A Sparsity-Aware MOR Methodology for Fast and Accurate Timing Analysis of VLSI Interconnects

Dimitrios Garyfallou, Charalampos Antoniadis, Nestor Evmorfopoulos and Georgios Stamoulis
Department of Electrical and Computer Engineering, University of Thessaly, Volos, Greece
Email: {digaryfa, haadonia, nestevmo, georges}@e-ce.uth.gr

Abstract—Signoff timing analysis is essential in order to verify the proper operation of VLSI circuits. As process technologies scale down towards nanometer regimes, the fast and accurate timing analysis of interconnects has become crucial, since interconnect delay represents an increasingly dominant portion of the overall circuit delay. It is a common view that traditional SPICE transient simulation of very large interconnect models is not feasible for full-chip timing analysis, while static Elmore-based methods can be inaccurate by orders of magnitude. Model Order Reduction (MOR) techniques are typically employed to provide a good compromise between accuracy and performance. However, all established MOR techniques result in dense system matrices that render their simulation impractical. To this end, in this paper we propose a sparsity-aware MOR methodology for the timing analysis of complex interconnects. Experimental results demonstrate that the proposed method achieves up to 30x simulation time speedups over SPICE transient simulation of the initial model, maintaining a reasonable typical accuracy of 4%.

Index Terms—Circuit Simulation, MOR, Timing Analysis, Interconnect

I. Introduction

Timing closure is the major milestone which dictates when a chip can be released to the semiconductor foundry for fabrication. As a consequence, static and dynamic timing analysis (STA/DTA) are critical parts of the design flow. There are two components in the signal path delays, which are the gate delay and the interconnect delay. In nanometer process technologies, interconnect delay has become the decisive factor of performance, since it contributes almost 70% of the total delay [1]. This is due to the increased interconnect average length and routing density, and the fact that interconnect capacitance dominates gate capacitance. Thus, we cannot neglect the effects of interconnect parasitics in timing analysis.

Accuracy is the main consideration in signoff timing analysis. It is well known that existing fast delay estimation methodologies based on Elmore delay model [2] are inadequate for signoff timing analysis of interconnects, since they rely on simplistic assumptions that are invalid in deep submicron technologies. Another important aspect of timing analysis is the efficiency of the interconnect delay estimation method. Traditional transient analysis using SPICE-like simulators offers golden accuracy results, but fails to meet the performance and memory requirements for full-chip analysis of current designs which include very large interconnects.

Model Order Reduction (MOR) techniques have been employed to provide a good compromise between accuracy and performance. MOR techniques substitute the large scale parasitic network with a model of lower order with similar response at the input/output ports. However, all established MOR methodologies [3] [4] [5] result in dense system matrices that render their simulation impractical, since the simulation cost can easily overshadow the benefits obtained from dimension reduction. Previous methods attempting to address the problem of dense matrices resulting from MOR [6] [7] [8] do not produce very sparse and accurate models, or they rely on heuristics and circuit specific criteria. A rigorous mathematical approach for the sparsification of dense MOR circuit matrices was proposed only recently in [9]. The aforementioned method employs a sequence of algorithms based on the computation of the nearest Laplacian matrix and the subsequent sparsification of the corresponding graph.

In this paper, we present a sparsity-aware MOR methodology, based on [9], for the efficient signoff timing analysis of VLSI interconnects. When the proposed methodology is incorporated in iterative optimization flows, the convergence rate of the optimization can be improved due to the more accurate estimation of the critical paths [10]. On top of that, it can be embedded in a framework like [11] for the timing analysis of large interconnects with many ports (denoted hereafter as complex interconnects). In addition, the sparsified reduced order models have a straightforward realization to equivalent interconnect networks that can be dumped into a more compact Standard Parasitic Exchange Format (SPEF) file and be used in industrial design flows.

The rest of the paper is organized as follows. Section II provides some background material on the interconnect modelling and delay estimation methods. Section III presents the proposed sparsity-aware MOR methodology for the timing analysis of complex interconnects. In Section IV, we demonstrate the performance and accuracy of the proposed methodology. Finally, Section V concludes the paper.

II. On the Interconnect Delay Estimation

A. Electrical Modelling of an IC Interconnect

In digital designs, a wire connecting pins of gates (standard cells in EDA terminology) is referred to as an interconnect. An interconnect typically has only one driver (input port), while it can drive a number of fanout cells (output ports). For equivalent electrical representation, IC interconnects are typically represented by RC networks. For complex large interconnects driving a big number of cells, an accurate representation can be obtained by breaking its total capacitance C_{total} and total resistance R_{total} into multiple sections, creating a distributed
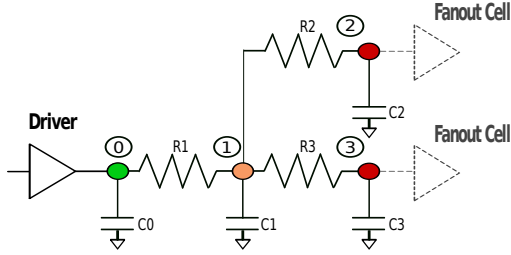
Fig. 1: A distributed RC interconnect of a gate driving a fanout of two cells.

RC-tree model as shown in Fig. 1. Such RC networks of current VLSI circuits may contain up to hundreds of thousands internal nodes, and thousands of output ports. Typically, the effect of inductance can be ignored within the chip and is only considered for package and board level analysis.

B. Interconnect Delay Estimation Methods

Elmore Delay Model is the most widely known static interconnect delay estimation method. For a distributed RC-tree (like the interconnect shown in Fig. 1), Elmore delay $T_{0 \to i}$ from the input port 0 to the output port i can be given by:

$$T_{0 \to i} = \sum_{k \in N} R_{ki} C_k$$

where N is the set of all nodes in the RC network, C_k is the lumped capacitance at node k, and R_{ki} represents the total resistance of the common path between the paths from input port 0 to node k and from input port 0 to output port i.

Although Elmore delay model is simple and fast to evaluate (linear time complexity with respect to the interconnect size), it considers only a step input ignoring effects that are input slew dependent. It is well established that this model can be off by orders of magnitude in some cases.

SPICE Transient Analysis of Initial Interconnect Model is the most accurate but time-consuming approach for output waveform calculation and delay estimation. Using the voltage waveform at the input port of the interconnect, which is computed during timing analysis of the previous gate, the voltage waveform on each output port can be obtained by SPICE transient analysis of the RC model. Given these waveforms, the corresponding delay is computed as the difference between the time instants when the output and input voltage crosses the 50% of the supply voltage.

Let the RC model of an interconnect be composed of n total nodes (excluding ground), where p nodes correspond to input and output ports. Given the excitation source vector of the circuit, the node voltages can be obtained by solving the following system of ordinary differential equations, arising after the Modified Nodal Analysis (MNA) [12]:

$$\mathbf{G}\mathbf{x}(t) + \mathbf{C}\frac{d\mathbf{x}(t)}{dt} = \mathbf{u}(t) \tag{1}$$

where $\mathbf{G} \in \Re^{n \times n}$ and $\mathbf{C} \in \Re^{n \times n}$ are the node conductance and capacitance matrices, $\mathbf{x} \in \Re^n$ is the vector of unknown node voltages, and $\mathbf{u} \in \Re^p$ is the vector of excitations from independent sources at the input ports. Note that the vector of excitations $\mathbf{u}(t)$ has only one non-zero entry, corresponding to the node where the independent current source is connected to, because the model of interconnects, as we described in the previous subsection, includes only one input port.

For the transient analysis, we can use the Backward-Euler numerical integration method to obtain a system of linear algebraic equations to be solved at any time instant [12]. Several efficient solution methods and preconditioning techniques have been proposed in the literature for solving such linear systems [13]. However, even such efficient simulation techniques cannot provide the performance required in timing analysis of large interconnects.

Model Order Reduction techniques have been employed in the past, as an alternative to the time-consuming SPICE transient analysis of the initial RC interconnect model of (1). MOR aims at approximating the initial model by another model of reduced order $r << n$, such that the input-output behavior is preserved. The MOR approximation is performed through a process of projecting the system matrices onto lower-dimensional subspaces by suitable projection matrices $\mathbf{U}, \mathbf{V} \in \Re^{n \times r}$:

$$\widehat{\mathbf{G}} = \mathbf{U}^T \mathbf{G} \mathbf{V}, \quad \widehat{\mathbf{C}} = \mathbf{U}^T \mathbf{C} \mathbf{V}$$

The biggest problem from the projections inherent in MOR is that sparsity of the matrices is lost, which can render impractical any time or frequency domain simulation and offsets the benefits from order reduction.

III. PROPOSED SPARSITY-AWARE MOR METHODOLOGY

A. Laplacian matrices

Laplacian matrices play a central role in the proposed methodology. Therefore, prior to the description of our methodology, we provide a definition of the Laplacian matrix. **Definition 1**: Let $G = (V, E, w, d)$ be a weighted undirected graph on the vertex set $V = \{1, 2, \ldots, n\}$ with no self-loops, and with weight functions $w : E \to \Re_{>0}$ and $d : V \to \Re_{\geq 0}$. The Laplacian of G is the matrix $\mathbf{G}_L \in \Re^{n \times n}$ such that:

$$\mathbf{G}_L(i,j) = \begin{cases} d(i) + \sum_{(i,k) \in E} |w(i,k)| & \text{, if } i = j \\ -w(i,j) & \text{, if } (i,j) \in E \\ 0 & \text{, otherwise} \end{cases} \tag{2}$$

Thus, any Laplacian matrix corresponds to a graph and vice versa.

B. Proposed Methodology

An intriguing methodology for the sparsification of dense matrices in MOR has been recently proposed in [9] (see Alg. 1). This methodology, firstly computes the nearest Laplacian matrices (step 1) to the *Reduced Order Model* (ROM) $(\widehat{\mathbf{G}}, \widehat{\mathbf{C}})$, let $\mathbf{G}_L, \mathbf{C}_L$, and then sparsifies those Laplacians by exploiting efficient spectral graph techniques, so that to preserve the eigenvalues of $\mathbf{G}_L, \mathbf{C}_L$ in between given error bounds.

The projection to the nearest Laplacian matrices at step 2 (depending on the chosen MOR technique) may induce

Algorithm 1 Sparsification of dense models resulting in MOR

1: **function** $\widehat{\mathbf{G}}_{sp}, \widehat{\mathbf{C}}_{sp} = \text{MORSPARSE}(\widehat{\mathbf{G}}, \widehat{\mathbf{C}})$
2:     $[\mathbf{G}_L, \mathbf{C}_L]$ = Compute the nearest Laplacian matrices to $\widehat{\mathbf{G}}_n, \widehat{\mathbf{C}}_n$ (steps 2-5 in Algorithm 6 [9]).
3:     $[\widehat{\mathbf{G}}_{sp}, \widehat{\mathbf{C}}_{sp}]$ = Sparsify $\mathbf{G}_L, \mathbf{C}_L$ (step 6 in Algorithm 6 [9])
4: **end function**

---

unacceptable error in the sparse ROM, especially when the number of port nodes is large. A MOR technique that produces model matrices (of a system with many ports that are) close enough to Laplacians is PACT [4].

PACT preserves the dominant poles of the initial model admittance matrix in the ROM admittance matrix, given an error control parameter and a frequency band of operation, by applying a series of matrix transformations.

Before applying the matrix transformations, PACT rearranges the equations of (1) so that the first p equations correspond to port nodes, while the rest of them correspond to internal nodes. Thus, (1) can be re-written as:

$$\begin{bmatrix} \mathbf{G}_P & \mathbf{G}_C^T \\ \mathbf{G}_C & \mathbf{G}_I \end{bmatrix} \begin{bmatrix} \mathbf{x}_P(t) \\ \mathbf{x}_I(t) \end{bmatrix} + \begin{bmatrix} \mathbf{C}_P & \mathbf{C}_C^T \\ \mathbf{C}_C & \mathbf{C}_I \end{bmatrix} \begin{bmatrix} \frac{d\mathbf{x}_P(t)}{dt} \\ \frac{d\mathbf{x}_I(t)}{dt} \end{bmatrix} = \mathbf{u}(t) \quad (3)$$

where $\mathbf{x}_P$ and $\mathbf{x}_I$ represent the p port nodes and the i internal node voltages respectively, $\mathbf{G}_P$ and $\mathbf{C}_P \in \Re^{p \times p}$ represent the interconnections between the port nodes, $\mathbf{G}_I$ and $\mathbf{C}_I \in \Re^{i \times p}$ describe the interconnections between the internal nodes and $\mathbf{G}_C$ and $\mathbf{C}_C \in \Re^{p \times i}$ represent the interconnections between the port nodes and the internal nodes. After the rearrangement of equations, PACT transforms $\mathbf{G}, \mathbf{C}$ in (3) as follows:

$$\mathbf{G}' = \mathbf{X}^T \mathbf{G} \mathbf{X} = \begin{bmatrix} \mathbf{G}_P - \mathbf{G}_C^T \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_i \end{bmatrix} = \begin{bmatrix} \mathbf{G}_P' & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_i \end{bmatrix}$$

$$\mathbf{C}' = \mathbf{X}^T \mathbf{C} \mathbf{X} = \begin{bmatrix} \mathbf{C}_P - \mathbf{B}^T \mathbf{A} - \mathbf{A}^T \mathbf{C}_C & \mathbf{B}^T \mathbf{L}^{-T} \mathbf{U} \\ \mathbf{U}^T \mathbf{L}^{-1} \mathbf{B} & \mathbf{\Lambda} \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{C}_P' & \mathbf{C}_C'^T \\ \mathbf{C}_C' & \mathbf{\Lambda} \end{bmatrix} \quad (4)$$

with

$$\mathbf{X} = \begin{bmatrix} \mathbf{I}_p & \mathbf{0} \\ -\mathbf{A} & \mathbf{L}^T \mathbf{U} \end{bmatrix}$$

where $\mathbf{A} = \mathbf{G}^{-1} \mathbf{G}_C$, $\mathbf{B} = \mathbf{C}_C - \mathbf{C}_I \mathbf{A}$, $\mathbf{L}$ is the lower triangular matrix from the Cholesky factorization of $\mathbf{G}_I$ ($\mathbf{G}_I = \mathbf{L}\mathbf{L}^T$) and $\mathbf{U}$ and $\mathbf{\Lambda}$ are the matrices with the eigenvectors as columns and a diagonal matrix with the eigenvalues from the eigendecomposition of matrix $\mathbf{C}_I' = \mathbf{L}^{-1} \mathbf{C}_I \mathbf{L}^{-T}$ ($\mathbf{C}_I' = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$), respectively.

Notice, that both internal matrices of (4) (the $i \times i$ lower right submatrices of $\mathbf{G}'$ and $\mathbf{C}'$) are diagonal, and thus the admittance matrix $\mathbf{Y}(s)$ of (3), which can be derived by obtaining the Laplace transform ($\mathbf{Y}(s)\mathbf{x}(s) = \mathbf{B}\mathbf{u}(s)$) and then eliminating $\mathbf{x}_I$ from the first p equations, is written as:

$$\mathbf{Y}(s) = \mathbf{G}_P' + s\mathbf{C}_P' - \frac{s^2 \mathbf{r}_1^T \mathbf{r}_1}{1 + s\lambda_1} - \dots - \frac{s^2 \mathbf{r}_n^T \mathbf{r}_n}{1 + s\lambda_n} \quad (5)$$

where $\mathbf{r}_i$ is the ith row of $\mathbf{C}_C'$ and λ_i is the ith diagonal of $\mathbf{\Lambda}$. Now, it is also shown in [4] that if the terms associated with λ_i in (5) are dropped when $\lambda_i < \lambda_c$, the relative error of each of the individual element of $\mathbf{Y}(s)$ is bounded on the complex axis for $|\omega| \le \omega_c$ by $\epsilon \le \epsilon_c$ if :

$$\epsilon_c = \omega_c \lambda_c + \omega_c^3 \lambda_c^3 \quad (6)$$

Finally, the ROM is built by eliminating the rows and columns of $\mathbf{G}'$ and $\mathbf{C}'$ for which the corresponding diagonal of $\mathbf{\Lambda}$ is less than λ_c or equivalently only the fractional terms in (5) that contribute a pole closer than $-\frac{1}{\lambda_c}$ to the imaginary axis are kept in $\mathbf{Y}(s)$. The value of λ_c is determined by inserting the user-specified error ϵ_c and maximum frequency ω_c in (6).

Looking at the model in (4), it is apparent that matrix $\mathbf{G}'$ is of the Laplacian kind because it is consisted of the Schur complement of matrix $\mathbf{G}$, which is Laplacian itself, and the identity matrix. Therefore, it is not required to approximate $\mathbf{G}'$ with its nearest Laplacian matrix and as result the error induced due to that approximation can be avoided.

IV. EXPERIMENTAL EVALUATION

For the evaluation of the proposed methodology, we developed an in-house gate-level STA tool in C/C++. Our timing analysis tool implements the Synopsys Composite Current Source (CCS) model for the gate timing analysis. CCS model offers an acceptable trade-off between time to compute and signoff accuracy in voltage waveform estimation at the gate output. The estimated voltage waveform is then used for the timing analysis of the driving interconnect.

Given the interconnect parasitics netlist, we built the matrices of the initial model shown in (1). Since (1) assumes that an independent current source is connected to the input port, we transformed the CCS voltage waveform which is in series with the first resistor of the interconnect, to the equivalent current source parallel to the resistor. At the next step, by applying the PACT MOR and subsequent sparification, we generated the PACT and MORSparse ROMs. Note that for all examined benchmarks, we selected $\omega_c = 5\text{GHz}$ and $\epsilon_c = 5\%$ in (6), which resulted in obtaining the upper left $p \times p$ submatrices ($\mathbf{G}_P', \mathbf{C}_P'$) of the reduced matrices in (4).

In order to evaluate the accuracy and runtime of interconnect timing analysis using MORSparse, we implemented the Elmore delay model and a SPICE transient analysis for the simulation of the initial/full model and the PACT and MORSparse ROMs. Our SPICE simulator was verified against Synopsys HSPICE. For the solution of the linear systems and the rest of linear algebra operations, we used the Eigen library [14]. We tested the proposed methodology on the ISPD-2012 [15] benchmarks implemented in 45nm process technology, while we executed all experiments on a Linux workstation with an Intel 3.50GHz 8-core Xeon processor and 16GB memory.

To estimate the interconnect delays, we ran our STA tool using Elmore delay method and SPICE simulation of the initial

TABLE I: Comparison of ROM sparsity, simulation runtime and delay estimation accuracy between dense ROMs obtained from PACT and sparse ROMs obtained from MORSparse, with respect to full-SPICE simulation of the initial model

Bench.	#nodes	#ports	Model Sparsity		Runtime					Accuracy		
			PACT	MORSp.	Full-SPICE (ms)	PACT (ms)	MORSp. (ms)	PACT vs Full-SPICE Speedup	MORSp. vs Full-SPICE Speedup	Elmore Delay MRE	PACT MRE	MORSp. MRE
vga_lcd	2788	264	52.9%	99.2%	901	44.1	29.1	20.4X	30.9X	264 %	5.5e-3 %	1.5 %
netcard	1827	261	0%	96.6%	381	250.3	100.2	1.52X	3.8X	174 %	1.2e-1 %	4.3 %
mem_ctrl	1677	294	41.5%	98.8%	221	49.5	23.2	4.5X	9.5X	288 %	2.7e-3 %	1.8 %
leon2	1685	247	1.6%	98.3%	653	391	122	1.67X	5.4X	135 %	7.5e-2 %	6.8 %
leon3mp	3345	501	21.1%	97.9%	696	96.2	100.5	7.7X	6.9X	180 %	3.7e-2 %	7.2 %
b19	2050	309	0.6%	96.5%	1073	96.1	165	11.1X	6.5X	135 %	1.4e-1 %	3.6 %

model (full-SPICE), PACT ROM and MORSparse ROM. For the purpose of demonstration, we selected one interconnect from each benchmark. Table I reports the corresponding simulation runtime, delay estimation Mean Relative Error (MRE) among all output ports, with respect to full-SPICE simulation, as well as the sparsity ratio ($\frac{\mathbf{zeros}(\widehat{G}+\widehat{C})}{\mathbf{rows}(\widehat{G}+\widehat{C})\times\mathbf{cols}(\widehat{G}+\widehat{C})}$) of the PACT and MORSparse ROMs. We can observe that a sparsity ratio over 96.5% is achieved for the MORSparse ROM, which leads to runtime speedups up to 30X over full-SPICE simulation and up to 3.2X over PACT ROM SPICE simulation. The runtime of Elmore delay estimation is not reported since the delay can be computed once (before STA) and be retrieved directly during STA. Note that although those benchmarks do not include coupling capacitances since Elmore cannot deal with crosstalk, MORSparse is expected to provide even greater speedups for such cases, as the related models are much bigger.

The accuracy results reported in Table I indicate that our methodology provides an accurate approximation of the interconnect delay, with a typical MRE of 4%, when Elmore delay could deviate up to 280%. The output waveforms obtained from the SPICE simulation of the initial model and the MORSparse ROM are given in Fig. 2. It is apparent that the differences between the two waveforms are negligible.
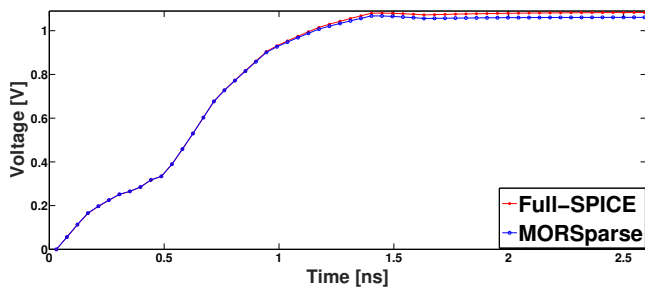

Fig. 2: Voltage response at the port inst_41217:A1 of vga_lcd.

V. CONCLUSION

This paper presents a fast and accurate methodology for the timing analysis of VLSI interconnects. The proposed methodology is based on a sparsity-preserving model order reduction algorithm to obtain sparse models of the interconnects to be simulated. When incorporated in a timing analysis tool, it can provide signoff accuracy results and accelerate interconnect simulation runtimes by orders of magnitude. Experimental results on complex interconnects of the ISPD-2012 benchmarks,

showed that our methodology introduces a negligible typical error of 4%, while achieving runtime speedups up to 30X over SPICE simulation of the initial model and up to 3.2X over SPICE simulation of the dense PACT ROM.

REFERENCES

[1] C. L. Ratzlaff, N. Gopal, and L. T. Pillage, "RICE: rapid interconnect circuit evaluator," in *Proceedings of the 28th ACM/IEEE Design Automation Conference*, 1991.

[2] W. C. Elmore, "The transient response of damped linear networks with particular regard to wideband amplifiers," *Journal of Applied Physics*, 1948.

[3] R. W. Freund, "Sprim: structure-preserving reduced-order interconnect macromodeling," in *IEEE/ACM International Conference on Computer Aided Design, 2004. ICCAD-2004.*, 2004.

[4] K. J. Kerns and A. T. Yang, "Stable and efficient reduction of large, multiport rc networks by pole analysis via congruence transformations," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 1997.

[5] P. Feldmann and R. W. Freund, "Efficient linear circuit analysis by pade approximation via the lanczos process," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 1995.

[6] Z. Ye, D. Vasilyev, and J. R. Phillips, "Sparse implicit projection (sip) for reduction of general many-terminal networks," in *2008 IEEE/ACM International Conference on Computer-Aided Design*, 2008.

[7] R. Ionutiu, J. Rommes, and W. H. A. Schilders, "Sparserc: Sparsity preserving model reduction for rc circuits with many terminals," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2011.

[8] D. Oyaro and P. Triverio, "Turbomor-rc: An efficient model order reduction technique for rc networks with many ports," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2016.

[9] C. Antoniadis, N. Evmorfopoulos, and G. Stamoulis, "Efficient sparsification of dense circuit matrices in model order reduction," in *Proceedings of the 24th Asia and South Pacific Design Automation Conference*, 2019.

[10] R. Puri, D. S. Kung, and A. D. Drumm, "Fast and accurate wire delay estimation for physical synthesis of large asics," in *Proceedings of the 12th ACM Great Lakes Symposium on VLSI*, 2002.

[11] S. Abbaspour, M. Pedram, A. Ajami, and C. Kashyap, "Fast interconnect and gate timing analysis for performance optimization," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2006.

[12] C.-W. Ho, A. Ruehli, and P. Brennan, "The modified nodal approach to network analysis," *IEEE Transactions on Circuits and Systems*, 1975.

[13] D. Garyfallou, N. E. Evmorfopoulos, and G. I. Stamoulis, "A combinatorial multigrid preconditioned iterative method for large scale circuit simulation on GPUs," in *15th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design*, 2018.

[14] G. Guennebaud, B. Jacob *et al.*, "Eigen v3," http://eigen.tuxfamily.org, 2010.

[15] M. M. Ozdal, C. Amin, A. Ayupov, S. Burns, G. Wilke, and C. Zhuo, "The ISPD-2012 discrete cell sizing contest and benchmark suite," in *Proceedings of the 2012 ACM International Symposium on International Symposium on Physical Design*, 2012.