ELSEVIER

Available at WWW.MATHEMATICSWEB.ORG POWERED BY SCIENCE DIRECT

Journal of Computational and Applied Mathematics 159 (2003) 341–364

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

www.elsevier.com/locate/cam

Monotone iterative methods for the adaptive finite element solution of semiconductor equations☆

R.-C. Chen, Jinn-Liang Liu*

Department of Applied Mathematics, National Chiao Tung University, 1001 Ta Hsueh Road, Hsinchu 300, Taiwan

Received 15 March 2002; received in revised form 19 April 2003

Abstract

Picard, Gauss–Seidel, and Jacobi monotone iterative methods are presented and analyzed for the adaptive finite element solution of semiconductor equations in terms of the Slotboom variables. The adaptive meshes are generated by the 1-irregular mesh refinement scheme. Based on these unstructured meshes and a corresponding modification of the Scharfetter–Gummel discretization scheme, it is shown that the resulting finite element stiffness matrix is an M-matrix which together with the Shockley–Read–Hall model for the generation–recombination rate leads to an existence–uniqueness–comparison theorem with simple upper and lower solutions as initial iterates. Numerical results of simulations on a MOSFET device model are given to illustrate the accuracy and efficiency of the adaptive and monotone properties of the present methods.

Keywords: Monotone iteration; Drift-diffusion model; Adaptive finite element

1. Introduction

Computer-aided simulation is one of the important processes in developing semiconductor devices. Numerical methods for the fundamental semiconductor equations play a significant role in this development. For most practical device structures, the electrostatic potential and carrier concentrations exhibit extreme layers [26], particularly in the neighborhood of p–n junctions and the oxide [9]. The presence of layers implies that adaptive mesh generation of unstructured grids is inevitable if an accurate and efficient device simulation platform is required [9].

To obtain numerical solutions of semiconductor equations, one must solve a system of nonlinear algebraic equations resulting from a discretization by, for example, the finite element method. The

☆ This work was supported by NSC under Grant 90-2115-M-009-028, Taiwan.
* Corresponding author.
E-mail address: jinnliu@math.nctu.edu.tw (J.-L. Liu).

0377-0427/03/$ - see front matter
doi:10.1016/S0377-0427(03)00538-7

standard method for the solution of the system is Newton's method or its variation. Newton's method is a local method that, in general, is very sensitive to the initial guess and converges quadratically in a sufficiently small neighborhood of the exact solution [28].

The method of monotone iterations is a classical tool for the study of the existence of the solution of semilinear PDEs of certain types [1,14,29,33]. It is also useful for the numerical solution of these types of problems approximated, for instance, by the finite difference [12,19,30], finite element [16], or boundary element [6,11,32] method. It is a constructive method that depends essentially on only one parameter, called the monotone parameter herein, which determines the convergence behavior of the iterative process. Based on adaptive 1-irregular finite element meshes, we extend this classical method to device simulation by exploiting a very special nonlinear property of the drift-diffusion model that the carriers satisfy Maxwell–Boltzmann statistical laws by which the model can be expressed in terms of the electrostatic potential and the Slotboom variables [5,17,35,39].

Embedded in the widely used Gummel's decoupling algorithm [13,34], three monotone iterative methods, namely, Picard, Gauss–Seidel, and Jacobi methods, are presented and analyzed in this paper. By extending the Scharfetter–Gummel discretization scheme proposed in [41] to the adaptive finite element approximation, it is shown that the resulting stiffness matrix is an M-matrix which together with the Shockley–Read–Hall model for the generation– recombination rate leads to an existence–uniqueness–comparison theorem with simple upper or lower solution as an initial guess. These methods also yield positivity of carrier concentrations under conditions of strong recombination similar to that of the method presented in [35] for the drift-diffusion model in one-space dimension.

In the next section, we state the model from the Van Roosbroeck system to the corresponding Slotboom-variable formulation and Gummel's decoupling algorithm. The model is subject to Dirichlet and Neumann types of conditions on various parts of the boundary of a real-life device domain. In Section 3, we first analyze the matrix properties of the resulting adaptive finite element systems for the Poisson equation, which then lead to the M-matrix properties for the semiconductor equations. Starting with the upper and lower solutions as initial guesses, it is shown in Section 4 that maximal and minimal sequences generated by Picard, Gauss–Seidel, and Jacobi iterations all converge monotonically from above and below to the unique solution of the resulting nonlinear system. We then summarize in Section 5 our implementation procedures into two algorithms, namely, monotone-Gummel and adaptive algorithms which combine Gummel's decoupling, monotone iterative, and adaptive methods. Section 6 represents a part of our extensive numerical experiments on various n-MOSFET device models to demonstrate the accuracy and efficiency of adaptive and monotone properties of the proposed methods. Moreover, numerical results of the Jacobi and Gauss–Seidel monotone iterations are also given to verify the theoretical results.

2. The drift-diffusion model

The steady-state Van Roosbroeck system [43], usually referred to as the drift-diffusion model of semiconductor devices, is

$$\Delta\phi = \frac{q}{\varepsilon_s}(n - p + N_A^- - N_D^+), \tag{2.1}$$

$$\frac{1}{q}\nabla \cdot J_n = R(\phi, n, p), \tag{2.2}$$

$$\frac{1}{q}\nabla \cdot J_p = -R(\phi, n, p), \tag{2.3}$$

where ϕ is the electrostatic potential, n and p are the electron and hole concentrations, q is the elementary charge, ε_s is the permittivity constant of the semiconductor, N_A^- and N_D^+ are densities of ionized impurities, J_n and J_p are the electron and hole current densities, and $R(\phi, n, p)$ is a function describing the balance of generation and recombination of electrons and holes. The current densities J_n and J_p are defined as follows:

$$J_n = -q\mu_n n\nabla\phi + qD_n\nabla n, \tag{2.4}$$

$$J_p = -q\mu_p p\nabla\phi - qD_p\nabla p, \tag{2.5}$$

where μ_n and μ_p are the field-dependent electron and hole mobilities and the diffusion coefficients of electrons and holes are expressed in the Einstein relations

$$D_n = V_T\mu_n, \quad D_p = V_T\mu_p \tag{2.6}$$

with $V_T = kT/q$ being the thermal voltage, k Boltzmann's constant, and T a constant temperature.

Based on Maxwell–Boltzmann statistical laws [17,39], we may write system (2.1)–(2.3) as

$$\Delta\phi = F(\phi, u, v), \tag{2.7}$$

$$\nabla \cdot \left(D_n n_i \exp\left(\frac{\phi}{V_T}\right) \nabla u \right) = R(\phi, u, v), \tag{2.8}$$

$$\nabla \cdot \left(D_p n_i \exp\left(\frac{-\phi}{V_T}\right) \nabla v \right) = R(\phi, u, v), \tag{2.9}$$

where

$$u = \exp\left(\frac{-\varphi_n}{V_T}\right), \tag{2.10}$$

$$v = \exp\left(\frac{\varphi_p}{V_T}\right) \tag{2.11}$$

are the Slotboom variables in which the quasi-Fermi potentials φ_n and φ_p are expressed as

$$n = n_i \exp\left(\frac{\phi - \varphi_n}{V_T}\right),$$

$$p = n_i \exp\left(\frac{\varphi_p - \phi}{V_T}\right),$$

and

$$F(\phi,u,v)=\frac{qn_i}{\varepsilon_s}\left(u\exp\left(\frac{\phi}{V_T}\right)-v\exp\left(\frac{-\phi}{V_T}\right)\right)+\frac{q(N_A^- - N_D^+)}{\varepsilon_s}, \tag{2.12}$$

$$R(\phi,u,v)=\frac{n_i^2(uv-1)}{\tau_n^0(n_i v\exp(-\phi/V_T)+p_T)+\tau_p^0(n_i u\exp(\phi/V_T)+n_T)}. \tag{2.13}$$

Here we consider particularly the Shockley–Read–Hall generation–recombination model with n_i being the intrinsic carrier concentration, τ_n^0 and τ_p^0 the electron and hole lifetimes, and p_T and n_T the electron and hole densities associated with energy levels of the traps.

In device simulation, most numerical methods have been developed for the approximation of system (2.1)–(2.3) with the primal state variables (ϕ,n,p) [3,36]. Nevertheless, there are also some methods developed for the Slotboom-variable formulation (2.7)–(2.9) with the state variables (ϕ,u,v) [35,39]. It has been observed in [3] that the choice of the variables u and v defined in (2.10) and (2.11) is preferable since they lead to self-adjoint and positive definite matrices and that the resulting matrices are better scaled than those of φ_n and φ_p. The theory and solution methods for systems of self-adjoint partial differential equations have reached a very high standard such that a solution of the static semiconductor equations in (ϕ,u,v) can be carried out very efficiently. However, the major drawback of the variables u and v lies in the enormous dynamic range required for real number representation in actual computations. By recalling definitions (2.10) and (2.11), for example, we find that the exponential terms vary more than 32 orders of magnitude for $\phi\in[-1,1]V$. It is, therefore, obvious that computations are limited to low-voltage applications. Although Newton's method can be successfully applied to (2.7)–(2.9) [3,25], it is very sensitive to the initial guess of those variables due to its local convergence property. In practical simulation, the device terminal characteristics of I–V curves (i.e., I–V points) is usually of interest. The conventional approach to obtain these curves is a homotopy process from lower biases to higher biases by Newton's method, which can be very costly in terms of computing time and human work load associated with the convergence problems of the method. On the other hand, with formulation (2.7)–(2.9), the monotone iterative method presented in this paper is a global method for which the initial guess can be chosen in a more arbitrary way, see Theorem 4.1 and Section 6 below. This then allows us to have a simultaneous (parallel) computing of multiple I–V points with various biasing conditions and with independent constant initial guesses for each I–V point calculation. The computational effort can thus be dramatically reduced [22].

System (2.7)–(2.9) is subject to some appropriate conditions on the boundary of a rectangular region denoted by $\Omega\subset\Re^2$ and shown in Fig. 1. The domain is bounded by the six segments $\Gamma_{AB}=\overline{AB}$, $\Gamma_{BC}=\overline{BC}$, $\Gamma_{CD}=\overline{CD}$, $\Gamma_{DE}=\overline{DE}$, $\Gamma_{EF}=\overline{EF}$, and $\Gamma_{FA}=\overline{FA}$, i.e., its boundary

$$\partial\Omega=\Gamma_{AB}\cup\Gamma_{BC}\cup\Gamma_{CD}\cup\Gamma_{DE}\cup\Gamma_{EF}\cup\Gamma_{FA}.$$

By assuming the charge neutrality condition and the mass-action law [40], the boundary conditions of the system in terms of the variables ϕ, u, and v are described as follows: The Dirichlet part of boundary conditions

$$\phi=V_O+V_b \quad \text{on } \partial\Omega_D, \tag{2.14}$$

$$u=\exp\left(\frac{-V_O}{V_T}\right) \quad \text{on } \partial\Omega_D, \tag{2.15}$$

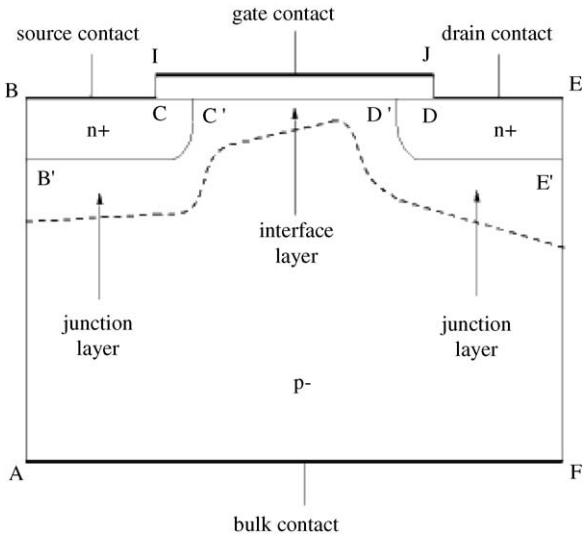

Fig. 1. Geometry of an n-MOSFET device.

$$v = \exp\left(\frac{V_O}{V_T}\right) \quad \text{on } \partial\Omega_D, \tag{2.16}$$

where $\partial\Omega_D = \Gamma_{BC} \cup \Gamma_{DE} \cup \Gamma_{FA}$, $V_O = V_S, V_D$, or V_B, are the source, drain, or substrate voltage applied on the ohmic contact parts Γ_{BC}, Γ_{DE}, or Γ_{FA}, respectively. V_b represents the built-in potential [40]. On the boundary Γ_{AB} and Γ_{EF}, we assume that the normal components of the electric field $E = -\nabla\phi$ and current densities are zero,

$$v \cdot E = 0, \quad v \cdot J_n = 0 \quad \text{and} \quad v \cdot J_p = 0.$$

These conditions lead to the Neumann boundary conditions

$$\frac{\partial\phi}{\partial v} = 0 \quad \text{on } \partial\Omega_N = \Gamma_{AB} \cup \Gamma_{EF}, \tag{2.17}$$

$$\frac{\partial u}{\partial v} = 0 \quad \text{on } \partial\Omega_N, \tag{2.18}$$

$$\frac{\partial v}{\partial v} = 0 \quad \text{on } \partial\Omega_N, \tag{2.19}$$

where v is the outward normal on $\partial\Omega$, and to the mixed and Neumann boundary conditions

$$\phi_- = \phi_+, \quad \varepsilon_s \partial_y \phi_- = \varepsilon_d \partial_y \phi_+ \quad \text{on } \partial\Omega_R = \Gamma_{CD}, \tag{2.20}$$

$$\frac{\partial u}{\partial v} = 0 \quad \text{on } \partial\Omega_R, \tag{2.21}$$

$$\frac{\partial v}{\partial v} = 0 \quad \text{on } \partial\Omega_R, \tag{2.22}$$

where V_G is the voltage applied on the gate, t_{ox} is the gate oxide thickness, ε_d is the permittivity constant of the gate oxide, and the $+$ and $-$ signs refer to as the limits from the oxide and the semiconductor regions, respectively, to the interface. The oxide region Ω_O (the rectangular region CIJD) is assumed to be free of charges, i.e., the Laplace equation $\Delta\phi=0$ holds there and $\phi_{IJ}=V_b+V_G$.

We now describe Gummel's decoupling algorithm for the DD model. With a given initial guess $(\phi^{(0)}, u^{(0)}, v^{(0)})$ and for each Gummel's iteration index g, $g=0,1,\ldots,\phi^{(g+1)}$ is computed by solving the nonlinear Poisson equation in Ω and the Laplace equation in Ω_O

$$
\begin{aligned}
&\Delta\phi^{(g+1)} = F(\phi^{(g+1)}, u^{(g)}, v^{(g)}) \quad \text{in } \Omega,\\
&\phi^{(g+1)} = V_O + V_b \quad \text{on } \partial\Omega_D,\\
&\frac{\partial\phi^{(g+1)}}{\partial v} = 0 \quad \text{on } \partial\Omega_N,\\
&\Delta\phi^{(g+1)} = 0 \quad \text{in } \Omega_O,\\
&\phi^{(g+1)} = V_b + V_G \quad \text{on } \Gamma_{IJ},\\
&\frac{\partial\phi^{(g+1)}}{\partial v} = 0 \quad \text{on } \Gamma_{CI}\cup\Gamma_{DJ},\\
&\phi_-^{(g+1)} = \phi_+^{(g+1)}, \quad \varepsilon_s\partial_y\phi_-^{(g+1)} = \varepsilon_d\partial_y\phi_+^{(g+1)} \quad \text{on } \partial\Omega_R
\end{aligned}
\tag{2.23}
$$

and then $u^{(g+1)}$ is computed by solving the electron current continuity equation, with now the known functions $\phi^{(g+1)}$ and $v^{(g)}$,

$$
\begin{aligned}
&\nabla\cdot\left(D_n^{(g+1)} n_i \exp\left(\frac{\phi^{(g+1)}}{V_T}\right)\nabla u^{(g+1)}\right) = R(\phi^{(g+1)}, u^{(g+1)}, v^{(g)}) \quad \text{in } \Omega,\\
&u^{(g+1)} = \exp\left(\frac{-V_O}{V_T}\right) \quad \text{on } \partial\Omega_D,\\
&\frac{\partial u^{(g+1)}}{\partial v} = 0 \quad \text{on } \partial\Omega_N\cup\partial\Omega_R,
\end{aligned}
\tag{2.24}
$$

and finally $v^{(g+1)}$ is computed by solving the hole current continuity equation, with both $\phi^{(g+1)}$ and $u^{(g+1)}$ known,

$$
\begin{aligned}
&\nabla\cdot\left(D_p^{(g+1)} n_i \exp\left(\frac{-\phi^{(g+1)}}{V_T}\right)\nabla v^{(g+1)}\right) = R(\phi^{(g+1)}, u^{(g+1)}, v^{(g+1)}) \quad \text{in } \Omega,\\
&v^{(g+1)} = \exp\left(\frac{V_O}{V_T}\right) \quad \text{on } \partial\Omega_D,\\
&\frac{\partial v^{(g+1)}}{\partial v} = 0 \quad \text{on } \partial\Omega_N\cup\partial\Omega_R
\end{aligned}
\tag{2.25}
$$

until all preset stopping criteria are satisfied.

This decoupling algorithm is widely used in practical simulations of semiconductor devices. Important analyses of the algorithm pertaining to the existence, stability, convergence, effectiveness, etc. have been thoroughly studied by Jerome and Kerkhoven [17,18,20]. The algorithm defines an outer iteration in a simulation procedure. The monotone method proposed here is applied individually to each one of the nonlinear algebraic systems resulting from the discretization of differential equations (2.23)–(2.25).

Before going into the discrete systems, we describe the essence of the monotone iterative method by using a nonlinear Poisson model problem [16,29,30]. Our description is primarily based on [30]. Consider the semilinear elliptic PDEs

$$\begin{aligned} &\Delta\phi = f(x, y, \phi) \quad \text{in } \Omega \\ &a\frac{\partial\phi}{\partial v} + b\phi = g(x, y) \quad \text{on } \partial\Omega, \end{aligned} \tag{2.26}$$

where $a \equiv a(x, y)$ and $b \equiv b(x, y)$ are nonnegative functions on $\partial\Omega$ with $a + b > 0$, and f and g are prescribed nonlinear and linear functions in their respective domains.

Applying the finite element method to (2.26) on a certain partition of the domain, we obtain a system of nonlinear algebraic equations in a compact form

$$\mathscr{A}Z = -F(Z), \tag{2.27}$$

where $\mathscr{A}$ is an $N \times N$ matrix, $Z \equiv (z_1, \ldots, z_N)^{\mathrm{T}}$ is an unknown vector, and $F(Z) \equiv (F_1(Z), \ldots, F_N(Z))^{\mathrm{T}}$ is a vector associated with both functions f and g. We denote by $\bar{\Omega}^h$ the set of grid points associated with the partition of $\bar{\Omega} = \Omega \cup \partial\Omega$, i.e.,

$$\bar{\Omega}^h = \{(x_i, y_i) \in \Omega \cup \partial\Omega : \ i = 1, 2, \ldots, N\},$$

where N is the total number of regular nodes (see Section 3 for the definition) of the finite element partition. The sets of grid nodes in Ω, $\partial\Omega$, $\partial\Omega_{\mathrm{D}}$, $\partial\Omega_{\mathrm{N}}$, and $\partial\Omega_R$ are similarly denoted by Ω^h, $\partial\Omega^h$, $\partial\Omega^h_{\mathrm{D}}$, $\partial\Omega^h_{\mathrm{N}}$, and $\partial\Omega^h_R$, respectively.

Starting with a given initial vector $Z^{(0)}$ for problem (2.27), the monotone iterative method generates a sequence of iterates $\{Z^{(m)}\}$, $m = 0, 1, \ldots$, by solving the equation

$$\mathscr{A}Z^{(m+1)} + \Lambda Z^{(m+1)} = \Lambda Z^{(m)} - F(Z^{(m)}), \tag{2.28}$$

where Λ is a nonnegative diagonal matrix in which its entries γ_k, $k = 1, \ldots, N$, are parameters that are determined by the nonlinear function f. Under various conditions on the matrices $\mathscr{A}$ and Λ; or equivalently on the discretization and the function f, it has been shown in [30] that, for the finite difference approximation, the sequence $\{Z^{(m)}\}$ generated by Eq. (2.28) converges monotonically to a solution of (2.27). Obviously, the convergence behavior of the monotone process (2.28) is essentially dedicated by Λ.

There are some variants of the Picard method (2.28), such as Jacobi, Gauss–Seidel, and block iterative monotone methods [30]. We shall discuss the Gauss–Seidel method and the Jacobi method below.

3. Matrix properties of the drift-diffusion model

For each Gummel's iteration and after the discretization, each one of the nonlinear problems (2.23)–(2.25) will result in a system of nonlinear algebraic equations similar to (2.27) with which our main concern now is the property of the resulting stiffness matrix $\mathscr{A}$. The discretization considered here is particularly based on the adaptive finite element method using the 1-irregular mesh refinement scheme [10,24].

Let T be a finite element partition of the domain Ω such that $T=\{\tau_j;\ j=1,\dots,M, \bar{\Omega}=\bigcup_{j=1}^{M}\bar{\tau}_j\}$, and $S_h(T)$ denote a finite element space on T. Let N^{a} be a set of N indices that are assigned to active degrees of freedom (i.e., regular nodes) and N^{c} assigned to constrained degrees of freedom (irregular nodes). By an *active* degree of freedom, we mean one that defines a parameter associated with the global stiffness matrix whereas a *constrained* degree of freedom is a linear combination of active degrees of freedom that are associated with the constrained node by element connectivity. For each $i\in N^{\mathrm{c}}$, there exists a set $A(i)\subset N^{\mathrm{a}}$ of corresponding active degrees of freedom such that the resulting finite element space $S_h(T)$ consists of continuous functions. Let ϕ_h be an arbitrary function in $S_h(T)$. If the element τ is a square, then ϕ_h is of the following form:

$$\begin{aligned}\phi_h &= \sum_{i\in N^{\mathrm{a}}}\phi_i\hat{b}_i+\sum_{j\in N^{\mathrm{c}}}\phi_j\hat{b}_j\\ &= \sum_{i\in N^{\mathrm{a}}}\phi_i\hat{b}_i+\sum_{j\in N^{\mathrm{c}}}\sum_{k\in A(j)}\frac{1}{2}\phi_k\hat{b}_j,\end{aligned}$$

where ϕ_i are scalars and $\hat{b}_i$ are *unconstrained bilinear bases* which can be constructed via the following four shape functions:

$$\begin{aligned}s_1&=(1-\xi)(1-\eta)/4,\\ s_2&=(1+\xi)(1-\eta)/4,\\ s_3&=(1+\xi)(1+\eta)/4,\\ s_4&=(1-\xi)(1+\eta)/4\end{aligned}$$

defined on the reference element $\hat{\tau}=\{(\xi,\eta):\ |\xi|\leqslant 1,|\eta|\leqslant 1\}$. For every $i\in N^{\mathrm{a}}$, let

$$C(i)=\{j\in N^{\mathrm{c}}\mid i\in A(j)\}.$$

We rewrite ϕ_h in the form

$$\begin{aligned}\phi_h &= \sum_{i\in N^{\mathrm{a}}}\phi_i\hat{b}_i+\sum_{k\in N^{\mathrm{a}}}\sum_{j\in C(k)}\frac{1}{2}\phi_k\hat{b}_j\\ &= \sum_{i\in N^{\mathrm{a}}}\phi_i\left(\hat{b}_i+\sum_{j\in C(i)}\frac{1}{2}\hat{b}_j\right).\end{aligned}$$

Thus, the functions

$$b_i = \hat{b}_i + \sum_{j \in C(i)} \frac{1}{2} \hat{b}_j \quad \forall i \in N^a \tag{3.1}$$

form *constrained bilinear bases.*

Let $(x_i, y_i) \in \bar{\Omega}$ be a mesh point in T. For each $i \in N^a$ and using the standard notation $\phi_i \approx \phi(x_i, y_i)$, there exists a set $V(i) \subset N^a$, $i \notin V(i)$, of active degrees of freedom such that the finite element approximation of problem (2.23) results in a system

$$\xi_i \phi_i - \sum_{k \in V(i)} \xi_k \phi_k = -F_i(\phi_i) + F_i^*, \tag{3.2}$$

where

$$\xi_k = -B_h(b_i, b_k) \equiv -\sum_{\tau \in T} \int_\tau \nabla b_i \cdot \nabla b_k \, \mathrm{d}x \, \mathrm{d}y,$$

$$\xi_i = B_h(b_i, b_i),$$

$$F_i(\phi_i) \equiv \sum_{\tau \in T} \int_\tau F(\phi_i, u_i, v_i) b_i \, \mathrm{d}x \, \mathrm{d}y$$

and F_i^* is associated with the boundary conditions in (2.23) if $(x_i, y_i) \in \partial\Omega$ and $F_i^* = 0$ if $(x_i, y_i) \in \Omega$.

Since the partition consists of rectangular elements, the following result can be easily proved (see, e.g., [2]) with each type of the 1-irregular elements as given in [24].

Theorem 3.1. *The matrix induced by* (3.2) *is diagonally dominant, i.e.,*

$$\xi_i \geqslant \sum_{k \in V(i)} \xi_k, \tag{3.3}$$

$$\xi_k \geqslant 0 \quad \forall k \in V(i). \tag{3.4}$$

Furthermore, the strict inequality in (3.3) *holds for at least one* $i \in N^a$.

However, for continuity equations (2.24) and (2.25), it is well known that the Scharfetter–Gummel discretization induces nonphysical diffusion in the direction normal to drift velocity for multidimensional problems, which has led to various modifications of the method [3,8,21,27,34,37,38,41,42]. In order to obtain the same matrix property as that of Theorem 3.1, we extend in particular the method proposed in [41] to the 1-irregular mesh refinement scheme. By analogy, it suffices to consider only the electron continuity equation.

For each $j \in N^a$ with $u_j \approx u(x_j, y_j)$, (2.24) is approximated by

$$L[u_j] \equiv \eta_j u_j - \sum_{k \in V(j)} \eta_k u_k = -R_j(u_j) + R_j^* \tag{3.5}$$

or in the more compact matrix form

$$\mathscr{A}U = -R(U) + R^*, \tag{3.6}$$

where

$$\eta_j = \sum_{k\in V(j)} \eta_k, \tag{3.7}$$

$$\eta_k = \xi_k d_k, \tag{3.8}$$

$$d_k = D_n|_{(k,j)} n_i B\left(\frac{\phi_j^{(g+1)} - \phi_k^{(g+1)}}{V_T}\right)\exp\left(\frac{\phi_j^{(g+1)}}{V_T}\right), \tag{3.9}$$

$$\xi_k = -B_h(b_j, b_k), \tag{3.10}$$

$$R_j(u_j) = R(\phi_j^{(g+1)}, u_j^{(g+1)}, v_j^{(g)}),$$

$$D_n|_{(k,j)} = (D_n(x_k, y_k) + D_n(x_j, y_j))/2,$$

$$U = (u_1, \ldots, u_N),$$

$$R(U) = (R_1(u_1), \ldots, R_N(u_N))$$

and $B(t) = t/(\mathrm{e}^t - 1)$ is the Bernoulli function for any real number t and R_j^* is associated with the boundary conditions in (2.24) if $(x_j, y_j) \in \partial\Omega$ and $R_j^* = 0$ if $(x_j, y_j) \in \Omega$. Note that, by the definition of the Bernoulli function and of the diffusion coefficient, the factors d_k in (3.9) are positive. We thus conclude the following result.

Theorem 3.2. *The matrix $\mathscr{A}$ in (3.6) is diagonally dominant, i.e.,*

$$\eta_j \geqslant \sum_{k\in V(j)} \eta_k,$$

$$\eta_k \geqslant 0 \quad \forall k \in V(j). \tag{3.11}$$

Furthermore, the strict inequality in (3.11) holds for at least one $j \in N^{\mathrm{a}}$.

4. Monotone convergence results

The diagonal dominance of the resulting matrices (i.e., M-matrices) of the model problems (2.23)–(2.25) provides not only stability of numerical solutions (i.e., no nonphysical oscillations) but also convergence of iterative procedures when the special properties of the nonlinearity in these problems are taken into account. Moreover, the existence and uniqueness of the solutions can also be guaranteed by means of the construction of lower and upper solutions which are defined as follows:

Definition 4.1. A vector $\tilde{U} \equiv (\tilde{u}_1, \ldots, \tilde{u}_N) \in \mathfrak{R}^N$ is called an upper solution of (3.6) if it satisfies the following inequality:

$$\eta_i \tilde{u}_i - \sum_{k\in V(i)} \eta_k \tilde{u}_k \geqslant -R_i(\tilde{u}_i) + R_i^* \tag{4.1}$$

and $\hat{U} \equiv (\hat{u}_1, \ldots, \hat{u}_N) \in \mathfrak{R}^N$ is called a lower solution if

$$\eta_i \hat{u}_i - \sum_{k \in V(i)} \eta_k \hat{u}_k \leqslant -R_i(\hat{u}_i) + R_i^* \tag{4.2}$$

for $i \in N^a$.

As in the previous section, we only consider the monotone convergence for the electron continuity equation (2.24) since the nonlinear functionals in (2.23)–(2.25) are at right-hand sides and these equations are all associated with self-adjoint operators. It is obvious that every solution of (3.6) is an upper solution as well as a lower solution. We say that $\hat{U}$ and $\tilde{U}$ are ordered if $\hat{U} \leqslant \tilde{U}$. Given any ordered lower and upper solutions $\hat{U}$ and $\tilde{U}$, we define

$$\langle \hat{U}, \tilde{U} \rangle \equiv \{U \in \mathfrak{R}^N;\ \hat{U} \leqslant U \leqslant \tilde{U}\}, \tag{4.3}$$

$$\langle \hat{u}_i, \tilde{u}_i \rangle \equiv \{w_i \in \mathfrak{R};\ \hat{u}_i \leqslant w_i \leqslant \tilde{u}_i\}. \tag{4.4}$$

We only consider positive solutions, i.e., $0 < \hat{U} \leqslant U \leqslant \tilde{U}$, due to physical relation (2.10). Choose the nonnegative scalars γ_i such that

$$\gamma_i \equiv \max \left\{ \frac{\partial R_i(w_i)}{\partial u_i};\ w_i \in \langle \hat{u}_i, \tilde{u}_i \rangle \right\}$$

or in matrix form

$$\Lambda \equiv \operatorname{diag}(\gamma_i)$$

for $i \in N^d$. Then by adding the term $\gamma_i u_i$ on both sides of (3.5) we obtain the equivalent system

$$L[u_i] + \gamma_i u_i = \gamma_i u_i - R_i(u_i) + R_i^*. \tag{4.5}$$

It is easily seen from the definition of γ_i that $\gamma_i > 0$ and

$$\gamma_i u_i - R_i(u_i) + R_i^* \geqslant \gamma_i v_i - R_i(v_i) + R_i^* \quad \text{when } \tilde{u}_i \geqslant u_i \geqslant v_i \geqslant \hat{u}_i. \tag{4.6}$$

Let $\bar{U}^{(0)} = \tilde{U}$ be an initial iterate. We construct a sequence $\{\bar{U}^{(m+1)}\}$ by solving the linear system

$$\eta_i \bar{u}_i^{(m+1)} - \sum_{k \in V(i)} \eta_k \bar{u}_i^{(m)} + \gamma_i \bar{u}_i^{(m+1)} = \gamma_i \bar{u}_i^{(m)} - R_i(\bar{u}_i^{(m)}) + R_i^* \tag{4.7}$$

for $m = 0, 1, 2, \ldots$ and $i \in N^a$. Similarly, by using $\underline{U}^{(0)} = \hat{U}$ as another initial iterate, we obtain a sequence $\{\underline{U}^{(m+1)}\}$ from the linear system

$$\eta_i \underline{u}_i^{(m+1)} - \sum_{k \in V(i)} \eta_k \underline{u}_i^{(m)} + \gamma_i \underline{u}_i^{(m+1)} = \gamma_i \underline{u}_i^{(m)} - R_i(\underline{u}_i^{(m)}) + R_i^* \tag{4.8}$$

for $m = 0, 1, 2, \ldots$ and $i \in N^a$. We refer to $\{\bar{U}^{(m)}\}$ and $\{\underline{U}^{(m)}\}$ as the maximal and minimal sequences. We now show that these two sequences are monotone and converge to a solution of (3.6).

Lemma 4.1. *The maximal and minimal sequences* $\{\bar{U}^{(m)}\}$ *and* $\{\underline{U}^{(m)}\}$ *given by* (4.7) *and* (4.8) *with* $\bar{U}^{(0)} = \tilde{U}$ *and* $\underline{U}^{(0)} = \hat{U}$ *possess the monotone property*

$$\hat{U} \leqslant \underline{U}^{(m)} \leqslant \underline{U}^{(m+1)} \leqslant \bar{U}^{(m+1)} \leqslant \bar{U}^{(m)} \leqslant \tilde{U}, \quad m = 0, 1, 2, \ldots . \tag{4.9}$$

Moreover for each m, $\bar{U}^{(m)}$ *and* $\underline{U}^{(m)}$ *are ordered upper and lower solutions.*

Proof. Let $w_i^{(0)} = \bar{u}_i^{(0)} - \bar{u}_i^{(1)} = \tilde{u}_i - \bar{u}_i^{(1)}$. By (4.7)

$$(\eta_i + \gamma_i) w_i^{(0)} = (\eta_i + \gamma_i)\tilde{u}_i - \left[\sum_{k \in V(i)} \eta_k \bar{u}_k^{(0)} + \gamma_i \bar{u}_i^{(0)} - R_i(\bar{u}_i^{(0)}) + R_i^* \right]$$

$$= \eta_i \tilde{u}_i - \sum_{k \in V(i)} \eta_k \tilde{u}_k - [-R_i(\tilde{u}_i) + R_i^*] \geqslant 0.$$

In view of Theorem 3.2, $w_i^{(0)} \geqslant 0$ for all $i \in N^a$. This leads to $\bar{u}_i^{(0)} \geqslant \bar{u}_i^{(1)}$. A similar argument using relations (4.8) and (4.2) gives $\underline{u}_i^{(0)} \leqslant \underline{u}_i^{(1)}$. Let $w_i^{(1)} = \bar{u}_i^{(1)} - \underline{u}_i^{(1)}$. By (4.7) and (4.8), we have

$$(\eta_i + \gamma_i) w_i^{(1)} = \sum_{k \in V(i)} \eta_k (\bar{u}_k^{(0)} - \underline{u}_k^{(0)}) + \gamma_i (\bar{u}_i^{(0)} - \underline{u}_i^{(0)}) - [R_i(\bar{u}_i^{(0)}) - R_i(\underline{u}_i^{(0)})].$$

It then follows from the relation $\bar{u}_i^{(0)} \geqslant \underline{u}_i^{(0)}$, the nonnegativity of η_k and (4.6) that

$$(\eta_i + \gamma_i) w_i^{(1)} \geqslant 0,$$

which again leads to $w_i^{(1)} \geqslant 0$ and hence $\underline{u}_i^{(0)} \leqslant \underline{u}_i^{(1)} \leqslant \bar{u}_i^{(1)} \leqslant \bar{u}_i^{(0)}$ for all $i \in N^a$. Assume, by induction, that $\underline{u}_i^{(m-1)} \leqslant \underline{u}_i^{(m)} \leqslant \bar{u}_i^{(m)} \leqslant \bar{u}_i^{(m-1)}$ for some $m > 1$. By (4.7), $w_i^{(m)} = \bar{u}_i^{(m)} - \bar{u}_i^{(m+1)}$ satisfies

$$(\eta_i + \gamma_i) w_i^{(m)} = \sum_{k \in V(i)} \eta_k (\bar{u}_k^{(m-1)} - \bar{u}_k^{(m)}) + \gamma_i (\bar{u}_i^{(m-1)} - \bar{u}_k^{(m)}) - [R_i(\bar{u}_i^{(m-1)}) - R_i(\bar{u}_k^{(m)})].$$

It follows again from (4.6) and Theorem 3.2 that

$$(\eta_i + \gamma_i) w_i^{(m)} \geqslant 0.$$

This yields $w_i^{(m)} \geqslant 0$ which shows that $\bar{u}_i^{(m)} \geqslant \bar{u}_i^{(m+1)}$. A similar argument gives $\underline{u}_i^{(m)} \leqslant \underline{u}_i^{(m+1)}$ and $\bar{u}_i^{(m+1)} \geqslant \underline{u}_i^{(m+1)}$. The monotone property (4.9) thus follows by induction.

To show that $\bar{U}^{(m)}$ and $\underline{U}^{(m)}$ are upper and lower solutions for each m, we observe from (4.7) that

$$\eta_i \bar{u}_i^{(m)} = \sum_{k \in V(i)} \eta_k \bar{u}_k^{(m-1)} + \gamma_i (\bar{u}_i^{(m-1)} - \bar{u}_i^{(m)}) - R_i(\bar{u}_i^{(m-1)}) + R_i^*.$$

By (4.6), (4.9) and Theorem 3.2, we have

$$\eta_i \bar{u}_i^{(m)} \geqslant \sum_{k \in V(i)} \eta_k \bar{u}_k^{(m)} - R_i(\bar{u}_i^{(m)}) + R_i^*.$$

This shows that $\bar{U}^{(m)}$ is an upper solution. The proof for the lower solution $\underline{U}^{(m)}$ is similar. □

Based on the monotone property of Lemma 4.1, we have the following monotone convergence result.

Theorem 4.1. *Let $\tilde{U}, \hat{U}$ be a pair of ordered upper and lower solutions of* (3.6). *Then the sequences $\{\bar{U}^{(m)}\}$ and $\{\underline{U}^{(m)}\}$ generated by solving* (4.7) *and* (4.8) *with $\bar{U}^{(0)} = \tilde{U}$ and $\underline{U}^{(0)} = \hat{U}$ converge monotonically to the solutions $\bar{U}$ and $\underline{U}$ of* (3.6), *respectively. Moreover*

$$\hat{U} \leqslant \underline{U}^{(m)} \leqslant \underline{U}^{(m+1)} \leqslant \underline{U} \leqslant \bar{U} \leqslant \bar{U}^{(m+1)} \leqslant \bar{U}^{(m)} \leqslant \tilde{U}, \quad m = 1, 2, \ldots \tag{4.10}$$

and if U^* is any solution of (3.6) *in $\langle \hat{U}, \tilde{U} \rangle$ then $\underline{U} \leqslant U^* \leqslant \bar{U}$.*

Proof. By Lemma 4.1, the limits

$$\lim \bar{U}^{(m)} = \bar{U} \quad \text{and} \quad \lim \underline{U}^{(m)} = \underline{U} \quad \text{as } m \to \infty$$

exist and satisfy relation (4.10). Letting $m \to \infty$ in (4.7) and (4.8) shows that $\bar{U}$ and $\underline{U}$ are solutions of (4.5). The equivalence between (3.6) and (4.5) ensures that $\bar{U}$ and $\underline{U}$ are solutions of (3.6). Now if $U^* \in \langle \hat{U}, \tilde{U} \rangle$ is a solution of (3.6) then $\tilde{U}$ and U^* are ordered upper and lower solutions. Using $\bar{U}^{(0)} = \tilde{U}$ and $\underline{U}^{(0)} = U^*$, Lemma 4.1 implies that $\bar{U}^{(m)} \geqslant U^*$ for every m. Letting $m \to \infty$ gives $\bar{U} \geqslant U^*$. A similar argument using U^* and $\hat{U}$ as ordered upper and lower solutions yields $U^* \geqslant \underline{U}$. This proves the theorem. □

In view of the relation $\bar{U} \geqslant U^* \geqslant \underline{U}$ for any solution U^* in $\langle \hat{U}, \tilde{U} \rangle$, $\bar{U}$ and $\underline{U}$ are often called maximal and minimal solutions in $\langle \hat{U}, \tilde{U} \rangle$, respectively. In general, these two solutions are not necessarily the same. Nevertheless, they are the same for model problems (2.23)–(2.25).

Theorem 4.2. *If the conditions in Theorem* 4.1 *hold, then* $\bar{U} = \underline{U}$ *is the unique solution of* (3.6).

Proof. Let $W = \bar{U} - \underline{U}$. Then $W \geqslant 0$ and by (3.6)

$$\mathscr{A}W = -R(\bar{U}) + R(\underline{U}) \leqslant \sigma(\bar{U} - \underline{U}) = \sigma W,$$

where

$$\sigma \equiv \max \left\{ -\frac{\partial R_i(w_i)}{\partial u_i};\ w_i \in \langle \hat{u}_i, \tilde{u}_i \rangle,\ i \in N^{\mathrm{a}} \right\}. \tag{4.11}$$

Equivalently, we have $(\mathscr{A} - \sigma I)W \leqslant 0$ where I is the identity matrix. Hence, the inverse $(\mathscr{A} - \sigma I)^{-1}$ exists and is nonnegative since $\sigma \leqslant 0$ and $\mathscr{A}$ is symmetric due to (3.9) and (3.10). This implies that $W \leqslant 0$ which leads to $W = 0$, i.e., in $\langle \hat{U}, \tilde{U} \rangle$, $\bar{U} = \underline{U} = U^*$ is the unique solution of (3.6). □

Finally, we give some comparison results for the monotone sequences obtained by the three basic iterative methods of Picard, Jacobi, and Gauss–Seidel. The iterative methods are based on system (3.6) with $\mathscr{A}$ written in the split form $\mathscr{A} = \mathscr{D} - \mathscr{L} - \mathscr{U}$, where $\mathscr{D}$, $\mathscr{L}$ and $\mathscr{U}$ are the diagonal, lower-off diagonal and upper-off diagonal matrices of $\mathscr{A}$, respectively. By Theorem 3.2, the elements of $\mathscr{D}$ are positive and those of $\mathscr{L}$ and $\mathscr{U}$ are nonnegative.

Using the split form of $\mathscr{A}$ the three iterative schemes are given as follows:

(a) Picard method:

$$(\mathscr{A} + \Lambda)U^{(m+1)} = \Lambda U^{(m)} - R(U^{(m)}) + R^*. \tag{4.12}$$

(b) Gauss–Seidel method:

$$(\mathscr{D} - \mathscr{L} + \Lambda)U^{(m+1)} = \Lambda U^{(m)} + \mathscr{U}U^{(m)} - R(U^{(m)}) + R^*. \tag{4.13}$$

(c) Jacobi method:

$$(\mathscr{D} + \Lambda)U^{(m+1)} = \Lambda U^{(m)} + (\mathscr{L} + \mathscr{U})U^{(m)} - R(U^{(m)}) + R^*. \tag{4.14}$$

It is clear that, following Theorem 3.2, the inverses of the matrices $(\mathscr{A} + \Lambda)$, $(\mathscr{D} - \mathscr{L} + \Lambda)$ and $(\mathscr{D} + \Lambda)$ all exist and are nonnegative.

Using either $\tilde{U}$ or $\hat{U}$ as initial iterates, we can construct a sequence from each one of the iterative schemes (4.12)–(4.14). Denote the respective sequences by $\{\bar{U}_{\mathrm{P}}^{(m)}\}$, $\{\bar{U}_{\mathrm{G}}^{(m)}\}$ and $\{\bar{U}_{\mathrm{J}}^{(m)}\}$ with $\bar{U}^{(0)}=\tilde{U}$ and by $\{\underline{U}_{\mathrm{P}}^{(m)}\}$, $\{\underline{U}_{\mathrm{G}}^{(m)}\}$ and $\{\underline{U}_{\mathrm{J}}^{(m)}\}$ with $\underline{U}^{(0)}=\hat{U}$, and refer to them again as maximal and minimal sequences, respectively. The convergence of these sequences and the uniqueness of the limiting solutions can be shown in a similar way as in Theorems 4.1 and 4.2. Furthermore, these three minimal or maximal sequences exhibit an ordered behavior as given in the following theorem for which a proof can be found in [30].

Theorem 4.3. *Let $\tilde{U},\hat{U}$ be a pair of ordered upper and lower solutions of* (3.6). *Then the three pairs of sequences* $(\{\bar{U}_{\mathrm{P}}^{(m)}\},\{\underline{U}_{\mathrm{P}}^{(m)}\})$, $(\{\bar{U}_{\mathrm{G}}^{(m)}\},\{\underline{U}_{\mathrm{G}}^{(m)}\})$ *and* $(\{\bar{U}_{\mathrm{J}}^{(m)}\},\{\underline{U}_{\mathrm{J}}^{(m)}\})$ *generated by solving* (4.12), (4.13), *and* (4.14), *respectively, with $\bar{U}^{(0)}=\tilde{U}$ and $\underline{U}^{(0)}=\hat{U}$ converge monotonically to the solution of* (3.6). *Moreover*

$$\bar{U}_{\mathrm{P}}^{(m)} \leqslant \bar{U}_{\mathrm{G}}^{(m)} \leqslant \bar{U}_{J}^{(m)} \quad \text{and} \quad \underline{U}_{\mathrm{P}}^{(m)} \geqslant \underline{U}_{\mathrm{G}}^{(m)} \geqslant \underline{U}_{J}^{(m)} \quad \text{for } m=1,2,\ldots. \tag{4.15}$$

5. Implementation algorithms

We briefly summarize our implementation procedures for the proposed methods into two algorithms. The first algorithm is an adaptive process based on the general framework of the weak residual error estimation developed in [23] and on the object-oriented programming (OOP) prototype proposed in [24]. The data structure of the prototype is designed to combine 1-irregular mesh refinement scheme, various types of PDEs, and various numerical methods in an integrated computational platform.

An adaptive algorithm.

Step 1: *Initial mesh*: Generate a coarse and structured mesh for which the number of nodes can be chosen as small as possible.

Step 2: *Junction refinement*: Since the initial mesh is usually very coarse, a preprocessing refinement to capture irregularities caused by the junction layers of the doping profile and by the inversion layer due to the applied voltages proves to be an essential step for more effective refinement and faster convergence in the subsequent computations. For the junction layers, Poisson's problem in (2.23) with low biasing conditions is pre-solved few times with the same procedures of error estimation and refinement as that of Steps 5 and 6 below.

Step 3: *Interface refinement*: For the inversion layer, several refinement iterations are performed specifically along the interface boundary. The refinement procedure is the same as that of Step 6. We now have a nonuniform mesh with a better resolution in the vicinity of the interface and the junction.

Step 4: *Solution*: On the current mesh, each one of nonlinear problems (2.23)–(2.25) is then approximated by FEM. The resulting nonlinear algebraic equations are then solved by some monotone iterative method. Note that this solution procedure consists of an outer loop associated with Gummel's iteration solving (2.23)–(2.25) consecutively and an inner loop associated with the monotone iteration. Moreover, the assembly of global stiffness and mass matrices of the resulting approximation is not required, that is, the solution of the discretized nonlinear systems is performed

on a node-by-node (regular node) basis. This step is described in more details in the following monotone-Gummel algorithm.

Step 5: *Error estimation*: All adaptive methods require more or less a posteriori information about the computed solution for optimizing overall computational effort in the sense that the methods deliver a given level of accuracy with a minimum of degrees of freedom. In essence, the a posteriori error estimation can be regarded as the driving force of adaptive mechanism. Several methods for error estimation in semiconductor simulations have been proposed and incorporated with adaptive grid refinement. PISCES [31], for example, limits the variation of the quasi-Fermi potential over one element; the element is refined if this limit is exceeded. Bank and Weiser [4] proposed a method to estimate the error based on the computation of the norm of the local residual of the elliptic equation and the jump in the normal derivative of the computed solution at inter-element boundaries. An estimation based on computation of the error in the current continuity equation is developed in [9]. We found that the variation of the approximate potential or electron concentration is much easier and inexpensive to obtain and yet sufficiently effective (since the variations differ drastically from low to high concentration) for adaptive mesh generation. Variations are calculated with respect to every two nodes in each element, from which the largest one is chosen as an error indicator for the element. Error indicators are obtained in an element-by-element manner according to the hierarchical tree structure of the elements in the OOP database (see [24] for more details). A set of criteria on such as global error (maximum) norms of approximated solutions in inner iteration and outer iteration will also be verified (see the next section). If none of the stopping criteria is satisfied, the adaptive process will continue to Step 6, otherwise it will go to Step 7 for postprocessing the computed solutions.

Step 6: *Refinement*: For each element, if its error indicator is larger than a preset error tolerance, it is divided into four subelements according to the rules of the 1-irregular mesh refinement scheme as given in [24]. We then move on to Step 4.

Step 7: *Postprocessing*: All computed solutions are then postprocessed for further analysis of the physical phenomena.

The second algorithm illustrates how the monotone iterative methods are embedded into the Gummel decoupling algorithm. Here we use the notation g as Gummel's (outer) iteration index and m as the monotone (inner) iteration index.

A monotone-Gummel algorithm

Step 0: Set $g := 0$.

Step 1: Solve the Poisson and Laplace equations in (2.23).

Step 1.1: Set $m := 0$ and set the initial guess

$$\phi_j^{(m)} = \begin{cases} \tilde{\phi}_j \text{ or } \hat{\phi}_j & \text{if } g = 0, \\ \phi_j^{(g)} & \text{otherwise} \end{cases}$$

for all $(x_j, y_j) \in \bar{\Omega}^h$, where $\tilde{\phi}_j$ and $\hat{\phi}_j$ are constant values that can be easily verified to be an upper and lower solution of ϕ, respectively.

Step 1.2: If $g = 0$, set $u^{(g)}$ and $v^{(g)}$ by the charge neutrality condition.

Step 1.3: Compute $\phi_j^{(m+1)}$ (an unknown real value) by solving the discrete potential system

$$\xi_j \phi_j^{(m+1)} + \gamma_j(\phi)\phi_j^{(m+1)} = \sum_{k \in V(j)} \xi_k \phi_k^{(m)} - F_j(\phi_j^{(m)}, u_j^{(g)}, v_j^{(g)}) + \gamma_j(\phi)\phi_j^{(m)} \quad \forall (x_j, y_j) \in \Omega^h,$$

$$\phi_j^{(m+1)} = V_{\mathrm{O}} + V_{\mathrm{b}} \quad \forall (x_j, y_j) \in \partial\Omega_{\mathrm{D}}^h,$$

$$\frac{\partial \phi_j^{(m+1)}}{\partial v} = 0 \quad \forall (x_j, y_j) \in \partial\Omega_{\mathrm{N}}^h,$$

$$\xi_j \phi_j^{(m+1)} = \sum_{k \in V(j)} \xi_k \phi_k^{(m)} \quad \forall (x_j, y_j) \in \Omega_{\mathrm{O}}^h,$$

$$\phi_j^{(m+1)} = V_{\mathrm{G}} + V_{\mathrm{b}} \quad \forall (x_j, y_j) \in \Gamma_{\mathrm{IJ}},$$

$$\frac{\partial \phi_j^{(m+1)}}{\partial v} = 0 \quad \forall (x_j, y_j) \in \Gamma_{\mathrm{CI}} \cup \Gamma_{\mathrm{DJ}},$$

$$\varepsilon_{\mathrm{s}} \partial_y \phi_-^{(m+1)} = \varepsilon_d \partial_y \phi_+^{(m+1)} \quad \forall (x_j, y_j) \in \partial\Omega_R^h \tag{5.1}$$

with

$$\gamma_j(\phi) = \max\left\{ \frac{qn_i}{\varepsilon_{\mathrm{s}} V_T} \left(u_j^{(g)} \exp\left(\frac{\phi_j}{V_T}\right) + v_j^{(g)} \exp\left(\frac{-\phi_j}{V_T}\right)\right);\ \hat{\phi}_j \leqslant \phi_j \leqslant \tilde{\phi}_j \right\}.$$

Step 1.4: Set $\phi_j^{(m)} := \phi_j^{(m+1)}\ \forall j$ and $m := m + 1$. Go to Step 1.3 until the stopping criteria of the inner iteration are satisfied.

Step 1.5: Set $\phi_j^{(g+1)} := \phi_j^{(m+1)}\ \forall j$.

Step 2: Solve the electron continuity equation in (2.24).

Step 2.1: Set $m := 0$ and set the initial guess

$$u_j^{(m)} = \begin{cases} \tilde{u}_j \text{ or } \hat{u}_j & \text{if } g = 0, \\ u_j^{(g)} & \text{otherwise} \end{cases}$$

for all $(x_j, y_j) \in \bar{\Omega}^h$, where $\tilde{u}_j$ and $\hat{u}_j$ are constant values for all $(x_j, y_j) \in \bar{\Omega}^h$ that can be easily verified to be an upper and lower solution of u, respectively.

Step 2.2: Compute $u_j^{(m+1)}$ (an unknown real value) by solving the discrete electron system

$$\eta_j u_j^{(m+1)} + \gamma_j(u) u_j^{(m+1)} = \sum_{k \in V(j)} \eta_k u_k^{(m)} - R_j(\phi_j^{(g+1)}, u_j^{(m)}, v_j^{(g)}) + \gamma_j(u) u_j^{(m)} \quad \forall (x_j, y_j) \in \Omega^h,$$

$$u_j^{(m+1)} = \exp\left(\frac{-V_{\mathrm{O}}}{V_T}\right) \quad \forall (x_j, y_j) \in \partial\Omega_{\mathrm{D}}^h,$$

$$\frac{\partial u_j^{(m+1)}}{\partial v} = 0 \quad \forall (x_j, y_j) \in \partial\Omega_{\mathrm{N}}^h \cup \partial\Omega_R^h \tag{5.2}$$

with

$$\gamma_j(u)=\max\left\{\frac{\tau_n^0(n_i v_j^{(g)}\exp(-\phi_j^{(g+1)}/V_T)+p_T)n_i^2 v_j^{(g)}+\tau_p^0 n_T n_i^2 v_j^{(g)}+n_i^3\tau_p^0\exp(\phi_j^{(g+1)}/V_T)}{(\tau_n^0(n_i v_j^{(g)}\exp(-\phi_j^{(g+1)}/V_T)+p_T)+\tau_p^0(n_i u_j\exp(\phi_j^{(g+1)}/V_T)+n_T))^2}\right\}.$$

Step 2.3: Set $u_j^{(m)} := u_j^{(m+1)}\ \forall j$ and $m := m+1$. Go to Step 2.2 until the stopping criteria of the inner iteration are satisfied.

Step 2.4: Set $u_j^{(g+1)} := u_j^{(m+1)}\ \forall j$.

Step 3: The hole continuity equation in (2.25) is solved analogously as done in Step 2.

Step 4: Set $g := g+1$ and go to Step 1 until the stopping criteria of the outer iteration are satisfied.

Note that the solution procedure in the above algorithm is completely similar to that of the standard Gummel algorithm except that the initial iterates for the Gummel iterations in Steps 1.1 and 2.1 are chosen by the lower and upper solutions of the corresponding semilinear PDEs. Moreover, it can be seen from (5.1) and (5.2) that the unknown scalars are calculated in a node-by-node manner without explicitly computing the Jacobian matrix as required by Newton's method. As shown in Theorem 4.1, we have rather large freedom to choose the values of the initial iterates. For elliptic PDEs, the upper or lower solution is readily determined by the boundary conditions. This in turn suggests that the initial guesses in the above algorithm can be deduced from applied voltages and built-in potentials. For example, we can choose $\tilde{\phi}_j = V_D + V_n$ and $\hat{\phi}_j = V_B + V_p$, $\tilde{u}_j = \exp(-V_B/V_T)$, $\hat{u}_j = \exp(-V_D/V_T)$, $\tilde{v}_j = \exp(V_D/V_T)$ and $\hat{v}_j = \exp(V_B/V_T)$, where V_n is the built-in potential in n-region and V_p is the built-in potential in p-region.

6. Numerical example

We consider an n-MOSFET device with a channel length of 0.34 μm and with the gate oxide thickness of 7 nm. The doping profile for the test device is shown in Fig. 2, which is an elliptical Gaussian distribution with the concentration 10^{20} cm^{-3} in the source and drain regions and 10^{16} cm^{-3} in the p-substrate region. The shallow implantation is needed to obtain a 'normal-off' device with positive threshold voltage and the deep implantation is necessary to avoid punchthrough. The junction depth is 0.2 μm and the lateral diffusion under the gate is 0.08 μm.

With the drain voltage $V_D = 1.0$ V and the gate voltage $V_G = 1.0$ V, Fig. 3 illustrates a typical adaptive final mesh which was generated from an initial mesh of 16 elements. The inversion and junction layers are effectively captured by the adaptive process. Corresponding to the final mesh, these computed solutions were obtained by the Gauss–Seidel monotone iteration with, for example, an initial guess of the upper solution shown in Fig. 4 for the electron continuity equation. The final results of the potential and electron concentration are shown in Figs. 5 and 6, respectively.

In order to observe more clearly the monotone convergence behavior, we present some numerical results of Jacobi and Gauss–Seidel monotone iterations for the electron continuity equation. Since $\exp(-V_D/V_T) \simeq 1.7e-17$, we have chosen $\tilde{u} = 1$ and $\hat{u} = 1.0e-18$ as an ordered pair of upper and lower solutions for the iterations. The approximate maximal and minimal solutions $\bar{u}_i$ and $\underline{u}_i$ for some coordinates (x_i, y_i) are given in Tables 1 and 2 for Jacobi and Gauss–Seidel iterations,

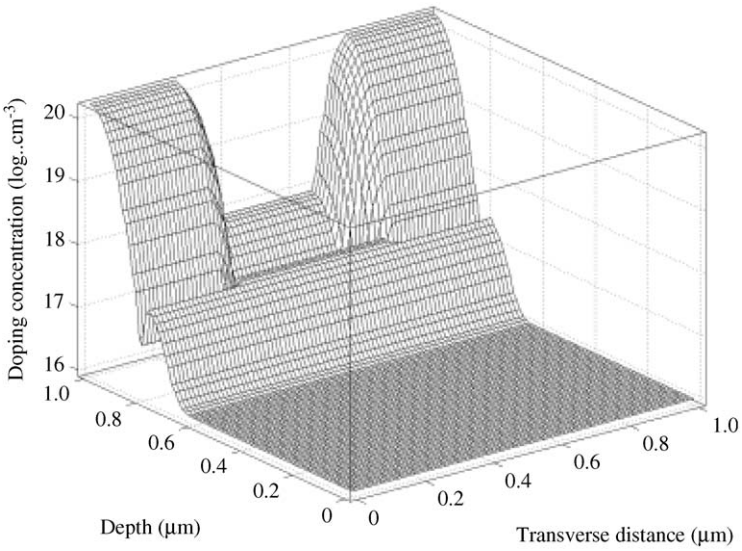

Fig. 2. Doping concentration.

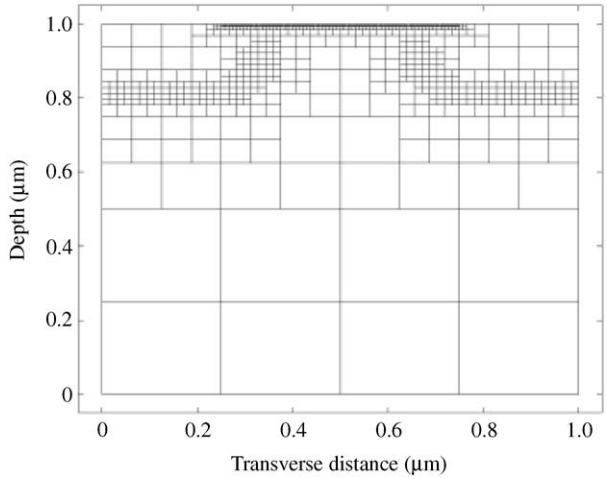

Fig. 3. The final adaptive mesh.

respectively. The last row in these tables gives the number of iterations for the maximal and minimal sequences $\{\bar{u}^{(m)}\}$, $\{\underline{u}^{(m)}\}$. It is clearly seen from these tables that $\bar{u}_i$ is always larger than $\underline{u}_i$ at every mesh point i. Moreover, the number of iterations for the Gauss–Seidel iteration is smaller than that of the Jacobi iteration. The stopping criterion for these iterations is $\|u^{(m)} - u^{(m-1)}\| \leqslant 0.001 V_T$ where $\|\cdot\|$ is the maximum error norm. Note that, for the hole current continuity equation, the initial values are determined by $\exp(V_D/V_T)$. Numerical results for this case are similar to those of Tables 1 and 2.

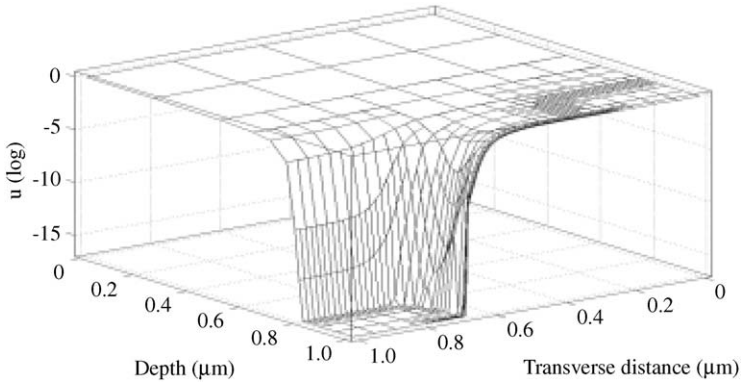

Fig. 4. The upper solution for the Gauss–Seidel iteration.

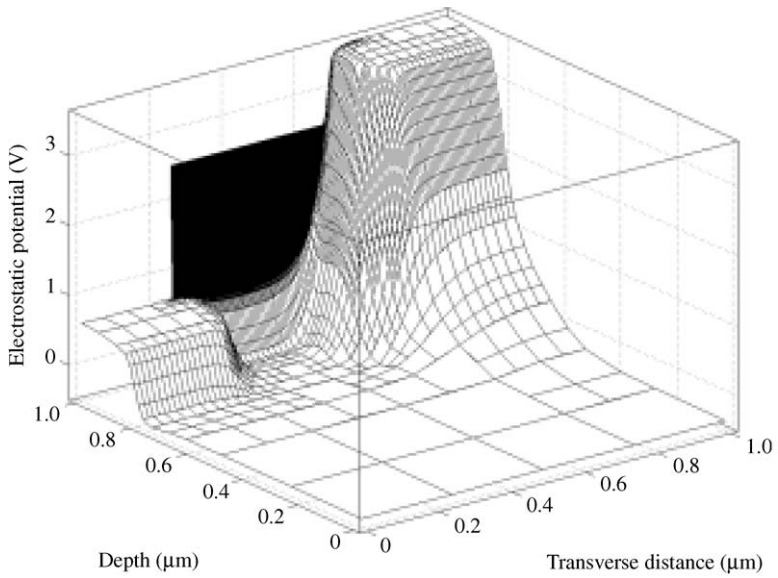

Fig. 5. Electrostatic potential.

We make a concluding remark on some numerical aspects in connection with the model formulation. Obviously, the dynamic range of the Slotboom values of u and v in (2.10) and (2.11) is enormously large in computations. For this, various scaling strategies have been proposed in the literature to avoid catastrophic roundoff effect [3]. The worst case of the numerics for the variables u and v that we have experienced during the course of the development of our code is about of order 10^{100} on our computing systems (Unix on DEC workstations and Linux on Pentium III) with the machine number of order 10^{300}. The ranges of applied voltages that have been tested with our code are -10 V (the reverse bias) to 10 V (the forward bias) for a p–n diode, 0–5 V (the drain bias) and 0–4 V (the gate bias) for a MOSFET.

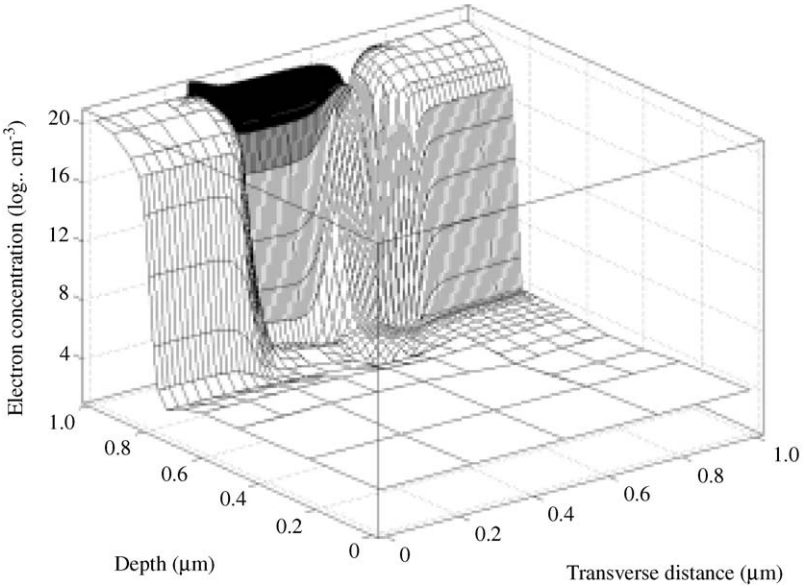

Fig. 6. Electron concentration.

Table 1
Upper and lower solutions (in log scale) at nodes (x_i, y_i) by the Jacobi monotone iteration (No.: Number of iterations)

	$y_i \setminus x_i$	0.05	0.5	0.7	0.75	0.8	0.95
$\bar{u}_i$	0.95	−4.78E − 10	−0.2423	−16.790	−16.794	−16.797	−16.799
$\underline{u}_i$		−2.07E − 9	−0.4330	−16.799	−16.799	−16.799	−16.799
$\bar{u}_i$	0.85	−1.33E − 9	−0.281	−16.253	−16.795	−16.796	−16.798
$\underline{u}_i$		−5.80E − 9	−0.310	−16.278	−16.799	−16.799	−16.799
$\bar{u}_i$	0.8	−2.17E − 6	−0.247	−6.168	−11.796	−13.217	−13.746
$\underline{u}_i$		−2.18E − 6	−0.259	−6.169	−11.797	−13.218	−13.746
$\bar{u}_i$	0.75	−1.38E − 2	−0.189	−1.040	−1.594	−1.9250	−2.2226
$\underline{u}_i$		−1.39E − 2	−0.193	−1.041	−1.595	−1.9254	−2.2229
$\bar{u}_i$	0.5	−6.69E − 2	−0.1133	−0.1782	−0.1945	−0.1945	−0.1945
$\underline{u}_i$		−6.72E − 2	−0.1136	−0.1785	−0.1947	−0.1947	−0.1947
$\bar{u}_i$	0.05	−8.16E − 3	−0.0111	−0.0139	−0.0146	−0.0146	−0.0146
$\underline{u}_i$		−8.19E − 3	−0.0112	−0.0139	−0.0146	−0.0146	−0.0146
No.		1619 ($\bar{u}_i$)	1750 ($\underline{u}_i$)				

To quantitatively discuss the issue of matrix conditioning [15] associated with the implementation of these Slotboom variables, we present the conditioning numbers of the stiffness matrices of (5.1) and (5.2) in Table 3 for several bias conditions.

The second row from the bottom of the table clearly shows that matrix conditioning deteriorates dramatically for large biases due to the term $\exp(\phi/V_T)$ in (3.9). To improve the conditioning, we

Table 2
Upper and lower solutions by the Gauss–Seidel iteration

	$y_i \setminus x_i$	0.05	0.5	0.7	0.75	0.8	0.95
$\bar{u}_i$	0.95	−6.46E − 11	−0.1736	−16.794	−16.796	−16.798	−16.799
$\underline{u}_i$		−2.00E − 9	−0.4274	−16.799	−16.799	−16.799	−16.799
$\bar{u}_i$	0.85	−1.80E − 10	−0.2681	−16.245	−16.797	−16.797	−16.799
$\underline{u}_i$		−5.61E − 9	−0.3099	−16.277	−16.799	−16.799	−16.799
$\bar{u}_i$	0.8	−2.17E − 6	−0.241	−6.167	−11.796	−13.217	−13.746
$\underline{u}_i$		−2.18E − 6	−0.259	−6.169	−11.797	−13.218	−13.746
$\bar{u}_i$	0.75	−1.38E − 2	−0.188	−1.040	−1.594	−1.9249	−2.2224
$\underline{u}_i$		−1.39E − 2	−0.193	−1.041	−1.595	−1.9254	−2.2229
$\bar{u}_i$	0.5	−6.68E − 2	−0.1131	−0.1781	−0.1943	−0.1943	−0.1943
$\underline{u}_i$		−6.71E − 2	−0.1136	−0.1785	−0.1947	−0.1947	−0.1947
$\bar{u}_i$	0.05	−8.15E − 3	−0.0111	−0.0139	−0.0146	−0.0146	−0.0146
$\underline{u}_i$		−8.19E − 3	−0.0112	−0.0139	−0.0146	−0.0146	−0.0146
No.		826 ($\bar{u}_i$)	1627 ($\underline{u}_i$)				

Table 3
The matrix conditioning associated with the Slotboom variables ($V_S = 0$ and $V_B = 0$)

Bias conditions	$V_D = 0$ $V_G = 0$	$V_D = 1$ $V_G = 1$	$V_D = 3$ $V_G = 0$	$V_D = 3$ $V_G = 2$	$V_D = 5$ $V_G = 3$
Degrees of freedom	749	749	3452	3452	4393
ϕ Matrix in (5.1)	2.346E+2	2.346E+2	2.443E+2	2.443E+2	8.646E+2
u Matrix in (5.2)	3.487E+17	1.587E+39	4.419E+187	7.274E+121	1.396E+188
u Matrix in (6.1)	1.515E+4	5.062E+4	1.133E+5	1.106E+5	1.706E+5

can divide (5.2) by this term and n_i to obtain

$$\hat{\eta}_j u_j^{(m+1)} + \hat{\gamma}_j(u) u_j^{(m+1)} = \sum_{k \in V(j)} \hat{\eta}_k u_k^{(m)} - R(\phi_j^{(g+1)}, u_j^{(m)}, v_j^{(g)}) \Bigg/ \left[n_i \exp\left(\frac{\phi_j^{(g+1)}}{V_T} \right) \right] + \hat{\gamma}_j(u) u_j^{(m)}, \tag{6.1}$$

where $\hat{\eta}_j = \exp(-\phi_j^{(g+1)}/V_T)\eta_j/n_i$, $\hat{\eta}_k = \exp(-\phi_j^{(g+1)}/V_T)\eta_k/n_i$ and $\hat{\gamma}_j = \exp(-\phi_j^{(g+1)}/V_T)\gamma_j/n_i$. Before solving (6.1), the division is performed node-by-node for all the known terms in the equation. The conditioning is indeed improved as shown in the last row of the table. Note that the stiffness matrix is an M-matrix and Theorem 4.1 still holds with this scaling technique. Another way to improve the conditioning is to perform (at the discrete level) the change of the variables $n = n_i u e^{\phi/V_T}$ and $p = n_i v e^{-\phi/V_T}$ back to their primitive forms as suggested in [7]. The resulting stiffness matrix

with

$$\hat{\eta}_k = \xi_k D_n|_{(k,j)} B\left(\frac{-\phi_j + \phi_k}{V_T}\right), \tag{6.2}$$

$$\hat{\eta}_j = \sum_{k\in V(j)}^{(m)} \xi_k D_n|_{(k,j)} B\left(\frac{\phi_j - \phi_k}{V_T}\right) \tag{6.3}$$

is then no longer an M-matrix. For example, for some node i, if $B((\phi_j - \phi_k)/V_T) = 1$ for $k \neq i$ and $\phi_j > \phi_i$ then $B((\phi_j - \phi_i)/V_T) < B((-\phi_j + \phi_i)/V_T)$ and $\hat{\eta}_j < \sum_{k\in V(j)}^{(m)} \hat{\eta}_k$. Nevertheless, the mixed or hybrid methods proposed in [7] can be used to recover the M-matrix property and furthermore to have the current conservation property. However, the implementation of these methods is more complicated than that of (6.1) since the discrete system is enlarged by these methods and the matrix reduction by means of static condensation requires an element-wise inversion of the block-diagonal matrix associated with the auxiliary variable. Moreover, a suitable numerical integration formula for the local and global matrices and for the right-hand side vector is required (see [7] for more details). The monotone parameters $\gamma_j(u)$ in (5.2) will also be more involved with these methods.

Acknowledgements

The authors would like to thank the referees for many valuable and helpful comments and suggestions on the manuscript.

References

[1] H. Amann, Supersolution, monotone iteration and stability, J. Differential Equations 21 (1976) 367–377.
[2] O. Axelsson, V.A. Barker, Finite Element Solution of Boundary Value Problems, Academic Press, New York, 1984.
[3] R.E. Bank, D.J. Rose, W. Fichtner, Numerical methods for semiconductor device simulation, IEEE Trans. Electron Dev. ED-30 (1983) 1031–1041.
[4] R.E. Bank, A. Weiser, Some a posteriori estimators for elliptic partial differential equations, Math. Comp. 44 (1985) 283–301.
[5] M. Berger, Nonlinear mathematical phenomena associated with semiconductor devices, AMS Lectures Appl. Math. 25 (1990) 99–106.
[6] C.A. Brebbia, S. Walker, Boundary Element Techniques in Engineering, Newnes-Butterworths, London, 1980.
[7] F. Brezzi, L.D. Marini, P. Pietra, Two-dimensional exponential fitting and applications to drift-diffusion models, SIAM J. Numer. Anal. 26 (1989) 1342–1355.
[8] J.F. Bürgler, R.E. Bank, W. Fichtner, R.K. Smith, A new discretization scheme for the semiconductor current continuity equation, IEEE Trans. Comput. Aided Des. 85 (1989) 479–489.
[9] J.F. Bürgler, W.M. Coughran Jr., W. Fichtner, An adaptive grid refinement strategy for the drift-diffusion equations, IEEE Trans. Comput.-Aided Des. 10 (1991) 1251–1258.
[10] L. Demkowicz, J.T. Oden, W. Rachowicz, O. Hardy, Toward a universal h–p adaptive finite element strategy. Part 1. Constrained approximation and data structure, Comput. Methods Appl. Mech. Eng. 77 (1989) 79–112.
[11] Y. Deng, G. Chen, W.M. Ni, J. Zhou, Boundary element monotone iteration scheme for semilinear elliptic partial differential equations, Math. Comp. 65 (1996) 943–982.
[12] D. Greenspan, S.V. Parter, Mildly nonlinear elliptic partial differential equations and their numerical solution II, Numer. Math. 7 (1965) 129–146.
[13] H.K. Gummel, A self-consistent iterative scheme for one-dimensional steady-state transistor calculations, IEEE Trans. Electron Dev. ED-11 (1969) 455–465.

[14] S. Heikkila, V. Lakshmikantham, Monotone Iterative Techniques for Discontinuous Nonlinear Differential Equations, Marcel Dekker, New York, 1994.
[15] N.J. Higham, Fortran codes for estimating the one-norm of a real or complex matrix, with applications to condition estimation, ACM Trans. Math. Software 14 (1988) 381–396.
[16] K. Ishihara, Monotone explicit iterations of the finite element approximations for the nonlinear boundary value problem, Numer. Math. 43 (1984) 419–437.
[17] J.W. Jerome, Consistency of semiconductor modeling: an existence/stability analysis for the stationary van Roosbroeck system, SIAM J. Appl. Math. 45 (1985) 565–590.
[18] J.W. Jerome, Analysis of Charge Transport, Springer, Berlin, Heidelberg, 1996.
[19] R. Kannan, M.B. Ray, Monotone iterative methods for nonlinear equations involving noninvertible linear part, Numer. Math. 45 (1984) 219–225.
[20] T. Kerkhoven, On the effectiveness of Gummel's method, SIAM J. Sci. Statist. Comput. 9 (1988) 48–60.
[21] J.P. Kreskovsky, A hybrid central difference scheme for solid state device simulation, IEEE Trans. Electron Dev. ED-345 (1987) 1128–1133.
[22] Y. Li, J.-L. Liu, S.M. Sze, T.-S. Chao, A new parallel adaptive finite volume method for the numerical simulation of semiconductor devices, Comput. Phys. Comm. 142 (2001) 285–298.
[23] J.-L. Liu, On weak residual error estimation, SIAM J. Sci. Comput. (1996) 1249–1268.
[24] J.-L. Liu, I.-J. Lin, M.-Z. Shih, R.-C. Chen, M.-C. Hsieh, Object oriented programming of adaptive finite element and finite volume methods, Appl. Numer. Math. 21 (1996) 439–467.
[25] P.A. Markowich, The Stationary Semiconductor Device Equations, Springer, Vienna, 1986.
[26] P.A. Markowich, C.A. Ringhofer, S. Selberherr, M. Lentini, A singular perturbation approach for the analysis of the fundamental semiconductor equations, IEEE Trans. Electron Dev. ED-30 (1983) 1165–1180.
[27] C.C. McAndrew, K. Singhal, E.L. Heasell, A consistent nonisothermal extension of the Scharfetter–Gummel stable difference approximation, IEEE Trans. Electron Dev. Lett. EDL-6 (1985) 446–447.
[28] J. Ortega, W.C. Rheinboldt, Iterative Solutions of Nonlinear Equations in Several Variables, Academic Press, New York, 1970.
[29] C.V. Pao, Nonlinear Parabolic and Elliptic Equations, Plenum Press, New York, 1992.
[30] C.V. Pao, Block monotone iterative methods for numerical solutions of nonlinear elliptic equations, Numer. Math. 72 (1995) 239–262.
[31] M.R. Pinto, C.S. Rafferty, R.W. Dutton, PISCES II: Poisson and Continuity Equation Solver, Stanford University, Stanford, CA, 1984.
[32] M. Sakahihara, An iterative boundary integral equation method for mildly nonlinear elliptic partial differential equations, in: C.A. Brebbia, G. Maier (Eds.), Boundary Elements IX, Vol. II, Springer, Berlin, Heidelberg, 1985, pp. 13.49–13.58.
[33] D. Sattinger, Topics in Stability and Bifurcation Theory, in: Lecture Notes in Mathematics, Vol. 309, Springer, Berlin, Heidelberg, New York, 1973.
[34] D.L. Scharfetter, H.K. Gummel, Large-signal analysis of a silicon Read diode oscillator, IEEE Trans. Electron Dev. ED-16 (1969) 64–77.
[35] T.I. Seidman, S.C. Choo, Iterative scheme for computer simulation of semiconductor devices, Solid-State Electron. 15 (1972) 1229–1235.
[36] S. Selberherr, Analysis and Simulation of Semiconductor Devices, Springer, New York, 1984.
[37] S. Selberherr, A. Schutz, H.W. Potzl, MINIMOS—a two-dimensional MOS transistor analyzer, IEEE Trans. Electron Dev. ED-278 (1980) 64–77.
[38] M. Sharma, G.F. Carey, Semiconductor device simulation using adaptive refinement and flux upwinding, IEEE Trans. Comput. Aided Des. 8 (1989) 590–598.
[39] J.W. Slotboom, Computer-aided two-dimensional analysis of bipolar transistor, IEEE Trans. Electron Dev. ED-20 (1973) 669–679.
[40] S.M. Sze, Physics of Semiconductor Devices, 2nd Edition, Wiley Interscience, New York, 1981.
[41] G.-L. Tan, X.-L. Yuan, Q.-M. Zhang, W.H. Ku, A.-J. Shey, Two-dimensional semiconductor device analysis based on new finite-element discretization employing the S-G scheme, IEEE Trans. Comput. Aided Des. 8 (1989) 468–478.

[42] T.W. Tang, Extension of the Scharfetter–Gummel algorithm to the energy balance equation, IEEE Trans. Electron Dev. ED-31 (1984) 64–77.
[43] W.V. Van Roosbroeck, Theory of flow of electrons and holes in germanium and other semiconductors, Bell Systems Tech. J. 29 (1950) 560–607.