

Parallel Algorithms for Power and Signal Integrity Analysis of High-Speed Designs

R. Achar, H. Dhindsa, N. Nakhla, D. Paul and M. Nakhla

*Department of Electronics, Carleton University
1125 Colonel By Dr, Ottawa, Ontario K1S5B6, Canada*

Abstract— Modern VLSI designs are posing ever increasing challenges for computer-aided design tools. This paper provides an overview of recent developments in parallel algorithms for signal and power integrity analysis in high-speed designs. Efficient partitioning and simulation algorithms have been developed employing waveform relaxation iterations for application to signal and power integrity analysis. Unlike direct solvers, the described algorithms are highly parallelizable and yield significant speed-ups. Numerical examples are presented to demonstrate the validity and efficiency of the presented methods.

I. INTRODUCTION

The recent trend in the VLSI industry toward miniature designs, low power consumption, and increased integration of analog circuits with digital blocks has made the signal and power integrity analysis a challenging task. The quest for high-speed applications has highlighted the previously negligible effects of interconnects such as ringing, signal delay, distortion, reflections, and crosstalk. Interconnects can exist at various levels of design hierarchy such as on-chip, packaging structures, multichip modules, printed circuit boards, and backplanes. It is predicted that interconnects will be responsible for majority of signal degradation in high-speed systems [1].

High-speed interconnect problems are not always handled appropriately by conventional circuit simulators, such as SPICE. If not considered during the design stage, these interconnect effects can cause logic glitches that render a fabricated digital circuit inoperable or they can distort an analog signal such that it fails to meet specifications. Since extra iterations in the design cycle are costly, accurate prediction of these effects is a necessity in high-speed designs. Hence, it becomes extremely important for designers to simulate the entire design along with interconnect subcircuits as efficiently as possible while retaining the accuracy of simulation.

The major difficulty in simulating high-frequency interconnect networks is due to the fact that, while described in terms of partial differential equations, distributed nature of interconnects is best represented in the frequency-domain and they do not have a direct representation in the time-domain. On the other hand, nonlinear devices can only be described in the time-

domain. These simultaneous formulations are difficult to handle using a traditional ordinary differential equation solver such as SPICE. Several methods can be found in literature to transform the interconnect models into a set of coupled ordinary differential equations, so as to be compatible with circuit simulators [2].

Also, the coupling between the lines is one of the major causes of the high computational cost of simulating large interconnect networks. For example, a 64-line system requires the approximation of 8320 transfer functions. Each of these approximations must not only be accurate, but the resulting functions must preserve the passivity and the causality of the overall macromodel. The average cost of simulating an N coupled line circuit is proportional to N^β where $3 < \beta < 4$. This results in a prohibitively time-consuming simulation task compared to the simple case of simulating a single line [3].

In addition, preserving power integrity has become one of the most challenging tasks facing designers of modern multifunction and miniature electronic products. As devices scale and more transistors are integrated into a single integrated circuit, the power and current levels are expected to increase with a corresponding decrease in voltage. With gigabit signals being propagated through the package and board, the ability to supply clean power to the transistor circuits becomes very critical. These issues make the design of robust and reliable power distribution networks (PDNs), consisting of chips, packages and printed circuit boards, essential to the success of the high-speed products [4]-[6].

Modelling of PDNs in modern VLSI circuits leads to large circuits, often consisting of millions of circuit elements. Hence, the simulation of PDNs becomes computationally expensive and time consuming. Several techniques have been proposed in the literature to address this issue. In general, these techniques can be categorized into *direct-solver* methods [4-5] and *iterative* methods [6-7]. In [4], it was shown that direct solvers offer speed-ups over iterative solvers for the analysis of power grids, where very smooth spatial variations in potential are observed. However, these methods suffer from high memory consumption and are unyielding to implementation in parallel domains.

Iterative solvers, on the other hand, converge very slowly but are memory efficient.

In this paper, an overview of recent developments in parallel algorithms for signal and power integrity analysis in high-speed designs is presented. The described algorithms are based on the concept of waveform relaxation using transverse partitioning [3], [8-13]. The presented algorithms are hybrid in nature and exploit the advantages of both classes of methods (i.e. direct solvers and iterative solvers). In addition, emphasis is placed on partitioning schemes as well parallelization of simulation tasks. The described techniques reduce the large/coupled simulation problem into a series of simulation steps of smaller complexity. In addition, these steps are performed in parallel and, thus, providing further significant reduction in the computational time.

The rest of the paper is organized as follows. Section II describes high-speed interconnect analysis based on waveform-relaxation and transverse partitioning. Section III describes the power integrity analysis based on waveform-relaxation and block-row partitioning. Section IV presents the numerical results.

II. WAVEFORM RELAXATION AND TRANSVERSE PARTITIONING FOR SIGNAL INTEGRITY ANALYSIS

In this section, high-speed interconnect analysis based on waveform-relaxation and transverse partitioning is described.

II.A Formulation of MTL Equations

Consider the MTL system with N conductors. The transmission line can be represented using Telegraphers equations as [2]

$$\frac{\partial}{\partial z} \begin{bmatrix} \mathbf{v}(z, t) \\ \mathbf{i}(z, t) \end{bmatrix} = - \begin{bmatrix} \mathbf{0} & \mathbf{R} \\ \mathbf{G} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v}(z, t) \\ \mathbf{i}(z, t) \end{bmatrix} - \begin{bmatrix} \mathbf{0} & \mathbf{L} \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \frac{\partial}{\partial t} \begin{bmatrix} \mathbf{v}(z, t) \\ \mathbf{i}(z, t) \end{bmatrix} \quad (1)$$

where vectors $\mathbf{v}(z, t)$ and $\mathbf{i}(z, t)$ represent the voltage and current along the transmission line. $\mathbf{R} \in \mathbb{R}^{n \times n}$, $\mathbf{L} \in \mathbb{R}^{n \times n}$, $\mathbf{G} \in \mathbb{R}^{n \times n}$ and $\mathbf{C} \in \mathbb{R}^{n \times n}$ are the per unit length (p.u.l) parameter matrices of the MTL. Using the terminal conditions, the solution of (1) in frequency domain can be written as

$$\begin{bmatrix} \mathbf{V}(d, s) \\ \mathbf{I}(d, s) \end{bmatrix} = e^{(A+sB)d} \begin{bmatrix} \mathbf{V}(0, s) \\ \mathbf{I}(0, s) \end{bmatrix} \quad (2)$$

where d is the length of the MTL and

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & -\mathbf{R} \\ -\mathbf{G} & \mathbf{0} \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} \mathbf{0} & -\mathbf{L} \\ -\mathbf{C} & \mathbf{0} \end{bmatrix}$$

This matrix exponential form in (2) does not have a direct time domain representation. Conventional methods, such as lumped segmentation, when applied to distributed MTLs result in large circuits or need higher order approximation which makes them computationally inefficient [1].

II.B Waveform Relaxation and Transverse Partitioning for Analysis of Large coupled Interconnects

The problem of large coupled Interconnects is addressed via waveform relaxation and transverse partitioning where the original simulation task is divided into number of smaller problems (subcircuits) which can be solved in parallel [3]. The interaction between the subcircuits is represented using relaxation sources. The subcircuits are solved independently, assuming initial values for relaxation sources and later updating the values of relaxation sources using solutions from previous iterations. This process is repeated until the solution convergence. Exploiting the weak coupling between the lines, MTLs are divided into subcircuits using transverse partitioning, where each individual line is considered as a subcircuit. The subcircuit containing single line can be described by rewriting (2) as

$$\frac{\partial v_j}{\partial z} = -r_{jj}i_j - l_{jj}\frac{\partial i_j}{\partial t} - \sum_{\substack{k=1 \\ k \neq j}}^N (r_{jk}i_k + l_{jk}\frac{\partial i_k}{\partial t}) \quad (3)$$

$$\frac{\partial i_j}{\partial z} = -\hat{g}_{jj}v_j - \hat{c}_{jj}\frac{\partial v_j}{\partial t} - \sum_{\substack{k=1 \\ k \neq j}}^N (g_{jk}(v_j - v_k) + l_{jk}(\frac{\partial v_j}{\partial t} - \frac{\partial v_k}{\partial t})) \quad (4)$$

where $\hat{g}_{jj} = \sum_{k=1}^N g_{jk}$ and $\hat{c}_{jj} = \sum_{k=1}^N c_{jk}$ are the self admittance and capacitance of the line, respectively. The summation terms in (2) and (4) represents the coupling effect of other lines on j th line. The equations (3) and (4) can be rewritten as

$$\frac{\partial v_j}{\partial z} = -r_{jj}i_j - l_{jj}\frac{\partial i_j}{\partial t} + e_j(z, t) \quad (5)$$

$$\frac{\partial i_j}{\partial z} = -\hat{g}_{jj}v_j - \hat{c}_{jj}\frac{\partial v_j}{\partial t} + q_j(z, t) \quad (6)$$

where $e_j(z, t)$ and $q_j(z, t)$ represent the coupling between the lines and are defined as

$$e_j(z, t) = - \sum_{\substack{k=1 \\ k \neq j}}^N (r_{jk} i_k + l_{jk} \frac{\partial i_k}{\partial t}) \quad (7)$$

$$q_j(z, t) = - \sum_{\substack{k=1 \\ k \neq j}}^N (g_{jk} (v_j - v_k) + l_{jk} (\frac{\partial v_j}{\partial t} - \frac{\partial v_k}{\partial t})) \quad (8)$$

The relaxation sources can be calculated by solving coupled differential equations (5)-(8) in frequency domain as

$$\delta_j(z, s) = \left[\frac{V_{src}(z, s)}{I_{src}(z, s)} \right] = \int_0^z e^{F_j(z-\eta)} \Psi_j(\eta, s) \quad (9)$$

where

$$F_j(s) = \begin{bmatrix} 0 & -r_{jj} - s l_{jj} \\ -\hat{g}_{jj} - s \hat{c}_{jj} & 0 \end{bmatrix}$$

$$\Psi_j(\eta, s) = \sum_{\substack{k=1 \\ k \neq j}}^N \begin{bmatrix} 0 & -r_{jk} - s l_{jk} \\ -g_{jk} - s c_{jk} & 0 \end{bmatrix} \begin{bmatrix} V_j(z, s) - V_k(z, s) \\ I_k(z, s) \end{bmatrix}$$

Further details on computing and updating the sources based on desired type of macromodels can be found in [3].

Correspondingly, each line is simulated independently on a different processor, during each WR iteration. Multiple independent simulations can be executed in parallel [13]. After simulating each line, the corresponding relaxation sources are updated to prepare for the next iteration. Once this is complete for all of the lines, the next iteration can begin.

III. POWER GRID ANALYSIS AND WAVEFORM RELAXATION TECHNIQUES

In this section, a brief review of power grid modelling and analysis issues is given. Also power integrity analysis based on waveform-relaxation and block-row partitioning is described.

III.A Power Grid Analysis and Problem Formulation

In this section, a brief review of power grid modeling and analysis issues is given. Typically, power distribution within an integrated circuit is done hierarchically, starting from the top-level metal layer, which is connected to the package, down through interlayer vias, and finally to the active devices. Adjacent layers have metal lines laid out at right angles to each other. The power is fed to the top layers through external voltage sources called V_{DD} contacts. The functional blocks such as logic gates, memory units, latches etc. in the active layers act as power drains, drawing current from the PDN at the bottommost layer.

The type of models that are used to describe power grids depends on the purpose, accuracy and efficiency desired by the designer. Since the on-chip metal interconnects are predominantly resistive and capacitive in nature, RC models are typically employed to describe the power grid in our analyses. Metal segments between vias are modeled as lumped resistors. The via resistances are typically ignored [4]. Capacitances are connected between each node in the grid and the ground to account for the capacitive effects in the PDN. V_{DD} contacts from the package, which are placed on top metal layer at uniform intervals in a flip-chip package, are modeled as ideal voltage sources. The nonlinear switching devices (i.e. active power drains) are replaced by time-varying current sources consisting of static and dynamic switching currents drawn by these devices.

Next, the MNA-based transient behavior of such an $N \times N$ grid can be described using the following system of ordinary differential equations:

$$GX(t) + CX(t) = Bu(t) \quad (1),$$

where X is the vector of unknown node voltages, G is a block-tridiagonal conductance matrix of the power grid circuit, C is a diagonal matrix of the capacitances at each node, $u(t)$ is the vector of inputs and B is a selection matrix. Fig.1 shows a typical RC structure resulting from power grid extraction. For modern integrated circuits such as microprocessors, a typical power grid network can easily include millions of nodes. Conventional circuit simulation techniques are inadequate to handle such a large problem size.

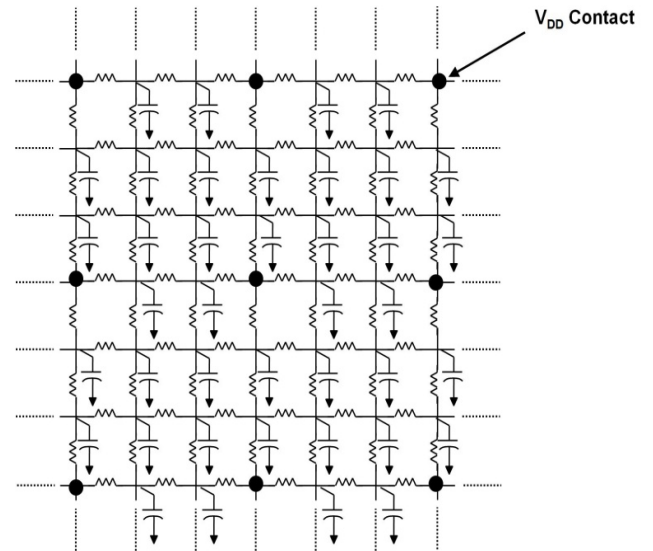


Figure 1: A schematic description of a power grid mesh

III.B Waveform Relaxation for Power Integrity Analysis

To address the issues of large size of power grid networks, waveform relaxation (WR) techniques have been recently developed [10-12]. Here, the given large power grid circuit is partitioned into smaller subcircuits. The coupling between the various subcircuits is represented by WR sources which can be voltage or current sources depending on the type of coupling between the subcircuits.

The rate of convergence of the WR algorithm depends largely on the strength of the coupling between the subcircuits, the type of partitioning, the method used for relaxation and the physical properties of the original circuit. There are two commonly used variations of waveform relaxation techniques: 1) Gauss Jacobi (GJ) and 2) Gauss Seidel (GS). In Gauss-Jacobi relaxation the WR sources are updated at the end of the iteration, while in Gauss-Seidel relaxation the sources are updated immediately after a subcircuit is simulated.

III.C Block Row Partitioning

The partitioning of the problem plays an important role in the performance of WR based algorithms. The idea is to partition the power grid circuit along the weak coupling between the subcircuits, while enabling parallel transient analysis. For this purpose, the given circuit is partitioned into subcircuits consisting of blocks of rows, where the boundaries of the partition correspond to the V_{DD} contacts in the original power grid, as shown in Fig. 2 [12]. Waveform relaxation sources are attached (which accounts for the coupling between the neighboring BRP blocks) at the boundary nodes.

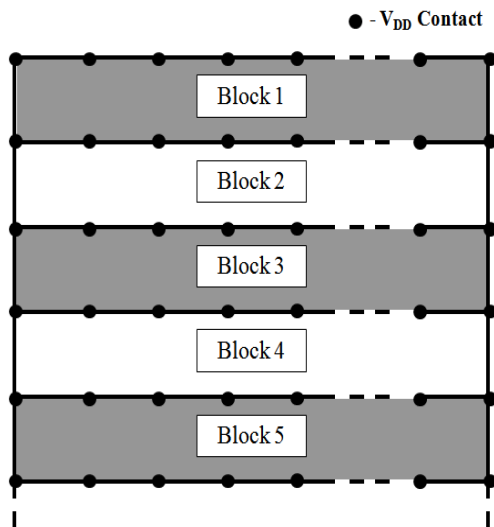


Figure 2: BRP scheme for power grid analysis

III.D Parallel Power Grid Analysis

In this section, parallel transient analysis algorithm for power grids is described. The algorithm divides the power grid circuit into smaller subcircuits and applies parallel Gauss-Seidel waveform relaxation on these subcircuits. In the case of power grids, partitioning the power grid circuit into subcircuits consisting of blocks of rows (BRP scheme) gives us the opportunity to parallelize even the GS-WR algorithm. This is because, the coupling of individual subcircuits is limited to the immediate neighboring subcircuits. Exploiting this aspect and suitably altering the order of subcircuits to be simulated (within a given iteration), parallel algorithm is developed follows.

In the algorithm, the odd blocks are simulated concurrently during the first phase of a given iteration. Once the solutions for these subcircuits have been updated, the even blocks are simulated concurrently during the second phase of the iteration. This results in a re-ordered parallel Gauss-Seidel relaxation process. Within each phase, individual subcircuits are simulated independently and simultaneously by different processors, without affecting the Gauss-Seidel relaxation process (since in a given iteration, the solution of odd blocks are not affected by the solution of other odd blocks and the same holds for even blocks).

III.E Time Partitioning for Enhanced Parallelization

To enhance the CPU utilization of the proposed parallel GS-WR algorithm, physical- and time-partitioning-based pipelining techniques are developed [12-13]. Next, the simulation tasks are scheduled efficiently on all the processors such that the processor idle time is minimized. For this purpose, given simulation time is divided into several small time-partitions. It is to be noted that, in a transient simulation, the solution of a particular time partition of a given subcircuit depends upon the solution of the subcircuit at the end of the previous time partition. Hence, these two operations in a given iteration must always be in sequence. The following conditions have to be met for assigning a particular simulation task to an available processor in the proposed GS-WR algorithm with physical and time partitioning:

- (a) For odd subcircuits, for simulating a specific time partition (say T_p) of a particular subcircuit (say i):
 1. The simulation of the previous time partition (T_{p-1}) of this subcircuit (i) must have been completed in the current iteration.

2. The simulation of this time partition (T_p) of this subcircuit (i) must have been completed for the previous iteration.
3. The simulation of this time partition (T_p) of the neighboring even subcircuits ($i-1$ and $i+1$) must have been completed for the previous iteration.

Similar conditions can be developed for even numbered blocks. The number of time partitions can be so chosen (equal to number of processors available) such that the CPU utilization of the proposed parallel GS-WR algorithm is always 100%.

IV. NUMERICAL RESULTS

In this section, numerical results validating the parallel algorithms described in earlier sections are provided. The parallel algorithm was implemented on a C++ based linear circuit simulator (internal) with OpenMP multithreading extensions. This circuit simulator uses backward Euler integration method for time-domain analysis. The KLU package (direct solver) [16] is used to solve the resulting system of linear equations. All the experiments and results reported in this section have been executed on the high performance computing virtual laboratory's (HPCVL) [17] SunFire cluster machines. The SunFire cluster is a symmetric multiprocessor (SMP) system running Solaris 10 UNIX. The cluster used for simulation in this experiment contained 72X dual core UltraSPARC-IV+ 1.5 GHz processors with 576 GB of RAM.

To validate the performance of the parallel waveform relaxation based algorithm, a large industrial power grid circuit, PG_2 , was chosen [18]. PG_2 is a power grid structure generated from a 7-metal layer PDN. It belongs to a flip-chip package based microprocessor of dimension 10mm X 10mm, built around a 65nm process. The V_{DD} contacts are made at about 300 μ m on the surface of the chip, at the top metallization layer. The resulting circuit PG_2 contained 7.5 million nodes.

This circuit was simulated using both the proposed parallel WR algorithm and the direct-LU solver (KLU package [16]). It was found that the proposed parallel algorithm converges to within 0.5mV of the exact solution of circuit PG_2 , in five iterations. This is illustrated in Fig. 4, via the transient voltage plot at a sample node in circuit PG_2 .

Next, the scalability of both these methods is investigated by running the experiment on different number of processors and the corresponding CPU costs are given in Table 1 and the resulting speed-ups are plotted against the number of CPUs, in Fig. 5. As seen, there is a significant reduction in the simulation times as a result of using the proposed parallel algorithm, with the increase in the number of processors.

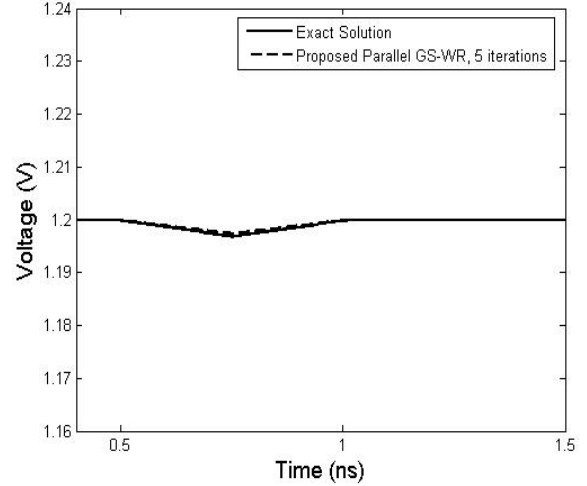


Figure 4: Transient convergence waveform at a sample node in circuit PG_2 .

V. CONCLUSIONS

An overview of recent developments in parallel algorithms for signal and power integrity analysis is presented in this paper. Efficient partitioning and simulation algorithms have been described employing waveform relaxation iterations for application to signal and power integrity analysis. Unlike direct solvers, the described algorithms are highly parallelizable and yield significant speed-ups.

Table 1: CPU cost comparison for simulating circuit PG_2 on a parallel platform

Circuit PG_2 : # of nodes = 7.5 millions			
Proposed parallel WR method		Direct-LU method (hr:min:sec)	Speed-up w.r.t Direct-LU method
# of CPUs	CPU time (hr:min:sec)		
1	3:31:10	8:14:07	2.34
3	1:14:38		6.62
5	0:45:12		10.93
6	0:40:43		12.14
8	0:30:39		16.12
10	0:24:45		19.96
12	0:20:14		24.41
16	0:16:49		29.39
18	0:14:58		33.01

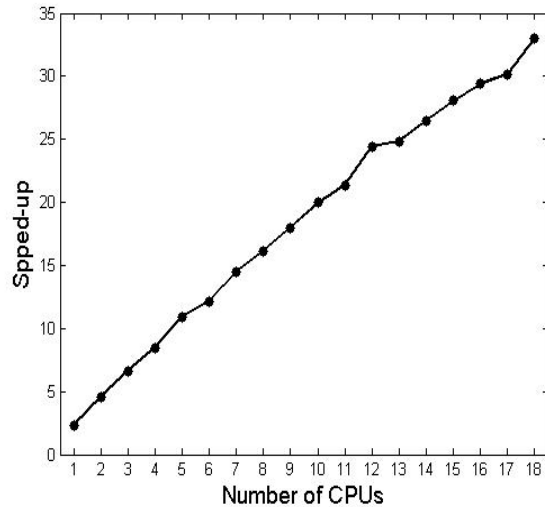


Figure 5: Speed-ups for circuit PG₂ using the proposed parallel WR algorithm with respect to direct-LU solver versus varying number of processors.

ACKNOWLEDGMENT

The authors would like to thank Koko Mihan of Javelin Design Automation, Ottawa, Eli Chiprout of Intel, Oregon, and Albert Ruehli of IBM T. J. Watson Research Center, NY, for their valuable inputs and suggestions.

REFERENCES

- [1] R. Achar and M. S. Nakhla, "Simulation of high-speed interconnects," *Proc. IEEE*, vol. 89, pp. 693–728, May 2001.
- [2] C. R. Paul, *Analysis of Multiconductor Transmission Lines*. New York: Wiley, 1994.
- [3] N. Nakhla, A. Ruehli, M. Nakhla, and R. Achar, "Simulation of coupled interconnects using waveform relaxation and transverse partitioning," *IEEE Trans. Adv. Packag.*, vol. 29, no. 1, pp. 78–87, Feb. 2006.
- [4] J. N. Kozhaya, S. R. Nassif, and F. N. Najm, "A multigrid-like technique for power grid analysis," *IEEE Trans. Computer-aided Design*, vol. 21, pp. 1148–1160, Oct. 2002.
- [5] J. Silva, J. M. S. Phillips, R. Joel, and S. L. Miguel, "Efficient representation and analysis of power grids," *Proceedings of the Design, Automation and Test in Europe*, pp. 420–425, 2008.
- [6] Y. Zhong and M. D. F. Wong, "Efficient second-order iterative methods for IR drop analysis in power grid", *IEEE/ACM Conference on Computer-Aided Design*, pp. 768–773, San Jose, Nov. 2007.
- [7] H. Qian, S. Nassif, and S. Sapatnekar, "Power grid analysis using random walks", *IEEE Transactions on Computer-Aided Design*, vol. 24, no. 8, pp. 1204–1224, August 2005.
- [8] E. Lelarsmee, A. Ruehli and A. Sangiovanni-Vincentelli, "The Waveform Relaxation method for Time-Domain analysis of Large Scale Integrated Circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, pp. 676–682, No. 5, May 2005.
- [9] A. Ruehli and T. A. Johnson, *Circuit Analysis Computing by Waveform Relaxation in Encyclopedia of Electrical and Electronics Engineering*, J. Webster, Ed. New York: Wiley, 1999, vol. 3.
- [10] H. Dhindsa, A. Sridhar, R. Achar, M. Nakhla, and D. Paul, "Transient analysis of power grid networks via waveform relaxation techniques," in *Proc. IEEE MMT-S International workshop series on Signal Integrity and High-Speed Interconnects (IMWS)*, Guadalajara, Mexico, February 2009, pp. 91–94.
- [11] R. Achar, M. Nakhla, A. Sridhar, H. Dhindsa, and D. Paul, "Fast analysis of power distribution networks using waveform relaxation," in *IEEE workshop on Signal Propagation on Interconnects*, May 2009, pp. 1–4.
- [12] R. Achar, M. Nakhla, H. Dhindsa, D. Paul and N. Nakhla, "Parallel and Scalable Transient Simulator for Power Grids via Waveform Relaxation", *IEEE Trans. on VLSI*, In Press.
- [13] D. Paul, N. Nakhla, R. Achar and M. Nakhla, "Parallel Simulation of Massively Coupled Interconnect Networks", *IEEE Trans. Adv. Packag.*, In Press.
- [14] M. Fiedler, *Special Matrices and Their Applications in Numerical Mathematics*. New York: Dover, 2008.
- [15] Y. Saad, *Iterative Methods for Sparse Linear Systems*. Philadelphia: SIAM, 2003.
- [16] T. A. Davis, *Direct Methods for Sparse Linear Systems*, Philadelphia: SIAM, 2006.
- [17] High Performance Computing Virtual Laboratory, <http://www.hpcvl.org>.
- [18] Private communication: Javelin Design Automation, Ottawa, Canada.