JOURNAL OF COMPUTATIONAL PHYSICS 120, 117–127 (1995)

A Discretized *k*-Space Method for Charge Transport in Semiconductors

D. W. BAILEY

Department of Electrical and Computer Engineering, University of South Carolina, Columbia, South Carolina 29208

AND

J. M. HIGMAN*

Beckman Institute, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801

Received July 5, 1994; revised November 4, 1994

We describe a method that has procedures similar to the Monte Carlo method for solving the Boltzmann transport equation, but is deterministic, and thus fundamentally different. In this new method, the distribution function is discretized and tracked in phase space. The scattering rates couple initial and final **k** states, eliminating the need for final-state calculations. The advantages this affords are numerous, although it is disk-space and memory intensive.

1. INTRODUCTION

The Monte Carlo method (MCM) [1–3] has gained widespread acceptance for solving the Boltzmann transport equation in semiconductors for research applications. In general, the MCM is a semiclassical method applicable to a wide variety of useful problems, and modifying the program, for example, by adding a scattering mechanisms is comparatively easy. These advantages often make the MCM the "method of choice" for new structure and transport conditions [4, 5].

It is not always, however, the best choice. A common complaint about the MCM is that it is computationally expensive, in terms of both memory usage and CPU time [6, 7]. The time step is often orders of magnitude less than the desired simulation time [8]. Also, the MCM is based on physical statistics, so stochastic noise is present in the results, and more precise results require greater computational resources. There are other limitations, but most of these are more closely attributable to limitations of the Boltzmann transport equation and the independent electron approximation, rather than the MCM [9].

Our intent in this work is to describe a method that retains the advantages of the MCM, yet eliminates the stochastic noise, and that can exploit modern vector and parallel computer architectures to gain a potential speed advantage. Our method, the discretized *k*-space method (DKSM), meets these expectations, but nonetheless introduces new problems. We believe these problems are tractable and will eventually be overcome. At the very least, the DKSM might lead to the development of related methods that mitigate the problems of the MCM. In this paper, we show that the DKSM is a promising future alternative to the MCM.

The objectives of this paper are to (1) show our complete method, (2) show realized and potential advantages of the DKSM over the MCM, (3) candidly discuss the current shortcomings and pitfalls of the DKSM, and (4) demonstrate that the DKSM is feasible by calculating steady-state, homogeneous electron transport in Si.

2. METHOD

2.1. *Background*

There exists a wide, eclectic range of methods for solving the Boltzmann transport equation for charge transport in semiconductors. A short list of solution procedures includes the MCM, drift-diffusion [6, 10], iterative integral [11], direct integration [12], scattering matrix [13, 14], and basis-function expansion [15, 16] methods. One of the methods very similar to the DKSM is briefly described by Fernandes and Santos [17] who first introduce and realize the idea of a scattering matrix connecting states in **k** space. Another similar method was recently presented by Nougier *et al.* [18]. Cellular automata has also been successfully applied to charge transport in semiconductors with numerous advantages over the MCM, including speed due to a pretabulated *k*-dependent scattering matrix [19]. Although not applied to semiconductor charge transport, the Monte Carlo flux method described by Schaeffer and Hui [20] is like the DKSM in that **k** space is discretized. In the category

* Present address: Motorola Corporation, APRDL MD K-10, 3501 Ed Bluestein Boulevard, Austin, Texas 78721.

0021-9991/95 $12.00

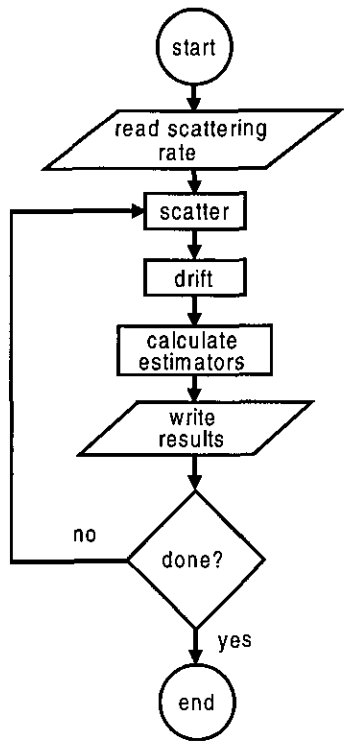

FIG. 1. Coarse flow chart for both MCM and DKSM. Both methods model transport in the Boltzmann transport equation by separate drift and scattering events.

of well known and common methods, however, the DKSM is most similar to the MCM. To clarify our description of the DKSM, we compare it with a typical MCM throughout the paper.

In the MCM, the trajectory of N charge particles are tracked in 7-parameter phase space; i.e., $f(\mathbf{x}_i, \mathbf{k}_i, t)$ is solved where f is the discrete distribution function, equal to 0 or 1, $\mathbf{x}$ is a spatial vector, $\mathbf{k}$ is a crystal momentum vector, and t is time. It is a semiclassical method in that each charge particle is drifted according to quasi-classical mechanics [1] and scattered according to quantum mechanics.

The DKSM is also a semiclassical method, but it is not a particle simulation. As shown in Fig. 1, the DKSM uses the same coarse flowchart as the MCM. The MCM models discrete carriers in (an approximated) k-space continuum, whereas the DKSM uses a discrete k space with a continuous occupation probability. Instead of tracking the trajectories of charge carriers, in the DKSM the distribution function in *cells* of the Brillouin zone is tracked subject to drift and diffusion. Drift and scattering refer to $f(\mathbf{x}_i, \mathbf{k}_i, t)$ in each k-space cell, rather than to simulation electrons. The distribution function is continuous in the range [0,1] and is assumed to be uniform in each cell. During the drift stage, $f(\mathbf{x}_i, \mathbf{k}_i, t)$ in every cell is shifted the same amount according to the field at $\mathbf{x}_i$ and the time step. At the end of each drift stage, the aggregate scattering in and out of every cell, i.e., the change in $f(\mathbf{x}_i, \mathbf{k}_i, t)$, is determined according to a precalculated scattering matrix.

In our calculations, we divide the Brillouin zone into cubic cells, each with edge length Δk. Our primary criterion for the selection of a gridding scheme is programming ease. A cubic cell reflects the crystal symmetry of the most important semiconductors and fills all the k space without overlap. Orthorhombic and tetragonal cells have no general-purpose advantages in a cubic semiconductor, and the interpolation simplicity of tetrahedral cells does not outweigh the added complexity in the DKSM, particularly as it relates to drift. Translational symmetry in the orthogonal axes directions with identical lattice constants make the cubic mesh the easiest to program, verify, and maintain.

2.2. *Initialization*

In a MCM simulation, if starting at equilibrium, simulation electrons are distributed according to the Fermi–Dirac distribution,

$$f(E) = [1 + e^{(E-E_F)/k_BT}]^{-1}, \tag{1}$$

where E_F is the chemical potential, k_B is the Boltzmann constant, and T is the carrier temperature. For clarity we consider only homogeneous transport, and so we neglect $\mathbf{x}$. The chemical potential is determined by the electron density n from the relation

$$n = \int f(E)g(E)\,dE, \tag{2}$$

where the integral is over the conduction band, and $g(E)$ is the density of states. To initialize the electron energy states from Eq. (1), either the *rejection technique* or a numerical form of the *direct technique* [2] can be used. Once the initial energy for a simulation electron is found, $\mathbf{k}$ is selected randomly and uniformly on that isoenergy surface. Thus, in a MCM, both initial energy and initial k states are determined stochastically.

In comparison, in the DKSM the average occupation probability $f(\mathbf{k}_i)$ is initialized as a thermal distribution according to a Fermi–Dirac distribution,

$$f(\mathbf{k}_i) = [1 + e^{(E(\mathbf{k}_i)-E_F))/k_BT}]^{-1}, \tag{3}$$

where here $E(\mathbf{k}_i)$ is the energy at the center of cell i. We assume that the cell is small enough such that $f(\mathbf{k}_i)$ can be assumed as uniform in each cell. The Fermi level in Eq. (4) is determined such that the numerically summed density is equal to the nominal (input) density, n. If Ω_i is the volume of cell i, the electron density is given by

$$n = \frac{1}{4\pi}\sum_i f(\mathbf{k}_i)\Omega_i, \tag{4}$$

where the summation is over all cells in the Brillouin zone.

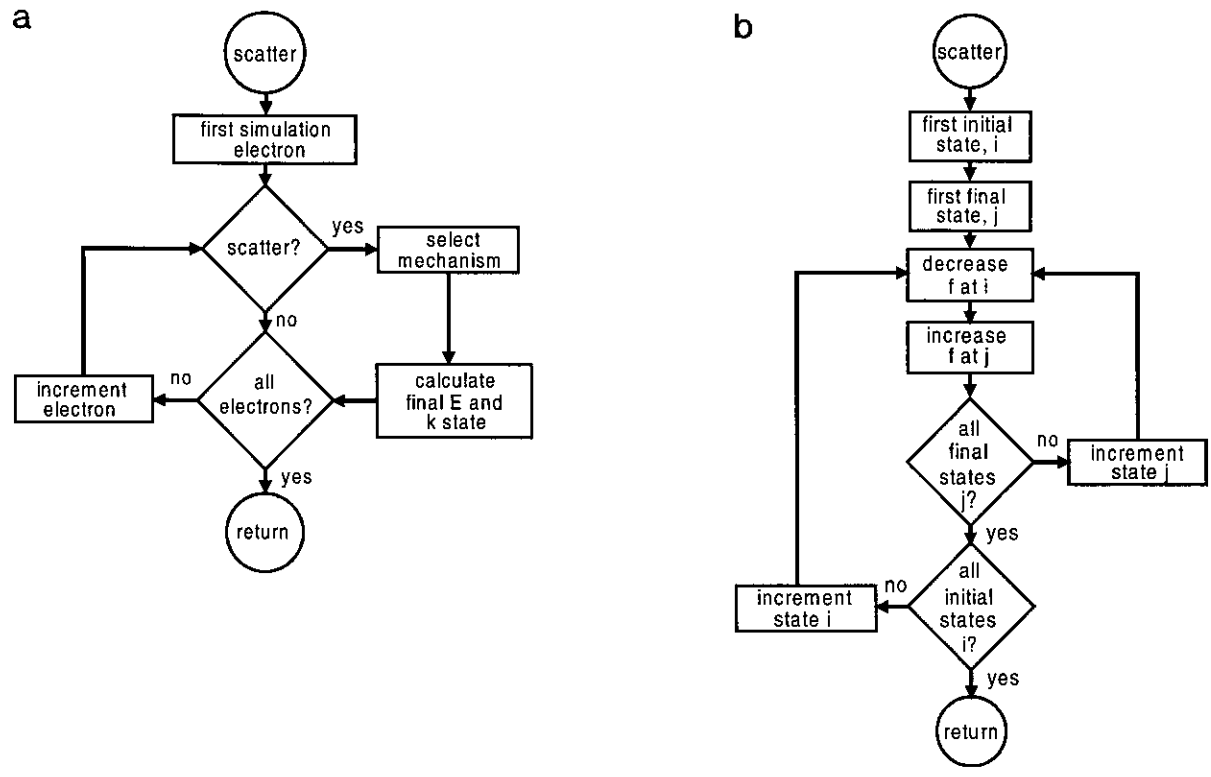

FIG. 2. Flow chart for scattering in (a) the MCM and (b) the DKSM. In the discrete time step MCM, each electron in the simulation is checked for scattering. If scattering (other than self-scattering) occurs, then the electron's final state must be calculated for the chosen scattering mechanism. The check for scattering, the scattering mechanism choice, and the calculation of the final state each involve a weighted random selection. The DKSM, in contrast, is completely deterministic. For each initial state i in the Brillouin zone, there is a loop over final states j as defined by non-zero scattering rates S_{ij}.

Cells on the boundary of the Brillouin zone may be cut by the faces of the zone and, therefore, Ω_i is not a constant, even for a constant grid.

2.3. *Scattering*

The physical approximations behind scattering are the same for the DKSM and the typical MCM. Scattering rates are calculated in the Born approximation and the independent electron approximation [21]. Scattering rates can be calculated prior and independent to the transport calculation, except for carrier–carrier scattering which must be integrated over the distribution function, although near equilibrium this too can be done analytically. The scattering process is treated as an instantaneous event, independent of the field. This *separation of drift and scattering* is the essence of why it is easy to add physical rigor in the MCM. (Since the DKSM is a variant of the MCM, this advantage still holds.) The collision term of the Boltzmann transport equation is a non-local integral and is usually at the root of the difficulty in direct solution methods.

The scattering procedure in the MCM is shown in the flowchart in Fig. 2(a). Scattering rates are normally stored as a function of initial-state energy [22] and initial valley or band. The discrete-time step version of the MCM [23, 24] is described here since it is most similar to the DKSM. In the discrete-time step method, each electron is stochastically checked for scattering after a short constant time step, Δt, according to its scattering rate. If it does not scatter, it takes very little run time. The primary constraint on the time step is that it be much shorter than the inverse scattering rate, $\Delta t \ll \min(\tau)$, where $1/\tau$ is the total scattering rate. After an electron is chosen to scatter, a scattering mechanism is randomly chosen according to the relative scattering rates. The final state is then calculated, which can consume considerable run time, particularly if a q-dependent scattering mechanism is chosen. An advantage to the discrete-time step method is that it permits a straightforward synchronization of scattering events, which is helpful when including carrier–carrier scattering and when vectorizing the code [24, 25].

Since most carriers undergo self-scattering at the end of each drift time step, however, waiting for the final state calculation of a small fraction of the electrons is a bottleneck in a finely parallelized MCM computation. That is, even though a particle simulation suggests parallelization, because the particle transport *calculations* are poorly synchronized, the MCM can be expected to have a performance handicap.

Figure 2(b) shows an expansion of the *scatter* block from Fig. 1 for the DKSM. The scattering rate is calculated and stored for $\mathbf{k}$ states and not integrated over the energy as is

usually done in the MCM. Next, the distribution function $f(\mathbf{k}_i)$ is redistributed according to the scattering matrix, S_{ij}. The scattering procedure for one time step is finished when all the appropriate final states for each initial state have been considered.

The scattering rate matrix is calculated independently of the transport calculation and is material dependent only, i.e., in this approximation it does not depend on the field. To calculate transport in a different semiconductor, the only change is to generate new S_{ij} and band structure files for the different material; *no* macroscopic material parameters (such as mobility) are input to the calculation.

The fraction of the distribution function in cell i redistributed by scattering to cell j is

$$\Delta f_{ij}|_{\text{scatter}} = f(\mathbf{k}_i)\{1 - \exp[-S_{ij}(1 - f(\mathbf{k}_j))\,\Delta t]\}. \tag{5}$$

$\delta f_{ij}|_{\text{scatter}}$ is subtracted from $f(\mathbf{k}_i)$ and added to $f(\mathbf{k}_j)$ for all cells i and j for which $S_{ij} \neq 0$ and $f(\mathbf{k}_i) \neq 0$. To avoid corruption of the distribution function, $f(\mathbf{k}_i)$ and $f(\mathbf{k}_j)$ are used at the beginning of the time step. To comply with the Boltzmann transport equation, the time step must satisfy

$$1 \gg \max\left(\sum_j^N (1 - e^{-S_{ij}\Delta t})\right) \tag{6}$$

for all cells i, or to first order in $S_{ij}\,\Delta t$,

$$\Delta t \ll 1/\max\left(\sum_j^N S_{ij}\right). \tag{7}$$

The scattering procedure in DKSM is the same for each initial cell, and each scattering of an initial state to a final state takes the same amount of run time. There are no (random) selection processes in the DKSM, nor extra calculations for a particular cell. This, and the independence between cell processing, makes the DKSM a good candidate for parallelization.

2.4. *Drifting*

In the drift stage of the DKSM, $f(\mathbf{k}_i)$ in each cell is shifted to other cells according to

$$\hbar \frac{d\mathbf{k}}{dt} = q\boldsymbol{\varepsilon}, \tag{8}$$

where q is the electron charge, $\boldsymbol{\varepsilon}$ is the field vector, and $\hbar$ is Planck's constant. We restrict drift in one time step to the adjacent cells. This is an algorithm limitation and is not intrinsic to the method. This restriction simplifies the drift algorithm and only slightly lengthens run time. The portion of a cell drifted in each direction is then given by

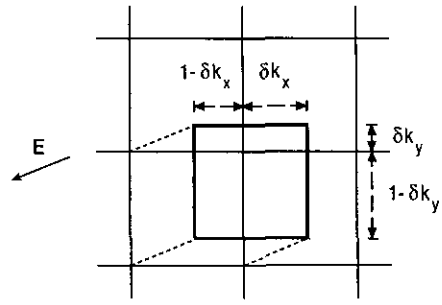

FIG. 3. Two-dimensional schematic of drifting. According to the field vector and time step, portions of $f(\mathbf{k})$ in each cell are distributed to adjacent cells.

$$\delta\mathbf{k} = \frac{q\boldsymbol{\varepsilon}}{\hbar}\Delta t. \tag{9}$$

The vector components δk_x, δk_y, and δk_z are dimensionless values normalized to the edge length Δk. To limit the drift process to (at most seven) adjacent k-space cells, the time step must meet the constraint

$$\Delta t \leq \frac{\hbar\,\Delta k}{q|\max(\varepsilon_x, \varepsilon_y, \varepsilon_z)|}, \tag{10}$$

Normally the scattering constraint, Eq. (7), limits the time step except for very high fields.

The fraction of $f(\mathbf{k}_i)$ redistributed to cell j because of drift is, therefore,

$$\Delta f_{ij}|_{\text{drift}} = f(\mathbf{k}_i)\Omega_i\,\Delta\Omega_{ij}, \tag{11}$$

where $\Delta\Omega_{ij}$ is the fraction of the volume of cell i that drifts to cell j. If cell i is represented by the vector $(\mathbf{k}_{ix}, \mathbf{k}_{iy}, \mathbf{k}_{iz})$, then

$$\Delta\Omega_{ij} = \delta\Omega_x\,\delta\Omega_y\,\delta\Omega_z, \tag{12}$$

where for the x component,

$$\delta\Omega_x = \begin{cases} \delta k_x, & \text{if } k_{jx} = k_{ix} \\ 1 - \delta k_x, & \text{if } k_{jx} = k_{ix} + \Delta k, \end{cases} \tag{13}$$

and similarly for the other components. These terms are defined in two dimensions in Fig. 3; the third dimension is omitted for clarity. As an example, the portion of cell i_0 that is shifted to the cell directly along the y axis is

$$\Delta\Omega_{i_0 j_0} = (1 - \delta k_x)\,\delta k_y(1 - \delta k_z), \tag{14}$$

where $j_0 = (k_{ix}, k_{iy} + \Delta k, k_{iz})$.

Drift on the Brillouin zone borders must be dealt with carefully. First, the number of electron states in each *cubic* cell is $(\Delta k)^3/4\pi^3$, but cells cut by the L-point face of the Brillouin

zone are not cubic. We treat these cut cells as cubes during drift, but we limit the number of available states in each according to their k-space volume, Ω_i. Second, periodic boundary conditions are enforced for the reduced Brillouin zone by a one-to-one pairing of each border cell with a cell offset by one reciprocal lattice vector. Third, if $\delta k_{x,y,\text{or}\,z} \neq 1$, drifting in the DKSM leads to a non-physical k-space redistribution because $f(\mathbf{k}_i)$ is forced to be uniform in each cell at the end of each time step. A similar problem is well known in cellular automata calculations of electron transport [19]. We currently avoid this problem by considering only fields along {100}, {110}, or {111} and choosing Δt such that $\Delta\Omega_{ij}$ equals 0 or 1. This method works because one of the grid axes is parallel to the field. As with every other problem in the DKSM, reducing the cell size improves the physical rigor and adds acceptable alternative approaches.

2.5. *Conserving Memory*

Perhaps the most overriding burden with the practical application of the DKSM is memory usage. As is common to many calculations, increasing memory usage can increase program speed, e.g., by using look-up tables, or it can improve the accuracy of the results. Since one of our principal goals is to show the feasibility of the DKSM, program speed is a secondary concern, although competitive calculations of MCM and DKSM show comparable run times. The greater memory requirements of the DKSM improve the precision of the solution over the MCM. In this section, we discuss techniques for reducing the memory needs for the DKSM while maintaining physical rigor. In Section 3.2, we outline other possible trade-offs for conserving memory.

The transport program size is primarily determined by four arrays: the scattering matrix S_{ij} (each element of which is a real number) and vector components of the final state, k_{jx}, k_{jy}, and k_{jz} (each element of which we identify with a small integer). The size of each of these arrays is given by the product of the number of initial states in the primary wedge and the maximum number of final states, both of which are largely determined by the cell size.

The constraints on the cell size are

(i) the energy resolution needed for absorption and emission processes,

(ii) drift and scattering conditions,

(iii) representation of $f(\mathbf{k})$ fine enough for accurate calculation of macroscopic observables, and

(iv) the maximum practical program size permitted by the computer.

The first three are upper bounds; the last is a lower bound. The number of initial states is proportional to the volume and so it increases as $\sim 1/(\Delta k)^3$, and the number of final states increases with $\sim 1/(\Delta k)^2$ because the density of states is an areal sum. Therefore, unless final states are independently limited, the program size increases as $\sim 1/(\Delta k)^5$. For example, for 10 intervals between Γ and X points, i.e., $\Delta k = \frac{1}{10}(2\pi/a)$, there are 4600 cells in the Brillouin zone; for 30 intervals, or $\Delta k = \frac{1}{30}(2\pi/a)$, there are 113,400 cells, almost 3^3 more. After much experimentation, we chose a grid with 30 intervals between Γ and X points. We explain this choice near the end of this section.

Concern over the memory requirements is first encountered in the disk usage of the scattering rate matrix. S_{ij} is calculated and stored to disk for later use by the transport calculation. For each initial state, scattering rates are separately sorted for emission and absorption (using a linked list) from highest to lowest. The information stored is the initial state i, the number of final states with non-zero scattering rates for i, followed by a list of each final state j with its scattering rate. This sequence is repeated for each state i in one irreducible wedge of the Brillouin zone, here called the *primary wedge*. Any of the 48 wedges can be designated the primary wedge. To save disk space, unique integer codes are used to define i and j, and "0." is written for repeated scattering rates. Binary file storage saves little disk space over a carefully formatted ASCII file. Other more elaborate coding methods can be used, but likely at the expense of readability.

The cubic symmetry of the semiconductor is exploited to save memory and disk space through the use of state transformations. A closely related transformation technique is commonly employed in full-band MCM simulations in the use of $E(\mathbf{k})$ [9, 26]. Figure 4 shows two Brillouin zones, each with two of the irreducible wedges shaded, to illustrate state transformations in an f-type scattering. For a cell i anywhere in the Brillouin zone, the transform T to the primary wedge is determined such that

$$Ti = i'. \tag{15}$$

T is a set of symmetry operations belonging to the point group of the lattice. The scattering matrix S_{ij} is only calculated and stored for transformed initial states i' in the primary wedge, although final states j' can be anywhere in the Brillouin zone. This is shown in Fig. 4(a). The final state j' is determined relative to the primary wedge, so it must be inverse transformed to find the actual final state j, as shown in Fig. 4(b), i.e.,

$$T^{-1}j' = j. \tag{16}$$

The transforms T and T^{-1} are kept as stored arrays to save on run-time transformation calculations; this is important since the CPU time for the DKSM is largely spent in the scattering procedure. By using a primary wedge and this transformation procedure, memory storage is reduced by a factor of 48 because S_{ij} is not stored for every possible initial state in the Brillouin zone.

To further control program size, we take two steps to limit the number of final states in S_{ij}. First, scattering is limited to energies near the center of final-state cells. Eligible final states are determined according to the energy and gradient at the

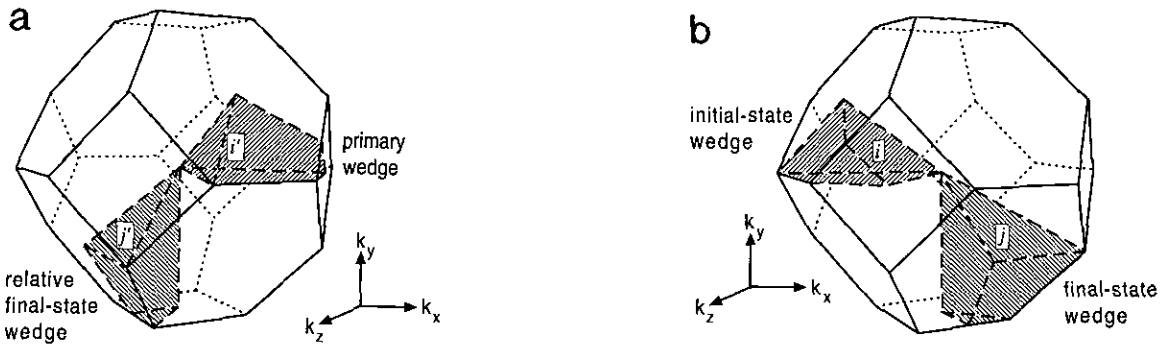

FIG. 4. Illustration of a scattering transformation for an example set of states i and j relative to (a) the primary wedge and (b) the initial state wedge. For any initial state i in the Brillouin zone, as shown in (b), a point transformation to the primary wedge is determined. Since S_{ij} is calculated for (transformed) states i' in the primary wedge, j' is inverse transformed to determine j, which is the final state relative to the (original) initial state i.

center of each cell. We have chosen to consider energies within an arbitrary radius of 35% of the k-space distance from the center to the face of each cell. This has the effect of lowering the number of possible final states, as well as reducing the number of absorption events wherein the distribution loses energy, and emission events that cause the distribution to gain energy. This type of absorption-emission mix-up occurs because the cell is too large compared to $\hbar\omega_0$; i.e., the energies near the walls of the cell are not close enough to the energy at the center. Therefore, emission and absorption processes place an upper bound on the cell size (see list near beginning of this section). Figure 5 illustrates the case when no scattering occurs and shows that reducing the cell size completely mitigates this type of problem.

Second, to control the size of S_{ij}, we limit the maximum number of final states to a total of 1200 for each initial state. The number of emission and absorption events are allocated in proportion to the total *possible* number of emission and absorption events. For example, if there are twice as many emission states as absorption states for an initial state i, then the 800 largest emission final states and the 400 largest absorption final states are included for i. In our calculation of S_{ij} for 30 intervals between Γ and X points, the maximum possible number of final states is 12,472, and the average number is 7626. In comparison, the limit of 1200 final states seems severe. By sorting from the highest to lowest scattering rate, however, the 1200-final-states limited total rate is not far from the realizable total rate except at high energy, as shown in Fig. 6. (The scattering rate calculation is discussed in more detail in Section 3.1.) Most of the individual rates are very small and do not contribute much to the total scattering rate. As clearly seen in Fig. 6, limiting the scattering rate to 100 final states, however, is too severe even at moderate energies.

With the above upper limits in place, the DKSM transport

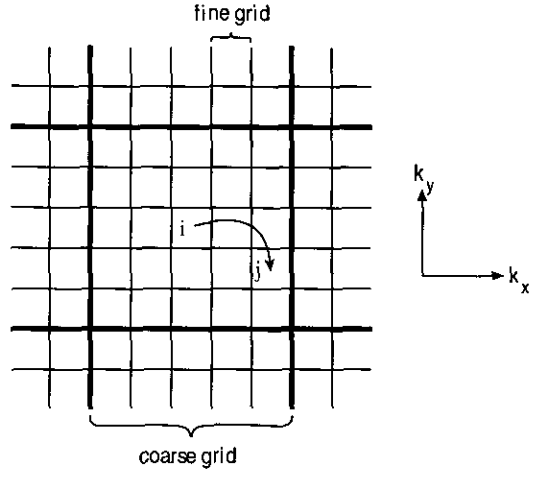

FIG. 5. Two-dimensional representation of scattering on a coarse grid (thick lines) and on a fine grid (thin and thick lines). The example of scattering from state i to state j shows the failure of the coarse grid in that $j = i$; i.e., initial and final states may be degenerate. For example, a grid with 10 cells between Γ and X points is too coarse compared with $\hbar\omega_0$, and this type of scattering is excessive.

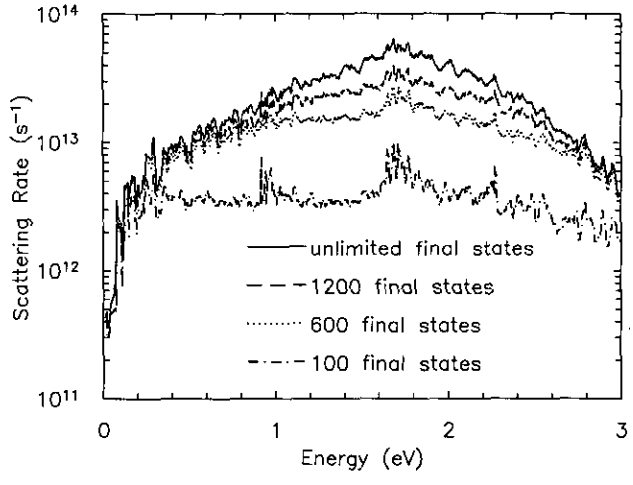

FIG. 6. Total scattering rate for an unlimited number of final states, and for a maximum of 1200, 600, and 100 final states. The scattering rates deviate from the unlimited rate at increasing energy as the number of final states increases.

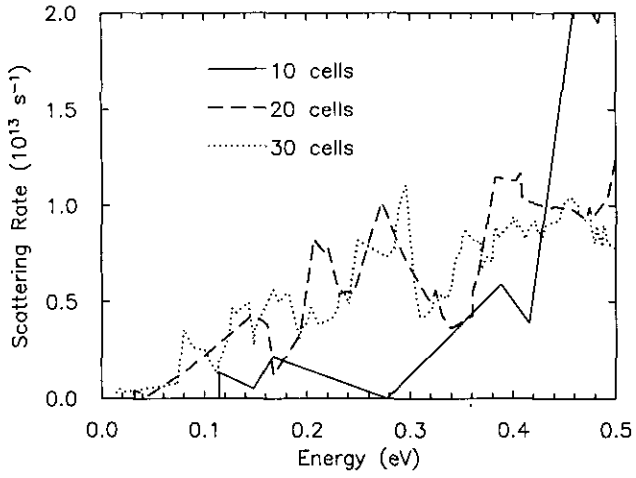

FIG. 7. Scattering rate at low energy for 10, 20, and 30 cells between Γ and X points. Minimizing the cell size, i.e., increasing the number of cells between Γ and X points, lowers the minimum energy and decreases the spread in scattering rates.

program is less than 40 Mbytes, and the ASCII S_{ij} file is 32 Mbytes. It is undesirable in terms of physical rigor to limit the scattering rate at all, but for the computer system we have available, it is a practical necessity. Increasing the cell size, or equivalently decreasing the number of cells in the Brillouin zone, causes the low-field transport to be compromised. Figure 7 shows the scattering rates on a linear scale at low initial-state energy for 10, 20, and 30 cells between the Γ and X points. Not only does a finer mesh increase the number of scattering rate points in the critical low-energy region, it decreases absorption–emission mix-up by providing energy separations between cell centers comparable to the phonon energy, $\hbar\omega_o$.

3. RESULTS

3.1. *Steady-State Homogeneous Transport in Si*

We have chosen steady-state, homogeneous transport in Si at 300 K to compare the DKSM against the MCM. Intervalley deformation potential scattering, using Fermi's Golden rule, is given by [2, 3]

$$S_{ij} = \frac{D^2}{8\pi^2\rho\omega_o}\left(N_q + \frac{1}{2} \pm \frac{1}{2}\right)\int_{j\text{cell}} \delta(E(\mathbf{k}) - E(\mathbf{k}_i) \mp \hbar\omega_o)\, d\mathbf{k}, \tag{17}$$

where upper and lower signs refer to emission and absorption, respectively, D is the effective deformation potential constant, ρ is the mass density, ω_o is the phonon frequency, and N_q is the equilibrium phonon occupation given by

$$N_q = [e^{\hbar\omega_o/k_BT} - 1]^{-1}. \tag{18}$$

TABLE I

Material Parameters Used in the DKSM and *Comparable MCM* Scattering Rate Calculations

Parameter	Value	Units
m_{xl}^*	0.737 (from $\mathbf{k}\cdot\mathbf{p}$)	m_0
m_{xt}^*	0.177 (from $\mathbf{k}\cdot\mathbf{p}$)	m_0
a	5.64 [2]	10^{-8} cm
ρ	2.33 [2]	g/cm
f1: D @ $\hbar\omega$	0.3 @ 19.0 [2]	10^8 eV/cm @ meV
f2: D @ $\hbar\omega$	2 @ 47.5 [2]	10^8 eV/cm @ meV
f3: D @ $\hbar\omega$	2 @ 59.1 [2]	10^8 eV/cm @ meV
g1: D @ $\hbar\omega$	0.5 @ 12.1 [2]	10^8 eV/cm @ meV
g2: D @ $\hbar\omega$	0.8 @ 18.6 [2]	10^8 eV/cm @ meV
g3: D @ $\hbar\omega$	11 @ 62.1 [2]	10^8 eV/cm @ meV

Note. Masses are extracted from the $\mathbf{k}\cdot\mathbf{p}$ calculation and not explicitly used in the S_{ij} calculation.

A valley degeneracy factor is not needed in Eq. (17) because the final state is completely specified. The integration over the k-space j cell, i.e., the final state, in Eq. (17) is calculated by determining the area of intersection between a plane (the local approximation for the isoenergy surface) and a cube (the j cell) [27, 28]. The scattering rate is very noisy at low energy because the isoenergy surface is ellipsoidal in Si with a radius of curvature small compared to Δk, and so it is poorly approximated by planar cuts. Material parameters used in the calculation of S_{ij} are listed in Table I. The band structure is calculated with a 30-band $\mathbf{k}\cdot\mathbf{p}$ method [29]. The total emission and absorption scattering rates are plotted in Fig. 8 as a function of energy.

We include only intervalley deformation potential optical phonon scattering in our calculation of transport in Si. Acoustic intravalley and impurity scattering are very important for the

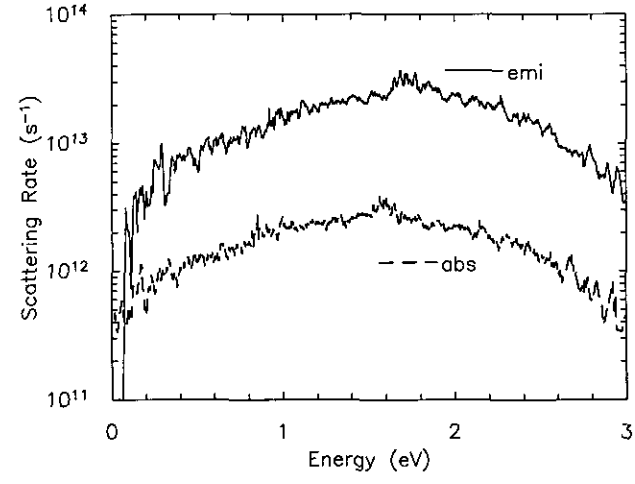

FIG. 8. Intervalley scattering rate separated into emission and absorption terms. Rates are plotted vs energy, although S_{ij} is calculated and stored for $\mathbf{k}$ states.

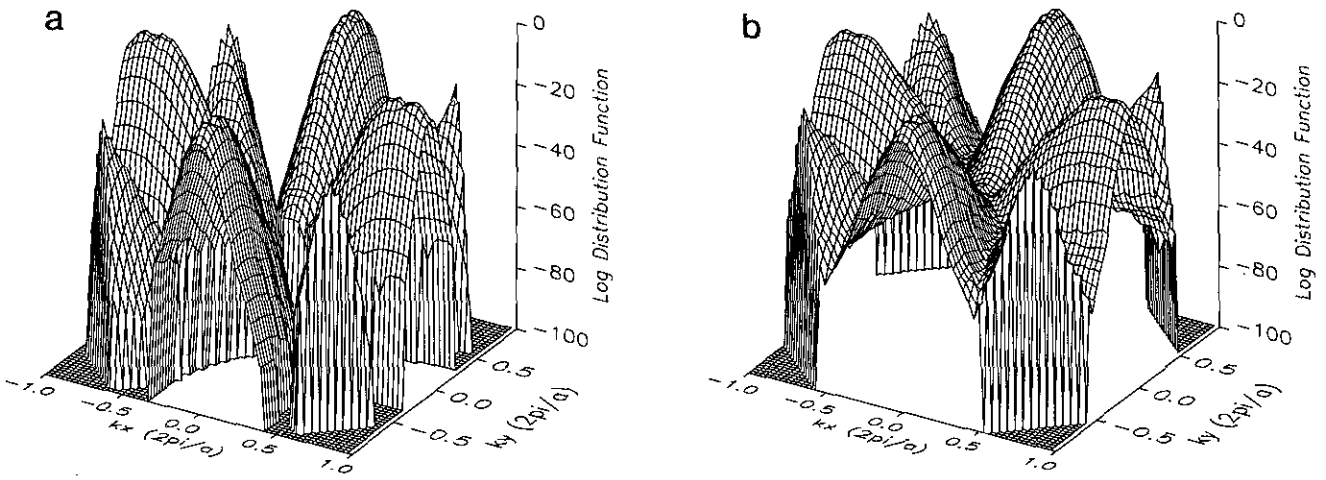

FIG. 9. The distribution function in the k_x–k_y plane of the Brillouin zone at fields of (a) 10 V/cm and (b) 1 kV/cm applied in the x direction. The z-axis is logarithmically scaled, highlighting the many orders of magnitude $f(\mathbf{k})$ is kept in the DKSM. The X valleys are most heavily populated, but electrons can also be found at the K point. At 1 kV/cm, electrons can be found throughout the Brillouin zone.

accurate calculation of mobility, but are far less important in the determination of average energy [2, 21]. A more complete simulation that can accurately model Si at modestly high fields should include acoustic phonon, impurity, and impact ionization scattering, and it should include at least two conduction bands.

The calculation of S_{ij} for all final states is carefully done to prevent double counting. The total scattering rate for an initial state to a final state is summed over all scattering mechanisms, which in our case means summing over deformation potentials. f- and g-type scattering specify the irreducible wedges for which the scattering rate is repeated and added. There are 32 wedges for f-type scattering (transverse valleys), and 8 for g-type scattering (opposite valley).

Figure 9 shows the calculated distribution function in a cross section of the Brillouin zone for 10 and 1000 V/cm applied in the x-axis direction. This cross section is in the k_x-k_y plane, so it includes X and K points. Figure 9 is somewhat misleading in that $f(\mathbf{k}_i)$ is not explicitly shown as flat in each cell, but it directly represents the DKSM results, i.e., the solution of the distribution function. Noise seen in the valleys is a manifestation of the discretization of the k-space grid. (See Figs. 6 and 7 for another example.) Even at only 1 kV/cm, regions between the valleys are populated, showing the importance of using a full band method.

The calculated energy distribution, $f(E)$, is an important basis for the verification of a transport calculation because it is more readily comparable than $f(\mathbf{k})$. $f(E)$ is the occupation probability of all $\mathbf{k}$ states for which $E = E(\mathbf{k}_i)$ integrated over the Brillouin zone. In the DKSM, the output energy distribution function is calculated from

$$f(E) = \frac{\sum_i f(\mathbf{k}_i)\Omega_i \delta(E - E(\mathbf{k}_i))}{\sum_i \Omega_i}, \tag{19}$$

where $\delta(E - E(\mathbf{k}_i))$ is a delta function defined here as unity if $E = E(\mathbf{k}_i)$, and as zero otherwise. Since $f(\mathbf{k}_i)$ is uniform in each cell and $E(\mathbf{k}_i)$ is discrete, $f(E)$ is a piecewise continuous (step-like) function.

Figure 10 shows $f(E)$ for fields from 0 to 10 kV/cm. Data points are shown as continuous lines for clarity. The 0-field curve represents the converged steady-state solution starting from a (straight line) Fermi–Dirac distribution function and has the same slope. The most striking aspect of Fig. 10 is that it shows 14 orders of magnitude. The DKSM is deterministic in that no random selections are made, unlike a Monte Carlo

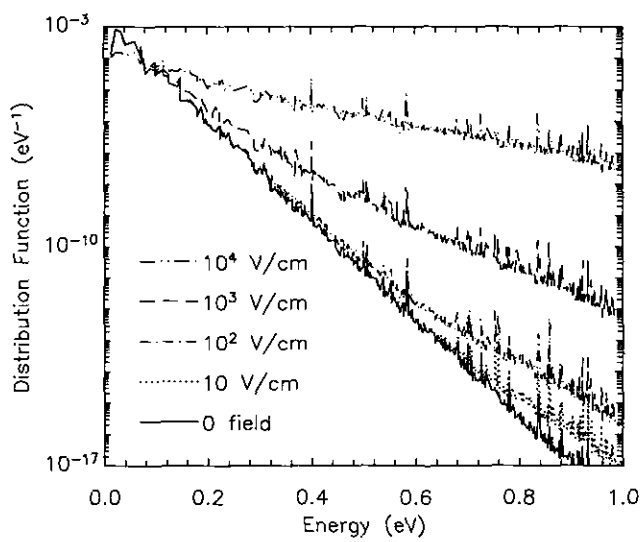

FIG. 10. Steady-state electron distribution function vs energy at fields of 0, 10, 10^2, 10^3, and 10^4 V/cm. For low fields, the only noticeable change in $f(E)$ from equilibrium is in the tail, i.e., for energies far above k_BT. The DKSM is a deterministic, full band method that allows $f(E)$ to be calculated over many orders of magnitude.

calculation. The "noise" seen in $f(E)$ is present because of limits in the precision of the $E(\mathbf{k}_i)$ and S_{ij} calculations and because of the mesh discreteness. The precision of f is limited by the precision of the computer and accumulated numerical error. This is because the DKSM is a deterministic method and so can store $f(E)$ over many more orders of magnitude than even a very large MCM program.

The high-level of precision in the distribution function suggests that DKSM might be an appropriate method in the study of low-probability events, such as electron emission into oxide or impact ionization. This level of precision in a method that includes full band structure makes the DKSM exceptional.

The most telling comparison of the DKSM is in the calculations of macroscopic observables, such as drift velocity. In the DKSM, the drift velocity v_d is calculated from

$$v_d = \frac{\sum_i \mathbf{v}_i \cdot \boldsymbol{\varepsilon} f(\mathbf{k}_i)\Omega_i}{\sum_i |\boldsymbol{\varepsilon}| f(\mathbf{k}_i)\Omega_i}, \tag{20}$$

where the group velocity at the center of each cell is proportional to the k-space derivative of the energy, i.e.,

$$\mathbf{v}_i = \frac{1}{\hbar}\frac{\partial E(\mathbf{k}_i)}{\partial \mathbf{k}}. \tag{21}$$

Since our DKSM calculations do not consist of a full complement of scattering mechanisms, we performed full-band MCM calculations using *only* intervalley scattering calculated with the same material parameters (here and in the plots referred to as *comparable MCM*). These results are displayed in Figure 11, along with results from a previously published MCM calculation [30] (here called *complete MCM*) for overall comparison. The *complete MCM* results are expectedly low because additional scattering mechanisms reduce the mobility. The DKSM agrees well with the *comparable MCM* until about 10^4 V/cm. We attribute this failure to the 1200-final-states limit placed on the S_{ij} calculation because of a program size restriction, as described in Section 2.5. This claim is further supported by DKSM calculations that differ only in the maximum number of final states. The 600-final-states calculation in Fig. 11 shows divergence from the 1200-final-states calculation only at high fields. The 100-final-states calculation is erratic because the approximation is too extreme. In summary, the DKSM performs as expected, and extrapolation suggests valid high-field results if the number of final states is increased.

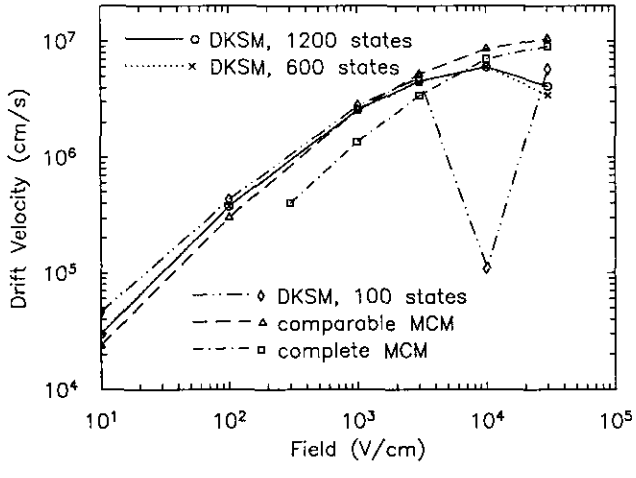

FIG. 11. Drift velocity vs field comparing calculations from the DKSM, a *comparable MCM* using a scattering rate calculation equivalent to the DKSM, and a *complete MCM* from another publication [30]. The *complete MCM* results are shown to establish the legitimacy of the other two calculations, although, unlike the DKSM and the *comparable MCM* calculations, it is not a full band method. The 1200-final-states DKSM calculation agrees with the *comparable MCM* until about 10^4 V/cm. Decreasing the number of final states decreases the agreement first at high fields.

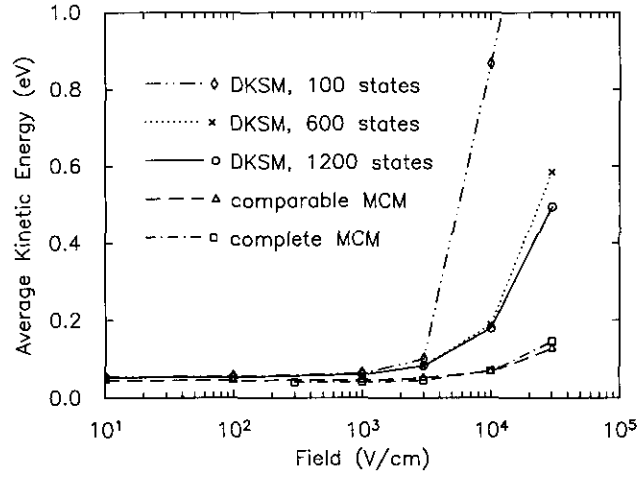

FIG. 12. Average electron energy vs field comparing calculations from the DKSM, a *comparable MCM* using a scattering rate calculation equivalent to the DKSM, and a *complete MCM* from another publication [30]. Failure of the DKSM average energy calculation using 1200 final states occurs at about the same field as for the drift velocity calculation (see Fig. 11). The 600- and 100-final-states DKSM average energies diverge from the 1200-final-states calculated results at decreasing fields.

Figure 12 shows the average electron energy versus field calculations from the DKSM compared with the same MCM calculations. The average electron energy in the DKSM, $\langle E \rangle$, is given by

$$\langle E \rangle = \frac{\sum_i E(\mathbf{k}_i) f(\mathbf{k}_i)\Omega_i}{\sum_i f(\mathbf{k}_i)\Omega_i}. \tag{22}$$

The 1200-final-states DKSM calculation again agrees well with the comparable MCM until about 10^4 V/cm. Agreement for the DKSM at high fields decreases as the number of final states decreases.

3.2. *Possible Compromises*

Except for some limited applications, the DKSM in the form presented here is inappropriate for most transport calculations on standard computers. Although it shows promise for advanced computers and for future computers as standard disk space and memory increase, it is too large for common use on present-day PCs and workstations. Several compromises are currently available, however, that can make the DKSM far more usable without waiting for the next generation of computers. The appropriateness of the compromise depends on the application and calculation goals.

The most direct way to limit the program size is to limit the number of final states. Reducing the number of final states also speeds the calculation, since profiling indicates that, overwhelmingly, most CPU time is spent in the scattering routine. The simplest way to decrease the number of states is to not use a full band calculation. A two-valley effective mass model is a reasonable alternative [17]. High field transport and calculations involving low-probability events are sacrificed in this approach.

Gridding the Brillouin zone with a non-constant cell size is another way to decrease the number of final states. The density of states increases sharply above the band edge, reducing the need for closely spaced cells in k space at high energy. Therefore, the most obvious, least compromising gridding scheme is to use small cells near the valley extrema and large cells otherwise.

Program speed will become an issue in the future development of the DKSM. Run time is a sensitive function of the number of final states and convergent criterium. For example, for 30 intervals between Γ and X points, to converge the average energy to within 0.005% across a time step, the DKSM normally takes half a day on an HP720 (comparable to a MCM calculation). The scattering procedure is simpler in the DKSM than in the MCM, but there is a speed cost for increased precision.

Several changes can be made to decrease the run time. First, drift does not have to be limited to adjacent cells. This is not expected to have a significant effect, however, since in our experience almost all the CPU time is spent in the scattering routine. Efforts to improve memory stride by sequencing the S_{ij} array might be worthwhile. Besides decreasing the number of final states in S_{ij}, perhaps the most promising approach is to limit the precision of $f(\mathbf{k}_i)$ to reduce the number of scatterings.

If the memory problems are overcome, then the DKSM will extend well to certain regimes of device modeling. Applying the DKSM to device simulations will require the distribution function to include a spatial dimension. Heterojunctions will require multiple S_{ij} calculations. Each will further stress memory needs. A deterministic, full-band method, however, is an ideal tool for studying rare events, such as oxide emission or impact ionization, and for high-field transport. As described in Section 2.3, the DKSM might also be the best choice for massively parallel calculations because of synchronization advantages.

4. CONCLUSION

The DKSM is very similar to the MCM in the way it *solves* the Boltzmann transport equation, yet it is profoundly different in the way it *stores* the solution, the distribution function. Like the MCM, the DKSM is computationally expensive, even more so in terms of memory usage. Unlike the MCM, the DKSM is deterministic and therefore lacking stochastic noise. It also promises to be more adaptable to modern vector and parallel computers than the MCM.

In the process of describing the DKSM in detail, we have openly discussed the shortcomings of the DKSM. We have also shown, in comparisons to the MCM, that the DKSM performs well for homogeneous steady-state transport, especially considering program size limitations. Overall, we have shown that the DKSM is a viable method and is a strong alternative candidate to the MCM for algorithm development on advanced computer architectures.

ACKNOWLEDGMENTS

We are thankful to P. D. Yoder for discussions and collaboration attempts, and we readily acknowledge that this work would not have been possible without early nurturing and direction from K. Hess. This study was supported by the National Science Foundation through the National Center for Computational Electronics at the University of Illinois at Urbana-Champaign.

REFERENCES

1. P. J. Price, in *Semiconductors and Semimetals*, edited by R. K. Willardson and A. C. Beers (Academic Press, New York, 1979), Vol. 14.
2. C. Jacoboni and L. Reggiani, *Rev. Mod. Phys.* **55**, 698 (1983).
3. K. Tomizawa, *Numerical Simulation of Submicron Semiconductor Devices* (Artech House, Boston, 1993).
4. K. Yokoyama and K. Hess, *Phys. Rev. B* **33**, 5595 (1986).
5. M. V. Fischetti and S. E. Laux, *IEEE Trans. Electron Devices* **38**, 650 (1991).
6. S. Selberherr, *Analysis and Simulation of Semiconductor Devices* (Springer-Verlag, Vienna, 1984).
7. J. M. Higman, K. Hess, C. G. Hwang, and R. W. Dutton, *IEEE Trans. Electron Devices* **36**, 930 (1989).
8. The time step is at most on the order of the fastest dephasing time in the semiconductor, much less than a picosecond. Simulation times for devices are commonly in the nanosecond range.
9. M. V. Fischetti and S. E. Laux, *Phys. Rev. B* **38**, 9721 (1988).
10. R. E. Bank, D. J. Rose, and W. Fichtner, *IEEE Trans. Electron Devices* **30**, 1031 (1983).
11. H. D. Rees, *J. Phys. C: Solid State Phys.* **6**, 262 (1973).
12. P. A. Lebwohl and P. M. Marcus, *Solid-State Commun.* **9**, 1671 (1971).
13. A. Das and M. S. Lundstrom, *Solid-State Electron* **33**, 1299 (1990).
14. A. Das and M. S. Lundstrom, *IEEE Trans. Electron Devices* **39**, 1157 (1992).
15. S. Krishnamurthy, A. Sher, and A.-B. Chen, *Appl. Phys. Lett.* **55**, 1002 (1989).
16. Y. L. LeCoz, Ph.D. thesis, Massachusetts Institute of Technology, 1988 (unpublished).

17. C. F. Fernandes and H. A. Santos, *IEE Proc.* **134**, 148 (1987).

18. J. P. Nougier, L. Hlou, P. Houlet, J. C. Vaissiere, and L. Varani, *Proc. Third International Workshop on Computational Electronics, Portland, Oregon, May 18–20, 1994.*

19. K. Kometer, G. Zandler, and P. Vogl, *Phys. Rev. B* **46**, 1382 (1992).

20. G. Schaeffer and P. Hui, *J. Comput. Phys.* **89**, 1 (1990).

21. B. K. Ridley, *Quantum Processes in Semiconductors* (Clarendon Press, Oxford, 1988).

22. An exception, that is, k-dependent scattering rates, is P. D. Yoder, Ph.D. thesis, University of Illinois at Urbana-Champaign, 1990 (unpublished).

23. T. Wang and K. Hess, *J. Appl. Phys.* **57**, 5336 (1985).

24. U. Ravaioli, in *Monte Carlo Device Simulation: Full Band and Beyond*, edited by K. Hess, J.-P. Leburton, and U. Ravaioli (Kluwer Academic, Boston, 1991).

25. U. A. Ranawake, P. Lenders, and S. M. Goodnick, in *Computational Electronics*, edited by K. Hess (Kluwer Academic, Boston, 1991).

26. H. Shichijo and K. Hess, *Phys. Rev. B* **23**, 4197 (1981).

27. G. Gilat and L. J. Raubenheimer, *Phys. Rev.* **144**, 390 (1966).

28. G. Gilat and Z. Kam, *Phys. Rev. Lett.* **22**, 715 (1969).

29. F. H. Pollak, C. W. Higginbotham, and M. Cardona, *J. Phys. Soc. Jpn., Suppl.* **21**, 20 (1966).

30. C. Canali, C. Jacoboni, G. Ottaviani, and A. Alberigi-Quaranta, *Appl. Phys. Lett.* **27**, 278 (1975).