

Statistic Analysis of Power/Ground Networks Using Single-Node SOR Method*

Zuying Luo¹, Sheldon X.-D. Tan²

¹ Dept. of Electronics, Beijing Normal University, Beijing 100875, P.R. China

² Dept. of Electrical Engineering, University of California at Riverside, CA 92521, USA

Email: luozy@bnu.edu.cn

Abstract

In this paper, we propose an efficient statistical analysis method for analyzing on-chip power grids. The new method, called SN-SOR (and its faster version, PSN-SOR), is based on a novel localized relaxed iterative approach and it can perform variational analysis on one node at a time. PSN-SOR further speeds up the analysis by using a refined conditioner, where the initial solution of SN-SOR is used as the pre-conditioner for the later iterations. Experimental results show that PSN-SOR is about two orders of magnitude (186X) faster than Monte-Carlo method with slight errors less than 5.685% on maximum and is about one order magnitude (41X) faster than general global successive over relaxation (SOR) method. PSN-SOR is more accurate and efficient than the recently proposed random walk method for localized statistical analysis.

1. Introduction

Process-induced variability becomes the major design concern in the current 65 nm and coming 45nm VLSI technologies [1]. The process-induced variations manifest themselves from wafer to wafer, die-to-die and device to device within a die [2,3]. Given the high sensitivity of today's chip performance to even minor voltage fluctuations, it is important to characterize the impact of process variations on on-chip power grids.

Several recent work considered the influence of process variations on power grids. Reference [4] propose a matrix-conversion based method to analyze variances of IR droops on resistor-only networks. In [5,6], impulse responses are used to compute the means and variances of node voltage responses due to general current variations. But this method needs to know the impulse response from all the current sources to all the nodes, which is expensive to compute for a large network. Method based on orthogonal polynomials of statistical processes has been proposed also [7]. But those methods still can't solve a few nodes in large power grid networks. Recently a statistical analysis method based on a random-walk (RW)

concept [8] was proposed to analyze voltage variance for a few nodes of large networks [9]. This method explores the local nature of voltage changes due to current sources in today's flip-chip technique so that it can efficiently analyze voltage variance for a single node. But the RW-based method does not consider the spatial correlations in intra-die variations.

In this paper, we first propose a novel single-node SOR method (SN-SOR) to efficiently analyze the variances of node voltages in P/G networks based on the global SOR method [10]. In SN-SOR, only one stimulus is applied at a node, saying q , to solve the response vector R_q . SN-SOR performs relaxation on nodes from q to its surround nodes in a wave-transmission style and the wave stops at some nearby nodes whose voltage drops are less than a pre-defined value. SN-SOR dynamically reduces the number of nodes used in the relaxation process based on their value changes during process. As number of relaxation nodes will become less as SOR iterations proceed, SN-SOR becomes more efficient than the RW method in the sense that it only *walks* partial paths while the RW method has to walk through the whole paths (from sources to some voltage-known nodes).

Similar to the SOR method [10], initial states of all nodes are assigned as V_{DD} value in the SN-SOR solving process. To further speed up the process, a pre-conditioned SN-SOR method (PSN-SOR) is further proposed. We first use SN-SOR to solve the response vector R_{qt} for a node qt close to the center of the P/G network. Then the position deviation is added and R_{qt} replaces V_{DD} as initial states in the R_{qr} SN-SOR solving processes for most remaining nodes with the similar topology as qt . Because R_{qt} is much closer to R_{qr} than V_{DD} value, PSN-SOR needs far less relaxation iterations (about 45%) than SN-SOR. Thus, PSN-SOR is more efficient than SN-SOR.

Both SN-SOR and PSN-SOR can efficiently compute voltage variation for few problematic nodes in P/G networks due to process variations. Experiment results show that the proposed method compares very favorably with the existing methods. Compared to the RW method, SN-SOR and PSN-SOR only relax 38.17% and 20.59%

* This work was supported in part by National High-Tech Research and Development (863) Program of China (No. 2007AA01Z109, No. 2006AA01Z223), NSF CAREER Award under grant CCF-0448534, NSFC(No.60473032, No.40672195) and NCET-06-0131.

nodes and can reduce the average error from 0.887% to 0.139% and 0.173% respectively. Compared to the Monte Carlo (5000 samplings) simulation, two methods are 139X and 186X faster with small accuracy losses. Compared to the previous global SOR method, SN-SOR and PSN-SOR can be 31X and 41X faster with about 1.041% errors. SN-SOR and PSN-SOR only takes 2.374 and 2.059 seconds in average respectively to statistically analyze one node for the circuit with 1.44 million nodes.

2. Background

In high-performance chips, P/G networks are always of mesh topology for high reliability [11], where arrays of floatable C4-PADs supply power to P/G networks. Shown in Fig.1, square nodes are C4-PADs and C4-PADs are organized into an array. As a result, a P/G network linked to a $(N_2+1) \times (N_2+1)$ array is divided into $M = N_2 \times N_2$ blocks. Thus, any node in the P/G network can get current from its neighbor PADs. And center nodes of blocks are marked as circle nodes in Fig.1 and usually suffer from large IR droops because they are far from its neighbor C4-

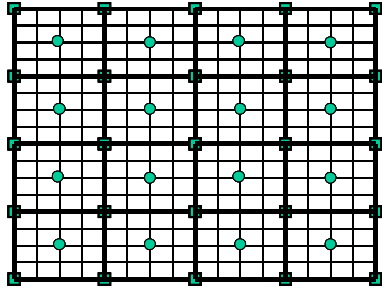


Figure 1. P/G meshes supplied by C4-PAD arrays

PADs.

A P/G network is a linear system whose input is the current source vector b and output is the nodal voltage vector x . The modified nodal analysis (MNA) method, $Gx = b$, is widely used to solve P/G networks, where G is the conductance matrix. In this paper, we focus on the DC analysis. The proposed method can be extended to the transient analysis for RC/RLC circuits as shown in [8]. The ICCG algorithm [12] is an iterative algorithm to efficiently solve the above MNA equation. Recently a global SOR algorithm was proposed for the efficient P/G analysis [10].

With technology scaling into nanometer regime, both absorption current b and conductance G in the P/G analysis are disturbed by rampant process variations. We model the variations as the first-order disturbance. Thus $Gx = b$ must be transformed for managing disturbances Δb and ΔG as follows.

$$(G_0 + \Delta G) \times (x_0 + \Delta x) = b_0 + \Delta b \quad (1)$$

where b_0 , G_0 , x_0 are nominal values of variations b , G , x

and Δx is the voltage response for Δb and ΔG . As a result, we can use Δb and ΔG to compute the Δx as following

$$\begin{aligned} \Delta x &= G_0^{-1} \cdot \Delta b - G_0^{-1} \{ \Delta G (x_0 + \Delta x) \} \\ &= G_0^{-1} \cdot \Delta b - G_0^{-1} \{ \Delta G \cdot x \} \end{aligned} \quad (2)$$

In this work, variations of absorption currents and conductance follow the normal distribution variations $N(b, \mu, b\sigma)$ and $N(g, \mu, g\sigma)$. We assume that two kinds of disturbances are independent. To consider the spatial correlations, we use a simple correlation model: all variations of absorption currents in a block are absolutely correlated while spatial correlations exist between different blocks. The disturbances of conductance are spatially correlated too. But we stress that our method can be used for more complicated correlation models.

3. Single-node statistical analysis method

(1) Response vector solving. Because the only unit stimulus (1A) is put at node q , the response vector R_q is defined as the nodal voltage vector x_q excited by the unit stimulus:

$$x_q = G^{-1} \cdot b_q = R \cdot b_q = R_q \quad (3)$$

where $R = G^{-1}$ is the resistance matrix, b_q is the input stimulus vector in which all elements are zero except for node q , which has value 1 in b_q . Although ICCG [12] and SOR [10] methods can solve R_q directly, they both become inefficient for solving a few node voltage as they have to solve for the whole circuit. The variance-computing RW method [9] is a localized single-node solving method but it still less efficient since it has to walk through the whole paths from the source node q to terminal nodes.

(2) Single-node P/G analysis due to Δb . According EQ(2), we can directly compute the standard voltage deviation $x\sigma_q$ of node q due to Δb as following.

$$(x\sigma_q)^2 = \sum_{bi=1}^M \left\{ x\sigma_{q,bi} \sum_{bj=1}^M (C_{bi,bj} \cdot x\sigma_{q,bj}) \right\} \quad (4)$$

where $M = N_2 \times N_2$ is the number of blocks, $C_{bi,bj}$ is the correlated coefficient between blocks bi, bj , and $x\sigma_{q,bi}$ is the contribution term of block bi to $x\sigma_q$. Because two nodes in a same block are totally correlated, $x\sigma_{q,bi}$ can be simply computed as follows:

$$x\sigma_{q,bi} = \sum_{p \in N_{bi}} (r_{p,q} \cdot b\sigma_p) \quad (5)$$

where N_{bi} is the node set of block bi and node p belongs to N_{bi} . $b\sigma_p$ is the standard current deviation of node p and $r_{p,q}$ is the resistance between nodes p, q . And $r_{p,q}$ is a p^{th} element in R_q .

(3) Single-node P/G analysis due to ΔG . According to EQ(3), we need the iterative method to compute $x\sigma_q$ caused by ΔG . Because $x = x_0 + \Delta x$, we have to use x_0 to

compute the initial value $\Delta x^{(0)} = -G^{-1} \times (\Delta G \times x_0)$ according to EQ(4)-EQ(5). After obtaining $\overline{x\sigma_i^{(k)}}$ as $k \geq 0$, we can refresh the equivalent current variation $\overline{b\sigma^{(k)}}$ as following. We first compute the ratio $\beta^{(k)} = \overline{x\sigma_q^{(k)}} \div \overline{x\sigma_q^{(0)}}$ and then refresh the current input variable $\overline{b\sigma_i^{(k+1)}} = \beta^{(k)} \bullet \overline{b\sigma_i^{(0)}}$. With $\overline{b\sigma_i^{(k+1)}}$, we can further obtain $\overline{x\sigma_i^{(k+1)}}$ and then $\beta^{(k+1)}$. As $\left| \overline{x\sigma_i^{(k+1)}} - \overline{x\sigma_i^{(k)}} \right|$ is less than an pre-defined small value ε_4 , we end the iteration and treat the $\overline{x\sigma_i^{(k+1)}}$ as $x\sigma_q$.

4. SN-SOR statistical method

After obtaining the nominal voltage vector x_0 , all nodes with large IR droops are marked as problematic nodes. The following SN-SOR algorithm is detail described for solving the response vectors, with which we can directly compute the standard voltage deviations.

Step1: Initialize the circuit. Voltage of each node is pre-defined as V_{DD} . Node q of the only 1A stimulus is stored into relaxation set A as an edge node.

Step2: If all nodes in A have been relaxed, turn to step 7. Else, sequentially take a node p from A and go to the next step.

Step3: If node p is a relaxation-ending node, return to step 2. Else, go to the next step.

Step4: Relax p and then do next according to following three cases. Case one, if p is an edge node and $\overline{\Delta x_p^{(k+1)}} = |x_p^{(k+1)} - x_p^{(k)}|$ is less than an pre-defined value $\varepsilon_2 \ll 1.0$, return to step 2. Case two, if p is an edge node and $\overline{\Delta x_p^{(k+1)}} \geq \varepsilon_2$, mark p as an internal node and go to step 5. Case three, if p is an internal node, go to step 6.

Step5: For the new-marked internal node, mark all un-relaxed neighbor nodes as edge nodes and push these new edge ones into A . Then, return to step 2.

Step6: if $\overline{\Delta x_p^{(k+1)}}$ is less than an pre-defined value $\varepsilon_3 \ll 1.0$ (in this work, $\varepsilon_3 = 0.1\varepsilon_2$), mark p as a relaxation-ending node and return to step 2.

Step7: If errors of all nodes are less than a pre-defined value ε_1 , end the algorithm and output the solution. Else, relax A again and return to step 2.

To demonstrate the effectiveness of SN-SOR, we use SN-SOR to calculate response vectors and observe the convergence and the number of the relaxation nodes of the method. With the increasing of the circuit size, the SOR iteration number increases too. But the iteration number of SN-SOR reach to 154 for the largest circuit of 1.44M nodes. Fig.2 show the convergence rate for the circuit with 202.44k nodes. Shown in the left hand side of

Fig.2, the number of SN-SOR iterations is 63 and the nodal voltage quickly converges to the final value before 22th iterations.

In Step4, if node p is marked as an edge node, p can't be further marked as an internal node in Step4 if the resistance $r_{p,q}$ is small enough. Thus, the relaxation wave will stop at the edge node, which limits relaxation only to some nodes around q . Shown in the right part of Fig.2, SOR nodes from this wave-ending technique are named as the candidate SOR nodes. Because the number of the candidate SOR nodes speedily converges to a number far less than the circuit nodes as iteration continues, the wave-ending technique in Step4 can limit SN-SOR into a small region.

If an internal node is marked as a relaxation-ending node in Step6, the node will not be relaxed in later SOR iterations according to the operation in Step3. More internal nodes will be marked as relaxation-ending ones in later SOR iterations. Thus, SN-SOR is much more efficient than the global SOR method. Shown in the right part of Fig.2, the SOR nodes, which are not relaxation-ending nodes, are named as the computation SOR nodes. With SN-SOR iteration increasing, the number of the computation SOR nodes reach to the maximum: 19774, at the 12th iteration and then it goes down to 1357 in the end. Compared with RW method [9] that always walks through paths from the source node to terminal nodes, SN-SOR only *walks* partial distances after excluding

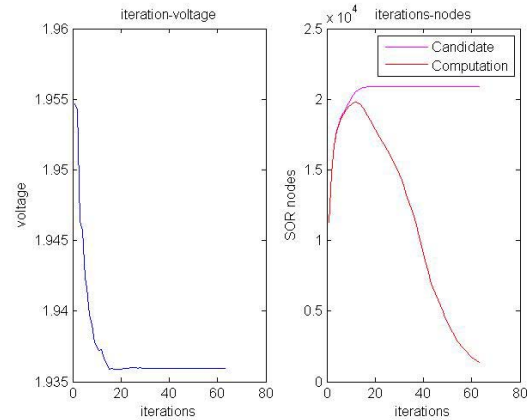


Figure 2. Convergence rate of SN-SOR relaxation-ending nodes in later SOR iterations.

For circuit with 202.44k nodes, we use SN-SOR to calculate voltage standard deviations (st.d) of 900 problematic nodes. Compared with the 5000 Monte-Carlo(MC) simulation method, we obtain the relative errors in standard deviation shown in Fig.3. Among all problematic nodes, most have relative errors from -2.6% to -2.8%. Thus, SN-SOR is accurate enough for practical P/G analysis.

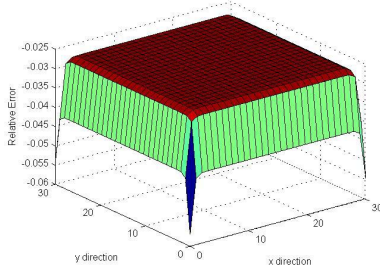


Figure 3. Errors of SN-SOR

5. PSN-SOR: pre-conditional SN-SOR statistical method

The SN-SOR method can work on all P/G networks with floatable C4-PADs, in which these C4-PADs are organized into any topologies. As to the regular topology of the C4-PAD array which is usually used in many P/G networks of high-performance chips, we improve the SN-SOR method and propose the pre-conditional SN-SOR method (PSN-SOR) that is more efficient than SN-SOR.

In P/G networks with a C4-PAD array shown in Fig.1, most of problematic nodes have similar topology around them. In PSN-SOR method, it first chooses the node q_c close to the center of a P/G network to compute the response vector R_{q_c} using SN-SOR. Then the remaining problematic nodes except border nodes are solved with the PSN-SOR algorithm in which R_{q_c} is used as the pre-condition. Finally, we use SN-SOR to solve all border problematic nodes. With these response vectors, we can directly compute the standard voltage deviations for these nodes. The errors given by PSN-SOR are shown in Fig.4. Comparing Fig.3 and Fig.4, we find that PSN-SOR is of

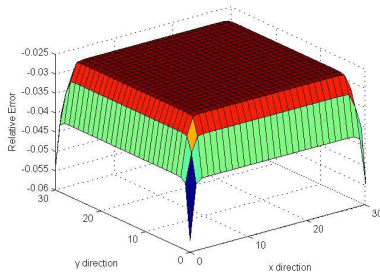


Figure 4. Errors of PSN-SOR

the nearly same accuracy as SN-SOR.

Compared with SN-SOR that use the V_{DD} as the initial states for all nodes, PSN-SOR is efficient because the pre-condition is close to the final result and less relaxation iterations are needed. To show the fast convergence rate of PSN-SOR, we use the method to compute response vector for a problematic node q in the circuit with 202.44k nodes used in Fig.2. The result is shown in left

hand side of Fig.5, since the node has the similar topology as the original node q_c , the voltage of node q nearly keeps constant and the iterations number is 43. Comparing right hand side parts of Fig.2 and Fig.5, both the number of candidate SOR nodes and the number of computation SOR nodes have been visibly reduced owing to the pre-

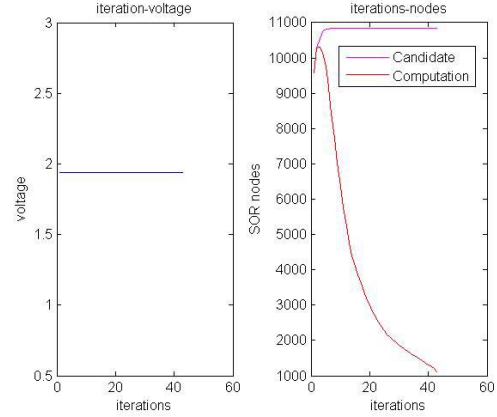


Figure 5. Convergence rate of PSN-SOR

condition.

6. Experimental Results

All experimental circuit includes 900 blocks. Each block has one problematic node. Each node is linked to a current source with the normal distribution $N(0.001,0002)$ and each conductance has the normal distribution $N(R_0,0.2 R_0)$. All variances in a same block are absolutely correlated. And there are spatial correlations among blocks. All the testing are performed on a desktop computer with 2×CPU(3.00GHz)-1G(memory). The program has been implemented in C.

In order to show the effectiveness of SN-SOR and PSN-SOR, we compare it with 5000 Monte-Carlo(MC) simulation method, the global SOR solving method, the global ICCG solving method. In Monte-Carlo (MC) simulation method, we sample 5000 times. It takes T_{sim} runtime to obtain standard deviations (Std) of all nodal voltages for accuracy comparison. For other methods, including Global ICCG, Global SOR, SN-SOR, and PSN-SOR, they directly compute standard voltage deviations(SVD) for 900 problematic nodes. The runtime to compute 900 SVDs is named as T_{std} .

(1) Efficiency comparisons are shown in Table.1. (A) Compared with the MC method, global methods of SOR and ICCG are far less efficient than the SN-SOR and the PSN-SOR methods. For the second circuit with 202.44k nodes, SN-SOR and PSN-SOR can be 139X and 186X times faster while the global SOR and ICCG methods can only deliver speedup 6.79 and 7.88 respectively as shown in columns 4,6,8,10. (B) Compare with the global SOR method, PSN-SOR can be 41X times faster for the circuit

with 518.88k nodes. (C) For the largest circuit of 1.44M nodes, PSN-SOR only takes 1853.09 seconds (30.9 minutes) to solve standard voltage deviations of 900 problematic nodes while SN-SOR takes a little more runtime 2136.84 seconds (35.6 minutes) in column 7.

(2) Accuracy comparisons are shown in Table.2. We not only care the average relative errors of 900 problematic nodes but also the maximum errors of them. (A) For two small-scale circuits, MC method is used as the golden to compute relative errors of other methods. All analytic methods show good accuracy with the average errors 2.381% - 2.890% and the maximum errors 2.672% - 5.695%. (B) For two medium-scale circuits with 360.24k and 518.88k nodes, the global SOR method is used as the comparison base. The localized SN-SOR and PSN-SOR methods are accurate with the maximum errors 0.029% - 1.041%. (C) Compared with SN-SOR for three large-scale circuits of up to 1.44M nodes, PSN-SOR has far less both average errors 0.077% - 0.331% and the maximum errors 0.776% - 1.159%.

(3) In order to explain reasons of high efficiency of localized SN-SOR and PSN-SOR methods, we compare three SOR solving methods: the previous global SOR, localized SN-SOR and PSN-SOR methods in Table.3. (A) Comparing the T_{std} per node(T_{std}/pn), both SN-SOR and PSN-SOR are much more efficient than the global SOR method. For the circuit with 518.88k nodes, the global SOR method must take 24.053 seconds to analyze one problematic node while SN-SOR and PSN-SOR only takes 0.752 and 0.575 second respectively. And for the largest circuit with 1.44M nodes, SN-SOR and PSN-SOR also only takes 2.374 and 2.059 seconds respectively to analyze a problematic node. (B) Compared the average nodes relaxed in a SOR iteration ($nodes/it$) term, SN-SOR and PSN-SOR become efficient owing to the locality. Both methods only relax nearly 5% nodes while the global SOR method must relax all nodes. (C) Compared with average SOR iterations per problematic nodes (nit) of the global SOR method in columns 3, PSN-SOR reduces nearly 50% nit in column 9 owing to the pre-condition while SN-SOR only slight improve the nit in column 6.

(4) we compare the efficiency of SN-SOR and PSN-SOR with the random walk(RW) method [9] as they are localized methods. We compare the computation of the voltage expectation $x\mu_q$ of a single problematic node q . After the mean value, a limited additional runtime is needed for both methods to compute the standard voltage deviation $x\sigma_q$, it is fair to compare them through $x\mu_q$ computation. The RW method takes 10000 walks and the longest walk is pre-defined as $10 \times ((N_2-1) \times (N_1-1)+1)$. Combining the comparison results in efficiency and accuracy in Table.4, SN-SOR and PSN-SOR are better than the existing RW method because they only need relax 38.17% and 20.59% nodes and can reduce the

average error from 0.887% to 0.139% and 0.173% respectively.

7. Conclusion

In this paper, we have proposed SN-SOR and PSN-SOR for efficient statistical analysis of on-chip power grids. Based on a localized relaxed iterative approach to perform variational analysis on one node at a time, SN-SOR and PSN-SOR are about two orders of magnitude faster than Monte-Carlo method with small errors and are about one order magnitude faster than general global successive over relaxation (SOR) method [10]. Thus, they are accurate and efficient to analyze fewer problematic nodes for a large-scale practical P/G network.

References

- [1] T.~W. Williams, "EDA to the rescue of the silicon roadmap," *Proc. ISQED*, keynote speech, 2007.
- [2] S. Nassif, "Design for variability in DSM technologies," *Proc. ISQED*, pp. 451-454, 2000.
- [3] S.B Samaan, "The impact of device parameter variation on the frequency and performance of VLSI chips," *Proc. ICCAD*, pp. 343-346, 2004.
- [4] I.A Ferzli, F.N Najm, "Statistical Verification of Power Grids Considering Process-Induced Leakage Current Variations," *Proc. ICCAD*, pp. 770-777, 2003.
- [5] S. Pant, D. Blaauw, V. Zolotov, et al, "A stochastic approach to power grid analysis," *Proc. DAC*, pp. 171-176, 2004.
- [6] A. Srivastava, R. Bai, D. Blaauw, et al, "Modeling and analysis of leakage power considering within-die process variations," *Proc. ISLPED*, pp. 64-67, 2002.
- [7] N. Mi, S. X.-D. Tan, P. Liu, et al, "Stochastic extended Krylov subspace method for variational analysis of on-chip power grid networks," *Proc. ICCAD*, pp. 48-53, 2007.
- [8] H.-F Qian , R Nassif , S.S Sapatnekar, "Random walks in a supply Network," *Proc. DAC*, pp. 93-98, 2003.
- [9] P Li, "Variational analysis of large power grids by exploring statistical sampling sharing and spatial locality," *Proc. ICCAD*, pp. 644-650, 2005.
- [10] Y Zhong, D.-F Wong, "Fast Algorithms for IR Drop Analysis in Large Power Grid," *Proc. ICCAD*, pp. 351-357, 2005.
- [11] E. Chiprout, "Fast Flip-chip Power Grid Analysis Via Locality and Grid Shells," *Proc. ICCAD*, pp. 485-488, 2004.
- [12] X.-H Wu, X.-L Hong, Y.-C Cai, et al, "Area Minimization of Power Distribution Network Using Efficient Nonlinear Programming Techniques," *IEEE TCAD*, Vol. 23, Issue 7, pp. 1086-1094, 2004.

Table 1.
Comparison on efficiency amid different methods

nodes	Global SOR		Global ICCG		SN-SOR		PSN-SOR	
	Ave. Err.	Max. Err.	Ave. Err.	Max. Err.	Ave. Err.	Max. Err.	Ave. Err.	Max. Err.
89640	2.381%	2.672%	2.382%	2.673%	2.388%	2.681%	2.331%	2.679%
176280	2.838%	5.693%	2.837%	5.693%	2.854%	5.695%	2.890%	5.695%
360240	Base	Base	0.000%	0.001%	0.024%	0.029%	0.079%	0.687%
518880	Base	Base	0.006%	0.006%	0.172%	0.189%	0.168%	1.041%
810840	NA	NA	NA	NA	Base	Base	0.077%	0.861%
1041480	NA	NA	NA	NA	Base	Base	0.138%	1.159%
1441440	NA	NA	NA	NA	Base	Base	0.331%	0.776%

Table 2.
Comparison on accuracy amid different methods

nodes	MC method	Global SOR		Global ICCG		SN-SOR		PSN-SOR	
	$T_{sim}(s)$	$T_{std}(s)$	speedup	$T_{std}(s)$	speedup	$T_{std}(s)$	speedup	$T_{std}(s)$	speedup
89640	6736.38	1006.61	6.69	850.36	7.92	75.84	88.82	53.86	125.07
176280	24961.03	3678.02	6.79	3167.89	7.88	178.53	139.81	133.53	186.93
360240	NA	10395.20	Base	7421.45	1.40	400.45	25.96	287.64	36.14
518880	NA	21647.72	Base	14313.83	1.51	676.84	31.98	517.89	41.80
810840	NA	NA	NA	NA	NA	1146.20	Base	798.33	1.44
1041480	NA	NA	NA	NA	NA	1705.89	Base	1237.48	1.38
1441440	NA	NA	NA	NA	NA	2136.84	Base	1853.09	1.15

Table 3.
Detail efficiency comparison of three SOR methods

nodes	Global SOR			SN-SOR			PSN-SOR		
	$T_{std}/pn(s)$	nit	nodes/ it	$T_{std}/pn(s)$	nit	nodes/ it	$T_{std}/pn(s)$	nit	nodes/ it
89640	1.118	42.00	89640	0.084	42.22	4786.5	0.060	15.16	5125.4
176280	4.087	58.24	176280	0.198	63.36	10198.9	0.148	22.03	12637.3
360240	11.550	83.92	360240	0.445	83.44	17117.8	0.320	31.50	22735.6
518880	24.053	109.49	518880	0.752	103.41	23739.7	0.575	49.30	27770.6
810840	NA	NA	NA	1.274	116.71	37503.7	0.887	52.39	40476.6
1041480	NA	NA	NA	1.895	136.79	43965.5	1.375	77.10	41803.3
1441440	NA	NA	NA	2.374	157.24	48720.7	2.059	79.95	72499.5

Table 4.
Efficiency comparisons with the RW method

nodes	10000 Random-Walk		SN-SOR			PSN-SOR		
	Avr. Err.	nodes/pn	Avr. Err.	nodes/pn	ratio	Avr. Err.	nodes/pn	ratio
89640	0.807%	926,460	0.000%	202,095	21.81%	0.004%	77,701	8.39%
176280	0.779%	2,183,309	0.015%	646,248	29.60%	0.055%	278,342	12.75%
360240	0.791%	3,991,288	0.021%	1,428,234	35.78%	0.077%	716,114	17.94%
518880	0.903%	6,326,452	0.154%	2,454,901	38.80%	0.155%	1,369,223	21.64%
810840	0.798%	9,239,225	0.140%	4,377,102	47.38%	0.207%	2,120,532	22.95%
1041480	1.002%	12,624,249	0.294%	6,014,232	47.64%	0.337%	3,222,958	25.53%
1441440	1.126%	16,583,440	0.347%	7,661,053	46.20%	0.375%	5,796,357	34.95%
Average	0.887%	100.000%	0.139%		38.173%	0.173%		20.593%