A Nested Dissection Approach to Modeling Transport in Nanodevices

U. Hetmaniuk, Y. Zhao*, and M. Anantram*
Department of Applied Mathematics, University of Washington,
Box 353925, Seattle, WA 98195-353925, U.S.A.
e-mail: hetmaniu@uw.edu
*Department of Electrical Engineering, University of Washington,
Box 352500, Seattle, WA 98195-2500, U.S.A.
e-mail: zhaoyq@uw.edu, anant@uw.edu

With the advent of smaller nanoelectronic devices, where quantum mechanics is central to the device operation, quantum mechanical simulations have become a necessity. The non-equilibrium Green's function (NEGF) method [1] has emerged as a powerful modeling approach for these nanodevices and nanomaterials.

A typical NEGF-based simulation solves the Green's function equations,

$$\begin{cases} \mathbf{A}\,(E)\,\mathbf{G}^r\,(E) = \mathbf{I} \\ \mathbf{A}\,(E)\,\mathbf{G}^<\,(E) = \mathbf{\Sigma}^<\,(\mathbf{G}^r\,(E))^\dagger \end{cases} \tag{1}$$

where the sparse matrix $\mathbf{A}$ is defined by

$$\mathbf{A} = E\mathbf{I} - \mathbf{H} - \mathbf{\Sigma}_{Lead} - \mathbf{\Sigma}_{Phonon}. \tag{2}$$

$\mathbf{G}^r(E)$ is the retarded Green's function, describing local density of states, and $(\mathbf{G}^r(E))^\dagger$ its Hermitian conjugate. $\mathbf{G}^<(E)$, the lesser Green's function, represents the electron correlation function for energy level E; the diagonal elements of $\mathbf{G}^<(E)$ represent the electron density per unit energy. $\mathbf{I}$ is the identity matrix and $\mathbf{H}$ the system Hamiltonian. $\mathbf{\Sigma}_{Lead}$ represents the self-energy matrix due to the leads and $\mathbf{\Sigma}_{Phonon}$ corresponds to the self-energy governing electron-phonon scattering. The matrix $\mathbf{\Sigma}^<$ corresponds to the lesser self-energy. Solving (1) for the diagonal of $\mathbf{G}^<$ at many energies E is a computationally intensive part of NEGF-based simulations.

The most common approach to compute blocks of $\mathbf{G}^r$ and $\mathbf{G}^<$ is the recursive Green's function method [4]. Recent advances utilize the nested dissection method [2] to exhibit a significant speedup. These new algorithms exploit a sparse block $\mathbf{LDL}^T$-factorization of $\mathbf{A}$ and re-use this factorization to fill in all diagonal blocks of the Green's functions in a specific order. The main difference between RGF and these methods is the replacement of *layers* of grid points organized along a specific direction with *arbitrarily-shaped clusters* of grid points organized in a binary tree. Such choice allows to *fold* and to *extract* in any physical direction when following the binary tree, generated by the nested dissection.

The present contribution introduces an algorithm for calculating diagonal blocks of $\mathbf{G}^<$ with partitions from METIS [3]. The developed method has a reduced complexity compared to the established recursive Green's function approach. For a device with N_y layers and N_x grid points per layer, as shown in Figure 1, the complexity for RGF is $\mathcal{O}(N_x^3 N_y)$ while the proposed algorithm exhibits a complexity $\mathcal{O}(N_x^2 N_y)$. Numerical experiments on a quantum well superlattice and a carbon nanotube demonstrate significant speedups over the recursive method.

References

[1] S. Datta, *The non-equilibrium Green's function (NEGF) formalism: An elementary introduction*, International Electron. Devices Meeting, pp. 703-706 (2002).
[2] A. George, *Nested dissection of a regular finite element mesh*, SIAM J. Numer. Anal, vol. 10, pp. 345-363 (1973).
[3] G. Karypis and V. Kumar, *A fast and high quality multilevel scheme for partitioning irregular graphs*, SIAM J. Sci. Comput., vol. 20, pp. 359-392 (1998).
[4] A. Svizhenko, M. Anantram, T. Govindam, B. Biegel, R. Venugopal, *Two-dimensional quantum mechanical modeling of nanotransistors*, J. Appl. Physics, vol. 91, pp. 2343 (2002).

978-3-901578-26-7

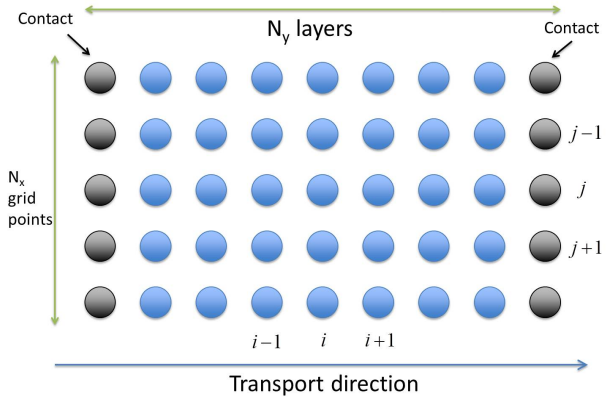

Fig. 1. Nano-device partitioned into N_y layers. Each layer contains N_x grid points.

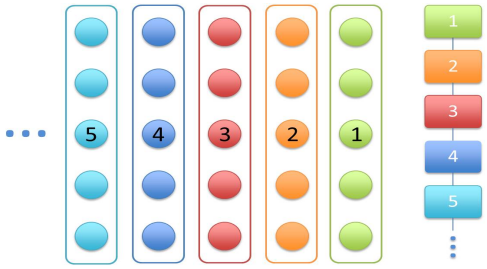

Fig. 2. Partition generating the RGF algorithm.. The first pass advances one layer at a time from left to right along the y-direction and, recursively, folds the effect of left layers into the current layer. The second pass marches one layer at a time from right to left along the y-direction and, recursively, extracts the diagonal blocks and the nearest neighbor off-diagonal blocks for the final result.

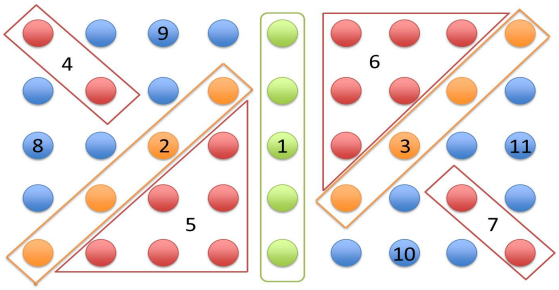

Fig. 3. Partition generated by METIS for system including dense layers at two ends.

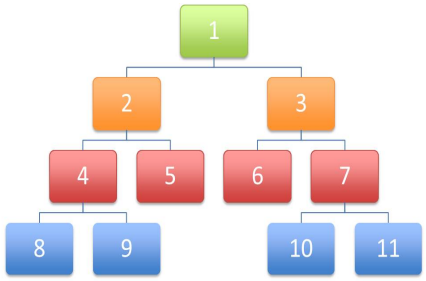

Fig. 4. Binary tree relating the different clusters of grid points. The first pass folds all *clusters* of grid points on the same level, while climbing up. The second pass extracts the diagonal blocks one level at a time, while marching down.

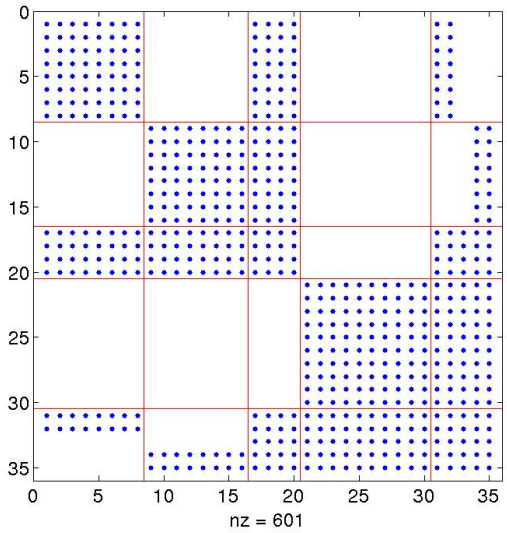

Fig. 5. Typical distribution of computed elements for $\mathbf{G}^r$ and $\mathbf{G}^<$.

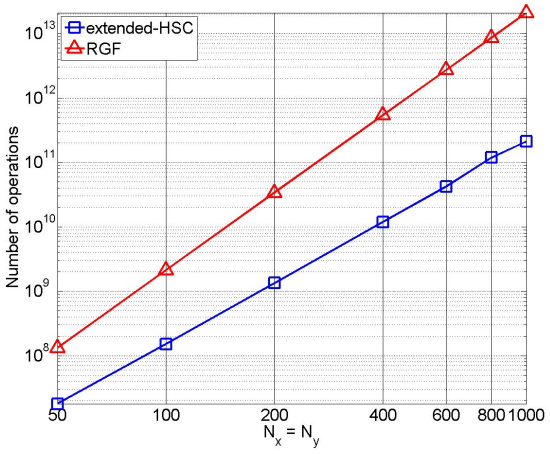

Fig. 6. Comparison of number of operations between our algorithm (blue) and RGF (red).