VLSI DESIGN
1998, Vol. 8, Nos. (1–4), pp. 337–341
Reprints available directly from the publisher
Photocopying permitted by license only

Numerical Evaluation of Iterative Schemes for Drift-diffusion Simulation

MAHESH B. PATIL [a], UMBERTO RAVAIOLI [b,*] and THOMAS KERKHOVEN [c]

[a] *Department of Electrical Engineering, Indian Institute of Technology at Kanpur, Kanpur, U.P. 208016, India;* [b] *Beckman Institute, University of Illinois at Urbana-Champaign, 405 N. Mathews Avenue, Urbana, IL 61801, USA;* [c] *Department of Computer Science, University of Illinois at Urbana-Champaign, 1304 W. Springfield Avenue, Urbana, IL 61801, USA*

We introduce an iterative scheme to solve the drift-diffusion device simulation problem, which combines the Gummel iteration with the "pointwise iteration", and then we compare its convergence behavior with other iteration strategies, for different test cases. Comparisons are made with the standard Gummel approach and a nonlinear multigrid iteration. The combined "Gummel-Pointwise" iteration has significantly better convergence characteristics than the Gummel iteration in all cases. The cost of the pointwise iteration varies only linearly with the number of grid points. While in terms of CPU time for the solution the pointwise iteration is not always faster in 2-D, in 3-D the combined technique becomes more and more advantageous as the grid size increases. It is found that the nonlinear multigrid scheme is overall not as effective as the combined iteration.

Keywords: Drift-diffusion, iterative methods, gummel iteration

1 INTRODUCTION

A major problem for the development of drift-diffusion solvers for 3-D applications is the difficulty in adapting existing 2-D schemes. The Gummel and Newton iteration methods are most commonly used to solve the nonlinear steady-state problem arising from the three semiconductor device equations [1], (Poisson equation, electron continuity and hole continuity, in the order)

$$\nabla^2\psi = -\frac{q}{\varepsilon}(p - n + N_d^+ - N_a^-) \tag{1}$$

$$\vec{\nabla} \cdot \mathbf{J}_n - qR = 0 \tag{2}$$

$$\vec{\nabla} \cdot \mathbf{J}_p + qR = 0 \tag{3}$$

where all symbols have their usual meaning.

* Corresesponding author. Tel.: (217) 244-5765, Fax: (217) 244-4333, e-mail: ravaioli@uiuc.edu.

In the case of Gummel iteration, the equations are solved in a decoupled manner. Poisson equation is solved at all grid points, followed by the electron continuity equation, and then by the hole continuity equation, repeating the procedure until convergence. In Newton's method the three device equations are solved simultaneously using a linearization procedure. The Gummel iteration always maintains the coupling between grid points, arising from discretization, while the equations are loosely coupled through successive updates of the variables ψ, n and p. In the Newton iteration, both the coupling between neighbouring grid points and that between the three equations is maintained during each iteration.

In practice, the Gummel method is found to be efficient if the coupling between the equations is weak; otherwise, the convergence may become quite slow. The Newton method is more robust, and converges in relatively few iterations, provided that the initial guess is sufficiently close to the solution. The major drawback of Newton's method is that it involves a matrix of size $3N \times 3N$ (where N is the number of grid points), which increases both memory requirements and CPU time at each iteration. These constraints are particularly severe in three dimensions if a direct method for matrix solution is to be used.

2 NUMERICAL APPROACH

Several strategies have been reported to circumvent the limitations of Gummel's and Newton's iterations [2–7], such as approaches for acceleration of convergence in Gummel's scheme, reduction of memory requirements in Newton's scheme, and so on. The purpose of this paper is to explore alternative iteration schemes to solve the device equations, and to evaluate their performance through numerical experiments.

A complementary approach to Gummel iteration is represented by the so called pointwise iteration. Here, the three equations are solved simultaneously at a given grid point, treating the variables at adjacent points as constants, and this procedure is repeated for all grid points. Thus, the coupling between the equations is maintained at each grid point, whereas the grid points are coupled to each other only loosely through the update of values at the neighbors.

The pointwise iteration does not need a large matrix to be stored as the equations are solved simultaneously one point at a time; this makes the iteration very attractive from the memory point of view. This procedure has been used as the smoothing iteration in nonlinear multigrid solution of the device equations [9–12]. Also, in the Alternate Block Factorization [7] and ILU/Knot [6] methods, the idea of coupling the three equations at each grid point has been used, although not explicitly as an iteration.

In our numerical experiments we have used rectangular grids, and it is convenient to adopt a *red-black* reordering as in [12], where the mesh points are arranged as red and black squares on a checker board. The three equations are solved for all red points first, followed by all black points. At each grid point, we have three equations in three unknowns (ψ, n, and p). In our specific implementation, we first perform a few Gummel-type iterations. If the residuals are not smaller than a specified tolerance, at the end of the Gummel-type iterations, we switch to Newton's method on the 3×3 system to solve the equations at that grid point.

Since the Gummel and the Pointwise iterations have complementary aspects, it is interesting to combine the two approaches to examine the effect on the convergence rate. This combined iteration, which we call the "GP_M" iteration consists of the following: (a) perform one Gummel iteration, (b) perform M times the pointwise iteration. As we will show later through numerical examples, this scheme can significantly accelerate the convergence.

To complete the comparison, we have also implemented a NonLinear MultiGrid method (NLMG) as described in [12]. Here, the pointwise iteration described above is used for smoothing,

full weighting for the restriction of residuals, injection for the restriction of solution, and bilinear interpolation for transferring error from coarse to fine level. We have implemented V-cycle with $\nu_1 = 2$ and $\nu_2 = 1$ where ν_1 and ν_2 are the number of pre-smoothing and post-smoothing steps respectively. For the NLMG iteration to converge, the coarse level cannot be arbitrarily coarse as pointed out in [12]. For our examples with a limited grid size, we found that the iteration did not converge with three levels, therefore, only two levels have been used. Two approaches can be used to solve the problem on the coarsest level: (a) The three equations may be solved simultaneously at all grid points using Newton's method as in [12]. (b) A large number (typically 50 to 100) of smoothing iterations may be performed to solve the coarse-grid problem approximately. This approach is also employed in [10] although, in that work, the grid points are scanned in a different order. Both approaches have been tried here. We will refer to one V-cycle as one Non-Linear MultiGrid (NLMG) iteration, NLMG with Newton's method at the coarse level as NLMG (N) and NLMG with m smoothing iterations at the coarse level as NLMG (mS).

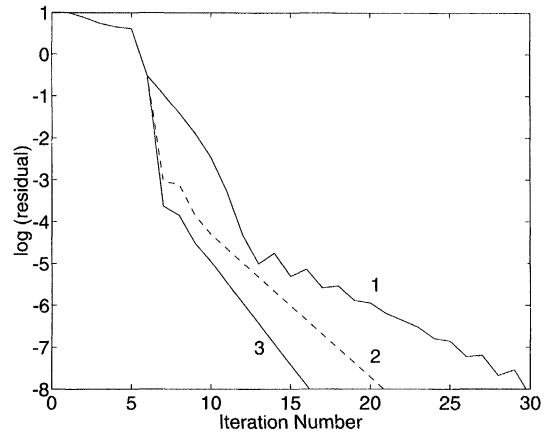

FIGURE 1 2-norm of residual versus iteration number for 2D p^+n device with a forward bias of 0.8 V and the solution for 0.6 V as the initial guess. (a) 1: Gummel, 2: GP_1, 3: GP_{25}. Damping was employed for the first 5 iterations. (b) 1: NLMG (N), 2: NLMG (50 S), 3: NLMG (100 S).

3 SIMULATION RESULTS

Because of space limitations, we present here only two examples of 2-D simulation, involving a silicon p^+n junction with a uniform substrate n-doping of 10^{17} cm^{-3}) and a Gaussian diffusion for the p^+ region with peak density 5×10^{18} cm^{-3} at the contact, and a 3-D example with a n^+n with the same doping levels. To facilitate the comparison, we use here a uniform grid with 33×33 points in 2-D and $17\times17\times17$ in 3-D. Similar tests conducted with non-uniform rectangular grids produced very similar behavior.

For the 2-D simulation we consider a situation of large forward bias (0.8 V) where the solution for 0.6 V forward bias is used as the initial guess (Figs. 1(a) and 1(b)), and also a reverse bias of 0.5 V with the equilibrium solution as the initial guess (Fig. 2). It is clearly seen that the GP_M iteration converges in significantly fewer iterations than the Gummel iteration in all cases. For the forward bias case, GP_1 and NLMG (N), respectively, are the most efficient in terms of number of iterations required for convergence. Note also that the convergence of the NLMG (mS) iteration is

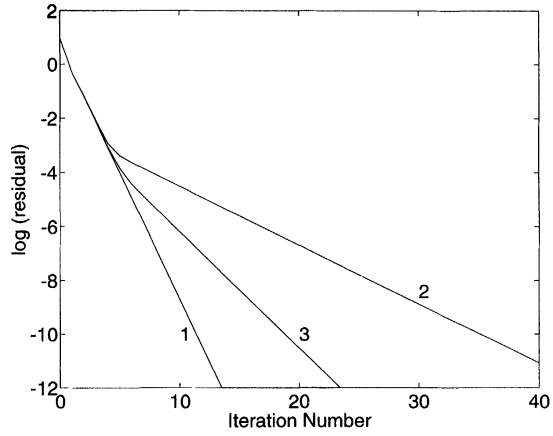

FIGURE 2 2-norm of residual versus iteration number for 2D p^+n device with a reverse bias of 0.5 V and equilibrium initial guess. Damping was employed for the first 5 iterations. 1: Gummel, 2: GP_1.

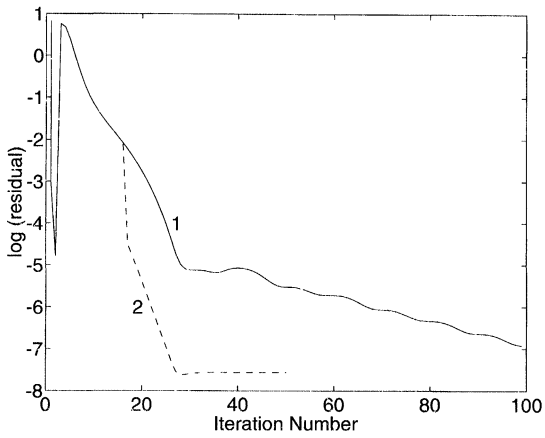

FIGURE 3 2-norm of residual versus iteration number for 3D n^+n device with 0.5 V and equilibrium initial guess.1: Gummel, 2: GP_{25}. Damping was employed for the first 15 iterations.

slow even for $m = 100$ which was found to be very effective instead in other simulations, not shown here, involving a forward biased $n^+ - n$ junction. For reverse bias, our simple NLMG iteration did not converge. We found that the coarse-grid problem for this case does not represent the fine-grid problem accurately enough because the depletion region on the n side is only two or three grid points wide. A locally adapted multigrid method [10] is more appropriate for this situation. In terms of CPU time for the example of Figure 1(a), the cost per iteration on the 33×33 grid was found to be: 0.155 s (Gummel); 0.118 s (Pointwise); 0.714 (NLMG(N)). This translates into total solution times of: 6.1 s (Gummel); 6.8 s (GP_1); 32.0 s (GP_25); 8.8 s (NLMG(N)); 24.0 s (NLMG(50S)); 22.0 s (NLMG(100s)). This shows that in 2-D problems there is little computational advantage in using alternative iteration schemes.

However, as drift-diffusion problems are pushed into 3-D, the combined Gummel-Pointwise iteration has immediately a marked advantage over the traditional Gummel iteration, since the computational cost of the pointwise iteration only grows linearly with the number of points. As a final example, we consider a 3D n^+n structure, with a $17 \times 17 \times 17$ grid which is uniform in each direction, with a voltage of 0.5 V applied to the n^+ contact. The equilibrium solution is used as the initial guess. For this example, Poisson's equation was solved using the multigrid method (correction scheme), and the continuity equations were solved using YSMP [8]. The residual for the Gummel and the GP_{25} iterations are shown in Figure 2. Again, the GP iteration converges in fewer iterations than Gummel. But while GP_{25} requires much more CPU time to solve the problem for a similar 2-D n^+n structure (83.0 s, as opposed to 29.0 s for Gummel iteration), in the 3-D case the Gummel solution requires 1065.0 s, and the GP_{25} solution only 595.0 s.

We conclude that the proposed Gummel-Pointwise iterative scheme in general improves the convergence behaviour of the Gummel scheme significantly. In terms of CPU time, the new scheme is advantageous for certain 2D cases; and for 3D simulation, it is expected to be always more effective than Gummel. It is suggested that implementation of the pointwise iteration on a parallel-architecture computer will enhance its effectiveness. The red black pointwise iteration can be naturally parallelized, as all red (or black) points can be processed in parallel and independently. This will drastically reduce the cost of the pointwise iteration, and the GP iteration will then be only about as expensive as Gummel iteration but with much better convergence characteristics.

Acknowledgements

The authors are thankful to Dr. Edwin Kan and Dr. Steve Laux for helpful comments. This work was supported by the National Science Foundation grants NSF ECS 95-26127 (NCCE) and ECS 95-09751.

References

[1] Selberherr, S. (1984). *Analysis and Simulation of Semiconductor Devices*, New York, Springer-Verlag.
[2] Toyabe, T., Masuda, H., Aoki, Y., Shukuri, H. and Hagiwara, T. (1985). "Three-dimensional device simulator CADDETH with highly convergent matrix solution algorithms", *IEEE Trans. Electron Devices*, **32**(10), 2038–2044.
[3] Wu, K.-C., Chin, G. R. and Dutton, R. W. (1991). "A STRIDE towards practical 3-D device simulation", *IEEE Trans. Computer-Aided Design*, **10**(9), 1132–1140.

[4] Pommerell, C. and Fichtner, W. (1991). "New developments in iterative methods for device simulation", *Proc. SISDEP 4 Conf.*, pp. 243–248.
[5] Simon, C., Sadkane, M. and Mottet, S. (1993). "Newton-GMRES method for coupled nonlinear systems arising in semiconductor device simulation", *Proc. SISDEP 5 Conf.*, pp. 243–248.
[6] Rafferty, C. S., Pinto, M. R. and Dutton, R. W. (1985). "Iterative methods in semiconductor device simulation", *IEEE Trans. Electron Devices*, **32**(10), 2018–2027.
[7] Bank, R. E., Chan, T. F., Coughran, W. M. Jr. and Smith, R. K. (1989). "The alternate-block factorization procedure for systems of partial differential equations", *BIT*, **29**, 938–954.
[8] Ellman, H. C. (1982). *Iterative methods for large, sparse,* nonsymmetric systems of linear equations, Yale University Computer Science Department Research Report 229.
[9] Constapel, R. and Berger, M. (1989). "A multigrid approach for device simulation using local linearization", in *Proc. NASECODE 6 Conf.*, pp. 355–359.
[10] Molenaar, J. and Hemker, P. W. (1990). "A multigrid approach for the solution of the 2D semiconductor equations", *Impact of Computing in Science and Engineering*, **2**, 219–243.
[11] De Zeeuw, P. M. (1992). "Nonlinear multigrid applied to a one-dimensional stationary semiconductor model", *SIAM J. Sci. Stat. Comput.*, **13**(2), 512–530.
[12] Joppich, W. and Mijalkovic, S. (1993). *Multigrid methods for process simulation*, New York, Springer-Verlag.

Authors' Biographies

Mahesh B. Patil received the B.Tech. degree from Indian Institute of Technology, Bombay, India, in1984, the M.S. dregree from the University of Southern California, Los Angeles, in 1987, and the Ph.D. degree from the University of Illinois at Urbana-Champaign in 1992, all in electrical engineering. In 1993 he was a visiting scientist with the Hitachi Central Research Laboratories, Japan. He is currently an Associate Professor of electrical engineering with Indian Institute of Technology, Kanpur, India.

Umberto Ravaioli received the Laurea in Electronics Engineering in 1980 and the Laurea in Physics in 1982, from the University of Bologna, Italy, and the Ph.D. in Electrical Engineering from Arizona State University, in 1986. He is now a Professor of Electrical and Computer Engineering at the University of Illinois, Urbana-Champaign.

Thomas Kerkhoven received the Drs. degree in Physics from the University of Amsterdam, Nederlands, in 1981, and the Ph.D. in numerical analysis from the Department of Computer Science of Yale University in 1985. He is now a Professor of Computer Science at the University of Illinois, Urbana-Champaign.