Fast Analysis and Optimization of Power/Ground Networks[*]

Haihua Su   Kaushik H. Gala   Sachin S. Sapatnekar
Department of Electrical and Computer Engineering
University of Minnesota, 200 Union Street SE,
Minneapolis 55455, USA

Abstract

This paper presents an efficient method for optimizing power/ground (P/G) networks by widening wires and adding decoupling capacitors (decaps). It proposes a structured skeleton that is intermediate to the conventional method that uses full meshes (which are hard to analyze efficiently), and tree-structured networks (which provide poor performance). As an example, we consider a P/G network structure modeled as an overlying mesh with underlying trees originating from the mesh, which eases the task of analysis with acceptable performance sacrifices. A fast and efficient event-driven P/G network simulator is proposed, which hierarchically simulates the P/G network with an adaptation of PRIMA to handle non-zero initial conditions. An adjoint network that incorporates the variable topology of the original P/G network, as elements switch in and out of the network, is constructed to calculate the transient adjoint sensitivity over multiple intervals. The gradients of the most critical node with respect to each wire width and decap are used by a sensitivity-based heuristic optimizer that minimizes a weighted sum of the wire and the decap area. Experimental results show that this procedure can be used to efficiently optimize large networks.

1. Introduction

With the rapid increases of signal frequency and reduction of feature sizes of high-speed electronic circuits, it is becoming more and more important to design and optimize P/G networks fast and accurately. Various algorithms and simplified device models for P/G networks have been explored in the past. Some of these techniques utilize the fact that most P/G networks tend to be tree-like structures so as to allow the use of path tracing algorithms for efficiency [SH90]; these techniques assume resistance-only models for the network. Other related work on optimizing P/G networks includes [SVR+94, TSL+99], which use techniques ranging from simulated annealing to the solution of a sequence of linear programs for wire widening, or [MK92], which optimizes the P/G network topology.

Most existing techniques have focused on methods that optimize a large and complex general mesh topology. In this work, we use a structured topology skeleton with a global mesh feeding local trees, as shown in Fig. 1. A similar method has been used in [SYSH98]. However, we emphasize that the approach can be modified to other topologies that are intermediate to the two extremes of full trees and full meshes: one such example is a global mesh that feeds smaller local meshes. The topology of the P/G network changes at the beginning of each switching event as new RC elements are added to the network or removed from it. Our task is to minimize the total P/G network cost, subject to the constraint that the maximum voltage drops of all the specified critical nodes are within certain values under the worst-case switching activities.

For the structure in Fig. 1, the task of analysis is rapidly performed using PRIMA [OCP98], a reduced order modeling technique that produces provably passive macromodels. The optimization of the objective function requires the computation of transient sensitivity for the specified critical nodes with respect to all the circuit elements, and we employ the adjoint method [PRV94]. Traditionally, transient sensitivity computation for a circuit with a fixed topology has been performed by a convolution between the forward-in-time voltage/current slope of the element (capacitor, resistor or inductor) in the original circuit and the backward-in-time voltage/current across the same element in the adjoint circuit, where the same fixed topology is chosen for the pair of circuits.

We present an extension of the adjoint network technique over multiple intervals for efficient sensitivity computation. Our sensitivity computation is coupled with an efficient PRIMA-based order reduction approach so that it can handle large-scale P/G circuits. A closed-form transient sensitivity expression is provided for a PRIMA approximation of a given order.
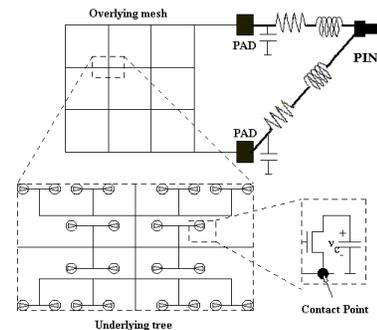

Fig. 1. A structured P/G bus topology

The following notation will be used throughout the paper. The voltages [currents] in the original network are denoted by v [i], while the voltages [currents] in the adjoint network are ψ [φ]. The symbols t [τ] denote the temporal variables in the original [adjoint] network.

2. Hierarchical analysis of the P/G network incorporating nonzero initial conditions

The procedures for analyzing the power and ground networks are symmetric, and a solution technique to optimizing either one can be extended to the other.

The interconnect in the P/G network is modeled as follows. Each wire on the mesh or tree structure is modeled as a set of connected segments under the π-model, with each segment modeled using lumped RLC parameters given by

$$R_s = \rho \, l_s \, / \, w_s \tag{1}$$
$$C_s = (\beta w_s + \alpha) \, l_s$$
$$L_s = \gamma l_s \, / \, w_s$$

where l_s and w_s are the length and the width of the segment, and the parameters ρ, β, α and γ are the sheet resistance per square, sheet capacitance per square, fringing capacitance per unit length and the inductance per square of the metal layer that is being used for routing the P/G network. Each package pin is modeled as an RLC branch connected to pads on the mesh.

The simulation is event-driven, proceeding one interval after another until the last, updating the corresponding switch states specified in the event list. The final state, i.e., the capacitor voltages and inductor currents at the end of each interval, constitutes the initial state for the next interval.

The entire system may be modeled as a linear system characterized by the equation

$$(G + sC)V(s) = J(s) \tag{2}$$

The objective is to reduce this system to a smaller one that captures the response of the system to the given set of inputs and the initial conditions in the system.

The hierarchical model reduction and simulation proceeds in three stages: first, all trees are reduced to an equivalent passive model. Next, the mesh is solved, along with these passive models to find all mesh voltages. Finally, these mesh voltages provide the voltage at the root of each tree and are used to solve each tree individually and independently. This hierarchical approach serves to reduce the amount of computation required during the analysis.

Reduction of the trees The MNA equation for each of the trees with initial conditions can be written as

$$(G_{Tree} + sC_{Tree})V_{Tree}(s) = [J_1(s) J_2(s)] \begin{bmatrix} I_{in}(s) \\ u(t-T) \end{bmatrix} \tag{3}$$

where $I_{in}(s)$ is the input excitation to the tree, which is the current injected from the mesh node into the tree from the root of the tree. An initial condition at time T on a capacitor C_i [inductor L_i] may be modeled as a voltage [current] source of value $V_{Ci}(T)$ [$L_iI_{Li}(T)$] in series [parallel] with a capacitor [inductor] with zero initial conditions. The vector $J_2(s)$ captures the initial conditions of the tree and has entries of the type $C_iV_{Ci}(T)$ and $L_iI_{Li}(T)$, where the multiplications by C_i and L_i correspond to conversions between Thevenin and Norton forms for ease of application for the formulation.

The PRIMA reduction procedure is applied to obtain a provably passive tree reduction. A RICE-like tree traversal [RGP91] computes the orthonormal basis **X** of the Krylov space, so that the procedure is extremely fast. The two-column right hand side matrix in Equation (3) tells us that two columns are added to **X** in each iteration and to obtain a reduction of order q, $\lceil q/2 \rceil$ iterations are required.

Solving the mesh Substituting the reduced order model of each tree can reduce the MNA equation (2) for the whole system to

$$(G_{Mesh} + sC_{Mesh})V_{Mesh}(s) = J_{Mesh}(s) \tag{4}$$

PRIMA is applied to (4) again to this reduced system to further reduce the system to a smaller order

$$(\tilde{G}_{Mesh} + s\tilde{C}_{Mesh})V_{Mesh}(s) = \tilde{J}_{Mesh}(s) \tag{5}$$

Since G_{Mesh} is sparse, sparse matrix technique can be used to compute the orthonormal basis X of the Krylov space. Since the overlying mesh is typically small in terms of the number of nodes and the order of the final system is small, the computational cost of this is also small.

Propagating waveforms down the trees The solution to equation (5) is used to set the mesh node voltages to the computed value. These values are used to recursively compute the voltage at each of the internal node in the local trees. The voltage at each node is computed as the sum of the zero input response and the zero initial condition response; note that the input for the tree is the voltage at the mesh node.

The transient response of each node voltage or branch current in the P/G net is thus found to have the following form:

$$g_1(t) = \sum_{i=1}^{q_k} r_i e^{\lambda_i t}, \quad 0 \le t \le T_f, \tag{6}$$

where q_k is the number of dominant poles for the node or branch, which is determined by the reduced order of each tree and the reduced order of the mesh matrix.

3. Adjoint sensitivity computation over multiple intervals

Adjoint sensitivity analysis is a standard technique for circuit optimization where the sensitivity of one output with respect to many parameter values is required [PRV94]. In adjoint sensitivity analysis, Tellegen's theorem is applied to a pair of circuits with the same topology by combining the branch currents and voltages at any two instants of time.

For our problem, we simulate the P/G network over the specified event list. At the beginning of each event, a set of switching activities occurs, with some RC elements switching out of the network and others switching in. This complicates the task of adjoint sensitivity computation since the topology changes for each interval. One contribution of this work is to extend adjoint analysis to this variable topology.

Similar derivation procedure of the transient adjoint sensitivity formula as in [PRV94] can be performed over multiple intervals. Due to space limitations, the detailed derivation is not discussed here, but only the final results are listed. Suppose there are a total of f events, and each event is lasting from t_{k-1} to t_k, $k=1$ to f, where $t_0=0$. The transient sensitivity formulas with respect to capacitor C, resistor R (inductor L) can be shown to be as follows:

$$\frac{\delta v^{(p)}(T_{peak})}{\delta C} = -\sum_{k=1}^{p} \int_{t_{k-1}}^{t_k} \psi_C^{(k)}(t_{k-1} + t_k - t)\dot{v}_C^{(k)}(t)dt \tag{7}$$

$$\frac{\delta v^{(p)}(T_{peak})}{\delta R} = \frac{\delta v^{(p)}(T_{peak})}{\delta L} = -\sum_{k=1}^{p} \int_{t_{k-1}}^{t_k} \varphi_{RL}^{(k)}(t_{k-1} + t_k - t)i_{RL}^{(k)}(t)dt \tag{8}$$

where $t_{p-1} < T_{peak} < t_p$. Notice that $\psi_C(\tau)$ and $\varphi_{RL}(\tau)$ are continuous over the period $t_f \ge \tau \ge t_f - T_{peak}$. In other words, $\psi_C(t)$ and $\varphi_{RL}(t)$ are continuous over the period $0 \le t \le T_{peak}$.

The simulation technique discussed in section 2 can be applied to analyze the adjoint P/G network in backward time. The event-driven simulation of the adjoint P/G network is performed in a backward order of the specified events, so that the topology of the adjoint network also changes in the reverse temporal order. The voltage $\psi(t)$ (for capacitors) and branch current $\varphi(t)$ (for resistors or inductors) also have the following form:

$$g_2(t) = \sum_{i=1}^{q_{ak}} r_{a_i} e^{\lambda_{a_i} t}, \quad 0 \le t \le T_{peak} \tag{9}$$

Transient sensitivity calculation is performed using expressions (7) and (8). For any of the element in the network, the $V(t)$ or $I(t)$ and $\psi(t)$ or $\varphi(t)$, of the form of expression (6) and (9), respectively, the transient sensitivity is represented by an expression similar to:

$$\frac{\delta V^{(p)}(T_{peak})}{\delta x} = -\sum_{k=1}^{p}\sum_{i=1}^{q_k}\sum_{j=1}^{q_{a_k}} \frac{r_i r_{a_j} \lambda_i}{\lambda_i + \lambda_{a_j}} e^{\lambda_{a_j}}\Big|_{t_{k-1}}^{t_k}, \qquad (10)$$

where V is the voltage response of the most critical node, and x may be any of the elements (capacitors, inductors or resistors) in the P/G network.

4. Heuristic optimization

The optimization technique used in this work is a sensitivity-based heuristic that is similar to the TILOS [FD85] algorithm, which is a greedy heuristic optimizer that changes the parameters that provide the "biggest bang for the buck."

The problem of optimizing a P/G network by varying wire widths can be formulated as

$$\text{Minimize} \qquad \text{Area} = \Sigma_i\, l_i w_i$$
$$\text{Subject to} \qquad \text{Max }(V_{drop}) \le V_{spec}$$

In the objective function, l_i represents the total length of a set of the P/G wire segments with width w_i. In each iteration, we first analyze the network to identify the most critical node, then determine the parameter of the network that this critical node is most sensitive to, and finally bump up the width of this parameter by a small amount, so that the most critical voltage drop is reduced.

In our method, we divide each wire in the mesh/trees into several π-segments, but model a set of adjacent wires as having the same width in order to reduce the network of optimization parameters. The gradients with respect to the area A_i of each set of N wires with width w_i is computed using the chain rule as follows:

$$\frac{\partial V(T_{peak})}{\partial A_i} = \frac{\partial V(T_{peak})}{\partial w_i}\frac{\partial w_i}{\partial A_i} = \frac{\partial V(T_{peak})}{\partial w_i}\frac{1}{\sum_{j=1}^{N} l_j} \qquad (11)$$

and

$$\frac{\partial V(T_{peak})}{\partial w_i} = \sum_{j=1}^{N}\left[\begin{array}{c}\frac{\partial V(T_{peak})}{\partial C_{j1}}\frac{\partial C_{j1}}{\partial w_j} + \frac{\partial V(T_{peak})}{\partial C_{j2}}\frac{\partial C_{j2}}{\partial w_j} \\ + \frac{\partial V(T_{peak})}{\partial R_j}\frac{\partial R_j}{\partial w_j} + \frac{\partial V(T_{peak})}{\partial L_j}\frac{\partial L_j}{\partial w_j}\end{array}\right] \qquad (12)$$

where the set of wires with width w_i consists of N wire segments; each of the segment j has resistance R_j, inductance L_j, and capacitance C_{j1}, C_{j2} at each terminal of the wire. From (1), it is easy to find $\partial R_j/\partial w_j$, $\partial L_j/\partial w_j$, $\partial C_{j1}/\partial w_j$ and $\partial C_{j2}/\partial w_j$.

The overall optimization procedure is as follows:

- Simulate the original P/G network using the hierarchical simulation method discussed in section 2 and determine the peak voltage V_{peak} and the time point T_{peak} where it occurs.
- Save the voltage approximants for the C's and the current waveforms for the R's and L's in the network.
- Simulate the adjoint network backward in time with zero initial conditions and save voltage/current waveforms for the adjoint network.
- Compute the voltage sensitivities using formula (10) and the voltage sensitivities with respect to A_i using (11) and (12).
- Bump up the width of the set of wires with maximum sensitivity by multiplying it with a small factor (<1.1).

- Repeat the above procedure until the maximum voltage drop of the whole network is within certain constraint.

The above procedure can be extended to include the optimization of decoupling capacitors. The objective of this optimization is to determine appropriate sizes of each wire and each decoupling capacitor for the minimum area overhead. Initially, decoupling capacitors with some small values are connected to some user-specified nodes in the P/G network. The gradients of the most critical node with respect to these decoupling capacitors are exactly the transient adjoint sensitivities calculated in each iteration. The cost function for the optimization is a weighted sum of the wire area and the areas of all the decoupling capacitors. In each iteration, either the wire width or the decoupling capacitor with the maximum sensitivity with respect to the objective function will be increased with a small factor until the constraints are met.

5. Experimental results

The simulation and optimization procedure was implemented in C, and the results on several P/G networks were tested. The networks were constructed randomly for power delivery to a 2cm x 2cm chip in a 0.18μm technology with V_{dd} = 1.65V. The set of events is randomly generated and is different for each circuit. The results shown here can be considered to correspond to a top level P/G distribution network, since complete P/G networks may have several millions of nodes. The utilization points here would correspond to functional blocks, each of which is reduced to an equivalent RC representation.

We have used the commercial simulator, HSPICE, to analyze the speed and accuracy of our simulation results. All experiments are performed on Sun Ultra-60 Workstations.

The waveforms for two networks are shown in Fig. 2, with the waveforms using HSPICE plotted concurrently on the same figures using dotted lines. In each case, the largest error between our waveform and that of HSPICE is always within 10%, and most often, much better.

The comparison of the run-time for the two cases and the speedup are shown in Table I. It can be seen that our simulation runs significantly faster than HSPICE.

Table II lists the results of optimization and the run-time for five different P/G networks with and without decaps. The type of network (power (P) or ground (G)) is listed for each circuit, along with the total number of nodes ("total") and the number of user-specified critical nodes ("crt"). The result for the specific voltage constraint, listed in the "spec" column, is shown for each circuit, along with the total wire area. The "Init" column refers to the worst-case voltage when all wires are unsized. The "Opt" column shows the voltage drop after the specifications are met, and can be seen to always be higher than the spec for P networks and lower for G networks. The CPU times and the number of iterations of the heuristic optimizer are shown in the last two columns.

Table III shows the comparison between two networks with different topologies. Circuit 1 deviates from the one-level hierarchical scheme shown in Figure 1, and is a two-level hierarchy in which the top level is a 9-node mesh with a tree of 112 nodes originating from each mesh node; we will refer to such a structure as a "9x112 structure." Some of the tree nodes of this upper level are connected to separate 9x112 structures. Specifically, in the structure here, we have a total of nine such

9x112 bottom level structures. Pads are assigned to each of these bottom level networks. The optimization is performed hierarchically. The bottom-level net is first optimized to within 7% of Vdd with a voltage source of 0.07Vdd applied on the connecting node to the top-level network. The top-level net is then optimized to within 3% of Vdd with the reduced order model connected to the top-level network. For comparison, a one-level 90x112 network (circuit 2) is constructed and optimized. The optimization results show that the two-level hierarchy can be performed far more quickly than the one-level network with similar wire areas. The worst-case voltage drop shows that the two-level hierarchy is more reliable than the one-level network.

6. Conclusion

An efficient transient sensitivity computation method for P/G network design and optimization is presented. A fast and efficient event-driven P/G network simulator is developed. Experimental results show that the simulation is accurate and fast. The optimization procedure involves a procedure for fast calculation of adjoint sensitivities in a heuristic optimization loop. This procedure is illustrated on a specific family of topologies described in Fig. 1, with an example of two-level hierarchy of such a topology. It can also be extended to other mesh topologies that have an overall tree-like structure, e.g., a tree-like macro structure in which each vertex is a mesh.

References

[FD85] J. P. Fishburn and A. E. Dunlop, "TILOS: A posynomial programming approach to transistor sizing," *Proceedings of the 1985 International Conference on Computer-Aided Design*, pp. 326-328, 1985.

[MK92] T. Mitsuhashi and E. S. Kuh, " Power and Ground Network Topology Optimization for Cell-based VLSIs," *Proceedings of the ACM/IEEE Design Automation Conference*, pp. 524-529, 1992.

[OCP98] A. Odabasioglu, M. Celik, and L. T. Pilleggi, "PRIMA: Passive reduced-order interconnect macromodeling algorithm," *IEEE Transactions on Computer-Aided Design*, vol. 17, pp. 645-654, Aug. 1998.

[PRV94] L. T. Pillage, R. A. Rohrer and C. Visweswariah, "*Electronic Circuit and System Simulation Methods*," McGraw-Hill, New York, NY, 1994.

[RGP91] C. L. Ratzlaff, N. Gopal, and L. T. Pillage "RICE: Rapid interconnect circuit evaluator," *Proceedings of the ACM/IEEE Design Automation Conference*, pp. 555-561, 1991.

[SH90] D. Stark and M. Horowitz, "Techniques for calculating currents and voltages in VLSI power supply networks," *IEEE Transactions on Computer-Aided Design*, vol. 9, pp. 126-132, Feb. 19790.

[SVR+94] B. R. Stanisic, N. K. Verghese, R. A. Rutenbar, L. R. Carley, and D. J. Allstot, "Addressing substrate coupling in mixed-mode IC's: Simulation and power distribution synthesis," *IEEE Journal of Solid-State Circuits*, vol. 29, pp. 226-238, Mar. 1994.

[SYSH98] J. C. Shah, A. A. Younis, S. S. Sapatnekar, and M. M. Hassoun, "An algorithm for simulating power/ground networks using Padé approximants and its symbolic implementation," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 45, pp. 1372-1382, Oct. 1998.

[TSL+99] X. Tan, C. J. R. Shi, D. Lungeanu, J. Lee and L. Yuan, "Reliability-Constrained Area Optimization of VLSI Power/Ground Networks Via Sequence of Linear Programmings," *Proceedings of the ACM/IEEE Design Automation Conference*, pp. 78-83, 1999.
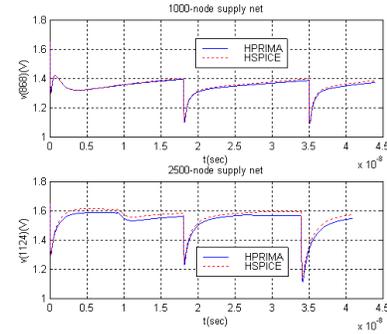
Fig. 2. Simulation results on a 1000-node and a 2500-node supply network. The reduced orders for the two networks are 30 and 36, respectively. HPRIMA stands for our Hierarchical PRIMA simulator. The value of V_{dd} is 1.65V.

Table I. Runtime Comparisons with HSPICE

Ckt	# of nodes		T_{HPRIMA} (s)	T_{HSPICE} (s)	*Speed Up*
	Mesh/Tree	crt			
1	9/1008	10	11.15	82.42	7.48
2	25/2500	25	22.45	232.09	10.34
3	25/3000	15	26.00	325.42	12.52
4	25/4000	20	40.94	499.53	12.20
5	25/5000	25	53.02	680.15	12.83
6	49/10800	50	93.39	1641.02	17.57

- Run time above includes evaluation time for the sum of exponentials.

Table II. Optimization Results (without decaps)

Ckt	Type	# nodes		Spec	Optimized results				CPU time (hrs)	# of iter
		total	crt		Init v_m (V)	Opt v_m (V)	Wire Area (cm^2)	Max Decap (nF)		
1	G	1017	10	0.206	0.673	0.197	0.121	-	0.56	155
1	G	1017	10	0.206	0.585	0.191	0.113	1.00	0.45	153
2	P	2016	20	1.444	0.963	1.453	0.249	-	2.64	288
2	P	2016	20	1.444	1.041	1.447	0.248	16.92	1.72	300
3	G	3025	15	0.206	0.644	0.203	0.483	-	3.20	391
3	G	3025	15	0.206	0.499	0.196	0.369	14.29	4.68	327
4	P	5025	25	1.444	1.068	1.448	0.621	-	4.93	285
4	P	5025	25	1.444	1.252	1.447	0.517	22.76	3.53	251
5	P	10849	50	1.444	1.273	1.449	1.558	-	5.78	185
5	P	10849	50	1.444	1.284	1.455	1.208	13.95	3.90	129

Table III. Topology Comparison

Ckt	# level	# nodes Mesh/Tree	Crt	Spec	Init V_m (V)	Opt V_m (V)	Wire Area (cm^2)	Max decap (nF)	CPU time (hrs)
1	2	90/10080	100	1.485	1.457	1.537	2.28	27.5	0.304
2	1	90/10080	100	1.485	1.353	1.488	2.55	72.6	10.26