

Latency Insertion Method (LIM) for DC Analysis of Power Supply Networks

Dmitri Klokov, Patrick Goh, and José E. Schutt-Ainé, *Fellow, IEEE*

Abstract—Process scaling in modern integrated circuits has led to multiple signal and power integrity issues. In particular, ensuring reliable performance of on-chip power delivery systems has become a major design challenge. Rigorous analysis and simulations are required at the design stage to ensure proper functionality of an on-chip power supply. This puts a strain on existing numerical tools due to the sheer size of the power grids. In this paper, a fast circuit simulation technique based on the latency insertion method (LIM) is proposed for the steady-state analysis of large-scale circuits, such as on-chip power distribution network. The proposed method is shown to be very efficient for modeling of networks with very large numbers of nodes. The comparison with one of the well-established methods used for the power grid analysis today, the Random-Walk algorithm, shows that LIM is almost two orders of magnitude faster.

Index Terms—DC analysis, IR-drop, latency insertion, latency insertion method, power distribution network, power integrity, power supply.

I. INTRODUCTION

DUE TO process scaling, the number of devices on a chip has increased dramatically. Consequently, modern on-chip power distribution networks (PDNs) are required to carry very large amounts of supply current. At the same time, power dissipation issues in high-performance, high frequency devices are forcing the use of lower supply voltages [1]. Also, as interconnects become narrower, wire resistances increase. As a result, large amounts of current are distributed through large-scale power grids with narrow interconnects at low voltages, which causes significant voltage drop (IR drop) on the grid, seriously compromising the performance of the chip. Therefore, efficient and accurate analysis of the IR drop in the power grid becomes one of the crucial steps in the design process.

Due to the extremely large size of the contemporary power grids, their analysis with traditional techniques becomes computationally inefficient or even impossible because of prolonged runtimes and excessive memory requirements. A number of methods have been proposed to address the size-based complexity of the problem. For example, hierarchical analysis technique [2] manages the complexity by solving local grids separately, but this can compromise accuracy.

Manuscript received December 16, 2010; revised May 19, 2011; accepted July 15, 2011. Date of publication September 6, 2011; date of current version November 8, 2011. Recommended for publication by Associate Editor M. Nakhla upon evaluation of the reviewers' comments.

The authors are with the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (e-mail: dklokot2@illinois.edu; pgoh@illinois.edu; jschutt@emlab.uiuc.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCPMT.2011.2162838

Another approach [3] employs a grid-reduction scheme by solving several coarsened grids and then extrapolating the results to the original fine-grain grid. Therefore, the method in [3] solves the network approximately and thus suffers from errors. The statistical Random-Walk method [4] guarantees a linear runtime and the availability of memory, but becomes inefficient when voltages at all the nodes have to be computed or when there is a need for high accuracy. The node-based iterative scheme [5] in effect implements the classical successive over relaxation iterative method for solving linear systems. The node-based method is efficient for steady-state analysis but not easily extendable into transient analysis. There also exist a number of techniques based on the conjugate gradient method [6], [7]. These methods demonstrate a lot of promise, but, as well as the node-based method in [5], are only applied to DC problems. In the design process of a PDN both types of analyses (static and dynamic) must be employed. Therefore, a method being proposed as an alternative to the industry-standard direct solvers (such as SPICE) must deliver similar general-purpose capabilities.

This paper presents an application of the latency insertion method (LIM) to the analysis of the steady-state PDN. The LIM was initially proposed as a transient simulation technique to efficiently model large networks [8]. The method is based on the finite-difference formulation similar to the one used by the well-known finite-difference time-domain method (FDTD) [9]. While FDTD solves the time-domain equations for electric and magnetic fields, LIM applies the same finite-difference technique to circuit equations for voltages and currents. LIM is meant to be an alternative to the traditional matrix-vector product-based circuit simulators such as SPICE, and is targeted primarily at the simulation of very large uniform networks, which makes it particularly suitable for modeling of on-chip power supply. LIM iteratively solves circuit equations using a time-stepping scheme, therefore, it avoids construction and computation of a large matrix (as opposed to SPICE). The method demonstrates linear numerical complexity, enables analysis of very large networks, and greatly outperforms conventional matrix-based methods.

In this paper, it is demonstrated that the LIM, an efficient transient method, can be successfully used for the DC part of the on-chip power supply analysis. For the problem of large circuit DC simulation, the method is compared to one of the well-established techniques the Random-Walk algorithm [4]. The Random-Walk method is an iterative algorithm based on a statistical solver. The comparison between the two methods shows that LIM significantly outperforms the Random-Walk method both in terms of speed and accuracy.

The rest of this paper is organized as follows. In Section II, the formulation of the LIM method is described. Sections III and IV demonstrate the application of the LIM method to the DC analysis of the PDN. In Section V simulation results are presented and benchmark comparison between the LIM and the Random-Walk method is made. Finally, in Section VI, the conclusions are reported.

II. LIM FORMULATION

LIM treats a circuit as a grid composed of nodes interconnected with branches. The approach is based on utilizing an equivalent model for circuit interconnects (distributed RLGC-based transmission line model). The RLGC equivalent circuit representation of integrated circuit interconnects can be obtained by running a layout extraction procedure in a computer aided design tool used in the design process.

Each branch (a section of an interconnect) can be represented by a distributed series resistor-inductor model as a combination of a voltage source, an inductor, and a resistor in series (as shown in Fig. 1). Current I_{ij} is assumed to be directed from node i at voltage V_i to node j at potential V_j .

Each node is modeled as a combination of a current source, a conductance, and a capacitor to the ground (as in Fig. 2).

Using the Kirchhoff's voltage law for the circuit in Fig. 1 the discrete time equation can be written as

$$V_i^{n+1/2} - V_j^{n+1/2} = L_{ij} \left(\frac{I_{ij}^{n+1} - I_{ij}^n}{\Delta t} \right) + R_{ij} \left(\frac{I_{ij}^{n+1} + I_{ij}^n}{2} \right) - E_{ij}^{n+1/2}. \quad (1)$$

Superscripts in (1) represent the discrete time. Currents are computed at whole time intervals $t = n\Delta t$ voltages at half intervals $t = (n + 1/2)\Delta t$, where Δt is the time step. This way voltages and currents are staggered by half a time step. This is known as the leapfrog scheme or semi-implicit scheme [9].

Solution of the discrete time (1) for the branch equivalent circuit in Fig. 1 yields the following expression for branch currents:

$$I_{ij}^{n+1} = 2\Delta t \frac{(V_i^{n+1/2} - V_j^{n+1/2})}{(2L_{ij} + R\Delta t)} + I_{ij}^n \frac{(2L_{ij} - R\Delta t)}{(2L_{ij} + R\Delta t)}. \quad (2)$$

Using the Kirchhoff's current law for the LIM node equivalent circuit in Fig. 2 the discrete time equation can be written as

$$C_i \left(\frac{V_i^{n+1/2} - V_i^{n-1/2}}{\Delta t} \right) + G_i \left(\frac{V_i^{n+1/2} + V_i^{n-1/2}}{2} \right) - H_i^n = - \sum_{k=1}^{M_i} I_{ik} \quad (3)$$

where M_i is the total number of branches connected to node i .

Solution of (3) yields the expression for node voltages

$$V_i^{n+1/2} = V_i^{n-1/2} \frac{(2C_i - G_i\Delta t)}{(2C_i + G_i\Delta t)} - 2\Delta t \frac{\left(\sum_{k=1}^{M_i} I_{ik}^n - H_i^n \right)}{(2C_i + G_i\Delta t)}. \quad (4)$$

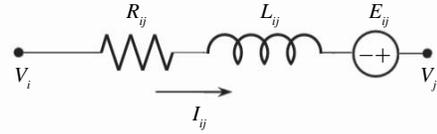


Fig. 1. LIM branch equivalent circuit.

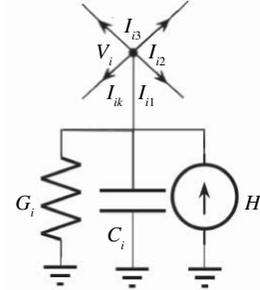


Fig. 2. LIM node equivalent circuit.

If the circuit model does not include shunt node conductance G_i (which, as we will show further, is the case for the power supply network model), expression in (4) can be simplified to

$$V_i^{n+1/2} = V_i^{n-1/2} - \frac{\Delta t}{C_i} \left(\sum_{k=1}^{N_a} I_{ik}^n - H_i^n \right). \quad (5)$$

In the LIM transient simulation, node voltages and branch currents are computed at each time step using the update expressions (2) and (4). First all node voltages are calculated through (4) next, all branch currents are updated through (2).

As it can be seen from (2) and (5), the presence of latency generated by the reactive elements L_{ij} and C_i is required for the algorithm to function. If the reactive elements are not present in the circuit, small fictitious inductors must be inserted in each branch and capacitors must be added at every node to enable LIM formulation. It can be shown that if the values of those fictitious elements are small enough, the accuracy of the transient simulation is not compromised [8].

The LIM algorithm uses central difference-based finite-difference approximation, which is conditionally stable. In the simple 1-D case (no more than two branches connected at each node) the condition for the numerical stability of the method can be written as [8], [10]

$$\Delta t < \min_{i=1}^{N_b} \left(\sqrt{L_i \min(C_i, C_{i+1})} \right) \quad (6)$$

where N_b is the number of branches in the network, L_i is the inductance of the branch i , C_i , and C_{i+1} are shunt capacitors connected on either side of the branch i . Condition (6) is analogous to the Courant criterion for wave propagation in a discrete grid [11], that is used to determine the limit on the time step in the FDTD method. In the FDTD method the limit on the time step is based on the fact that numerical waves in a computational space cannot propagate faster than the speed of light. The same idea applies to the LIM, the speed of signal propagation is limited by the latency present in the circuit. However, in the case when there are more than two branches

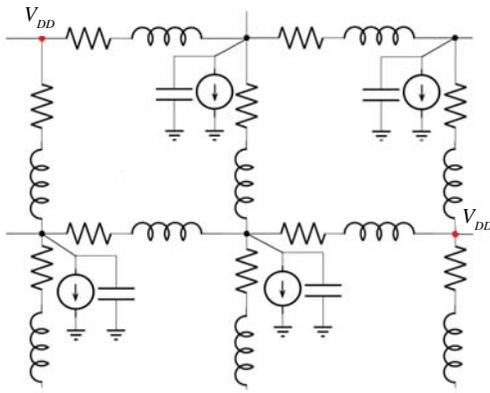


Fig. 3. Equivalent circuit model for a part of an on-chip power supply network.

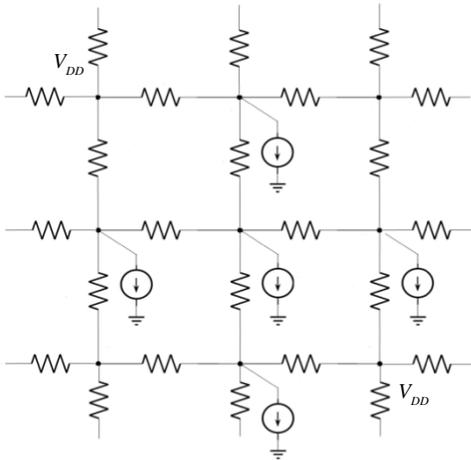


Fig. 4. Circuit model for a steady-state PDN.

connected at a single node, the problem of determining the stable simulation time step is not as straightforward. Again using the analogy with the FDTD method, we note that the limit on the time step reduces as the dimension of the computational space increases. The same idea applies to the LIM. However, in the LIM model network the number of connections at a single node is not limited to tree, and the notion of dimension in such a model does not have a clear physical meaning (as opposed to the FDTD case). The problem of finding a suitable value for the time step of the simulation is further discussed in Section IV.

III. ON-CHIP PDN

A typical power grid model consists of wire resistances, wire inductances, wire capacitances, decoupling capacitors, V_{DD} pads, and current sources that represent currents drawn by logic gates or functional blocks [4]. The equivalent model for a part of PDN is shown in Fig. 3. The model in Fig. 3 is somewhat simplified and can be viewed as the representation of a single layer of power lines with only line-to-ground capacitance taken into account.

Analysis of a PDN is often done in two steps. The steady-state (DC) solution is found first and then the transient analysis is performed. The transient simulation is naturally carried out

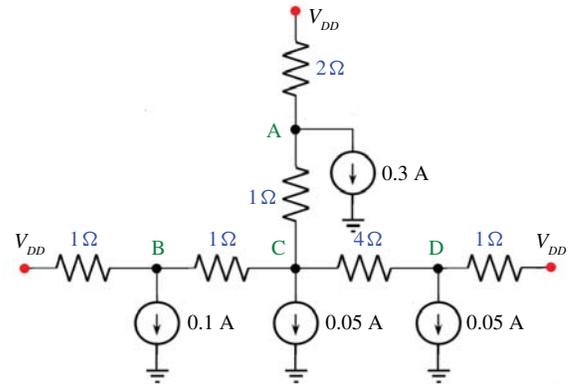


Fig. 5. Simple circuit example.

using the standard LIM formulation [8], [12]–[14]. In this paper, we focus on the first stage, the problem of computing the steady-state node voltages, the DC solution. Due to the sheer size of contemporary PDNs even the initial DC analysis becomes a challenging problem that is not suitable for direct solvers and requires the use of alternative more efficient algorithms [4], [5]. We show that the LIM, while being primarily a transient simulation technique, can be efficiently applied to the DC steady-state analysis problem. A transient simulation in the network that contains only constant current and voltage sources rapidly converges to the steady-state voltage and current values.

In the steady state all capacitors are open circuited and inductors are short circuited. The model in Fig. 3 then simplifies to the one shown in Fig. 4.

The power grid is then represented as a network of purely resistive branches and nodes of two types: constant voltage sources (V_{DD} pads) and constant current sources.

IV. STEADY-STATE PDN ANALYSIS USING LIM

In order to show how LIM can be applied to the DC analysis of a power grid, we use a simple circuit example [4]. Fig. 5 shows a simple circuit that represents a small section of a PDN.

If the supply voltage V_{DD} is chosen to be 1 V, the voltages at nodes A, B, C, and D can be easily calculated: 0.6, 0.8, 0.7, and 0.9 V respectively.

A. Enabling LIM Formulation: Latency Insertion

The DC PDN model does not contain latency. In order to enable the LIM formulation for the circuit in Fig. 5, latency elements must be inserted in all the branches and at all the nodes. The resulting LIM-enabled augmented circuit corresponding to Fig. 5 is shown in Fig. 6.

Since the insertion elements are fictitious, they must be made as small as possible, so that the accuracy of the solution is not compromised. In general, the values of the fictitious reactive elements are chosen to be much smaller than those of the actual inductances and capacitances present in the circuit. On the other hand, the choice of the values for fictitious reactive elements affects the stability for the simulation. In a

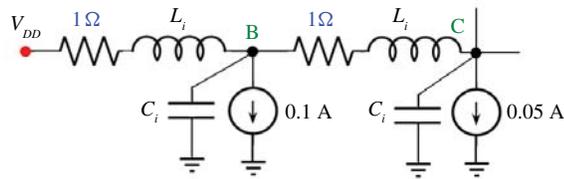


Fig. 6. Augmented section of the example circuit from Fig. 5 with fictitious latency elements in place.

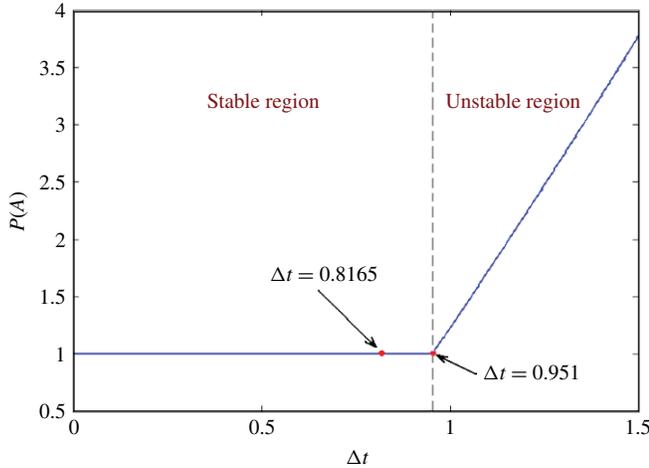


Fig. 7. Spectral radius of the amplification matrix \mathbf{A} as a function of the time step.

transient LIM simulation values of the reactive elements determine the limit on the maximum possible time step of the simulation. Therefore, it is desirable to have a lot of latency in the circuit that will allow using the largest time step and hence, achieving the shortest runtime.

However, in the case of DC analysis the actual values of the insertion elements are not important since only steady-state voltages are computed and not transient waveforms. Therefore, the values of insertion elements can be chosen arbitrarily. Considerations involved in the choice of the optimum time step are discussed in some detail further.

B. Choosing the Time Step of the LIM Simulation

In the case of the DC analysis with an iterative method the duration of the simulation is determined by the number of iterations required for the solution to converge to the steady-state values. In our case, since the simulation is actually transient, there is still the issue of choosing the appropriate increment for the time step. First of all, the Δt must be chosen so that the simulation remains stable, also a fast rate of convergence is desirable.

The exact limit on the time step of the LIM simulation can be found using the amplification matrix approach [15], [16]. The amplification matrix approach is based on transforming the LIM update equations into the matrix form, such that voltage and current vector values at the next time step are found by multiplying the values from the previous step with the matrix \mathbf{A} , the amplification matrix. That matrix depends on the structure of the network and the time step value. According

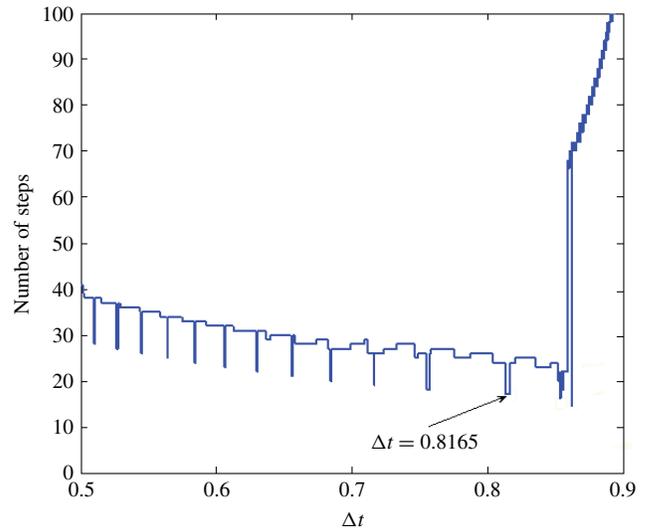


Fig. 8. Convergence of the LIM simulation as a function of the time step.

to the method in [15] the time step of the simulation must be chosen so that the spectral radius of the amplification matrix is less than or equal to unity. If any of the eigenvalues of the matrix \mathbf{A} exceeds 1, the solution will grow uncontrollably. Fig. 7 shows the dependence of the spectral radius of the amplification matrix for the circuit in Fig. 5 on the value of the time step. It can be seen in Fig. 7 that the system becomes unstable as the value of the time step exceeds $\Delta t = 0.95$.

While the amplification matrix approach can produce the exact limit on the time step, it is not always convenient to derive the matrix and solve for its eigenvalues. In [12] and [17] the upper bound on the time step of the transient LIM simulation was derived using the direct Lyapunov method (also known as the energy method). In the case when more than two branches are connected to a single node the stability condition can be written as

$$\Delta t \leq \sqrt{2} \min_{i=1}^{N_n} \left(\sqrt{\frac{C_i}{N_b^i} \min_{p=1}^{N_b^i} (L_{i,p})} \right) \quad (7)$$

where N_n is the total number of nodes in the circuit, C_i is the shunt capacitance at node i , N_b^i denotes the number of branches connected to node i , $L_{i,p}$ denotes the value of p th inductor connected to node i . The result of (7) is the square root of the smallest $L_{i,p}$, C_i product among all the nodes of the circuit divided by the number of connections at the node. In 1-D case (7) becomes (6). The limit in (7) is easy to obtain in the process of parsing a circuit netlist. Condition in (7) is only a sufficient one. If the time step exceeds the limit given by (7), the simulation does not necessarily become unstable.

Condition (7) can be used for the calculation of the time step for the steady-state PDN simulation. As was mentioned before, in the DC case actual values of the fictitious latency elements are not relevant. Therefore, for simplicity and convenience all insertion elements throughout the circuit can be assigned to a value of unity. Then, using (7), we can write the expression

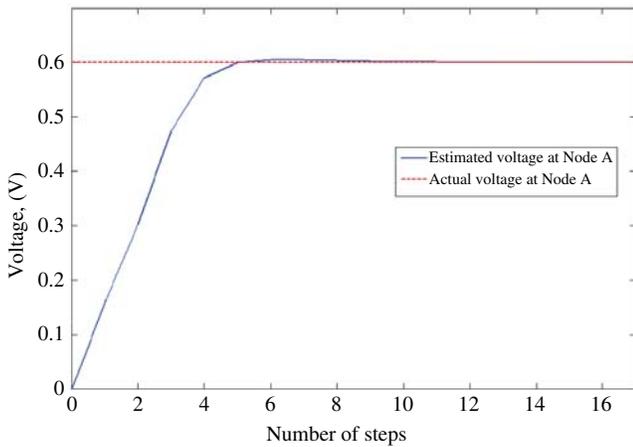


Fig. 9. Convergence of the simulation using the LIM algorithm.

TABLE I
LIM RESULTS FOR THE BASIC CIRCUIT

Nodes	Estimated Voltage (V)	Actual Voltage (V)
A	0.600007	0.6
B	0.800009	0.8
C	0.700010	0.7
D	0.900013	0.9

for the time step as

$$\Delta t = \sqrt{\frac{2}{\max_{i=1}^{N_n} (N_b^i)}}. \quad (8)$$

It can be demonstrated using the circuit in Fig. 4 that the time step resulting from (8) is stable and closely approximates the optimum step for fastest convergence. Node C in the circuit has the highest number of connections $N_b^C = 3$ then, from (8) the time step for the circuit is $\Delta t = 0.8165$. From Fig. 7 it can be seen that the time step from (8) is in the stable region.

It is also desirable to use the time step that results in the fast convergence of the simulation. The simulation starts with zero initial conditions and then converges to a certain DC voltage level. Larger time step results in larger increments of voltage values at each step of the simulation. Therefore, in general, a larger time step allows the simulation to reach the steady-state voltage value in fewer iterations. However, if the time step is too close to the stability limit, voltages oscillate around their steady-state values, which results in prolonged run times. The trend is shown in Fig. 8.

First, as the time step increases, the number of iterations required for convergence decreases. Then, as the value of Δt approaches 0.9, there is a steep increase in the number of steps required for convergence. The value of Δt predicted by (8) results in the near optimum number of 17 iterations required for voltages at all nodes of the circuit example to converge.

C. Numerical Simulation Results for the Basic Circuit

After latency is created and the time step is determined, the augmented circuit can be simulated with LIM. Table I shows the simulation results for the circuit in Fig. 5.

TABLE II
RUNTIME RESULTS FOR LARGE CIRCUITS

Number of Nodes	LIM (CPU seconds)	Random-Walk (CPU seconds)
10 K	< 1	10
250 K	3	258
500 K	6	509
1 M	13	1126
2 M	28	2528

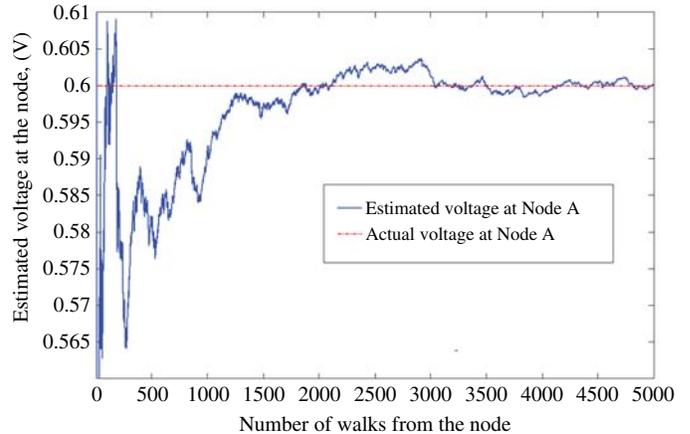


Fig. 10. Convergence of the Random-Walk algorithm.

It took only 17 iterations to achieve the relative error on the order of 0.001%. Fig. 9 shows the convergence of the simulation for a single node of the basic circuit.

From Fig. 9 it can be seen that the LIM simulation quickly reaches the correct steady-state value of the node voltage and stays stable at that value. For the comparison, the same example was simulated using the Random-Walk algorithm.

In Fig. 10 the convergence of the simulation performed using the Random-Walk algorithm is shown. Fig. 10 demonstrates that it is relatively harder to obtain high-precision solution using the Random-Walk algorithm due to the random nature of the method. Random-Walk method is based on the statistical approach “the random walk game.” The solution converges to the correct value as more “walks” are taken from the node, and the results are averaged over a large number of iterations. However, even after a large number of “walks” the solution still oscillates considerably.

V. NUMERICAL RESULTS OF THE DC POWER GRID ANALYSIS

As was mentioned above, the main advantage of the iterative techniques in general and LIM in particular is the ability to efficiently analyze the networks with very large numbers of nodes that are not suitable for matrix-based techniques (such as SPICE). Several large circuits were simulated using LIM and Random-Walk methods. All circuits used for simulation had the form of resistive grids with constant voltage (V_{DD}) sources and constant current sources (power drains), similar to the model in Fig. 4. Runtime results are summarized and compared in Table II. Runtimes as functions of the number of

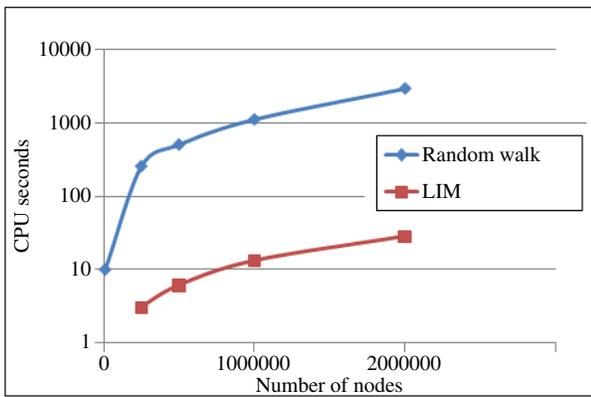


Fig. 11. Runtime as a function of the number of nodes.

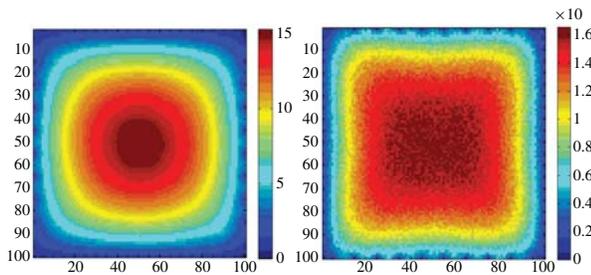


Fig. 12. Voltage drop profile (on the left) and the corresponding error profile (on the right) for a 10 K circuit. Color scale on the voltage drop plot corresponds to the percent difference from the supply voltage. Color scale on the error profile plot shows simple difference between the LIM simulation values and the ones produced by the HSPICE.

nodes in the circuit are also shown in Fig. 11 on a semi-log scale.

Table II can be extrapolated as both algorithms have linear numerical complexity and runtimes scale nearly linearly. All computations were carried out on a Windows workstation with 2.4 GHz CPU and 2 GB of RAM.

In the above simulations, the error margin was set to 1 mV. It is possible to reduce the runtime of the Random-Walk method if high accuracy is not required or if only a few node voltages need to be computed. Otherwise, the LIM method demonstrates almost two orders of magnitude speedup over the conventional Random-Walk algorithm.

In the Random-Walk method, the variance is used as the measure of convergence of a node voltage [4]. LIM is not a statistical method and does not involve random number generation. In the LIM program convergence is claimed when the difference between the new voltage value and the old one does not exceed a certain error margin (user specified). Normally the simulation runs until the voltage values at all nodes stay within the error margin. A significant speedup can be achieved if the nodes at which the convergence is registered are removed from the computation. Such nodes are assigned some special “untouchable” type values and are not processed any more. A simulation is stopped once there are no more nodes left to process.

There is a certain error introduced every time a node is considered converged and voltage at that node is fixed.

However, the error can be kept small. Fig. 12 demonstrates the error values for the 10 K node circuit (with HSPICE simulation results treated as exact) produced by the LIM simulation. The largest error is still less than 1.7 mV.

With modifications the LIM simulation of a circuit with 500 K nodes can be reduced to 2 CPU seconds, while it takes 509 CPU seconds with the Random-Walk method.

The circuit has a favorable structure (large number of supply nodes), but it is favorable for both the LIM and the Random-Walk method (as well as for any other iterative method).

VI. CONCLUSION

The LIM was applied to the analysis of steady-state on-chip PDNs. Despite being initially developed as a purely transient technique, the method was shown to be an efficient tool for the DC analysis of PDNs. Therefore, the complete analysis of a PDN (both DC and transient) can be carried out using a single method. The LIM algorithm exhibits fast convergence, high accuracy, and short runtime. The algorithm has linear numerical complexity and is suitable for the analysis of very large networks. Experimental results show that LIM significantly outperforms the conventional Random-Walk method, particularly for circuits with very large numbers of nodes.

REFERENCES

- [1] M. K. Govan, L. L. Biro, and D. B. Jackson, “Power considerations in the design of the alpha 21264 microprocessor,” in *Proc. 35th Des. Autom. Conf.*, Jun. 1998, pp. 726–731.
- [2] M. Zhao, R. V. Panda, S. S. Spatnekar, and D. Blaauw, “Hierarchical analysis of power distribution networks,” *IEEE Trans. Comput.-Aid. Des. Integr. Circuits*, vol. 24, no. 5, pp. 1204–1224, Aug. 2005.
- [3] J. Kozhaya, S. R. Nassif, and F. N. Najm, “A multigrid-like technique for power grid analysis,” *IEEE Trans. Comput.-Aided Des. Integr. Circuit*, vol. 21, no. 10, pp. 1148–1160, Oct. 2002.
- [4] H. Qian, S. R. Nassif, and S. S. Sapatnekar, “Random walks in a supply network,” in *Proc. 40th Des. Autom. Conf.*, Jun. 2003, pp. 93–98.
- [5] Y. Zhong and M. D. Wong, “Fast algorithms for IR drop analysis in large power grid,” in *Proc. Int. Conf. Comput.-Aid. Des.*, 2005, pp. 351–357.
- [6] J. Xie and M. Swaminathan, “DC IR drop solver for large scale 3-D power delivery networks,” in *Proc. IEEE Electr. Perform. Electron. Packag. Syst.*, Nov. 2010, pp. 217–220.
- [7] H. Dag and F. L. Alvarado, “Toward improved uses of the conjugate gradient method for power systems applications,” *IEEE Trans. Power Syst.*, vol. 12, no. 3, pp. 1306–1314, Aug. 1997.
- [8] J. E. Schutt-Ainé, “Latency insertion method (LIM) for the fast transient simulation of large networks,” *IEEE Trans. Circuit Syst.*, vol. 48, no. 1, pp. 81–89, Jan. 2001.
- [9] K. S. Yee, “Numerical solution of initial boundary value problems involving Maxwell’s equation in isotropic media,” *IEEE Antennas Propagat.*, vol. 14, no. 3, pp. 303–307, May 1996.
- [10] Z. Deng and J. E. Schutt-Ainé, “Stability analysis of latency insertion method (LIM),” in *Proc. IEEE 13th Electr. Perform. Electron. Packag. Conf.*, Oct. 2004, pp. 167–170.
- [11] R. Courant, K. Friedrichs, and H. Lewy, “On partial difference equations of mathematical physics,” *IBM J.*, vol. 11, pp. 215–234, Mar. 1967.
- [12] S. L. Lalgudi, M. Swaminathan, and Y. Kretchmer, “On-chip power-grid simulation using latency insertion method,” *IEEE Trans. Circuit Syst.*, vol. 55, no. 3, pp. 914–931, Apr. 2008.
- [13] Z. Deng and J. E. Schutt-Ainé, “LIM-SPICE for the analysis of power distribution networks,” in *Proc. 9th IEEE Signal Propagat. Interconn. Workshop*, May 2005, pp. 17–20.
- [14] J. E. Schutt-Ainé, “LIM-based algorithms for the transient simulation of large networks,” in *Proc. IEEE 13th Electr. Perform. Electron. Packag. Conf.*, Oct. 2007, pp. 233–236.
- [15] J. E. Schutt-Ainé, “Stability analysis for semi-implicit LIM algorithm,” in *Proc. IEEE Asia Pacific Microw. Conf.*, Dec. 2009, pp. 1270–1272.

- [16] P. Goh, J. E. Schutt-Ainé, D. Klokov, J. Tan, P. Liu, W. Dai, and F. Al-Hawari, "Partitioned latency insertion method (PLIM) with stability considerations," in *Proc. IEEE 15th Workshop Signal Propagat. Interconn.*, May 2011, pp. 107–110.
- [17] S. Lalgudi and M. Swaminathan, "Analytical stability condition of the latency insertion method for nonuniform GLC circuits," *IEEE Trans. Circuits Syst.*, vol. 55, no. 9, pp. 937–941, Sep. 2008.



Dmitri Klokov received the B.S. degree in physics from Saratov State University, Saratov, Russia, in 2001, and the M.S. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign, Urbana, in 2007. He is currently pursuing the Ph.D. degree in the same university.

His current research interests include numerical methods, computational electromagnetics, circuit simulation, interconnect analysis, and signal and power integrity.



Patrick Goh received the B.S. and M.S. degrees in electrical engineering from the University of Illinois at Urbana-Champaign (UIUC), Urbana, in 2007 and 2009, respectively, where he is currently pursuing the Ph.D. degree.

He has been a Teaching Assistant with the Department of Electrical and Computer Engineering, UIUC, since August 2007. His current research interests include circuit simulation for high-speed interconnects, modeling, and computer-aided design tools.

Mr. Goh was a recipient of the H. L. Olesen Award from UIUC for outstanding effort in undergraduate teaching in 2010.



José E. Schutt-Ainé (S'86–M'86–SM'98–F'07) received the B.S. degree in electrical engineering from the Massachusetts Institute of Technology, Cambridge, in 1981, and the M.S. and Ph.D. degrees from the University of Illinois at Urbana-Champaign (UIUC), Urbana, in 1984 and 1988, respectively.

He joined Hewlett-Packard Technology Center, Santa Rosa, CA, as an Application Engineer, involved with microwave transistors and high-frequency circuits. In 1983, he joined UIUC and then joined the Electrical and Computer Engineering

Department as a member of the Electromagnetics and Coordinated Science Laboratories, where he currently specializes in the study of signal integrity for high-speed digital and high frequency applications. He has been a Consultant for several corporations. His current research interests span the spectrum from microwave measurements to the generation of computer-aided design tools for electronic systems.

Dr. Schutt-Ainé has been a recipient of several research awards including the National Science Foundation (NSF) Magnetic Resonance Imaging Award in 1991, the National Aeronautics and Space Administration Faculty Award for Research in 1992, the NSF Marine Corps Aviation Association Award in 1996, and the UIUC-National Center for Superconducting Applications Faculty Fellow Award in 2000.