BENCHMARKING LINEAR SOLVERS WITH SEMICONDUCTOR SIMULATION EXAMPLES

Stephan Wagner[1], Tibor Grasser[1], and Siegfried Selberherr[2]

[1] Christian Doppler Laboratory for TCAD in Microelectronics
at the Institute for Microelectronics
[2] Institute for Microelectronics, Technical University Vienna, A-1040 Vienna, Austria
E-mail: Wagner@iue.tuwien.ac.at, Phone: +43(1)58801-36037, Fax: +43(1)58801-36099

ABSTRACT

Engineering tools to solve the semiconductor device equations create large nonlinear equation systems. Since a significant part of the simulation time is spent with solving linear equation systems inside a damped Newton algorithm, the choice of the employed linear solver is rather important. This choice not only depends on the computational power available, but also on the type of the simulation example. We evaluated and compared the performance of several solvers. Rather than using a set of single matrices, this work is based on complete simulations with consistent settings.

KEYWORDS

Partial differential equations, Linear Equation Systems, Numerical methods, Iterative Methods, Direct Methods, Parallelization, Cluster Computing

MOTIVATION

A significant part of the computation time of a numerical semiconductor device or circuit simulation is spent with solving linear equation systems. Since the remarkably increased performance of toady's average computers inspires to even more costly simulations (e.g. optimizations and mixed-mode device/circuit simulations), a further speed-up of the simulators is highly appreciated. Thus, the on-going development of highly-optimized mathematical code must not be neglected. With regard to the growing demand for computational power, an evaluation of different solvers yields information on their specific advantages to minimize the overall simulation time.

INTRODUCTION

Semiconductor device and circuit simulations require the solution of a nonlinear system of partial differential equations discretized on a grid [1]. We deal here with point-residual methods: finite differences and finite boxes discretization schemes. The non-symmetric system matrices are sparse (local connections between unknowns only) and the nonlinear problem is solved by a damped Newton algorithm demanding the solution of a linear equation system at each step. Matrices assembled during mixed device/circuit simulations (mixed-mode) normally contain one or several distributed devices in combination with the circuit equations and have therefore a different structural type.

At our institute a proprietary assembly and solver system [2] has been developed, which is currently employed by the device and circuit simulator Minimos-NT [3]. The assembly module provides an application programming interface to the simulator models, several conditioning measures, sorting, and scaling. The solver module offers the two iterative solvers BI-CGSTAB [4] and GMRES [5] (restarted) in an adaptive loop with an ILUT-preconditioner as well as a direct Gaussian solver (LU factorization). In addition, external solver modules can be plugged-in by an interface.

At the moment, two external modules can be employed: the Parallel Sparse Direct Linear Solver PARDISO [6,7,8] and the Algebraic Multigrid Methods for Systems SAMG [9,10]. The former provides a multi-threaded direct Gaussian solver as well as an LU-CGS implementation. The latter is a library of different solvers. For the current evaluation the two iterative solvers BI-CGSTAB and GMRES in combination with an ILU-preconditioner have been selected.

SETUP AND DEVICES

We present results of an extensive performance evaluation based on twelve examples which were taken from current scientific projects at our institute. They were simulated on an IBM AIX 5.2 cluster (four nodes based on Power4+ architecture; 192 GB memory) and on a 2.4 GHz single Intel Pentium IV (1 GB memory) running under Suse Linux 8.2. For compiling and linking the native xlc/xlC/xlf compilers (32-bit; optimization level O5: linked against the ESSL library) and the Intel 7.1 compilers (IA32: optimization level 03) were used. The simulation time was measured with the *time* command, the fastest of three consecutive runs was taken.

Rather than using a set of single matrices, this evaluation is based on complete simulations with consistent settings, as typically encountered during daily work. Four two-dimensional, four mixed-mode device/circuit (the number of devices is given in brackets), and four three-dimensional simulations were used for evaluating the solver performance. The examples with additional data are summarized in Table 1.

RESULTS

In Figure 1 a comparison of different solvers for selected simulations on the Intel computer is given. Due to the large simulation time differences, all times are scaled to the in-house ILU-BI-CGSTAB in the center of the graph. Interesting results are the superiority of the advanced implementations of LU factorization and iterative solvers for circuits and three-dimensional devices, respectively. The in-house GMRES solver has advantages for circuits also, whereas the direct solver on the left hand side can in fact only be used for quality assessment of two-dimensional simulations.

To show the relative impact of multi-threading, the PARDISO-LU solving ratios (referring to the single-threaded version) against the number of processors/threads are shown in Figure 2. For the three-dimensional examples, also the PARDISO-LU-CGS ratios are given. In addition to the real (wall clock) time required for solving the example, the cumulated user (CPU) times are shown, which are increasing due to the parallelization overhead. Whereas for two-dimensional device and circuit simulations too many processors can be even contra-productive, the marginal additional utility for three-dimensional simulations is drastically diminishing. Thus, for the average simulation four processors should be sufficient. Under scarce conditions, e.g. during optimizations, assigning two tasks per node of eight processors appears to minimize the real time effort.

CONCLUSION

Iterative methods still show a significant performance advantage over direct solvers. However, the 1983 quotation "In 3D sparse direct solution methods can safely be labeled a disaster" [11] describes the experiences (in regard to both time and memory consumption) with the classically implemented, in-house LU factorization, but does not embrace the recent developments. Especially for mixed-mode device/circuit simulations the advanced direct methods show an interesting performance advantage.

REFERENCES

[1]: S. Selberherr, *Analysis and Simulation of Semiconductor Devices,* Springer, Wien-New York, 1984.

[2]: S. Wagner, T. Grasser, C. Fischer, and S. Selberherr, "A Simulator Module for Advanced Equation Assembling", in Proc. 15th European Simulation Symposium, 55-64, 2003.

[3]: http://www.iue.tuwien.ac.at/software/minimos-nt, *Minimos-NT 2.0 User's Guide*, Institut für Mikroelektronik, Technische Universität Wien, Austria, 2002.

[4]: H. A. van der Vorst, "BI-CGSTAB: A Fast and Smoothly Converging Variant of BI-CG for the Solution of Nonsymmetric Linear Systems", SIAM Journal on Scientific and Statistical Computing, **13 (2)**, 631-644, 1992.

[5]: Y. Saad and M. Schultz, "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems", SIAM Journal on Scientific and Statistical Computing, **7 (3)**, 856-869, 1986.

[6]: G. Karypis and V. Kumar, "A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs", SIAM Journal on Scientific Computing, **20 (1)**, 359-392, 1998.

[7]: O. Schenk and K. Gärtner, "Solving Unsymmetric Sparse Systems of Linear Equations with PARDISO", Future Generation Computer Systems, 2003, accepted, in press.

[8]: O. Schenk, K. Gärtner, and W. Fichtner, "Efficient Sparse LU Factorization with Left-Right Looking Strategy on Shared Memory Multiprocessors", BIT, **40 (1)**, 158-176, 2003.

[9]: T. Clees and K. Stüben, "Algebraic Multigrid for Industrial Semiconductor Device Simulation", in Proc. Challenges in Scientific Computing, 110-130, 2002.

[10]: T. Füllenbach (now: Clees), K. Stüben, and S. Mijalkovic, „Application of an Algebraic Multigrid Solver to Process Simulation Problems", in Proc. Simulation of Semiconductor Processes and Devices, 225-228, 2000.

[11]: R. Bank, D. Rose, and W. Fichtner, "Numerical Methods for Semiconductor Device Simulation", IEEE Journal on Electrical Devices, **30 (9)**, 1031-1041, 1983.

Simulation	Data	Dimension	Entries ([‰])	Remark
MOSFET	2D device	2.704	23.662 (3.24)	L_G = 1 µm
Flash cell	2D device	5.967	47.956 (1.35)	tunneling effects
Pin diode	2D device	6.335	56.127 (1.40)	optical generation
SiGe HBT	2D device	19.313	210.915 (0.57)	self-heating
Colpitts oscillator	Circuit (1)	3.928	35.002 (2.27)	transient (400 steps)
Amplifier	Circuit (1)	6.391	35.291 (0.86)	hydrodynamic
Ring oscillator	Circuit (10)	25.246	226.931 (0.36)	transient (100 steps)
2-input nand gates	Circuit (8)	146.614	1.347.138 (0.06)	transient (50 steps)
MagFET	3D device	85.308	921.860 (0.13)	magnetic field
FinFET	3D device	81.037	807.150 (0.12)	thin SOI finger
SOI	3D device	87.777	997.296 (0.13)	two operating points
HBT	3D device	119.098 175.983	1.257.714 (0.09) 1.833.138 (0.06)	two iteration schemes, 2nd one with selfheating

Table 1: Four two-dimensional, four mixed-mode device/circuit (the number of devices is given in brackets), and four three-dimensional simulations were used for evaluating the solver performance. The dimension of the linear equation system, the number of non-zero entries in the sparse matrices and some remarks on the simulation type are given.

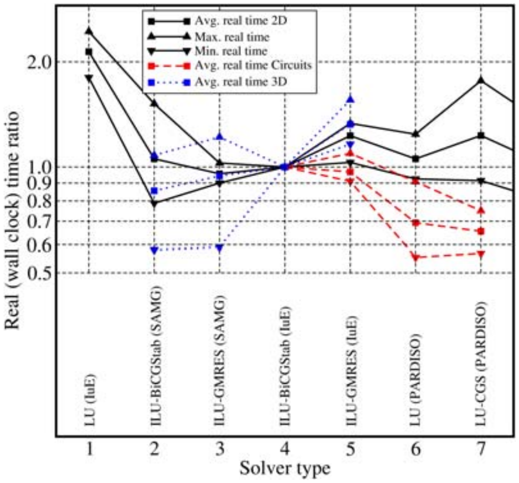

Figure 1: Solving times (average, minimum, maximum) for selected simulations on the Intel computer. All times are scaled to the in-house ILU-BI-CGSTAB in the center.

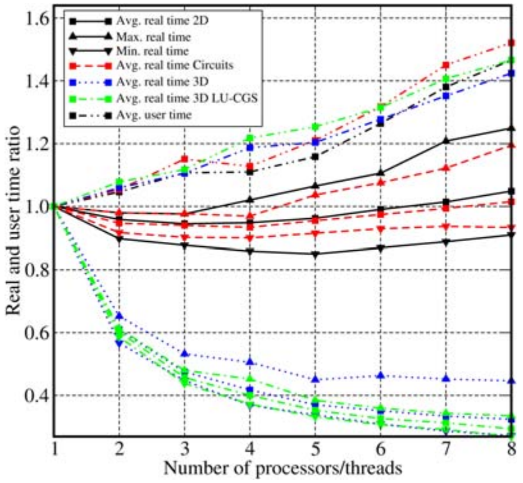

Figure 2: The relative PARDISO real/wall clock times (average, minimum, maximum) and average user time versus the number of processors/threads on the IBM cluster.