Contemporary Mathematics
Volume 180, 1994

Domain Decomposition Methods for Device Modelling

R.K. COOMER AND I.G. GRAHAM

ABSTRACT. We give an overview of some recent work [3] on the parallel solution of the drift diffusion equations for semiconductor device modelling. Discretization is by a variant of the finite element method. The resulting nonlinear equations are solved by Gummel's iteration, with the associated linear systems resolved by an additive Schwarz method. The algorithm has been implemented on a MasPar MP-1.

1. Drift-diffusion equations: discretization, outer iteration

In this paper we are concerned with the iterative solution of the (scaled) steady-state drift-diffusion equations for device modelling [8]:

$$-\lambda^2\Delta\psi + \delta\{\exp(\psi - v) - \exp(w - \psi)\} - d = 0, \tag{1.1}$$

$$-\nabla.(\exp(\psi - v)\nabla v) - \sigma\rho_v r(\psi, v, w) = 0, \tag{1.2}$$

$$-\nabla.(\exp(w - \psi)\nabla w) + \sigma\rho_w r(\psi, v, w) = 0. \tag{1.3}$$

Here ψ is the electrostatic potential and v and w are the electron and hole quasi-Fermi potentials respectively. The parameters λ, δ, σ, ρ_v and ρ_w are determined by the physics of the device, d is the (scaled) doping profile and r is the recombination/generation rate. We consider this system on a polygonal domain $\Omega \subset R^2$ with boundary $\partial\Omega$. At the *contacts* $\partial\Omega_D := \cup_i \partial\Omega_{D_i}$ (with the $\partial\Omega_{D_i}$ closed and non-empty subsets of $\partial\Omega$), we have Dirichlet conditions: $v|_{\partial\Omega_{Di}} = w|_{\partial\Omega_{Di}} = \alpha_i$, for each i, with the constants α_i corresponding to (scaled) applied voltages. Then ψ is specified at the contacts by requiring that the *space charge* (i.e. the zero-order term in (1.1)) should vanish there. Homogeneous Neumann conditions are imposed on $\partial\Omega\backslash\partial\Omega_D$. The function d typically has sign changes across thin transition regions or interfaces (between "p" and "n" regions

1991 *Mathematics Subject Classification.* Primary 65N55, 65N22; Secondary 65H10, 65F10.
This paper is in final form and no version of it will be submitted for publication elsewhere.

0271-4132/94 $1.00 + $.25 per page

of the device), and as a consequence ψ, v, w have interior layers at or near these interfaces, with width determined by the small parameter λ ([8]).

To solve this system we first subdivide Ω into open convex pairwise disjoint quadrilateral *substructures* $\Omega^{(i)}$ such that $\overline{\Omega} = \cup_i \overline{\Omega^{(i)}}$. Subdividing each substructure into two triangles yields a "coarse grid" with diameter H which is further subdivided to give a fine triangular grid with diameter h. We assume for theoretical purposes that the refinement is *quasi-uniform* with respect to both H and h and that the fine grid is of *weakly acute type*. We also assume that the *collision points* $\partial\Omega_D \cap \overline{\partial\Omega_N}$ are vertices of substructures and hence are nodes of both the coarse and fine grids.

We discretize (1.1)-(1.3) in S_h, the space of piecewise linear finite elements with respect to the fine grid. We denote by ϕ_p the usual nodal basis functions in S_h, where p ranges over all nodes of the fine grid. Let $(\cdot,\cdot)$ denote the usual L_2 inner product, and define the corresponding discrete inner product $\langle f, g\rangle$ to be the integral of the piecewise linear interpolant of fg over Ω. These inner products can be extended to vector-valued functions f, g in the obvious way. Also, if $X \in S_h$, we define the piecewise constant function $\overline{X}$ so that $\exp(\overline{X})$ is the *harmonic average* of $\exp(X)$, i.e. for each triangle T, $\exp(\overline{X}|_T) = (\mathcal{A}(T)^{-1} \int_T \exp(-X))^{-1}$, where $\mathcal{A}(T)$ is the area of T.

We now define an iterative method for finding discrete solutions of (1.1)-(1.3). This version of the so-called *Gummel's method* consists of iterating the map $\mathcal{G} : (V, W) \longmapsto (\tilde{V}, \tilde{W})$ on $(S_h)^2$, defined as follows.

Step 1. (Fractional Step) Find $\tilde{\Psi} \in S_h$ such that for all $p \notin \partial\Omega_D$,

$$\lambda^2(\nabla\tilde{\Psi}, \nabla\phi_p) + \langle \delta\{\exp(\tilde{\Psi} - V) - \exp(W - \tilde{\Psi})\} - d,\ \phi_p\rangle = 0. \tag{1.4}$$

Step 2. Find $\tilde{V} \in S_h$ such that for all $p \notin \partial\Omega_D$,

$$(\exp(\overline{\tilde{\Psi} - V})\nabla\tilde{V}, \nabla\phi_p) - \langle \sigma\rho_v r(\tilde{\Psi}, V, W), \phi_p\rangle = 0. \tag{1.5}$$

Step 3. Find $\tilde{W} \in S_h$ such that for all $p \notin \partial\Omega_D$,

$$(\exp(\overline{W - \tilde{\Psi}})\nabla\tilde{W}, \nabla\phi_p) + \langle \sigma\rho_w r(\tilde{\Psi}, \tilde{V}, W), \phi_p\rangle = 0. \tag{1.6}$$

All iterates are assumed to satisfy the essential boundary conditions on $\partial\Omega_D$. If the iterates converge then the limit $(\Psi, V, W) \in (S_h)^3$, satisfies a finite element discretization of (1.1)-(1.3) with two modifications. Firstly the zeroth order terms have been "mass lumped" using the discrete inner product. As well as providing a simple pointwise evaluation for complicated nonlinear terms, this mass-lumping facilitates the formulation of globally convergent monotone iterative schemes for calculating the fractional step (1.4) (see §2). Secondly the exponential coefficients in the continuity equations (1.2), (1.3) are replaced by their harmonic averages. This can be interpreted as a certain generalisation to $2D$ of the classical Scharfetter-Gummel discretization for these equations. It can also be interpreted in terms of a "hybrid' mixed finite element method and, as

such, ensures that the resulting (piecewise constant) approximations to the electron and hole currents ($\exp(\psi - v)\nabla v$ and $\exp(w - \psi)\nabla w$) have weak conservation properties ([**2**]). For more details see [**3**], [**4**].

2. Theoretical results

Since the work of Kerkhoven and Jerome ([**7**], [**6**], and the references therein) it has been known that, under appropriate assumptions, Gummel's map $\mathcal{G}$ is a contraction (on an appropriate space) for fixed h, provided the *applied bias* $\alpha := \max\{|\underline{\alpha}|, |\overline{\alpha}|\}$ is sufficiently small, where $\underline{\alpha} = \min_i \alpha_i$ and $\overline{\alpha} = \max_i \alpha_i$. Since numerical device modellers are often concerned with algorithmic complexity (i.e. with the cost in CPU time for a specified accuracy), it is also of interest to study how the Lipschitz constant of $\mathcal{G}$ changes as the mesh is refined. In [**3**] we have studied this question using a refinement of the techniques of Kerkhoven and Jerome. We also make use of a discrete Sobolev inequality well-known in the domain decomposition literature, namely that the uniform norm of an arbitrary element of S_h (which vanishes at least at one point of Ω) can be bounded in terms of its $H^1(\Omega)$ seminorm, where the constant of proportionality grows logarithmically with h as the mesh is refined (see, for example [**1**], [**5**]). Let B denote the set $\{(V, W) \in (S_h)^2 : \underline{\alpha} \leq V, W \leq \overline{\alpha}\}$, equipped with the norm $\|(V, W)\|_B = \|V\|_{H^1(\Omega)} + \|W\|_{H^1(\Omega)}$. Then we have the following result.

THEOREM 2.1. Let $r = 0$ in (1.2),(1.3). Then, $\mathcal{G} : B \to B$ and for each $M > 0$ there exists a constant $C > 0$, independent of h, such that for all $(V^i, W^i) \in B$, $i = 1, 2$ and all $\alpha \leq M$, we have

$$\|\mathcal{G}(V^1, W^1) - \mathcal{G}(V^2, W^2)\|_B \leq C\alpha(1 - \log h)^{1/2}\|(V^1, W^1) - (V^2, W^2)\|_B.$$

The assumption $r = 0$ plays a crucial role in the proof. It yields a discrete maximum principle for the solutions of (1.5), (1.6) which in turn implies that $\mathcal{G} : B \to B$. This assumption, which is made by all the other convergence analyses of Gummel's method of which we are aware, can be physically justified to some extent for devices with small current flow. However in general r is an essential part of the physical model and cannot be neglected. Thus it remains an important open question to extend the present analysis to include r.

Theorem 2.1 shows that the convergence of Gummel's method only degrades logarithmically with h as the mesh is refined, for fixed α. Inside each Gummel iterate we have to solve the semilinear problem (1.4). Since this may be regarded as singularly perturbed with respect to the small parameter λ, standard analyses of Newton's method will predict a convergence ball with radius dependent on λ (as well as h). Instead in [**3**], [**4**] we exploit the monotonicity in (1.4) to devise a monotone quasi-Newton scheme with an arbitrarily large radius of convergence. To introduce this, think of (1.4) as the problem of finding a solution $\tilde{\boldsymbol{\Psi}}$ to the nonlinear problem $\boldsymbol{F}(\boldsymbol{\Psi}) = \boldsymbol{0}$, where $\boldsymbol{\Psi}$ denotes the vector of nodal values of $\Psi \in S_h$. Let $J(\boldsymbol{\Psi})$ denote the Jacobian of $\boldsymbol{F}$. Then our method is:

- Assume we have *lower* and *upper solutions* $\mathbf{\Lambda}^0, \mathbf{\Omega}^0$ which satisfy

$$\mathbf{\Lambda}^0 \leq \mathbf{\Omega}^0 \quad \textit{and} \quad \boldsymbol{F}(\mathbf{\Lambda}^0) \leq \mathbf{0} \leq \boldsymbol{F}(\mathbf{\Omega}^0).$$

- Then, for $k \geq 0$, set

$$\mathbf{\Lambda}^{k+1} = \mathbf{\Lambda}^k - (J^k)^{-1}\boldsymbol{F}(\mathbf{\Lambda}^k) \quad \textit{and} \quad \mathbf{\Omega}^{k+1} = \mathbf{\Omega}^k - (J^k)^{-1}\boldsymbol{F}(\mathbf{\Omega}^k),$$

where $J^k := \max\{J(\mathbf{\Lambda}^k), J(\mathbf{\Omega}^k)\}$, and the maximum is taken *elementwise*. It is shown in [3] that $\mathbf{\Lambda}^0$, $\mathbf{\Omega}^0$ (which are bounded independently of h) are easy to construct. For such starting vectors we prove in [3] the following (quadratic, mesh independent) convergence result.

THEOREM 2.2. The sequences $\{\mathbf{\Lambda}^k\}$, $\{\mathbf{\Omega}^k\}$ converge to the same limit $\tilde{\mathbf{\Psi}}$, which is the unique solution of $\boldsymbol{F}(\mathbf{\Psi}) = \mathbf{0}$. Moreover

$$||\mathbf{\Omega}^{k+1} - \mathbf{\Lambda}^{k+1}||_2 \leq C||\mathbf{\Omega}^k - \mathbf{\Lambda}^k||_2^2, \quad k \geq 0,$$

with a constant C depending on λ and δ but independent of h and k.

The proof of this result makes use of the monotonicity of the discretization of the zero-order term in (1.4). This property is present in the undiscretized equation (1.1) and has been preserved through the use of mass-lumping in the discretization. Thus $\boldsymbol{F}$ is monotone, but unfortunately $\boldsymbol{F}$ is neither convex or concave on any domain which contains the solution, and so the results of [9] on monotone Newton methods cannot be used. The special quasi-Newton method defined above gets around this difficulty and the results obtained can be thought of as a generalisation of those in [9].

3. Inner iteration, domain decomposition

Each step of the iteration (1.4)-(1.6) (combined with the quasi-Newton method for (1.4)) amounts to the finite element approximation of a mixed boundary value problem for a symmetric linear elliptic second-order PDE, with coefficients which may suffer severe (finite) jumps across narrow interior layers. For example in a simple $p-n$ diode with no applied bias at room temperature, the potential ψ has a layer around the $p-n$ interface in which ψ changes from about -18 to $+18$ [8]. Consequently, for small bias, $\exp(\psi - v)$ varies between 10^{-8} and 10^8 in this layer. This jump is correspondingly present in the coefficient of the discretized equation (1.5), and the associated linear system is thus severely ill-conditioned (similarly (1.6)). It is essential (even for two-dimensional applications) to find preconditioners which mollify the effects of these jumps.

Fortunately the theory of additive Schwarz methods provides us with a reasonable solution of this problem and also yields algorithms which are readily parallelisable. We adopt here an approach analogous to that proposed in [10]. Each of the finite element problems is equivalent to a large sparse symmetric positive definite (SPD) linear system. To solve this we first eliminate locally

the unknowns at interior nodes of substructures, yielding a new system $S\boldsymbol{x} = \boldsymbol{c}$ (where $\boldsymbol{x}$ now contains the unknown nodal values on the substructure boundaries, and S is the (SPD) Schur complement of the original system. We solve this latter system by the preconditioned conjugate gradient method (PCGM). The action of S can be computed by many local matrix-vector products plus nearest neighbour addition without assembling S explicitly.

Our preconditioner $\hat{S}$ is well-known among domain decomposition enthusiasts and goes back at least to the work of Bramble, Pasciak and Schatz [**1**]. It consists of the following steps: (i) For each substructure edge (except those at which essential boundary conditions are applied), invert the minor of S corresponding to interior nodes of that edge, and (ii) Invert the restriction of S to the coarse grid, with grid transfer operators defined by linear interpolation and its adjoint. Then add the results of (i) and (ii). Using a refinement of the elegant additive Schwarz analysis (e.g. [**5**], [**10**]), we can show that the condition number of $\hat{S}^{-1}S$ is bounded by $C(1 + \log(H/h))^2$, with C independent of H and h and also independent of the jumps of the coefficients of the underlying PDE across substructure boundaries.

Our implementation is on a MasPar MP-1 data parallel machine with $1K$ (= 1024) processors, arranged in a 32×32 array. We assign a (small) substructure to each processor. Computation of the action of S then requires many local actions of Schur complements on substructures, followed by local addition across substructure boundaries. This fits naturally into the "massively parallel" programming model. Important questions then arise concerning the implementation of the preconditioner, especially the coarse grid problem (which is still large and may be almost as badly conditioned as S). Although this question merits further research, we have chosen in the present work to use (inner) iterations for both the local edge solves and for the coarse grid solve. After extensive experiments we concluded that approximate inner solves have a detrimental effect on the performance of the (outer) CGM and consequently we have solved the inner problems to (essentially) machine precision also using CGM. In addition we precondition the coarse grid problem by diagonal scaling. All inner iterations can be done by parallel local operations. For example one multiplication by the coarse-grid operator involves many parallel 4×4 multiplications and local addition.

At present we have experiments only for uniform grids. The unit square is divided into $m \times m$ subdomains, each of which contains a uniform mesh of triangles with $n \times n$ interior nodes. Various tests have been performed ([**3**], [**4**]) on model scalar problems which show that the number of preconditioned outer CGM iterates predicted by the theory is sufficient for convergence. More interestingly, we find that CPU time only increases very modestly with the size of coefficient jumps across substructure boundaries, and that the algorithm scales nicely with machine size: With m fixed, the solution time grows with $O(n^2)$, which is the time needed for (local) matrix-vector multiplication with the (full) Schur complements.

As a more challenging problem we have solved the semiconductor problem (1.1)-(1.3) in the case of a reverse biased $p-n$ diode. Full details of the parameter values and device geometry are in [**3**]. With an applied bias of 0.02 volts and with $r = 0$ we obtained the results in Table 1.

m	n	**Quasi–Newton its**	**Gummel its**	**Time (s)**
8	1	54	8	341
16	1	55	8	798
32	1	56	8	1666

TABLE 1

The number of quasi-Newton iterates given is those required for the solution of (1.4) in the first Gummel iterate. After that very few are required. This number remains fixed as the mesh is refined (as implied by Theorem 2.2). Moreover the number of overall Gummel iterates also appears unaffected by the mesh refinement, which is slightly better than that predicted by Theorem 2.1. Since our machine has a 32×32 array of processors, the growth in solution time can be attributed almost entirely to the cost of solution of the coarse grid problems. Qualitatively similar results were obtained with the recombination r switched on. Full details are in [**3**], [**4**].

REFERENCES

1. J. H. Bramble, J. E. Pasciak and A. L. Schatz, *The construction of preconditioners for elliptic problems by substructuring I*, Math. Comp. **47** (1986), 103–134.
2. F. Brezzi, L.D. Marini and P. Pietra, *Numerical simulation of semiconductor devices*, Comp. Meth. Appl. Mech. Engrg. **75** (1989), 493–514.
3. R K. Coomer and I. G. Graham, *Massively parallel methods for semiconductor device modelling*. Mathematics Preprint Number 93/02, University of Bath, 1993 (submitted for publication).
4. R. K. Coomer, *Parallel iterative methods in semiconductor device modelling*, Ph.D. thesis, University of Bath, 1994.
5. M. Dryja and O. B. Widlund, *Some domain decomposition algorithms for elliptic problems*, Iterative methods for large linear systems (L. Hayes and D. Kincaid, eds.), Academic Press, Orlando, 1989.
6. J. W. Jerome and T. Kerkhoven, *A finite element approximation theory for the drift diffusion semiconductor model*, SIAM J. Numer. Anal. **28** (1991), 403-422.
7. T. Kerkhoven, *Coupled and decoupled algorithms for semiconductor simulation*, Ph.D. thesis, Yale University, 1985.
8. P.A. Markowich, C.A. Ringhofer and C. Schmeiser, *Semiconductor equations*, Springer, Wien - New York, 1990.
9. J. M. Ortega and W. C. Rheinboldt, *Iterative solution of nonlinear equations in several variables*, Academic Press, New York, 1970.
10. B. F. Smith, *An optimal domain decomposition preconditioner for the finite element solution of linear elasticity problems*, SIAM J. Sci. Stat. Comput. **13** (1992), 364–378.

SCHOOL OF MATHEMATICAL SCIENCES, UNIVERSITY OF BATH, BATH BA2 7AY, UNITED KINGDOM (CORRESPONDENCE TO: I.G. GRAHAM)
E-mail address: igg@maths.bath.ac.uk