SMM: Scalable Analysis of Power Delivery Networks by Stochastic Moment Matching

Andrew B. Kahng, Bao Liu and Sheldon X.-D. Tan†

CSE and ECE Departments, UC San Diego, La Jolla, CA 92093
† EE Department, UC Riverside, Riverside, CA 92521
abk@ucsd.edu, bliu@cs.ucsd.edu, stan@ee.ucr.edu

Abstract

*This paper proposes a novel method for analyzing large on-chip power delivery networks via a stochastic moment matching (*SMM*) method. The proposed method extends the existing direct stochastic random walk method that can be only applied to DC analysis in purely resistive networks or transient analysis of RC networks with low efficiency. The new method can analyze general structure RLC networks by combining the stochastic process with frequency domain moment matching technique. As a result, we achieve better scalability than traditional frequency domain P/G analysis approaches, and better efficiency than existing random walk transient analysis techniques. Our experimental results show that SMM can easily trade efficiency for accuracy or vise versa. SMM can easily deliver 10X-100X speedup over a LU-based direct solver and about 10X speedup over the pure random walk method with reasonable accuracy on large industry P/G networks.*

1 Introduction

Nanometer VLSI designs encounter increasingly severe power/ground (P/G) supply voltage degradation. Causes include several design trends: (1) technology advancement implies decreased wire widths and increased interconnect resistances along a P/G supply network, (2) increased device density causes increased supply current density on chip, and (3) a higher clock frequency leads to more significant inductance effect. At the same time, the scaling of technology entails a decreasing supply voltage and a decreasing noise margin for signal transition, which makes a transistor more vulnerable to supply voltage degradation. A degraded supply voltage leads to performance degradation or even malfunction. For example, a 10% supply voltage degradation could be responsible for more than 10% transistor performance degradation, and the effect is super-linear [20]. Therefore, P/G supply network design in a nanometer VLSI system is critical to system performance and functionality.

VLSI power/ground supply network analysis and system performance verification are closely related. P/G supply network analysis needs performance verification results as inputs, e.g., tap currents, transistor toggling rates or "timing windows"; performance verification needs to take into consideration the resultant supply voltage degradation [1, 2]. Conventional timing analysis techniques are applied to P/G network analysis. For example, static (input-vector-independent) analysis is enabled by stimulating a P/G supply network with maximum (worst case) tap current waveform envelopes [3]; model order reduction techniques extract electrical behavior characteristics of a P/G supply network and facilitate simulation [22]. However, power/ground networks are unique in their non-tree topology, large instance sizes, numerous voltage/current sources and output (observation) nodes, which makes conventional timing analysis techniques impractical to apply without significant efficiency and scalability improvements.

Scalability improvement of P/G network analysis includes multigrid-like [10, 12], hierarchical [21], and partition-based [11] approaches, which provide a global abstraction of a P/G network. Recently, a stochastic random walk technique [17] is proposed to improve scalability of P/G network analysis by exploiting locality of supply voltage degradation, i.e., supply voltage drop of a P/G node is largely determined by its neighboring region, e.g., from the node to a nearby P/G supply pad. This random walk method is applied to DC analysis and time domain transient analysis.

Transient random walk is less efficient especially for P/G grids with boundary P/G pads where the current sources in the companion models of the reactive elements (e.g., capacitors and inductors) vary with time, and one needs to compute node voltages at each time step. Inefficiency can be mitigated by introducing a hierarchical approach [18, 19], and combining random walk with a LU-based solver via a partitioning scheme [8].

The authors of [17] also suggest possible random walk in frequency domain. E.g., an existing stochastic moment generation method removes all capacitors except one or two at each step to facilitate computation in a large RC network [4].

In this paper, we propose a new stochastic moment computation method, SMM, by combining stochastic traversal with tree structure moment computation techniques. The resultant method boosts the computation efficiency of pure time-domain random walk by computing a small number of moments instead of large number of time steps. It also extends the existing moment matching method in achieving better efficiency for circuits with multiple inputs/outputs, e.g., P/G delivery networks. Our new analysis technique can be applied to supply voltage drop analysis in practical size P/G networks with either internal or boundary P/G pads.

We study SMM in three different iteration schemes, including Liebmann's method in solving partial differential equations. The best solution convergence occurs by following the random walk transition probability in traversing the netlist and computing the moments. We reduce the inherent instability of moment computation by (1) computing moments of all orders in the same random walk process, (2) avoiding the averaging of moments over different random walk processes, and (3) applying filtering schemes to exclude potentially inaccurate solutions, e.g., with zero moments, zero voltage, and positive poles.

Our experiments show that simulation results of SMM approach that of SPICE with increased number of iterations; and SMM runtime decreases with increased number of power supply pads (which help to confine random walks in a local region, and to improve P/G analysis scalability). Averaging over a number of iterations provides an easy accuracy-efficiency tradeoff; filtering out possible inaccurate solutions overcomes the inherent numerical instability of moment matching techniques, which makes SMM an efficient P/G network analysis approach.

The rest of the paper is organized as follows. In Section 2, we present P/G network modeling and existing random walk and moment matching techniques. Section 3 presents the stochastic moment matching method. Our implementation and experimental results are respectively presented in Sections 4 and 5, before we conclude in Section 6.

2 Background

A P/G supply network is modeled as a distributed RLC(K) network which includes interconnect resistors and ground capacitors. Power/ground pads are modeled as DC voltage sources, while active devices which draw supply currents are modeled as time-varying current sources (Fig. 1). Each active device injects a supply current, e.g., of maximum envelope current waveform [3], during signal transition in a "timing window", i.e., the time frame bounded by the earliest and the latest signal arrival times in timing analysis. We formulate the P/G supply network analysis problem as follows.

Problem 1 Given

1. *an RLC(K) netlist of a P/G supply network,*
2. *a subset of nodes with DC supply voltage sources, and*
3. *a subset of nodes with supply current sources,*

find *the voltages of a subset of observation nodes.*

We use the following notations in this paper.

- (p, q) = edge between nodes p and q,
- E = set of edges of P/G netlist,
- T_q = subtree rooted at node q in a tree structure netlist,
- C_q = ground capacitance at node q,

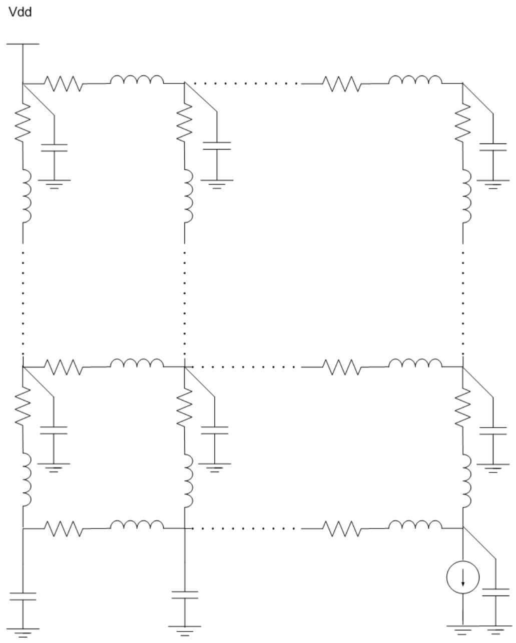

Fig. 1. A power supply network in a grid, with resistance and inductance on each grid segment, ground capacitance and possible voltage or current sources at each node.

- G_{pq} = conductance between nodes p and q,
- L_{pq} = inductance between nodes p and q,
- V_q = voltage of node q,
- I_q = current of the current source at node q,
- $m_i(q)$ = the i-th order moment of the voltage of node q,
- $m_i(I_q)$ = the i-th order moment of the source current I_q.

2.1 Random Walk in a Resistive Network

P/G network analysis can be performed by straightforward circuit nodal analysis, e.g., via a direct equation system solver [17, 22]. However, direct solvers suffer efficiency and scalability problems. A more efficient approach is to apply Monte Carlo methods of partial differential equations (PDEs), e.g., random walk [6].

For example, in a purely resistive P/G network (e.g., Fig. 1 with zero inductances and capacitances), for a node q, with conductance G_{pq} between q and each of its neighboring node p, and a current source I_q between q and the ground, Kirchoff's current law gives the ground current I_q and voltage V_q of node q as follow.

$$V_q = \frac{\sum_{(p,q)\in E} G_{pq} V_p - I_q}{\sum_{(p,q)\in E} G_{pq}} \tag{1}$$

This is the finite-difference form of a boundary value problem of partial differential equations, where each node voltage is associated with neighboring node voltages, and the known node voltages consist of the boundary conditions [10].[1] One of

[1]E.g., in a continuous resistive surface, (1) becomes a partial differential equation $I_q(x, y) = \frac{\partial^2 G(x,y)V(x,y)}{\partial x^2} + \frac{\partial^2 G(x,y)V(x,y)}{\partial y^2}$, where $I_q(x, y)$, $G(x, y)$ and $V(x, y)$ are ground current, conductance, and voltage at point (x, y).

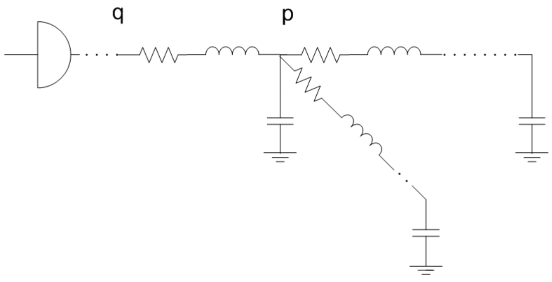

Fig. 2. An RLC interconnect tree with ground capacitance at each node and resistance and inductance on each segment. Moments of node q are given by its downstream neighbor node p.

the Monte Carlo methods of solving PDEs, i.e., random walk, is explained in [17] as follows. A traveler pays an amount $A(q) = \frac{I_q}{\sum_{(p,q)\in E} G_{pq}}$ (e.g., for lodging) at a node q, with a probability $Prob(p,q) = \frac{G_{pq}}{\sum_{(p,q)\in E} G_{pq}}$ of going to an adjacent node p, until he reaches a node of V_{dd} voltage source (home), where he stays and receives a reward (which equals the V_{dd} voltage). The average net gain (the reward minus the costs) of the trip approaches the voltage at node q (Algorithm 1). Such a random walk game avoids prohibitive full scale analysis and exploits locality of the problem (since most of the traversals are at neighboring nodes). Accuracy improves with increased number of random walk processes [17].

Algorithm 1: Random Walk in a Resistive Network

Input: Resistive netlist, boundary nodes with known voltages
Output: Voltage of the observation node

1. Starting from the observation node
2. While(not reaching a boundary node)
3. Pay $A(q)$ at node q
4. Walk to an adjacent node p with $Prob(p,q)$
5. Gain V_b of the voltage at boundary node b
6. Return the net gain

2.2 Moment Computation in an RLC Tree

Moment matching techniques (e.g., AWE [15]) derive poles and residues from moments of state variables in a circuit system. Each state variable, e.g., a node voltage V_q, can be expanded in moments $m_i(q)$. To compute these moments generally requires inversion of the $G^{-1}C$ matrix, which takes $O(n^3)$ time [15]. However, for an interconnect in a tree topology, a linear moment computation method is as follows [16].

For a node p and its parent node q in an RC tree (Fig. 2 with zero inductance), with a rising signal transition, the current through resistor R_{pq} goes to all the capacitors in the subtree T_p which is rooted at node p, i.e.,

$$V_q = V_p + R_{pq} \sum_{k\in T_p} sC_k V_k \tag{2}$$

where V_p and V_q are respectively voltages at nodes p and q, R_{pq} is the resistance between nodes p and q, C_k is the ground capacitance at node k, V_k is the voltage at node k, and T_p is the subtree rooted at node p. Expanding the voltages in moments gives

$$\begin{aligned} \sum_i m_i(q)s^i &= \sum_i m_i(p)s^i + R_{pq} \sum_{k\in T_p} sC_k \sum_i m_i(k)s^i \\ m_i(q) &= m_i(p) + R_{pq} \sum_{k\in T_p} C_k m_{i-1}(k) \end{aligned} \tag{3}$$

In the presence of inductance L_{pq} between nodes p and q, (Fig. 2), (3) becomes as follows.

$$\begin{aligned} V_q &= V_p + (R_{pq} + sL_{pq}) \sum_{k\in T_p} sC_k V_k \\ \sum_i m_i(q)s^i &= \sum_i m_i(p)s^i + (R_{pq} + sL_{pq}) \sum_{k\in T_p} sC_k \sum_i m_i(k)s^i \\ m_i(q) &= m_i(p) + R_{pq} \sum_{k\in T_p} C_k m_{i-1}(k) + L_{pq} \sum_{k\in T_p} C_k m_{i-2}(k) \end{aligned} \tag{4}$$

By (3) or (4), we can compute the m moments for each of the n nodes in a tree structure interconnect in mn time by m depth-first traversals (Algorithm 2).

Algorithm 2: Moment Computation in a Tree Structure Interconnect

Input: RLC netlist tree, input node voltage moments
Output: Output node voltage moments

1. For each moment order i
2. Depth-first traversal of the netlist tree
3. In pre-order, compute $m_{i-1}(p)$ for each node p by (3) or (4)
4. In post-order, compute $\sum_{k\in T_p} C_k m_{i-1}(k)$ for each T_p

3 Stochastic Moment Matching (SMM) Theory

3.1 Moment Computation in an RLC Network

We now apply moment computation technique to an RLC network of general structure (Fig. 1), where we do not have the equation between the current through a resistor and the current through all the capacitors in a subtree as in a tree structure interconnect. However, Kirchoff's current law reveals the direction to a random walk application.

For a node q in an RLC network, with conductance $G_{pq} = (R_{pq} + sL_{pq})^{-1}$ between nodes p and q, and a total ground current $\bar{I}_q = sC_qV_q + I_q$ through a ground capacitor C_q and a current source I_q at node q, Kirchoff's current law gives:

$$\sum_{(p,q)\in E} \frac{V_q}{R_{pq} + sL_{pq}} = \sum_{(p,q)\in E} \frac{V_p}{R_{pq} + sL_{pq}} - sC_qV_q - I_q \tag{5}$$

Expanding V_p, V_q, and I_q into moments $m_i(p)$, $m_i(q)$, and $m_i(I_q)$ respectively gives

$$\begin{aligned} \frac{1}{R_{pq} + SL_{pq}} &= \sum_i (-1)^i \frac{i!L_{pq}^i}{R_{pq}^{i+1}} s^i \\ \sum_{(p,q)\in E} \frac{1}{R_{pq}} m_j(q) &= -C_q m_{j-1}(q) - m_j(I_q) + \\ \sum_{(p,q)\in E} (\frac{1}{R_{pq}} m_j(p) &+ \sum_{i=1}^{j} (-1)^i (\frac{i!L_{pq}^i}{R_{pq}^{i+1}})(m_{j-i}(p) - m_{j-i}(q))) \end{aligned} \tag{6}$$

3.2 Random Walk in an RLC Network

Rewriting (6) as follows enables a random walk game.

$$m_j(q) \;=\; Prob(p,q)m_j(p) + A(q) \tag{7}$$

where

$$\begin{aligned}
Prob(p,q) &= \frac{R_{pq}^{-1}}{\sum_{(p,q)\in E} R_{pq}^{-1}} \\
A(q) &= \sum_{(p,q)\in E} \frac{1}{R_{pq}^{-1}} \sum_{i=1}^{j} (-1)^i (\frac{i! L_{pq}^i}{R_{pq}^{i+1}})(m_{j-i}(p) \\
&\quad -m_{j-i}(q)) - \frac{C_q m_{j-1}(q) + m_j(I_q)}{\sum_{(p,q)\in E} R_{pq}^{-1}}
\end{aligned}$$

In such a random walk game, a traveler pays an amount of $A(q)$ for lodging at node q, with a probability $Prob(p,q)$ of going to an adjacent node p, until he reaches home or a V_{dd} node. At each step, we have $m_j(q) = m_j(p) + A(q)$, where $A(q)$ includes low order moments and is known if we compute moments in an increasing order. Averaging over a number of random walks with $Prob(p,q)$ gives the same moment $m_j(q)$ as in (6).

4 Stochastic Moment Matching Implementation

4.1 Algorithms

Algorithm 3: P/G Analysis by SMM Method (Algorithm II)
Input: RLC P/G network, V_{dd} nodes V, current sources S
Output: Node voltage drops

1. For each current source node $s \in S$
2. Walk from s to a V_{dd} node with $Prob(p,q)$
3. For each node q in path from node v to node s
4. For each moment order j
5. Compute $m_j(q)$ according to (6)
6. Collect node moments
7. Compute poles/residues by moment matching
8. Calculate time domain waveforms and voltage drops

We implement the following three algorithms with different netlist traversal and moment computation methods, and compare their solution convergence in Section 5, which shows Algorithm II gives the best solution convergence.

I. A random walk based on transition probability $Prob(p,q)$; when visiting each node, moments are updated according to (7) with $Prob(p,q) = 1$. Using the principle of dynamic programming, we compute moments of all nodes in a random path, rather than generating a random path starting at each node.

II. A random walk based on transition probability $Prob(p,q)$; when visiting each node, moments are updated according to (6). We observe better solution convergence when updating moments as in (6) instead of $m_j(q) = m_j(p) + A(q)$.

III. In applying Liebmann's method of solving partial differential equations [6], we iteratively update moments for all nodes according to (6). [2]

4.2 Numerical Stability of SMM

Moments are sensitive to numerical stability. To achieve better accuracy, it is critical to compute moments of all orders of a node based on the same random walk process (see our implementation in Algorithm II, where the iteration over moment orders is the innermost loop). Taking averaged moments over different random walk processes does not help in improving solution accuracy.

It is also helpful to reduce numerical instability accumulation during moment computation, e.g., by reducing the necessary number of moment computation processes. Moments are converted to poles and residues by either single-node moment matching [15] or multi-node moment matching [9]. Multi-node moment matching has increased accuracy with increased number of inputs, which makes it preferred to apply to practical P/G networks. We implement single-node moment matching on three orders of poles and residues [15] with a scaling factor of $\frac{|m_0|}{|m_1|}$ for numerical stability improvement [7].

Random walk provides a framework to reduce inherent numerical instability and improve accuracy of moment computation based analysis approaches. E.g., we filter out potentially inaccurate moment computation solutions, e.g., with zero moments (of a node which may not be visited in a random walk), zero voltage drops, and positive poles during random walk processes. Averaging over a large number of remaining solutions provides increasingly accurate analysis results.

The total runtime of supply voltage drop analysis via SMM includes $O(MP)$ for random walk, where M is the number of moments, and P is the average path length for a random walk process; $O(r^3)$ for transforming moments into poles and residues, where r is the number of poles/residues, and $r = 3$ in our implementation; and an additional $O(L)$ time for binary search for the lowest voltage in a time domain waveform. The runtime is dominated by the average random walk path length P, which increases with instance size, but also decreases with an increased number of P/G supply pads. Hence, random walk techniques demonstrate better scalability with increased number of P/G supply pads.

5 Experimental Results

We construct our power supply network test cases as subgraphs of grids, where each grid segment consists of a resistor and a possible inductor, each grid node is attached with a ground capacitor, and possible voltage and current sources (Fig. 1). The initial power/ground network is stabilized at the supply voltage before current sources are activated. Upon activation, a current source injects a triangle waveform supply current to model toggling of a transistor [14].

Our first test case is a 10×10 power supply grids with a single power supply pad and a single current source at each of two opposite corners of the grid. Each segment in the grid has a uniform resistance of $1K\Omega$, and each node in the grid

[2] In solving the Dirichlet problem $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$, whose finite-difference form is $u(i-1,j) + u(i+1,j) + u(i,j-1) + u(i,j+1) - 4u(i,j) = 0$, Liebmann's method is to systematically approximate each interior grid point by taking the average of its four neighbors. It does not matter in what order this process is carried out. After a few iterations, this process will converge to an approximate solution of the problem [6].

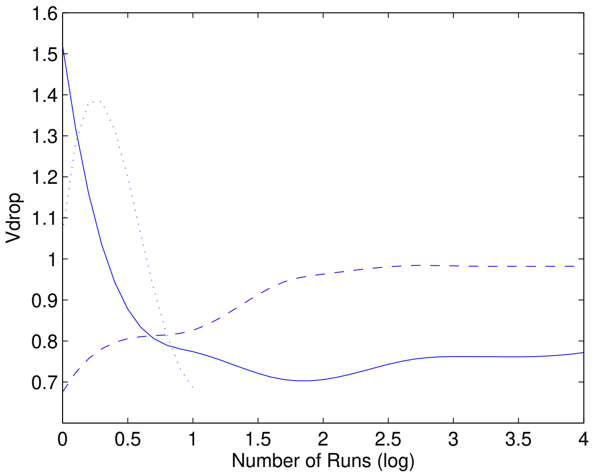

Fig. 3. Solution convergence of three moment computation algorithms: Algorithm I (in solid curve); Algorithm II (in dashed curve); and Algorithm III (in dotted curve). All supply voltage drop V_{drop} estimates are normalized to SPICE results.

has a uniform ground capacitance of $1pF$. The current source has a transition time $T_r = 1ns$ and a peak current of $1mA$. We compute the worst case supply voltage drop at the current source node.

We compare performance of the three algorithms in Section 4.1 by finding the average solution of i runs for each algorithm, where $i = 1, 10, 100, 1000$ or 10000. Fig. 3 shows that *Algorithm II has the best rate of convergence with respect to SPICE results*, while Algorithm I and III show much slower rates of convergence to the SPICE result, e.g., Algorithm III of Liebmann's method takes much longer runtime so that we only apply it for small numbers of runs. We apply Algorithm II for the rest of the experiments.

To access scalability, we apply SMM to power grids of different sizes with different number of supply voltage nodes. In the next experiment, we have a single current source located at a corner of a $N \times N$ power grid, where $N = 10, 20, 50$ or 100. We also have a $G \times G$ grid of power supply nodes, where $G = 1, 2, 3$, or 4. When $G = 1$, a single power supply node is located at the opposite corner to the current source. Otherwise, $G \times G - 1$ power supply nodes and the single current source are on the grid. Resistance on each segment is $1K\Omega$, and ground capacitance at each node is $1pF$. The current source provides a supply current in a triangle waveform with a transition time $T_r = 1ns$ and a peak current of $1mA$. We report supply voltage drop V_{drop} at the current source node and CPU runtime on a $i686$ computer with a $2.8GHz\ P4$ processor and $512MB$ memory in Table I. We observe that *SMM runtime increases with increased power grid dimension, and decreases with increased number of power supply nodes.* With more power supply nodes as possible destinations, random walks are more likely to be confined in a local region, which helps to improve efficiency and scalability of practical P/G network analysis.

We apply SMM to a randomly generated power supply network. The generated power supply network is a subgraph of a 100×100 grid, which models a top level power supply network with possible hard macro blocks as routing obstacles. Resistance on each segment ranges from 100Ω to $1K\Omega$, ground capacitance at each node ranges from $0.1pF$ to $1.0pF$, inductance on each segment ranges from $0.1pH$ to $1.0pH$. 100 supply voltage V_{dd} nodes are randomly located in the network. Each current source has a transition time T_r from $0.5ns$ to $2.5ns$, and a peak current from $0.5mA$ to $2.0mA$. Fig. 4 compares supply voltage drop estimates by 1000 SMM runs with SPICE simulation results, where each dot represents a sampling power network node, its x coordinate gives the SPICE simulated node voltage, while its y coordinate gives the SMM estimated node voltage. The SMM estimated voltage drops show good correlation with the SPICE results.

TABLE I
CPU RUNTIME (IN SECONDS) AND ESTIMATES OF SUPPLY VOLTAGE DROP AT THE CURRENT SOURCE NODE (NORMALIZED BY SPICE RESULTS) BY 50 RUNS OF SMM IN A $N \times N$ POWER GRID WITH A $G \times G$ POWER SUPPLY NODE (PAD) GRID.

	CPU	V_{drop}	CPU	V_{drop}	CPU	V_{drop}	CPU	V_{drop}
N \ G	1		2		3		4	
10	0.141	0.944	0.065	1.088	0.042	1.104	0.036	1.116
20	0.478	0.945	0.211	1.044	0.090	1.100	0.064	1.114
50	5.537	0.846	1.860	0.984	0.441	1.026	0.261	1.027
100	23.083	0.908	7.788	0.925	1.972	0.972	1.146	1.018

TABLE II
CPU RUNTIME (IN SECONDS) AND SUPPLY VOLTAGE DROP ESTIMATES (NORMALIZED BY SPICE RESULTS) OF SMM AND TRANSIENT RANDOM WALK (TRW) IN A RANDOMLY GENERATED POWER SUPPLY NETWORK.

	method \ node	1	2	3	4	5	6	7	8
CPU	SMM	12.8	7.3	9.5	12.8	4.4	4.6	6.9	12.4
Vdrop		1.05	0.97	0.94	1.04	0.97	0.96	1.03	1.04
CPU	TRW	142.1	141.5	139.3	135.0	192.6	107.6	100.3	100.6
Vdrop		1.12	1.15	1.09	1.21	1.32	1.09	0.94	1.07

Table II compares runtime of SMM with transient random walk (TRW) [17]. Each SMM run includes 100 iterations; each TRW run includes 100 iterations for each time step, each time step is of $5ps$. Although TRW saves runtime by not going through moment computation and Laplace transformation, and applying a bookkeeping process which reuses random walk paths in previous time steps instead of generating new random paths for each time step, SMM is much more efficient by not going through multiple time steps. We also observe SMM achieves better accuracy than TRW by avoiding the inaccuracy accumulation in going through multiple time steps.

Our last test case is a power supply netlist which includes 25118 ground capacitors and 65403 resistors, and is extracted using Cadence Fire&Ice from an industry design of $90nm$ technology with $13K$ components. Table III shows that SMM achieves increased estimation accuracy with increased iteration number, while the runtime and the accuracy of a random walk moment computation process vary with the distance between an observation node and a P/G supply pad. For this test

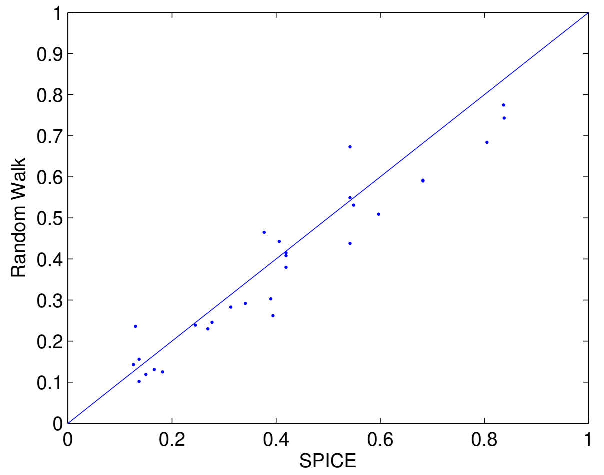

Fig. 4. Supply voltage drop (V) by 1000 SMM runs vs. SPICE results in a randomly generated power supply network.

case, standard moment computation methods [5] cannot be applied due to their scalability limits; SPICE simulation takes 579.50 seconds to perform $5000ps$ transient analysis in $1ps$ time steps; while random walk based frequency domain analysis provides accurate estimation with less runtime. SMM is especially efficient for instances with a large number of P/G supply pads.

TABLE III
CPU RUNTIME (IN SECONDS) AND SUPPLY VOLTAGE DROP ESTIMATES (NORMALIZED BY SPICE RESULTS) IN A PRACTICAL POWER SUPPLY NETLIST IN $90nm$ TECHNOLOGY WHICH IS EXTRACTED BY CADENCE FIRE&ICE.

	#iter \ node	1	2	3	4	5	6	7	8
CPU	10	6.12	8.60	4.74	5.88	4.93	2.12	6.78	7.90
Vdrop		0.90	1.13	0.49	0.84	0.48	0.59	1.36	0.96
CPU	100	65.32	71.83	89.02	66.98	92.30	29.04	46.71	74.66
Vdrop		1.01	0.99	0.93	0.96	0.94	0.92	0.94	0.96

6 Conclusion

In this paper, we proposed stochastic moment matching method (SMM) for analyzing large power deliver networks in time domain. The new method combines the stochastic random walk process with efficient moment matching technique in frequency domain. It achieves better efficiency than the existing transient analysis random walk method by using small number of moments instead of large number of time steps. SMM also extends the classic moment matching technique to achieve better scalability for linear circuits with multiple inputs/outputs, e.g., P/G delivery networks. In averaging over a large number of solutions, SMM provides an easy tradeoff between efficiency and accuracy.

By filtering out possible inaccurate solutions, SMM overcomes the inherent numerical instability of moment matching techniques and provides increasingly accurate analysis results. Experimental results show that the proposed SMM method can lead to 10X-100X speedup over LU-based simulation methods and about 10X speedup over the pure random walk method with reasonable accuracy for large industry power/ground networks. Further speedup can be obtained by trading more accuracy. This makes the proposed method very suitable for voltage drop estimation of power/ground networks at various physical stages for different accuracy requirements.

ACKNOWLEDGMENTS

The authors thank anonymous reviewers of an earlier draft for their helpful comments.

References

[1] G. Bai, S. Bobba and I. N. Hajj, "Simulation and Optimization of the Power Distribution Network in VLSI Circuits," in *Proc. International Conference on Computer-Aided Design*, pp. 481-486, 2000.

[2] G. Bai, S. Bobba and I. N. Hajj, "Static Timing Analysis Including Power Supply Noise Effect on Propagation Delay in VLSI Circuits," in *Proc. Design Automation Conference*, 2001.

[3] S. Bobba and I. N. Hajj, "Estimation of Maximum Current Envelope for Power Bus Analysis and Design," in *Proc. International Symposium on Physical Design*, pp. 141-146, 1998.

[4] Y. L. Le Coz, D. Krishna, D. M. Petranovic, W. M. Loh and P. Bendix, "A Sum-Over-Paths Impulse-Response Moment-Extraction Algorithm for IC-Interconnect Networks: Verification, Coupled RC Lines," in *Proc. ICCAD*, pp. 665-670, 2003.

[5] Y.-M. Lee, Y. Cao, T.-H. Chen, J. Wang and C. C.-P. Chen, "HiPRIME: Hierarchical and Passivity Preserved Interconnect Macromodeling Engine for RLKC Power Delivery," in *IEEE Trans. on Computer-Aided Design of Integrated Circuits And Systems (TCAD)*, 24(6), 2005, pp. 797-806.

[6] S. J. Farlow, *Partial Differential Equations for Scientists and Engineers*, Dover Publications, 1993.

[7] P. Feldmann and R. W. Freund, "Efficient Linear Circuit Analysis by Pade Approximation via the Lanczos Process," *IEEE Trans. on Computer-Aided Design on Integrated Circuits and Systems*, 14(5), pp. 639-649, 1995.

[8] W. Guo, S. X.-D. Tan, Z. Luo, X. Hong, "Partial Random Walk for Large Linear Network Analysis," in *Proc. International Symposium on Circuits and Systems (ISCAS)*, pp. 173-176, 2004.

[9] Y. Ismail, "Efficient Model Order Reduction via Multi-Node Moment Matching," in *Proc. International Conference on Computer-Aided Design*, pp. 767-774, 2002.

[10] J. Kozhaya, S. R. Nassif and F. Najm, "A Multigrid-like Technique for Power Grid Analysis," in *IEEE Trans. on Computer-Aided Design*, 21(10), pp. 1148-1160, 2002.

[11] H. Li, Z. Qi, S. X-D. Tan, L. Wu, Y. Cai, and X. Hong, "Partitioning-Based Approach to Fast On-Chip Decap Budgeting and Minimization," in Proc. *Design Automation Conference*, 2005, pp. 170-175.

[12] S. R. Nassif and J. N. Kozhaya, "Fast Power Grid Simulation," in *Proc. Design Automation Conference*, pp. 156-161, 2000.

[13] A. Odabasioglu, M. Celik and L. T. Pileggi, "PRIMA: Passive Reduced-Order Interconnect Macromodeling Algorithm," in *Proc. Intl. Conf. on Computer-Aided Design*, pp. 58-65, 1997.

[14] R. Panda, D. Blaauw, R. Chaudhry, V. Zolotov, B. Young and R. Ramaraju, "Model and Analysis for Combined Package and On-Chip Power Grid Simulation," in *Proc. International Symposium on Low Power Electronics and Design*, pp. 179-184, 2000.

[15] L. T. Pillage, R. A. Rohrer, "Asymptotic Waveform Evaluation for Timing Analysis", *IEEE Trans. on Computer-Aided Design*, 9(4), pp.352-366, 1990.

[16] L. T. Pillage, R. A. Rohrer and C. Visweswariah, *Electronic Circuit and System Simulation Methods*, McGraw-Hill, Inc., 1994.

[17] H. Qian, S. R. Nassif and S. S. Sapatnekar, "Random Walks in a Supply Network," in *Proc. DAC*, pp. 93-98, 2003.

[18] H. Qian and S. S. Sapatnekar, "Hierarchical Random-Walk Algorithms for Power Grid Analysis," in *Proc. ASP-DAC*, pp. 499-504, 2004.

[19] H. Qian, S. R. Nassif and S. S. Sapatnekar, "Power Grid Analysis Using Random Walks," in IEEE Trans. CAD, vol. 24, no. 8, August 2005.

[20] Synopsys, "PrimeTime-SI User Manual," 2004.

[21] M. Zhao, R. V. Panda, S. S. Sapatnekar, T. Edwards, R. Chaudhry and D. Blaauw, "Hierarchical Analysis of Power Distribution Networks," in *Proc. Design Automation Conference*, pp. 150-155, 2000.

[22] S. Zhao, K. Roy and C.-K. Koh, "Frequency Domain Analysis of Switching Noise on Power Supply Network," in *Proc. International Conference on Computer-Aided Design*, pp. 487-492, Nov. 2000.