Numerical parallel algorithms for large-scale nanoelectronics simulations using NESSIE

Eric Polizzi, and Ahmed Sameh
Purdue University, Department of Computer Sciences, 250 N. University Street,
West-Lafayette, IN 47907-2066,
email: epolizzi@purdue.edu

NESSIE is a parallel finite element simulator, presented in [1,2], for electron transport in 2D and 3D nanodevices using a full self-consistent Schrödinger-Poisson model (see Fig. 1). Starting from the physical description of realistic devices, to the discretization scheme of partial differential equations models, and the numerical algorithms, NESSIE makes use of both applied mathematics techniques and analysis, as well as high performance computing tools. The simulator has also been extended to operate within the NEGF formalism using finite element discretization [3]. The wave function and Green's function approaches are formally equivalent, but differ in the required numerical treatment. However, in both cases three types of numerical operations [5] are involved: (i) solving a linear system for the potential, (ii) solving a linear system for the Green's functions or wave functions, and (iii) solving the generalized eigenvalue problems resulting from decomposition by modes. In order to reduce the size of the linear system for the Green's function, a general subbands decomposition method and its asymptotic approach have been proposed and analyzed in [2]. These methods allow to drastically decrease the simulation times needed for: the 3D electron waveguide devices [1], the 2D DG-MOSFET [2], and the 3D Silicon nanowire transistors [4]. However, the size of the linear systems and the eigenvalue problems keep increasing either for large 3D systems (of the order of 10^{6+}) or depending on the level of sophistication of the model used to define the electronic structure i.e. whether effective mass, multi-band, or atomistic.

Here, we propose to enhance the numerical efficiency of the self-consistent procedure, using new high performance parallel algorithms. In NESSIE, the linear systems are currently solved on a uniprocessor using a preconditioned Krylov subspace iterative method. In order to compute the electron density, one needs to repeat this procedure for all the energy levels (of the order 10^{3+}). This can be trivially done independently on each processor of a Linux cluster using MPI directives. One advantage of this parallel procedure is that the communications between the processors are minimal. A significant drawback, however, to this simplistic solution strategy is that it may not be possible for each processor, with its limited local memory, to handle huge linear systems. For very large systems as outlined above, it is essential to adopt a strategy in which all processors of a high-end computing platform, cooperate on solving a single system. Towards this goal, we developed and applied (i) a parallel hybrid linear solver SPIKE [6] that represents an improvement over the classical direct solvers of ScaLAPACK (see Table 1), (ii) a parallel trace-minimization algorithm TRACEMIN [7] that is superior to Lanczos, subspace iteration, and LAPACK schemes for computing the lowest eigenpairs (see Table 2).

[1] E. Polizzi and N. Ben Abdallah, Phys. Rev. B, 66, 245301 (2002). [2] E. Polizzi and N. Ben Abdallah, J. Comp. Phys., submitted (2004). [3] E. Polizzi and S. Datta, IEEE-NANO Conf. (2003). [4] J. Wang, E. Polizzi, M. Lundstrom, IEDM conf. (2003). [5] E. Polizzi, H. Sun, A. Sameh, Nanotech conf. (2004). [6] A. Sameh and V. Sarin, Inter. J. of Comp. Fluid Dynamics, 12, 213, (1999). [7] A. Sameh and Z. Tong, , J. Comp. and Appl. Math, 123, 155, (2000).

A full journal publication of this work will be published in the Journal of Computational Electronics.

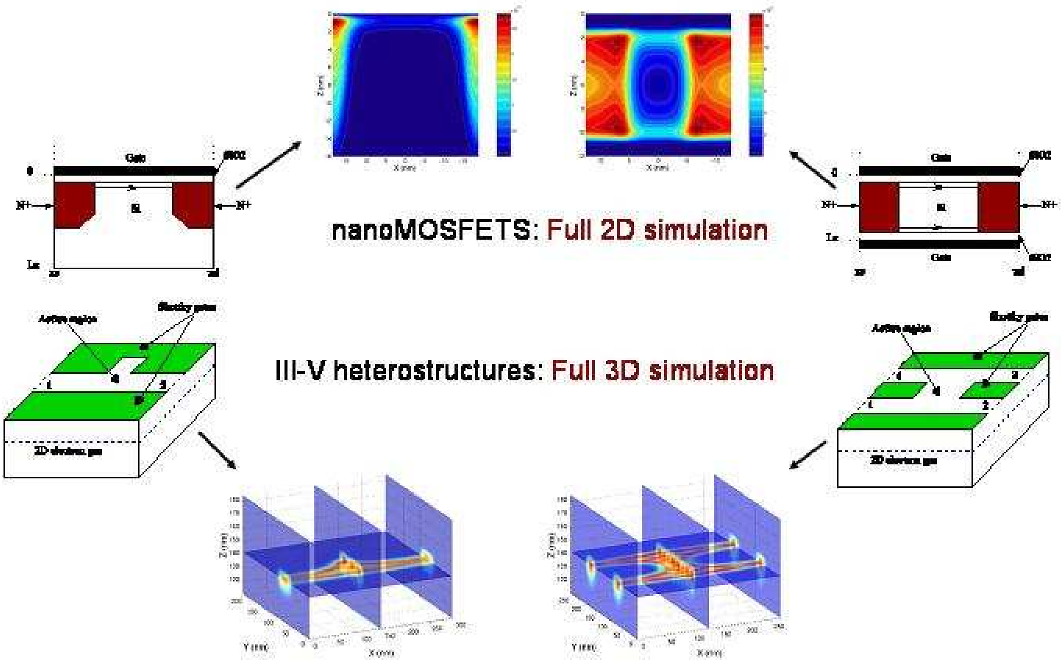

Figure 1: Simulation results of the electron density for 2D MOSFET devices and 3D electron waveguide devices, obtained using a full self-consistent quantum approach with NESSIE. The results are shown at equilibrium for a given applied gate potential.

CPU	2	4	8	16
Preprocessing	1.0/3.4	1.3/2.6	2.6/2.8	4.6/3.1
Solver 1 RHS	3.1/4.5	4.5/3.3	8.1/3.1	13.0/3.0
Solver 50 RHS	6.0/3.0	9.0/2.4	17.0/3.2	31.0/2.8

Table 1: Speed improvement obtained for non-diagonally dominant systems with the SPIKE algorithm versus LAPACK/ScaLAPACK on a IBM-SP platform. The size of the matrix is $n = 160,000$ and the bandwidth $b = 161$, and the solver is applied for different number of Right Hand Side (RHS). The algorithm shows both a perfect scalability compared to LAPACK (1 CPU), and a factor speed of improvement of ~ 3 versus ScaLAPACK.

CPU	1	2	4	6	8
n=707, b=45	3	4	3	3	*
n=1671, b=72	20	25	33	33	28

Table 2: Speed improvement obtained for two different eigenvalue problems (section of a nanowire [4]) with the TRACEMIN algorithm versus LAPACK direct routine. With increasing matrix size n, speedup and scalability improve significantly.

A full journal publication of this work will be published in the Journal of Computational Electronics.