA, August 8-10 2005.

[8] R. Saigal. *Linear Programming: a Modern Integrated Analysis*. Kluwer Academic Publishers, 1995.

[9] N. Srivastava, X. Qi, and K. Banerjee. Impact of on-chip inductance on power distribution network design for nanometer scale integrated circuits. In *IEEE International Symposium on Quality Electronic Design*, pages 341–351, San Jose, CA, March 21-23 2005.

[10] Q. K. Zhu. *Power Distribution Network Design for VLSI*. John Wiley and Sons, New Jersey, 2004.