Comparison of self-consistency iteration options for the Wigner function method of quantum device simulation

B. A. Biegel and J. D. Plummer
Center for Integrated Systems, Department of Electrical Engineering, Stanford University, Stanford, California 94305-4070
(Received 16 January 1996)

In the present work, we compare the efficiency, accuracy, and robustness of four basic iteration methods for implementing self-consistency in Wigner function-based quantum device simulation. These methods include steady-state Gummel, transient Gummel, steady-state Newton, and transient Newton. In a single mathematical framework and notation, we present the numerical implementation of each of these self-consistency iteration methods. As a test case to compare the iteration methods, we simulate the current-voltage (I-V) curve of a resonant tunneling diode. Standard practice for this task has been to rely solely on either a steady-state or a transient iteration method. We illustrate the dangers of this practice, and show how to take advantage of the complimentary strengths of both steady-state and transient iteration methods where appropriate. Thus, because the steady-state methods are vastly more efficient (i.e., have a much lower computational cost), and are usually equal in accuracy to the transient methods, the former are preferable for wide-ranging initial device investigations such as tracing the I-V curve. Implementation difficulties which we address here may have reduced the use of the steady-state methods in practice. On the other hand, the transient methods are inherently more robust and accurate (i.e., they reliably and correctly reproduce device physics). However, the high computational cost of the transient methods makes them more appropriate for a narrower range of directed investigations where transient effects are inherent or suspected, rather than for full I-V curve traces. Finally, we found the two Gummel methods to be generally preferable to their (theoretically more accurate) Newton counterparts, since the Gummel methods are equally accurate in practice, while having a lower computational cost. [S0163-1829(96)00135-X]

I. INTRODUCTION

The Wigner function formulation of quantum mechanics[1–3] has many useful characteristics for the simulation of quantum-effect electronic devices, including the natural ability to handle small-signal or transient conditions in self-consistent, dissipative, and open-boundary systems.[4–9] However, solving the Wigner function transport equation is a relatively computer-intensive proposition. Further, the inclusion of self-consistency[7–11] requires an iterative solution of the Wigner function transport equation and Poisson's equation, making the computational efficiency of the iteration method critically important. In the present work, we consider four basic iteration methods for implementing self-consistency in the Wigner function approach to quantum device simulation, including steady-state Gummel, transient Gummel, steady-state Newton, and transient Newton. In the first half of this paper (Secs. II–V), we present, in a single mathematical framework and notation, the analytical formulation and numerical implementation of each of these self-consistency iteration methods. In the second half (Sec. VI), we use simulation examples to compare the efficiency (computational cost), accuracy (ability to correctly reproduce device physics), and robustness (reliability) of these iteration methods.

Due to the difficulty of implementing and maintaining multiple self-consistency iteration approaches in a numerical simulator, most researchers using Wigner function simulation rely on a single implementation, usually the steady-state or transient Gummel approach, in their quantum device research. The simulation tool used in this work, SQUADS (Stanford quantum device simulator), has been designed for the investigation of quantum device *simulation* as much as for the investigation of quantum device *operation*. Its modular structure makes it ideally suited to the analysis of alternative simulation approaches, such as the comparison of self-consistency iteration methods in this work. Only by presenting the theory, numerical implementation, and simulation examples for all of these simulation alternatives in a cohesive framework is this comparison possible.

In selecting specific simulation examples for this comparison, we note that only transient iteration methods are suitable for time-dependent investigations, such as switching, small-signal, or large-signal simulations. However, for the very basic electronic device simulation task of tracing the current-voltage (I-V) curve, steady-state methods are also suitable. Therefore, we used the accurate generation of the I-V curve for the ''prototypical'' quantum device, the resonant tunneling diode (RTD),[12–14] as the test case for evaluating the four self-consistency iteration methods. In fact, this device and simulation task have been the most common in the Wigner function simulation literature. Figure 1 shows a ''typical'' measured RTD I-V curve.[15] Some features of note in this I-V curve are a negative differential resistance region and a bistable region. The ''plateau'' shape in the negative differential resistance region is actually the time average of a very fast oscillating current. The ability of the various self-consistency iteration methods to efficiently and reliably reproduce these features will be the basis for their comparison.

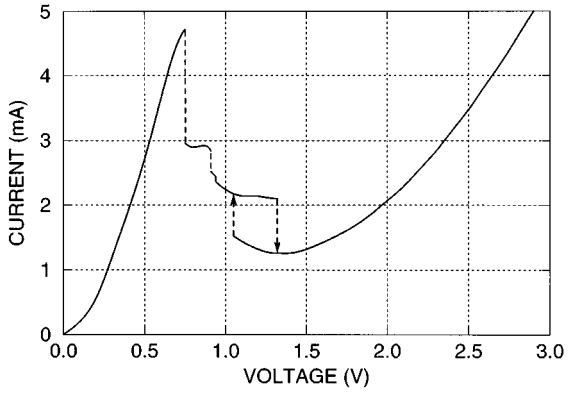

FIG. 1. Experimental RTD I-V curve (Ref. 15) showing the characteristic negative differential resistance region and plateau structure between 0.8 and 1.3 V. The plateau current is actually the time average of a high-frequency oscillating current. (Permission to reprint data given by T.C.L.G Sollner.)

II. WIGNER FUNCTION FORMULATION

The Wigner function approach models a quantum system by computing the evolution of the Wigner function $f(x,k,t)$, a phase-space state function, according to the Wigner function transport equation (WFTE). In one dimension and for a constant effective mass, the WFTE is

$$\frac{\partial f}{\partial t} = -\frac{hk}{2\pi m^*}\frac{\partial f}{\partial x} - \frac{1}{h}\int_{-\infty}^{\infty} dk' f(x,k') \times \int_{-\infty}^{\infty} dy[U(x+y)-U(x-y)] \times \sin[2y(k-k')] + \left.\frac{\partial f}{\partial t}\right|_{\text{coll}}, \quad (1)$$

where h is Planck's constant, m^* is the electron effective mass, and U is the conduction-band minimum. (We use an n-type device for all derivations and simulations in this work.) Scattering has been implemented in the Wigner function method using the relaxation time approximation, based on the analogy of the WFTE to the (classical) Boltzmann transport equation.[5,6] The resulting scattering term is

$$\left.\frac{\partial f}{\partial t}\right|_{\text{coll}} = \frac{1}{\tau}\left[\frac{f^{\text{eq}}(x,k)}{\int dk\, f^{\text{eq}}(x,k)}\int dk\, f(x,k) - f(x,k)\right], \quad (2)$$

where τ is the relaxation time and f^{eq} is the equilibrium (zero bias, no scattering) Wigner function. In this work, we use the upwind (i.e., incoming-carrier), equilibrium, Fermi-Dirac distribution boundary conditions given by Frensley:[4]

$$f^{\text{bc}}_{x=0,k>0} = \frac{4\pi m^* k_B T}{h^2}\ln\left\{1+\exp\left[-\frac{1}{k_B T}\left(\frac{h^2k^2}{8\pi^2 m^*}-\mu_0\right)\right]\right\}, \quad (3a)$$

$$f^{\text{bc}}_{x=L,k<0} = \frac{4\pi m^* k_B T}{h^2}\ln\left\{1+\exp\left[-\frac{1}{k_B T}\left(\frac{h^2k^2}{8\pi^2 m^*}-\mu_L\right)\right]\right\}. \quad (3b)$$

Enforcing self-consistency in an electronic device simulator means requiring that the energy-band profile $U(x)$ of the device be consistent with the charge-density profile $\rho(x)$ in the device. To ensure this, Poisson's equation (PE) must be satisfied simultaneously with the WFTE. The PE uses the charge density to determine the energy-band profile of the device, and the WFTE uses the energy-band profile to determine (among other things) the charge density. In one dimension, Poisson's equation can be written

$$\frac{d}{dx}\left[\varepsilon(x)\frac{d}{dx}u(x)\right] = q\rho(x) = q^2[C(x)-c(x)], \quad (4)$$

where ϵ is permittivity, u is the (Hartree, or mean-field) potential, q is the electronic charge, c is the free-electron density, and C is the fixed charge density (e.g., ionized dopants). The conduction-band minimum is calculated from the potential

$$U(x) = u(x) + \delta U(x), \quad (5)$$

where δU is the (fixed) heterostructure band offset. The boundary conditions on the PE as enforced in SQUADS are very simple: the applied bias strictly determines the potential at the contacts. We chose the $x=0$ Fermi level as the reference energy, so that $\delta U(0) = -\mu_0$. Thus the PE boundary conditions are:

$$u(0) = 0, \quad (6a)$$

$$u(L) = (\mu_0 - \mu_L) + [\delta U(0) - \delta U(L)] - qV_a. \quad (6b)$$

To complete the WFTE-PE interdependence, the carrier density is calculated from the Wigner function using

$$c(x) = \frac{1}{2\pi}\int dk\, f(x,k). \quad (7)$$

III. DISCRETIZATION

To solve the WFTE-PE system, we must discretize the simulation domain and, accordingly, these two equations. The details of this process are well described elsewhere,[4,16] so only a summary is reported here, as implemented in SQUADS. In this section, we complete most of this discretization process, leaving only those details which differ between the self-consistency iteration methods for the following sections. The simulation domain is discretized as follows:

$$f(x,k,t) \rightarrow f(x_i,k_j,t_n) \equiv f_{i,j,n}, \quad (8)$$

$$x_i = i\Delta_x, \quad i \in \{0,1,\ldots,N_x\}, \quad (L = N_x\Delta_x), \quad (9)$$

$$k_j = \frac{\pi}{N_k\Delta_x}\left[j - \left(\frac{N_k+1}{2}\right)\right], \quad j \in \{1,2,\ldots,N_k\}, \quad (10)$$

$$t_n = n\Delta_t, \quad n \in \{0,1,\ldots,N_t\}, \quad (11)$$

where L is the width of the simulation region.

In the present work, we compare to and extend the results of Jensen and Buot,[16] and, therefore, we use their device structure and WFTE discretization scheme. In particular, we

use a second-order upwind difference scheme to discretize the position derivative, and Cayley discretization for the time derivative. The discretized WFTE, a system of $(N_x+1)N_k$ simultaneous equations (one for each phase-space grid point), can be abbreviated

$$(\mathbf{T}+\mathbf{K}+\mathbf{P}+\mathbf{S})[F_{i,j,n}]=-(4/\Delta_t)f_{i,j,n}, \tag{12}$$

where, in the transient mode,

$$\mathbf{T}[F_{i,j,n}]=-(2/\Delta_t)F_{i,j,n}, \tag{13a}$$

$$F_{i,j,n}=f_{i,j,n+1}+f_{i,j,n}. \tag{13b}$$

[Note that, for a transient simulation with the Cayley time derivative (hereafter called a Cayley simulation), solving the system of equations yields F_n, from which the Wigner function f_{n+1} must be calculated.] In the steady-state mode,

$$\mathbf{T}=\mathbf{0}, \tag{14a}$$

$$F_{i,j,n}=f_{i,j}. \tag{14b}$$

Below, the time subscript n is suppressed, since the other terms in Eq. (12) are time independent. For these terms,

$$\mathbf{K}[F_{i,j}]=\frac{hk_j}{4\pi m^*\Delta_x}\begin{cases} F_{i+2,j}-4F_{i+1,j}+3F_{i,j} & (j\leqslant \frac{1}{2}N_k) \\ -F_{i-2,j}+4F_{i-1,j}-3F_{i,j} & (j>\frac{1}{2}N_k) \end{cases}, \tag{15}$$

$$\mathbf{P}[F_{i,j}]=\sum_{j'=1}^{N_k} V_{i,j-j'}F_{i,j'}, \tag{16a}$$

$$V_{i,j''}\equiv\frac{2}{N_k}\sum_{i'=1}^{N_k/2}\sin\left[\frac{2\pi}{N_k}i'j''\right](U_{i+i'}-U_{i-i'}), \tag{16b}$$

$$S[F_{i,j}]=\frac{1}{\tau}\left[\frac{F^{\rm eq}_{i,j}}{\left(\sum_{N_k}^{j''=1}F^{\rm eq}_{i,j''}\right)}\left(\sum_{j'=1}^{N_k}F_{i,j'}\right)-F_{i,j}\right], \tag{17}$$

where V is called the nonlocal potential. [Note that for Cayley simulations, $F^{\rm eq}=2f^{\rm eq}$, although the factor of 2 cancels out in Eq. (17). However, it is important to use $F^{\rm bc}=2f^{\rm bc}$ for all boundary conditions in Cayley simulations.]

Given the discrete Wigner function, the discrete carrier density is computed as

$$c_i=\frac{\Delta_k}{2\pi}\sum_{j=1}^{N_k}f_{i,j}. \tag{18}$$

For completeness, the expression for the discrete current density will be given. It is determined from the continuity equation, using the fact that the current density must be position independent at steady state.[4] The resulting expression depends on the form used for the discrete position derivative. For the second-order upwind derivative, the discrete current density is[6]

$$J_{i+1/2}=\frac{-qh\Delta_k}{8\pi^2m^*}\sum_{j=1}^{N_k}k_j\begin{cases} 3f_{i+1,j}-f_{i+2,j} & (j\leqslant\frac{1}{2}N_k) \\ 3f_{i,j}-f_{i-1,j} & (j>\frac{1}{2}N_k) \end{cases}. \tag{19}$$

We will discuss in somewhat more detail the discretization of the Poisson equation, since the implementation of self-consistency using the PE is the focus of this work. The PE has been discretized in two ways by researchers investigating the WFTE-PE system: the direct form[16] and the differential (or Newton) form.[9] The appropriate form of the PE depends on the self-consistency iteration method, as discussed in Secs. IV and V. Both forms are described here.

The direct Poisson equation can be written, for a position-dependent permittivity,

$$(1+a)u_{i+1}-2u_i+(1-a)u_{i-1}=(q^2\Delta_x^2/\varepsilon_i)[C_i-c_i], \tag{20a}$$

$$a\equiv(\varepsilon_{i+1}-\varepsilon_{i-1})/(4\varepsilon_i). \tag{20b}$$

[Note that for the position-independent permittivity assumed in the simulations of Sec. VI, $a=0$, and that the discrete PE (in both forms) is even simpler.]

The Newton form of the PE is more complicated but more flexible. Newton equations are inherently iterative, seeking to find the solution to a nonlinear system by successively better approximations. To derive the Newton PE, we first define the ''Poisson function'' $P(u)$, which is based on the PE and must evaluate to 0 when the self-consistent potential and carrier density are supplied as input. From Eq. (4),

$$P^{(n)}(u)\equiv\frac{d}{dx}\left[\varepsilon(x)\frac{d}{dx}u^{(n)}(x)\right]-q^2[C(x)-c^{(n)}(x)]. \tag{21}$$

In Eq. (21), n is the iteration index (which for transient simulations is also the time step). A Newton iteration is a two-step process. First, the Newton PE system of equations is solved for δu, the *change* in the potential,

$$\left[\frac{\partial P^{(n)}(u)}{\partial u^{(n)}}\right][\delta u^{(n+1)}(x)]=-[P^{(n)}(u)]. \tag{22}$$

Then the potential is updated,

$$u^{(n+1)}(x)=u^{(n)}(x)+\delta u^{(n+1)}(x). \tag{23}$$

If the Newton iteration converges to the self-consistent solution, $P^{(n)}(u)$ converges to 0, and therefore so will the updates, δu. We denote the converged self-consistent potential as $u^*(x)$.

In discrete form, the Newton PE becomes

$$\left[\frac{\partial P_i^{(n)}}{\partial u_{i'}^{(n)}}\right][\delta u_i^{(n+1)}]=-[P_i^{(n)}(u)], \tag{24a}$$

$$u_i^{(n+1)}=u_i^{(n)}+\delta u_i^{(n+1)}, \tag{24b}$$

where

$$P_i^{(n)}(u)=(1+a)u_{i+1}^{(n)}-2u_i^{(n)}+(1-a)u_{i-1}^{(n)} -(q^2\Delta_x^2/\varepsilon_i)[C_i-c_i^{(n)}], \tag{25}$$

$$\frac{\partial P_i^{(n)}}{\partial u_{i'}^{(n)}} = (1+a)\,\delta_{i+1,i'}^{(n)} - 2\,\delta_{i,i'}^{(n)} + (1-a)\,\delta_{i-1,i'}^{(n)} + \left(\frac{q^2\Delta_x^2}{\varepsilon_i}\right)\left[\frac{\partial c_i^{(n)}}{\partial u_{i'}^{(n)}}\right], \tag{26}$$

and $\delta_{i,i'}$ is the Kronecker delta function. Note that $\partial c/\partial u$ is left unspecified for now, since its value depends on which self-consistency iteration method is used. It is not difficult to show that the direct PE, Eq. (20a), is actually a special case of the Newton PE, where we take $\partial c/\partial u = 0$. This is how SQUADS implements the direct PE when needed.

Solving the direct PE yields the exact potential profile $u(x)$ for the given carrier density profile $c(x)$. The Newton PE, in contrast, uses the $\partial c/\partial u$ term to account for changes in $c(x)$ that will result from the $u(x)$ solution, and thereby attempts to predict a $u(x)$ which is closer to $u^*(x)$. In other words, the $\partial c/\partial u$ term (which should be based on a distillation of the WFTE) in the Newton PE provides some corrective feedback to achieve faster convergence to the self-consistent operating point.

An unresolved issue is what to use for the initial potential profile u^0 in the first solution of the Newton PE and the WFTE at each bias point. For steady-state I-V curve simulations SQUADS uses linear extrapolation from $u^*(x)$ at the previous two bias points. [At the first bias point, linear band bending is used, and at the second, a linear potential is added to $u^*(x)$ from the first bias point.] Transient I-V curve tracing is one continuous simulation, so the final potential profile $u^*(x)$ at one bias point is used to compute u^0 at the next. In particular, when the bias is incremented in a transient simulation, the potential profile is incremented linearly across the entire device.[17] [Again, linear band bending is used to initialize the potential profile at the first bias point.]

The combination of the WFTE and PE, when discretized for numerical solution, constitutes a nonlinear system of equations. The self-consistency iteration methods offer a means of solving this nonlinear system (which we cannot solve directly) by iteratively solving a set of linear equations (which we *can* solve directly). Sections IV and V detail the remainder of the numerical implementation of four self-consistency iteration methods for the WFTE-PE system. Because the mathematics of the steady-state and transient approaches of each method (Gummel or Newton) are similar, the two Gummel approaches are described together in Sec. IV, and the two Newton approaches in Sec. V. However, the task of tracing the self-consistent operating points along the I-V curve, which has been chosen for our iteration method comparison, is very different for the transient and steady-state approaches. The steady-state approaches try to locate the self-consistent operating point in as few iterations as possible, while the transient approaches seek to follow the actual time-dependent operation of the device until it evolves to the steady state. Therefore, when running simulations, the converse pairing is more appropriate, so in Sec. VI we consider the two steady-state methods together followed by the two transient methods.

IV. GUMMEL (PLUG-IN) APPROACH

The Gummel (a.k.a. plug-in) approach[18] to solving the WFTE-PE system is almost universally used to add self-consistency to the WFTE. This is due to the simplicity of the Gummel approach, since the two equations are solved independently,[19] and the PE is numerically much simpler to implement and solve than the WFTE. For the steady-state Gummel method,[8,9] the steady-state WFTE and the PE are iteratively and alternately solved, plugging in one equation's solution as input for the other. When the Wigner function and potential stop changing (within specified convergence criteria), the self-consistent operating point has been reached. For the transient Gummel method,[6,7] the only mathematical difference is that the transient WFTE is used, so that each iteration is a time step. That is, we alternately time step the WFTE and update the potential using the PE until steady-state operation is reached (again, within specified convergence criteria). The transient Gummel iteration is initiated by solving the WFTE once in steady-state mode.

We now consider whether the direct or Newton form of the PE [i.e., zero or nonzero $\partial c/\partial u$ term in Eq. (26)] should be used for the steady-state and transient Gummel iteration methods. We have observed that a steady-state Gummel iteration often diverges [consecutive Wigner function and $u(x)$ solutions oscillate wildly] unless some corrective feedback is supplied through a nonzero $\partial c/\partial u$. Thus we must use the Newton PE for the steady-state Gummel method. In general, there is no exact, closed-form expression for $\partial c/\partial u$ for a quantum system. This is why we solve the WFTE—it accounts for quantum effects such as tunneling and reflection, along with nonequilibrium carrier transport, to relate the energy bands to carrier concentration. So we seek an approximate form for $\partial c/\partial u$ that is easy to compute but still produces convergence. To this end, the SQUADS uses the classical, equilibrium expression for $\partial c/\partial u$. Any justification must be based on the transport equation. In this case, we see that the boundary conditions supply carriers to the device according to the classical relationship, even though quantum processes and nonequilibrium transport will distort this relationship as the distance from the contacts increases. Also, scattering (if included) tends to produce the classical result.

The standard approach (see, e.g., Ref. 9) in deriving $\partial c/\partial u$ is to assume classical Maxwell-Boltzmann statistics:

$$c(u) = N_c \exp[(u-u_0)/(k_B T)], \tag{27}$$

$$\frac{\partial c}{\partial u} = \frac{c(u)}{k_B T}. \tag{28}$$

Equation (28) results in relatively slow but reliable convergence to the self-consistent operating point. Note, however, that the boundary conditions in Eqs. (3a) and (3b) are based on Fermi-Dirac statistics, not Maxwell-Boltzmann statistics. We have observed that using Fermi-Dirac statistics to derive $\partial c/\partial u$ can significantly accelerate the convergence speed of the steady-state Gummel method. SQUADS uses the Joyce-Dixon approximation[20] to relate c and u according to Fermi-Dirac statistics. To determine $\partial c/\partial u$, we write $u(c)$, derive $\partial u/\partial c$, and invert. Thus

$$r \equiv c/N_c, \tag{29}$$

$$u - u_0 = k_B T\left[\ln(r) + \sum_{m=1} a_m r^m\right], \tag{30}$$

$$\frac{\partial u}{\partial c}=\frac{\partial u}{\partial r}\frac{\partial r}{\partial c}=k_BT\left[\frac{1}{r}+\sum_{m=1}\frac{a_m}{m}r^{m-1}\right]\frac{1}{N_c}, \tag{31}$$

$$\frac{\partial c}{\partial u}=\frac{N_c}{k_BT}\left[\frac{1}{r}+\sum_{m=1}\frac{a_m}{m}r^{m-1}\right]^{-1}. \tag{32}$$

We have found that using Joyce-Dixon terms above $m=3$ does not improve convergence speed. In fact, in cases where $r_i \gg 1$ for one or more position nodes x_i, including higher-order terms, may render the steady-state Gummel method nonconvergent. Therefore, SQUADS uses a third-order Joyce-Dixon approximation by default. If the iterates P_i^n are not converging toward 0, we drop back to Maxwell-Boltzmann statistics (zeroth-order Joyce-Dixon approximation) until progress toward convergence is maintained for several iterations. The algorithm by which the Joyce-Dixon order is dynamically chosen to accelerate convergence of the steady-state Gummel method in SQUADS is now rather complicated, being based more on experience than theory. To our knowledge, only the standard (i.e., Maxwell-Boltzmann) form of $\partial c/\partial u$ has been used in previous steady-state Gummel iterations of the WFTE-PE system. In Sec. VI E we show that our accelerated convergence algorithm greatly decreases the computational cost of the steady-state Gummel iteration method.

In contrast to the steady-state Gummel method, the transient Gummel method seeks to follow the exact evolution of the device. Since there is no closed form for $\partial c/\partial u$ in a general quantum system, and because the approximations typically used (such as those used with the steady-state Gummel method) are only heuristically correct, using them in the transient Gummel method is more likely to create physics than model it. To avoid this, we must use the direct PE ($\partial c/\partial u=0$). For the transient Gummel method, then, each iteration starts with the exact potential profile for the carrier density at the current time point, the system is evolved one time step with the transient WFTE, and then the potential is adjusted for the new (but only slightly different) carrier density. We present the results of particular transient and steady-state Gummel simulations in Sec. VI.

V. FULL NEWTON APPROACH

With the Gummel approach to solving the WFTE-PE system, two independent (i.e., uncoupled) sets of linear equations are alternately solved, one derived from the WFTE and resulting in an updated Wigner function, and the second derived from the PE and producing an updated potential. With the full Newton formulation,[21] we instead solve a *combined* (i.e., coupled) WFTE-PE linear system to produce simultaneous updates of both the Wigner function and potential. The advantage of the full Newton approach is that changes in one solution directly affect the outcome of the other, so the corrective feedback that we had to approximate in the steady-state Gummel method is inherent in the Newton formulation. This tends to produce much faster convergence with a steady-state Newton method than with the steady-state Gummel method. Like the transient Gummel method, the transient Newton method seeks to follow the exact evolution of the quantum system, so it evolves to the steady-state operating point only as quickly as a real device would. However, the transient Newton method should be more accurate than the transient Gummel method, but by how much is not yet clear.

Use of the Newton formulation for quantum self-consistency[5] requires us to define a WFTE function $W(F)$, just as we defined the PE function in Eq. (21). For this purpose, we simply use Eq. (12),

$$W(F)\equiv(\mathbf{T}+\mathbf{K}+\mathbf{P}+\mathbf{S})[F]+(4/\Delta_t)f_{i,j,n}. \tag{33}$$

The Newton formulation for the WFTE-PE system solves the following system:

$$\begin{bmatrix}\frac{\partial W}{\partial F} & \frac{\partial W}{\partial u}\\ \frac{\partial P}{\partial F} & \frac{\partial P}{\partial u}\end{bmatrix}^{(n)}\begin{bmatrix}\delta F\\ \delta u\end{bmatrix}^{(n+1)}=-\begin{bmatrix}W(F)\\ P(u)\end{bmatrix}^{(n)}, \tag{34}$$

where the leftmost matrix is the Jacobian, and P is the Poisson function defined in Eq. (21). After each solution of Eq. (34), the unknowns are updated as

$$F^{(n+1)}=F^{(n)}+\beta(\delta F)^{(n+1)}, \tag{35a}$$

$$u^{(n+1)}=u^{(n)}+\alpha(\delta u)^{(n+1)}. \tag{35b}$$

As with the Gummel iteration methods, convergence toward the steady-state, self-consistent operating point with the Newton iteration methods is determined by monitoring the progress of the Poisson function $P^{(n)}(u)$, and update $\delta u^{(n)}$ iterates towards 0.

The update scaling factors α and β in Eqs. (35a) and (35b) are used only for the steady-state Newton method. Because the transient Newton method attempts to follow the transient operation of the device exactly, one must not modify the updates that are computed. Even for the steady-state Newton method, these update factors are ideally unity, though they can be reduced to some fraction when the iterates are not converging. Frensley[10] used $\alpha=0.5$ and $\beta=0.1$. However, for the simulations reported herein, in the few cases when the steady-state Newton method could not locate the self-consistent operating point, reducing α and β did not help, and in fact usually made convergence less likely. Thus the simulations in this work always used $\alpha=\beta=1$. Instead, where convergence was not occurring with the steady-state Newton method, SQUADS uses the steady-state Gummel method until the iteration begins converging again. Finally, since the Newton update in Eq. (35a) requires a Wigner function to update *from*, both steady-state and transient Newton simulations begin with a single steady-state Gummel solution of the WFTE. Initialization of the potential profile was discussed in Sec. III.

The full Newton equation (34) must be discretized for numerical solution. In discrete form Eq. (34) is

$$\begin{bmatrix}\frac{\partial W_{i,j}}{\partial F_{i',j'}} & \frac{\partial W_{i,j}}{\partial u_{i'}}\\ \frac{\partial P_i}{\partial F_{i',j'}} & \frac{\partial P_i}{\partial u_{i'}}\end{bmatrix}^{(n)}\begin{bmatrix}\delta F_{i,j}\\ \delta u_i\end{bmatrix}^{(n+1)}=-\begin{bmatrix}W_{i,j}(F)\\ P_i(u)\end{bmatrix}^{(n)}. \tag{36}$$

Expressions for $W_{i,j}$ and $P_{i,j}$ were given in Sec. III. The Jacobian blocks have yet to be determined. Actually, the Jacobian block for $\partial W/\partial F$ is identical to the coefficient matrix used for the WFTE solution of a Gummel iteration, although the unknowns that we solve for in the Newton formulation are $\delta F_{i,j,n}$ instead of $F_{i,j,n}$. The only difference between the Gummel WFTE coefficient matrix and the Newton $\partial W/\partial F$ Jacobian block is that terms which become boundary conditions with the Gummel formulation are zero with the Newton formulation, since δF^{bc} is zero. Thus these terms do not appear in the right-hand-side vector as in the Gummel methods.

The $\partial P/\partial u$ Jacobian block is also slightly different from the PE coefficients used in the Gummel formulation. In particular, with the Newton formulation, we do not have to attempt to approximate the effect of the change in potential on the carrier concentration through $\partial c/\partial u$. This relationship is taken care of exactly through the off-diagonal Jacobian blocks. The $\partial P/\partial u$ block is therefore the same as that used for the direct PE:

$$\frac{\partial P_i}{\partial u_{i'}} = (1+a)\,\delta_{i+1,i'} - 2\,\delta_{i,i'} + (1-a)\,\delta_{i-1,i'}\,. \tag{37}$$

The more interesting Jacobian blocks in this case are the off-diagonal ones, if only because expressions for them have (to our knowledge) never been published, although Frensley has used the steady-state Newton method to solve the WFTE-PE system.[10] The $\partial W/\partial u$ block is somewhat complicated, due to the convoluted way in which the u_i values enter into the computation of the nonlocal potential. Note from the relationship between the band edge U and the potential energy u in Eq. (5) that

$$\frac{\partial W_{i,j}}{\partial u_{i'}} = \frac{\partial W_{i,j}}{\partial U_{i'}}. \tag{38}$$

After some effort, the $\partial W/\partial u$ Jacobian block is

$$\frac{\partial W_{i,j}}{\partial u_{i'}} = \frac{-4\pi}{N_k h} \sum_{j''=1}^{N_k} f_{i,j''} \sin\left[\frac{2(i-i')(j-j'')\pi}{N_k}\right], \tag{39a}$$

$$(1 \leqslant |i'-i| \leqslant N_k/2). \tag{39b}$$

The Jacobian block for $\partial P/\partial f$ is much simpler. Recalling from Eq. (7) how carrier concentration c is calculated, we find from the definition of the discrete Poisson function in Eq. (25) that

$$\frac{\partial P_i}{\partial f_{i',j'}} = \frac{q^2 \Delta_x}{2\varepsilon N_k}\,\delta_{i,i'}\,. \tag{40}$$

Combining all of these results, Fig. 2 gives an example of the structure and size of the discrete full Newton equation for $N_x=7$ and $N_k=6$. Since the $\partial W/\partial f$ Jacobian block is identical in the Gummel and Newton formulations, and because this block is by far the largest in the Jacobian matrix, one might expect that solving the WFTE-PE system by the two approaches should require roughly the same storage and CPU time. This is not at all the case, especially in SQUADS, where the storage and solution of the discrete WFTE (and thus the $\partial W/\partial f$ Jacobian block) have been highly optimized. The result is that the Newton formulation requires typically twice the storage five times as much CPU time per loop as the Gummel formulation. We present performance data for all self-consistency iteration methods along with simulation results in Sec. VI.

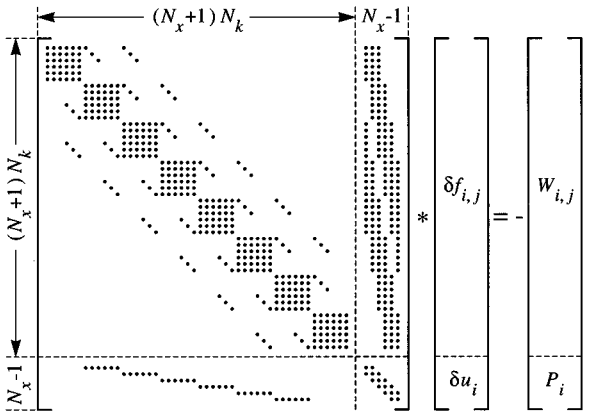

FIG. 2. Newton matrix equation for Wigner function method self-consistency: Jacobian matrix block sizes and nonzero coefficient structure for $N_x=7$ and $N_k=6$.

VI. RESULTS AND DISCUSSION

A. Simulated device and parameters

As stated previously, simulations in this work used the RTD device structure and simulation parameters of Jensen and Buot.[16] The simulated RTD, depicted in Fig. 3 at equilibrium, was composed of a 5-nm undoped GaAs quantum well between 3-nm undoped $Al_{0.3}Ga_{0.7}As$ tunnel barriers and 3-nm undoped GaAs spacer layers. The GaAs contact layers were 19 nm each, giving a total device width of $L=55$ nm. The electron effective mass was assumed constant at $0.0667m_0$, and the permittivity was also taken as constant at $12.9\varepsilon_0$. We used $N_x=86$, $N_k=72$, $\Delta_t=1$ fs, and $\tau=525$ fs (Ref. 22) at $T=77$ K.

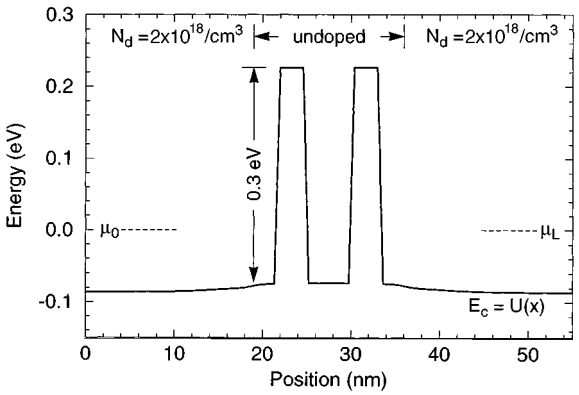

FIG. 3. Simulated GaAs RTD structure: equilibrium self-consistent conduction band, Fermi levels, and doping. The 0.3-eV $Al_{0.3}Ga_{0.7}As$ tunnel barriers are 3 nm thick, and the GaAs quantum-well width is 5 nm. The center 17 nm of the device (including 3 nm outside each tunnel barrier) are undoped.

B. Convergence criteria

The choice of convergence criteria for the WFTE-PE iteration presents a dilemma: too loose of criteria and the predicted self-consistent operating point is not trustworthy; too tight and the number of iterations required for convergence may rise dramatically. In this work, we chose to err on the side of too much computation rather than too little: our convergence criteria were relatively strict.

For steady-state simulations the proper convergence criterion is simply to verify that the (direct) Poisson equation is satisfied to a high degree. Thus we required that

$$P_i(u) < 10^{-8} \text{ eV} \quad (0 < i < N_x). \tag{41}$$

This convergence criteria, although necessary, was not sufficient in all cases. To assure that consecutive solutions were not oscillatory, and for steady-state Newton simulations where $P(u)$ is always very small (if update constant α is unity), we also required that the potential update at any point be very small:

$$\delta u_i < 10^{-6} \text{ eV} \quad (0 < i < N_x). \tag{42}$$

These relatively strict convergence criteria were feasible for the steady-state iteration methods because convergence tended to be very fast. Some researchers[9] have used criteria like Eq. (42) as their only indication of self-consistency, but this is not sufficient. It is possible, especially with an approximate iteration method such as the steady-state Gummel approach, for the potential updates to be small without actually having reached the self-consistent solution.

The convergence criteria in Eqs. (41) and (42) were also enforced for the transient iteration simulations in this work, but they are inadequate to guarantee that the steady-state, self-consistent operating point has been reached. The δu criterion is not especially revealing in a transient simulation because of the approximate proportionality of δu to the time step Δt. (A small time step gives little time for carriers to move, resulting in a correspondingly small change in the potential.) Also, because transient simulations tend to oscillate around the steady-state operating point as they relax toward it, satisfying the $P(u)$ criterion does not guarantee that a simulation has reached the steady state. A more definitive convergence criterion for transient simulations, also used by Jensen and Buot,[11] is based on the fact that the discrete current density for the WFTE is defined such that it is position independent at the steady state, as discussed in Sec. III. Thus a WFTE transient simulation can be said to have reached the steady state when the variation in current density δJ over the width of the device drops below some relatively small value. In this work, current densities were on the order of 10^5 A/cm^2, so our final transient simulation convergence criterion was

$$\delta J \equiv (J_{\max} - J_{\min}) < 1000 \text{ A/cm}^2. \tag{43}$$

This criterion was less strict than we would have liked, but tightening it would have led to much longer simulation times. Actually, when Eq. (43) was satisfied in a transient simulation, a steady-state simulation using the final potential profile usually differed from the actual steady-state result by less than 10 A/cm^2. Further, when it was necessary to verify controversial transient simulation results, we ran transient simulations in which all three convergence criteria were four orders of magnitude tighter.

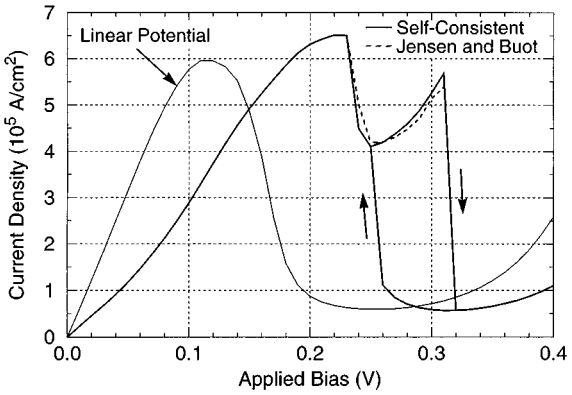

FIG. 4. Simulated RTD I-V curve using the Gummel and Newton steady-state self-consistency iteration methods. Jensen and Buot's up-trace (Ref. 11) (where different), and a non-self-consistent (linear potential) I-V curve are shown for comparison. (Permission to reprint data given by K. L. Jensen.)

C. Steady-state iteration method simulations

One purpose of this work was to examine when the (physically based) transient iteration methods are required to accurately reproduce the operation of an RTD, and when the computationally more efficient steady-state iteration methods may be used. The test device for this work was selected because of the very interesting I-V curve simulated by Jensen and Buot,[11] who used the transient Gummel method to implement self-consistency. Their simulations produced an I-V curve similar in shape to the experimental curve in Fig. 1 (although for a different RTD). In fact, they even observed persistent current oscillations for all biases in the plateau region of the I-V curve, concluding that ''intrinsic oscillations have a dominant influence on the plateaulike structure and hysteresis in the I-V characteristics.'' [11] Subsequent work by Buot and Rajagopal[23,24] described the physics behind this behavior.

Based on the results obtained by Jensen and Buot, it was not clear that the steady-state Gummel and Newton iteration method simulations would converge in the plateau region, since persistent oscillations indicate that no stable, self-consistent operating point exists. Although unstable equilibrium points should exist in this region, our otherwise convergent iteration methods could be rendered nonconvergent. In fact, both the accelerated Gummel and Newton simulations were unable to converge at some challenging points in the plateau. However, by automatically using the standard Gummel iteration method in these cases, the steady-state iteration methods did find self-consistent operating points over the entire simulated bias range. The resulting I-V curve (Fig. 4) was very similar to that of Jensen and Buot (also shown), and identical for the two steady-state iteration methods. The hysteresis loop in the I-V curve required the simulation of both the up-trace (0.0–0.4 V) and the down-trace (0.4–0.0 V).

It seems contradictory that the steady-state iteration methods found steady-state operating points in the plateau (0.24–

0.31 V on the up-trace and 0.25–0.24 V on the down-trace), while the transient simulation of Jensen and Buot did not. One possible explanation is that these oscillations, although persistent, are not perpetual. Jensen and Buot's conclusions seem to rule this out. If the oscillations are perpetual, the simultaneous WFTE and PE solutions found by the steady-state iteration methods must be unstable equilibrium operating points. Thus, given any impulse or even numerical noise, a system prepared according to the steady-state solution will begin to oscillate in a transient simulation. Determining whether one or both of these explanations are correct can only be accomplished with transient iteration simulations, which are described in Sec. VI D.

Before moving on to transient simulations, one conclusion can already be drawn based on the RTD I-V curves in Fig. 4. Also shown in Fig. 4 is a non-self-consistent simulated I-V curve for the RTD of Fig. 3. This simulation assumed a linear potential drop across the undoped (central) region of the RTD, and therefore did not require solution of the PE, and only a single WFTE solution per bias point. Comparing these simulated I-V curves with the experimental one in Fig. 1 (for a different RTD structure), we see that the linear potential simulation was able to predict a negative differential resistance region, but that is about the limit of its usefulness. On the other hand, the similarity between the simulated self-consistent I-V curve and the experimental curve clearly shows that enforcing self-consistency is necessary to reproduce some of the salient physics of real RTD's. The open question is whether the computationally expensive transient iteration methods can add any further detail.

D. Transient iteration method simulations

To compare self-consistency iteration methods, and now to investigate the nature of the plateau operating points, we used the transient Gummel iteration method to simulate the I-V curve of the RTD in Fig. 3 over the same bias range as for the steady-state simulations. We set a maximum limit of 4000 iterations (4 ps) per bias point. If the transient simulation did not converge in this time (e.g., due to sustained oscillations), the simulation moved to the next bias point anyway. Surprisingly, although the current oscillations observed by Jensen and Buot did occur in the plateau region, the simulation converged for all bias points except the first three in the plateau (0.24, 0.25, and 0.26 V). Further, the resulting I-V curve (except for those three points) was indistinguishable from the steady-state curve, as one would expect (assuming the convergence criteria are strict enough).

We noticed that the oscillations in the plateau region were progressively more persistent at lower biases. Whereas only 1300 iterations were required to reach convergence at 0.31 V, fully 3800 iterations were required at 0.27 V. The 0.26-V bias point was apparently on course to convergence at 4000 iterations. Indeed, further evolution resulted in full convergence after a total of 7008 iterations. To demonstrate these oscillations, Fig. 5 shows the complete plot of collector current versus time at 0.26 V on the up-trace. Both the oscillations and the convergence criteria decreased very regularly over the course of the simulation, with a decay constant of 0.2/ps. For example, for the oscillation amplitude,

$$A(t) \approx 0.8 \times 10^5 e^{-(0.2t/1\ \mathrm{ps})}\ \mathrm{A/cm^2}. \tag{44}$$

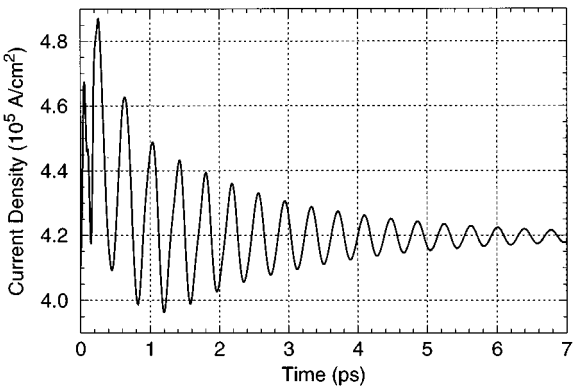

FIG. 5. Simulated transient collector current as RTD evolves to the steady state after switching from 0.25 to 0.26 V, showing that the RTD is stable at this bias.

Although the ultimate fates of the remaining points, 0.24 and 0.25 V on both curve traces, were inconclusive after 4000 iterations, we expected, extrapolating from the results and trends for the other plateau points, that their oscillations would simply be even more persistent, but not perpetual.

Our belief that the transient RTD simulation would eventually reach the steady state for 0.24 and 0.25 V turned out to be incorrect. Further evolution (in either curve-trace direction) led to oscillations of constant amplitude by about 8000 iterations at both biases. For example, Fig. 6 shows the transient current at 0.24 V on the up-trace. We allowed these simulations to run for several thousand more iterations to make certain that the oscillations were not slowly decreasing, as we had expected. Data on the final oscillations at these two points (independent of the trace direction) are given in Table I.

We used one additional test to assure that the 0.24- and 0.25-V bias points were unstable. As suggested in Sec. VI C, we ran transient Gummel simulations starting from the fully converged steady-state Gummel solution at 0.24 and 0.25 V, expecting them to diverge (oscillations to build). This was indeed the result. The collector current versus time for the 0.24-V simulation is shown in Fig. 7. The result for 0.25 V was similar. Divergence was very regular, with a decay constants of -0.4/ps at 0.24 V and -0.2/ps at 0.25 V. For the oscillation amplitude at 0.24 V,

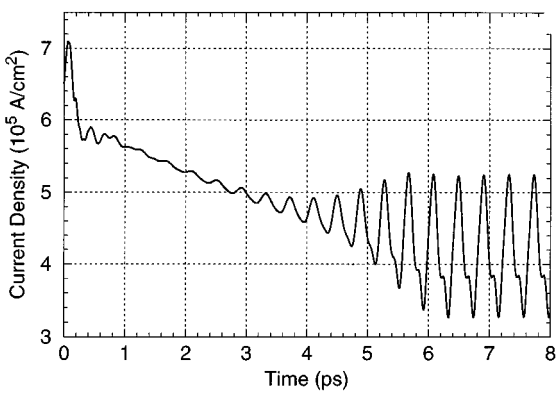

FIG. 6. Simulated transient collector current after switching from 0.23 to 0.24 V, showing sustained oscillations.

TABLE I. Collector current final oscillation data (after 10 ps) at applied biases of 0.24 and 0.25 V. Current density from steady-state simulations is appended for comparison.

Oscillation parameter	0.24 V	0.25 V
Amplitude (10^5 A/cm^2)	1.98	1.08
Period (ps)	0.413	0.374
Frequency (THz)	2.42	2.67
Time average (10^5 A/cm^2)	4.18	4.06
Steady-state current (10^5 A/cm^2)	4.50	4.10

$$A(t)\approx 6.31e^{(0.4t/1\ \mathrm{ps})}\ \mathrm{A/cm^2}. \tag{45}$$

Of course, the oscillation amplitude will be bounded, just as it was in Fig. 6. These results prove that the RTD is inherently unstable at these biases. To model this behavior, Buot and Jensen describe an equivalent circuit model for the RTD (Ref. 25) that reproduces the bounded instability depicted in Figs. 6 and 7.

The results at 0.24 and 0.25 V called into question our conclusion that the remainder of the plateau was stable. The convergence criterion in Eq. (43) is admittedly not as strict as we would like. It leaves open the possibility that the RTD might oscillate perpetually with an amplitude of less than 1000 A/cm^2. To verify that the upper portion (0.26–0.31 V) of the plateau was stable, we ran simulations at the lower end (0.26 V), middle (0.29 V), and top (0.31 V) of this region with four orders of magnitude stricter convergence criteria. Most importantly, the current variation was required to be less than 0.1 A/cm^2 for convergence. Throughout these simulations, the oscillations continued to decay regularly at all three bias points, reaching convergence at 27 906, 10 424, and 7522 iterations, respectively. To illustrate, Fig. 8 shows a plot of the current variation versus time for the 0.26-V simulation.

Based on the above transient simulations, we can now conclude that the plateau in the simulated RTD's I-V curve is composed of two parts: an unstable region (0.24–0.25 V) in which the RTD oscillates forever, and a stable region (0.26–0.31 V) where persistent oscillations eventually die out. Actually, these regions are simply the result of a monotonic increase in the exponential decay constant [see Eqs. (44) and (45)] from -0.4/ps at 0.24 V, through 0 at about 0.255 V, and up to about 0.67/ps at 0.31 V. The unstable region agrees with Jensen and Buot's results showing perpetual oscillations in the plateau, while the stable region contradicts their conclusion that these oscillations occur throughout the plateau and are required for the plateau to occur. In fact, these oscillations have only a minor effect on the value of the I-V curve in the unstable region of the plateau (see Table I), and no effect at all elsewhere. We suspect that Jensen and Buot's incorrect conclusions resulted either from premature termination of their transient simulations, or from their use of an accelerated convergence technique.[11]

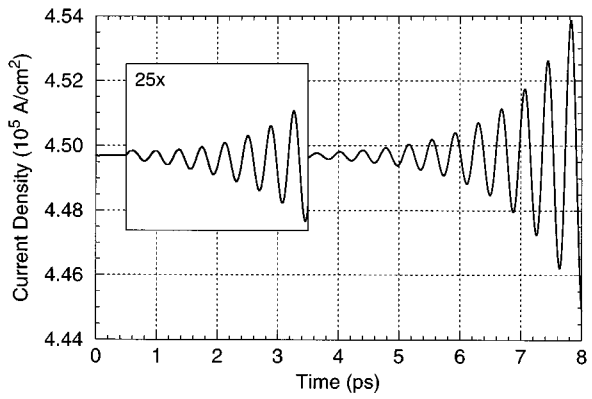

FIG. 7. Simulated transient collector current starting from a fully converged steady-state Gummel iteration simulation at 0.24 V, showing that the RTD is unstable at this bias.

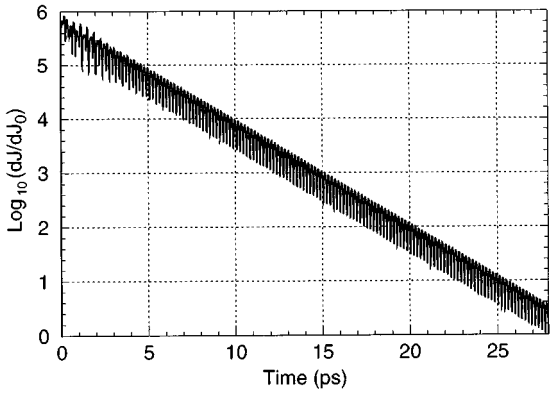

FIG. 8. Current variation vs time after switching from 0.25 to 0.26 V. dJ is the current variation, and dJ_0 is the convergence criterion of 0.1 A/cm^2. The simulation converges regularly, showing that the RTD is stable at this bias. The spikes in the curve are due to the decaying oscillations.

In the above discussion of transient self-consistency simulations, we did not mention the transient Newton iteration method. In fact, we only ran partial I-V curve traces (5–10 points in either direction and some plateau region points) using this iteration method. Based on these simulations, we determined that the RTD evolved almost identically with transient Newton method as with the transient Gummel method. For example, Fig. 9 compares the collector current from the I-V curve simulations at 0.06 V for the two transient iteration methods. Although the transient Newton method sometimes converged a few iterations faster, for the bias point shown in Fig. 9, the transient Gummel and Newton methods converged in exactly the same number of iterations (629).

We concluded from these observations that performing a full I-V curve trace with the transient Newton method would provide no additional information. Thus, although in theory the transient Newton approach is more accurate than the transient Gummel approach, for the relatively small time step used here, the improvement in accuracy was found to be equally small. Another reason we did not complete the transient Newton I-V curve simulation was, as we discuss in Sec. VI E, that it would have required an unreasonable amount of CPU time.

E. Computational efficiency

We have shown that essentially identical I-V curves are produced for the RTD in Fig. 3 by all four self-consistency

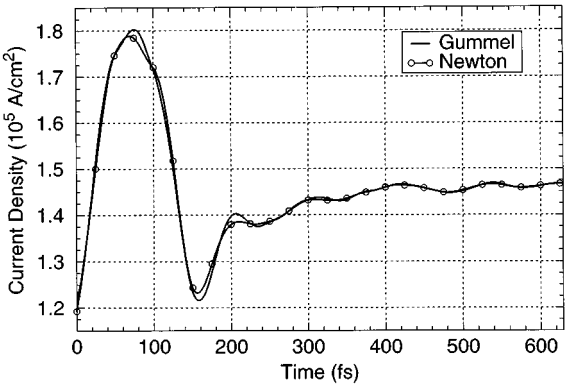

FIG. 9. Simulated collector current for transient Gummel and Newton iteration method simulations after switching from 0.05 to 0.06 V. This indicates that the Gummel approach is effectively as accurate at the Newton approach for the chosen simulation parameters.

iteration methods. It is reasonable in such a case to use the most efficient iteration method. Thus the relative efficiencies of the iteration methods is another main point of comparison. As one can surmise from the foregoing discussions, the computational costs of the four iteration methods are vastly disparate. The number of WFTE solves and total CPU time used by each of the iteration methods for the two-trace *I-V* curve is summarized in Table II. Data for the non-self-consistent simulation shown in Fig. 4 are also given for comparison. We have also included data for the standard steady-state Gummel implementation (see Sec. IV), for comparison to the accelerated implementation used in this work.

Some notes regarding the data in Table II are in order. The current was simulated at 0.01-V bias increments in both directions over the range 0.0–0.4 V, giving a total of 82 bias points plus the equilibrium solution needed for scattering calculations. The 140 steady-state Gummel iterations done during the course of the steady-state Newton simulation were a result of the Newton method's inability in some cases to locate the self-consistent operating point as it entered or exited the plateau region. The transient simulations used 100-fs bias slewing (rather than changing the applied bias in a single time step) to mitigate the ''shock'' of bias changes and thus to minimize convergence time. The transient simulations further assume that the four oscillating operating points (0.24 and 0.25 V in both trace directions) were terminated at 8000 iterations, while all other bias points were run to full convergence. Since we did not conduct a complete transient Newton *I-V* curve simulation, the data in Table II for this iteration method are estimates, but should be very close, based on the arguments at the end of Sec. VI D.

TABLE II. Number of WFTE solves and total CPU time required for a two-trace $I-V$ curve simulation for each self-consistency iteration method. Data are given for both the standard and accelerated steady-state Gummel approaches. The steady-state Newton simulation required several Gummel loops in some difficult cases. The transient Newton data are estimated. CPU times are for a DEC Alpha 3000/300 LX.

Simulation type (Iteration method)	WFTE solves (i.e., iterations)	CPU time (h)
Linear (non-self-consistent)	84	0.28
Steady-state Gummel (std)	4300	14.3
Steady-state Gummel (acc)	1450	5.0
Steady-state Newton	$410N+140G$	7.2
Transient Gummel	96 500	330
Transient Newton	~96 500	~1,650

We note that the simulations for this work were carried out on several platforms. The *I-V* curves for which data are reported in Table II were produced on independent processors of an SGI Challenge XL computer and on DEC Alpha 3000/300LX workstations. These platforms were roughly equivalent in performance, requiring about 12 CPU s per Gummel loop and 60 s per Newton loop. A Cray C-90 supercomputer was used for the longer, single-bias investigations (e.g., the detailed investigations at 0.24 and 0.25 V). The Cray required only 1.05 CPU s per Gummel loop.

Several factors determine the relative computational costs of the self-consistency iteration methods. Considering just the steady-state Gummel simulations, the importance of using our accelerated convergence implementation (see Sec. IV) is clear. In fact, the CPU time advantage of using Fermi-Dirac statistics is often even more dramatic than the roughly 3:1 ratio shown in Table II. Outside the plateau region, the average number of iterations required for convergence to the self-consistent solution was 41 using the standard approach, but only seven using our accelerated approach. However, for all iteration methods, most of the iterations took place in the challenging plateau region of the *I-V* curve. (One result of this was that the up-trace always took more CPU time than the down-trace.) For the accelerated Gummel simulation, locating operating points in the plateau often required dropping back to the more reliable standard approach. The result was only a 2.2:1 advantage in CPU time over the standard approach in the plateau region. With its faster convergence, the advantage of the accelerated Gummel implementation increases as convergence criteria become more strict.

A more general factor influencing the relative computational costs of the self-consistency iteration methods is the much greater CPU time required for a Newton loop than a Gummel loop. In this work, the ratio was 5:1. In spite of this, the full steady-state Newton simulation required only 44% more CPU time than the accelerated steady-state Gummel simulation, and only half the time of the standard Gummel simulation. This recoup by the steady-state Newton method was a result of yet another factor in the efficiency equation: the Newton method's more sophisticated solution update algorithm (see Sec. V), meaning that fewer iterations were required for convergence. In spite of the strict convergence criteria used, aside from the plateau region, almost all bias points required only three steady-state Newton iterations to meet these criteria. (The relatively low number of iterations required by both steady-state iteration methods was made possible by the initialization algorithm for u^0, as discussed in Sec. III.) Again, the faster convergence of the steady-state Newton approach improves its favorability in comparison to the steady-state Gummel approach as convergence criteria become more strict.

By far the most significant factor in the computational cost equation is whether the iteration method uses the steady-

state or transient approach to finding the self-consistent operating point. The mathematical descriptions, and thus the CPU time per iteration, of the steady-state and transient methods are very similar for each formulation (Gummel or Newton). However, Table II shows that the transient iteration methods require roughly two orders of magnitude more iterations (on average) than the steady-state methods to converge to the self-consistent operating point. The reason for the huge difference is that the transient iteration methods attempt to follow the exact evolution of the device as it relaxes towards the steady state after a bias change, so they must take as long (in simulation time) as a real device would to reach the steady state. Because of the extreme computational cost of the transient iteration method, to complete the transient Gummel simulation in an acceptable amount of real time, we ran several sections of each trace concurrently, using a steady-state Gummel-converged solution for the initial condition (except for the two points on each trace which did not converge).

F. Discussion

In this section, we discuss the strengths and weaknesses, in terms of efficiency, accuracy, and robustness, of the four self-consistency iteration methods considered in this work. From previous sections, the obvious strength of the steady-state methods is their relative computational efficiency. As we have also stated, the main strength of the transient methods is their direct physical basis, and their resulting ''exact'' adherence to the time-dependent operation of the device being simulated. These are clearly complementary strengths, so that both the steady-state and transient approaches have important uses. In particular, we recommend using a steady-state iteration method for wide-ranging initial investigations (e.g., to trace the I-V curve), thereby gaining the insight necessary to narrow the focus of a more detailed investigation where transient effects are inherent (e.g., switching) or suspected (e.g., oscillations). Strangely, we find the literature roughly equally divided between use of transient and steady-state Gummel approaches, with apparently no group simultaneously using the information and advantages provided by both. Hopefully this work will help to end that unnecessary exclusivity.

If a main strength of the steady-state methods is their relative efficiency, their main shortcoming, at least in some cases, is accuracy. The inability of the steady-state iteration methods to show the transient oscillations predicted by the transient iteration methods was to be expected: only transient simulations can model time-dependent effects. Much more of a concern was the fact that the steady-state methods offered no concrete indication that an unstable operating condition existed, and thus that a transient simulation should be used. For the simulations in this work, if we had not known to look for oscillations in the plateau, we would have been perfectly satisfied that our steady-state simulations told the entire story about the RTD's I-V curve. Admittedly, the actual I-V curve was only slightly different at two points, but the physics underlying those small differences was quite important.

Another shortcoming of the steady-state iteration methods is that convergence to a simultaneous solution of the steady-state WFTE and the PE cannot be guaranteed. There are several potential causes of this lack of ''robustness'' or reliability. First, there are almost certainly ''pathologic'' operating conditions for some quantum devices where the steady-state methods will be unable to converge. Even if a device is stable at a given bias, the operating point may not be found if the previous WFTE and PE solutions are far away from it. Incrementing the bias across a bistable operating point, of which there are three in Fig. 4, is the usual culprit here. Bistable operating points were, in fact, problematic for both the steady-state Newton method and the (accelerated) steady-state Gummel method. However, SQUADS detects nonconvergent behavior during steady-state self-consistent simulations and automatically switches (temporarily) to the standard (and more robust but slower) steady-state Gummel approach. In this way, potential divergence problems of the steady-state iteration methods were completely avoided in this work.

Just as blind faith in the results of steady-state self-consistent simulations is not advisable, so too is complete reliance on transient self-consistent simulations. Admittedly, the basic transient methods are always adequate in terms of reliability and accuracy (i.e., the ability to correctly reproduce device physics). However, their extreme computational cost has some harsh consequences. The first is that one cannot afford to undertake transient simulations such as those presented in this work without a good reason (and a very fast computer). The problem with this is that often there *is* no concrete reason *a priori* for running a simulation—only a vague notion of how the device might behave. Certainly it is currently completely unfeasible to run multiple week-long transient self-consistent I-V curve simulations to examine the effects of varying simulation or device parameters. In contrast, the decision to run the same steady-state simulations (in a few hours each) hardly merits a second thought.

The opposite side of the tendency for doing too few transient self-consistent simulations is trying to do too many. A good reason to limit reliance on transient simulation where appropriate is that inadequate computing resources invite unnecessary compromises to be made in the implementation of the simulator or in the execution of the simulation. For example, fewer bias points or time steps may be simulated than necessary, the time step or convergence criteria may be larger than accuracy dictates, and so on. One compromise we made that seems justified (as discussed in Sec. IV D) was the use of the transient Gummel method instead of the theoretically more accurate Newton method. On the other hand, our choice of slew rate based solely on achieving fast convergence, rather than modeling reality, is not so easily excused. In fact, investigations using a lower slew rate[17] show that transient current predictions like that in Fig. 5 may bear little resemblance to what a real RTD would do under test. However, in a circuit of RTD-like devices, 100-fs bias slewing may be reasonable. Since generating the I-V curve was the test case for this work, the details of the evolution to the steady state could be ignored in this case. In general, any compromises in implementation or execution should be considered carefully, so that they do not conspire to weaken the direct physical link which is the main advantage of the transient iteration methods over the steady-state approaches. The best defense against these compromises is to focus comput-

ing resources on a limited set of transient simulations that are expected to add value to the steady-state results.

We have advocated using the various self-consistency iteration methods in a hierarchical manner. An efficient steady-state approach should be used to investigate a broad range of operating conditions, and to narrow the scope for more exacting (and expensive) transient simulations. We now discuss the clues from steady-state simulations that indicate device operating conditions for which transient simulation might be warranted (i.e., where sustained, significant, or interesting transient effects might occur). Some of these clues are obvious. A negative differential resistance region is a known cause of oscillations, whether intrinsic to the device or a result of the device interacting with the (simulated or real) measurement apparatus. Also, any operating point at which the steady-state simulation has significant difficulty converging should raise a red flag. Obviously, if the steady-state iteration method completely fails to converge at a particular bias point, a transient simulation is necessary to determine device operation. Finally, only a transient iteration method can be used for inherently transient self-consistent simulations, such as switching, small-signal, or large-signal investigations.

G. Other iteration methods

As a final note, the Newton and Gummel methods presented above are certainly not the only possible ways to solve the WFTE-PE system and thereby implement self-consistency, although they are perhaps the most basic. Many variations on the Gummel and Newton methods are possible,[26] and other nonlinear system solving approaches may be used. For example, Jansen, Farid, and Kelly[27] used the conjugate-gradient method to compute the self-consistent I-V curve for a RTD. According to their analysis, this method is about an order of magnitude faster than the transient Gummel approach, making it about an order of magnitude slower than the steady-state Gummel and Newton iteration methods described herein. However, the conjugate-gradient method has the distinct advantage of a much smaller memory footprint. This would be useful for very large simulations (e.g., $N_x, N_k > 200$). Since memory usage for solving the WFTE-PE system has been highly optimized in the SQUADS, the conjugate-gradient method has not been considered necessary for our purposes.

VII. SUMMARY

We reviewed the theory and numerical implementation of four basic approaches to implementing self-consistency in the Wigner function approach to quantum device simulation. These approaches include steady-state and transient Gummel, and steady-state and transient Newton. To our knowledge, this is this first time that all these approaches have been described in a single mathematical framework and notation. In the process of describing the numerical implementations of these iteration methods, we gave expressions for the off-diagonal Jacobian blocks in the Newton formulation, apparently for the first time. We also presented an accelerated convergence algorithm for the steady-state Gummel approach which makes it the most efficient means of generating the self-consistent I-V curve for a RTD.

We also analyzed the strengths and weaknesses of the various self-consistency iteration methods. A large part of that analysis concerned relative computational costs. The computational efficiency of the steady-state methods makes them ideal for wide-ranging initial investigations, such as full I-V curve traces. There are undeniable difficulties in using the steady-state iteration methods, such as lack of robustness in the Newton and accelerated Gummel methods, and the relatively slow convergence of the standard Gummel approach. These problems may have discouraged the use of steady-state approaches in the past. We have demonstrated how these problems can be avoided, and we have shown the excellent results and efficiencies that the steady-state iteration methods can achieve.

We have also shown that even if a steady-state iteration method converges to a simultaneous solution of the steady-state WFTE and PE, there is no guarantee that this is a stable operating point. Transient iteration methods are inherently more accurate and reliable, and are required to treat time-dependent situations (such as unstable oscillations). However, we have shown that steady-state methods are just as important *in practice* in the investigation of quantum device physics. Efficient steady-state simulations can be used to determine the basic operation of the device (e.g., the I-V curve, possible unstable regions), allowing one to narrow the scope of (expensive) transient simulations. Those transient simulations which *are* done can then be implemented and executed without serious compromises so that they will correctly model device physics and add value to the steady-state results.

ACKNOWLEDGMENTS

B.A.B. would like to thank Dr. Kevin Jensen for his valuable input and technical assistance, especially in helping to identify an elusive bug in SQUADS, and for his critical reading of this manuscript. Consultations with Dr. Kiran Gullapalli and Dr. Zhiping Yu were also appreciated. Support for this research was provided in part by the Joint Services Electronics Program under Contract No. DAAL03-91-C-0010, and by the Computational Prototyping Program at Stanford University.

[1] E. Wigner, Phys. Rev. **40**, 749 (1932).
[2] V. I. Tatarskii, Usp. Fiz. Nauk **139**, 587 (1983) [Sov. Phys. Usp. **26**, 311 (1983)].
[3] M. Hillery, R. F. O'Connel, M. O. Scully, and E. P. Wigner, Phys. Rep. **106**, 121 (1984).
[4] W. R. Frensley, Phys. Rev. B **36**, 1570 (1987).
[5] W. R. Frensley, Rev. Mod. Phys. **62**, 745 (1990).
[6] F. A. Buot and K. L. Jensen, Phys. Rev. B **42**, 9429 (1990).
[7] N. C. Kluksdahl, A. M. Kriman, D. K. Ferry, and C. Ringhofer, Phys. Rev. B **39**, 7720 (1989).
[8] K. K. Gullapalli, D. R. Miller, and D. P. Neikirk, Phys. Rev. B **49**, 2622 (1994).

[9] H. Tsuchiya, M. Ogawa, and T. Miyoshi, IEEE Trans. Electron Devices **38**, 1246 (1991).
[10] W. R. Frensley, in *Computational Electronics: Semiconductor Transport and Device Simulation*, edited by K. Hess, J. P. Leburton, and U. Ravaioli (Kluwer, Boston, 1991), p. 195.
[11] K. L. Jensen and F. A. Buot, Phys. Rev. Lett. **66**, 1078 (1991).
[12] L. L. Chang, L. Esaki, and R. Tsu, Appl. Phys. Lett. **24**, 593 (1974).
[13] E. R. Brown, T. C. L. G. Sollner, C. D. Parker, W. D. Goodhue, and C. L. Chen, Appl. Phys. Lett. **55**, 1777 (1989).
[14] B. Ricco and M. Ya. Azbel, Phys. Rev. B **29**, 1970 (1984).
[15] E. R. Brown, W. D. Goodhue, and T. C. L. G. Sollner, J. Appl. Phys. **64**, 1519 (1988).
[16] K. L. Jensen and F. A. Buot, IEEE Trans. Electron Devices **38**, 2337 (1991).
[17] B. A. Biegel and J. D. Plummer (unpublished).
[18] M. R. Pinto, Ph.D. thesis, Stanford University, 1990, p. 371.
[19] H. K. Gummel, IEEE Trans. Electron Devices **11**, 455 (1964).
[20] W. B. Joyce and R. W. Dixon, Appl. Phys. Lett. **31**, 354 (1977).
[21] M. R. Pinto (Ref. 18), p. 304.
[22] K. L. Jensen and F. A. Buot, J. Appl. Phys. **67**, 7602 (1990).
[23] F. A. Buot and A. K. Rajagopal, Phys. Rev. B **48**, 17 217 (1993).
[24] F. A. Buot and A. K. Rajagopal, Mater. Sci. Eng. B **35**, 303 (1995).
[25] F. A. Buot and K. L. Jensen, COMPEL **10**, 241 (1991).
[26] M. R. Pinto (Ref. 18), Chap. 5.
[27] R.-J. E. Jansen, B. Farid, and M. J. Kelly, Physica B **175**, 49 (1991).