Solution of the Wigner-Poisson Equations for RTDs

M. S. Lasater[1], C. T. Kelley[1], A. G. Salinger[2], D. L. Woolard[3], and P. Zhao[4]

[1]Center for Research in Scientific Computation and Department of Mathematics
[4]Electrical and Computer Engineering Department
North Carolina State University, Raleigh, North Carolina, 27695-8205, USA

[2]Sandia National Laboratories P.O. Box 5800, MS-1111
Albuquerque, New Mexico, 87185, USA[1]

[3]U. S. Army Research Office
U. S. Army Research Laboratory, RTP, North Carolina, 27709-2211, USA

Email: tim_kelley@ncsu.edu Tel.: 1-919-515-7163

ABSTRACT

We will discuss a parametric study of the solution of the Wigner-Poisson equations for resonant tunneling diodes. These structures exhibit self-sustaining oscillations in certain operating regimes. We show numerically that the phenomenon corresponds to a Hopf bifurcation, using the bias across the device as a continuation parameter. We will describe the engineering consequences of our study and how it is a significant advance from some previous work, which used much coarser grids. We use the LOCA package from Sandia National Laboratory. This package, and the underlying NOX and Trilinos software, enable effective parallelization. We report on the scalability of our implementation.

Keywords: Wigner-Poisson Equations, Resonant Tunneling Diode, Hopf Bifurcation, Continuation.

1. INTRODUCTION

Semiconductor technology has developed to the point where the next generation of electronic devices will operate at the atomic level. Since the device scale is so small, design problems arise immediately. Currently, we do not have the technology to observe and collect all relevant data from such small devices. Furthermore, even if we had this capability, the device physics are determined by quantum mechanics and not by classical electromagnetism. A fundamental result of quantum mechanics is that the act of observing a quantum system will have an impact on the results we obtain. Thus, physically measuring how a normal quantum system is functioning would require an account of the effects of the observer on the reported data. To avoid this issue, engineers and physicists researching these quantum devices are working to develop an accurate model of these quantum systems from first-principle physics. One particular nanostructure we are interested in is the resonant tunneling diode (RTD).

A RTD is created by taking a slab of semiconductor and placing a second kind of semiconductor (one that has a larger band-gap) into this semiconductor. Since the second type has a larger band-gap, this effective creates potential barriers within the structure. Figure 1 is a diagram of an RTD.

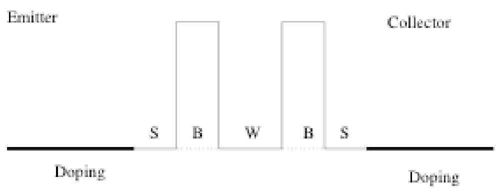

Figure 1: Diagram of RTD

The second type of semiconductor is represented by the dotted lines in the diagram. The potential barriers are also shown in the diagram and are represented by (B). Between the two barriers is a section of the original semiconductor. This is the quantum well (W) that is contained between the two barriers. Far from the barriers, the original semiconductor is doped (represented by the darker lines). Doping is where atoms that contain more (or less) electrons that the semiconductor itself are embedded into the semiconductor to create (or take away) extra electrons in the structure. Between the barriers and the doped regions are areas where the original semiconductor exists. These areas are called spacers (S).

Classically, if a particle runs into a potential barrier and it does not have enough speed, it will be reflected back. Since quantum mechanics treats electrons as waves instead of particles, an electron at any speed that encounters a barrier still has some probability of passing through the barrier. This effect is known as "quantum tunneling" and is the basis of this device. If a voltage difference is applied across the device, electrons will start to move along the device, tunnel through the barriers, and reach the other side, thus creating a current.

Numerical simulations [1], [2] have shown that current

[1] Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed-Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000

oscillation can be expected for certain voltage differences, and that these current oscillations have a high frequency in the terahertz (THz) regime. With these numerical simulations, engineers and physicists are hoping to understand what physical mechanism creates these intrinsic oscillations and determine what physical parameters (i.e. doping profile, barrier height and width, well width, etc.) are conducive to sustaining and controlling these oscillations in hopes of producing a viable high frequency power source. This work is attempt to create a faster and more accurate RTD simulator to aid the engineers in these goals.

2. MODEL DESCRIPTION

The model used to describe the electron transport in these devices is the Wigner-Poisson equations [3]. These equations consist of a nonlinear PDE that describes the time-evolution of the distribution of the electrons in the device coupled with Poisson's equation which incorporates the potential effects of the electrons into the model. The first of these equations can be given by

$$\frac{\partial f}{\partial t} = W(f) = K(f) + P(f) + S(f) \qquad (1)$$

Here, $f=f(x,k,t)$, is the distribution of the electrons. It is a function of the position of the electron, x, the momentum of the electron, k, and time, t. The position variable x ranges from 0 to L, the length of the device, and the momentum variable k ranges from $-\infty$ to ∞. The time-derivative of f is comprised of three terms. The first term, $K(f)$, represents the kinetic energy effects on the distribution and is given by

$$K(f) = \frac{-hk}{2\pi m^*}\frac{\partial f}{\partial x} \qquad (2)$$

Here, h is Planck's constant and m^* is the effective mass of the electron. The second term, $P(f)$, is the nonlinear term in the equation and is for the potential energy effects on the distribution

$$P(f) = \int_{-\infty}^{\infty} f(x,k')T(x,k-k')dk' \qquad (3)$$

where $T(x,z)$ is given by

$$T(x,z) = 4\int_{0}^{L_c/2}[U(x+y) - U(x-y)]\sin(2yz)dy$$

(4)
In this equation, $U(x)$ is the electric potential as a function of position, and L_c is the coherence length. $P(f)$ is nonlinear in f since $U(x)$ depends on f through Poisson's equation. The final part of the time-derivative describes the scattering processes that occur between the electrons

$$S(f) = \frac{1}{\tau}\left[\frac{\int_{-\infty}^{\infty} f(x,k')dk'}{\int_{-\infty}^{\infty} f_0(x,k')dk'} f_0(x,k) - f(x,k)\right] \qquad (5)$$

Here, τ is the relaxation time, and $f_0(x,k)$ is the equilibrium Wigner distribution. This is the steady state solution to Eq. (1) when there is no voltage difference across the device. The boundary conditions for f impose the incoming electron distributions. That is, at x=0 and for $k > 0$ (electrons with positive momentum that are moving right) we have

$$f(0,k) = \frac{4\pi m^* k_B T}{h^2}\ln\{1 + \exp[\frac{-1}{k_B T}(\frac{h^2 k^2}{8\pi^2 m^*} - \mu_0)]\}$$

and at $x=L$ and for $k < 0$ (electrons with negative momentum that are moving left) we have

$$f(L,k) = \frac{4\pi m^* k_B T}{h^2}\ln\{1 + \exp[\frac{-1}{k_B T}(\frac{h^2 k^2}{8\pi^2 m^*} - \mu_L)]\}$$

k_B is Boltzmann's constant, T is the temperature, μ_0 is the Fermi energy at x=0, and μ_L is the Fermi energy at $x=L$.

The electric potential $U(x)$ is made up of two parts. The first part is from the electrostatic potential created by the electrons in the device. We will denote this part by $u(x)$. The second part is from the potential barriers in the device created from the heterojunction of the two different semiconductor materials. We will denote this part by $\Delta(x)$. To get $u(x)$, we must solve Poisson's equation

$$\frac{d^2u}{dx^2} = \frac{q^2}{\varepsilon}[N_d(x) - \frac{1}{2\pi}\int_{-\infty}^{\infty} f(x,k')dk'] \qquad (6)$$

q is the charge of the electron, ε is the dielectric constant, and $N_d(x)$ is the doping profile. The boundary conditions for Poisson's equation are where the voltage difference across the device is incorporated. We have that

$$u(0) = 0, u(L) = -v \qquad (7)$$

where $v \geq 0$ is the applied voltage. Once we have solved for $u(x)$, we have $U(x) = u(x) + \Delta(x)$.

3. DISCRETIZATION

To numerically solve for the distribution, we discretize both the domain and equations using a finite difference method. For the x-domain, we use N_x grid points where $x_i = (i - 1)*\Delta\_x$, $i = 1,2,\ldots, N_x$ and $\Delta\_x = L/(N_x - 1)$. These grid points are evenly spaced across $[0, L]$. For the k-domain, we first truncate from $-\infty$ to ∞ to $-K_M$ to K_M, where K_M is a maximum momentum we consider. We use N_k grid points where $k_j = (2*j - N_k - 1)*\Delta\_k/2$, $j = 1,2,\ldots, N_k$ and $\Delta\_k = 2*K_M/N_k$. These grid points are evenly spaced across $(-K_M, K_M)$. So numerically we want to compute an approximation to the distribution at each grid point. That is for each $i = 1,2,\ldots, N_x$ and $j = 1,2,\ldots, N_k$, calculate a f_{ij} such that $f_{ij} \approx f(x_i, k_j)$.

To approximate the spatial derivative term in Eq. (2), we use a second-order upwind differencing scheme. For the integral terms in Eqs. (3) and (4), we use the midpoint rule in k and the trapezoid rule in x for their approximations. Finally, for solving Poisson's equation, we use a standard three-point central differencing scheme. This discretization converts the continuous nonlinear PDE problem to a nonlinear ODE for the solution for f at the grid points.

4. CONTINUATION METHODS

We are interested in studying the steady-state Wigner distribution, *f*, of the Wigner-Poisson equations, $\partial f/\partial t = W(f)$, as a function of a system parameter, *v*, which is the applied voltage difference across the RTD. So, in the end, we are trying to find the steady-state Wigner distribution, *f(v)*, which satisfies the nonlinear equation

$$W(f(v)) = 0 \tag{8}$$

as we vary the parameter *v*. To do this, we use continuation methods.

Continuation methods map out solutions to nonlinear equations that depend on a parameter as a function of this parameter. Continuation algorithms generate a sequence of parameters $\{v^m\}$ along with a corresponding sequence of steady-state solutions $\{f^n\}$ that satisfy $W(f^n(v^m)) = 0$. Since we know that when $v = 0$, the steady-state solution is given by the equilibrium Wigner distribution f_0, then the first terms in these sequences are $v^1 = 0$ and $f^1 = f_0$.

We will now present three common continuation methods. A standard technique for solving nonlinear methods are the zero-order continuation, first-order equations is through Newton's Method, and each continuation method uses it to solve the nonlinear equation. The three continuation methods are zero-order continuation, first-order continuation, and the pseudo-arclength continuation.

Assume we have just computed f^n for some v^m, and now we want to compute the next steady-state solution f^{n+1} for a v^{m+1} that is close to v^m. The zero-order continuation uses Newton's Method to solve $W(f^{n+1}(v^{m+1})) = 0$ using f^n as an initial iterate. This method is called zero-order since it does not attempt to incorporate the effects of changing the parameter v^m to v^{m+1} in our initial iterate for f^{n+1}.

The first-order continuation method considers such a change by trying to approximate the sensitivity of the steady-state solution *f* to the parameter *v*, given by $\partial f/\partial v$, at the previous steady-state solution f^n. To compute this value, we differentiate *W(f(v))* with respect to *v* to get

$$\frac{\partial W}{\partial v} = \frac{\partial W}{\partial f}\frac{\partial f}{\partial v} = W'(f^m)\frac{\partial f}{\partial v} \tag{9}$$

So solving for $\partial f/\partial v$ involves solving a linear equation where the coefficient matrix is the Jacobian of *W* with respect to *f* evaluated at f^n, denoted by $W'(f^n)$, and the right hand side is $\partial W/\partial v$. To evaluate $\partial W/\partial v$ at the previous steady-state solution f^n, we use a forward difference approximation

$$\frac{\partial W}{\partial v} \approx \frac{[W(f^m(v^m+\delta)) - W(f^m(v^m))]}{\delta} \tag{10}$$

where _ is some small perturbation. Once we have the approximation to $\partial f/\partial v$, the initial iterate for Newton's Method to solve for f^{n+1} at v^{m+1} will be $f^n + \partial f/\partial v * (v^{m+1} - v^m)$.

The final continuation method, pseudo-arclength continuation [4], is useful when continuing around turning points. Turning points are parts of the steady-state solution branches where the branch turns around. When a turning point occurs, the Jacobian matrix becomes singular. So applying Newton's Method is difficult as we approach the turning point since the Jacobian matrix is becoming singular, making the linear solves for the Newton steps harder. Pseudo-arclength continuation handles this problem by augmenting the nonlinear equation *W(f(v))* with an artificial parameter *s* (the arclength parameter) and an additional arclength equation. So the system we are solving now is

$$\begin{aligned} W(f(v,s)) &= 0 \\ n(f(s),v(s),s) &= 0 \end{aligned} \tag{11}$$

where the first equation specifies that the solution is on the steady-state solution branch, and the second equation specifies the step to take in the parameter *s*. Suppose we have the point (v^m, f^n) on the solution curve and the next solution point to be computed is (v^{m+1}, f^{n+1}). For the next continuation step, the arclength equation is given by

$$n(f(s),v(s),s) = \frac{\partial f}{\partial s}^{T}(f - f^{i}) + \frac{\partial v}{\partial s}(v - v^{i}) - \Delta s$$

where Δ_*s* is the step taken in the parameter *s*. A geometric interpretation of the *(v,f)* points that satisfy $n(f(s), v(s), s) = 0$ can be given. Suppose *a*, *b*, *c*, and *d* are real numbers and (x_0, y_0, z_0) is a three-dimensional vector. It is a result from analytic geometry that the three-dimensional vectors *(x, y, z)* that satisfy $a(x - x_0) + b(y - y_0) + c(z - z_0) - d = 0$ lie in the plane perpendicular to the three-dimensional vector *(a, b, c)* at a distance away from (x_0, y_0, z_0) which is determined by the size of *d*. Similarly, if (v^{m+1}, f^{m+1}) satisfy the arclength equation, then the point will lie in the *(v,f)* plane perpendicular to the gradient of *(v(s),f(s))* at some distance away from (v^m, f^n) which is determined by the size of Δs. An example of tracing a one-dimensional system with a one-dimensional parameter using pseudo-arclength continuation is given in Figure 2 to further illustrate this point.

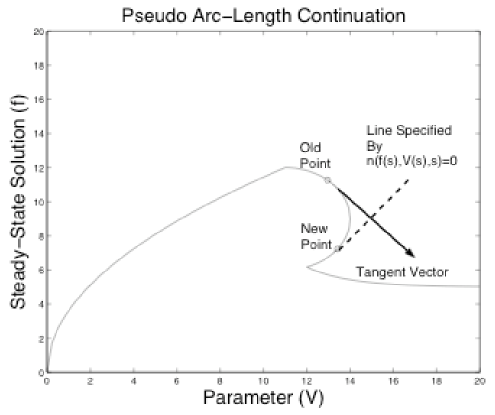

Figure 2: Arclength Continuation

5. LOCA – Library of Continuation Algorithms

To implement these continuation methods into our RTD simulator, we used LOCA (Library of Continuation Algorithms), a software library developed at Sandia National Laboratories [5]. This software library was created for large scale bifurcation and stability analysis. It is a part of Sandia's Trilinos project. Trilinos is a collection of Sandia's parallel solver algorithms, and LOCA uses several other parts of

Trilinos in its continuation methods. To solve the nonlinear equations, LOCA relies on NOX, Trilinos nonlinear solver. To solve the linear equations created in the Newton iterations, AztecOO, Trilinos's preconditioned Krylov solver, is used. To determine the stability of the computed steady-state solutions, LOCA determines the eigenvalues of the Jacobian of the nonlinear equation $W(f)$. For a given steady-state solution f^* of the nonlinear ODE $df/dt = W(f)$, it is a well-known result [] that the eigenvalues of the Jacobian of W at f^*, $W'(f^*)$, determines the stability of f^*. If all of the eigenvalues of $W'(f^*)$ have negative real part, then f^* is asymptotically stable. If any of the eigenvalues of $W'(f^*)$ have positive real part, then f^* is unstable. To check the eigenvalues of $W'(f^*)$, LOCA utilizes Trilinos's eigensolver Anasazi.

6. PRECONDITIONER DEVELOPMENT

The nonlinear solver in the continuation method used for our application was Newton-GMRES. This is an inexact Newton Method, where the linear solution for the Newton steps are solved the Krylov iterative method GMRES [6]. To reduce the number of iterations GMRES takes and therefore reduce the computational burden of the simulation, a preconditioner was developed. When solving the linear equation $Ax = b$, where A is a n by n matrix and x,b are n-dimensional vectors, a preconditioner is another matrix M multiplied into the equation (so now we solve $MAx = Mb$) where the new coefficient matrix MA is an easier matrix for an iterative method to handle. Usually, M is an approximate inverse to A. When solving the linear equations in Newton's Method, the coefficient matrix is always the Jacobian matrix. If we look at Eq. (1), and ignore the last two terms, we get the approximation that $W(f) \approx K(f)$. Since K defined in Eq. (2) is a linear operator, we know $\partial K/\partial f = K$. So an approximation to the Jacobian is $W'(f) \approx K$. Therefore, the preconditioner we use is $M = K^{-1}$.

7. PARALLEL SIMULATOR

To parallelize our evaluation of $W(f)$, we take our domain in (x, k) space and distribute among different processors. Here, we decided that each processor would get a contiguous block of x-space and all of the corresponding k-space that went with each. By splitting the data between the processors this way, we ensure that the integrals in k-space can be performed by each processor independently. This splitting, though, will require communication between the processors that calculate the spatial derivative term in Eq. (2). The Poisson solve was not parallelized and is performed by the main processor before everything else is calculated. Once $U(x)$ is known, the main processor sends out a copy to rest of the processors. The processors then compute their part of $W(f)$, and return this to the main processor.

The parallel runs reported in this paper were performed on a IBM Blade Center with Xeon 2.8 GHz processors at the North Carolina State University's High Performance and Grid Processing.

8. NUMERICAL RESULTS

The first thing we did with our simulator was to verify our numerical simulation with others that were previously published [1]. While these previously published used a very coarse grid (N_x = 86, N_k = 72), their computational time to analyze the current output for $v = 0$ to $v = 0.480$ volts took a few days. Our improved simulator was able to match these results while reducing the computational time to a few hours, while not using any parallel processing. Figure 3 is a plot of the current output versus applied voltage for the coarse grid. The results for the finer grids do not match those in Figure 3, and we are currently exploring the reasons for the difference, which we believe are new physics. We will report on this in future work.

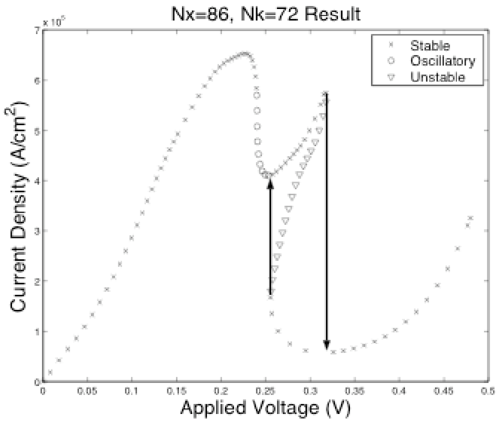

Figure 3: Coarse mesh simulation

Since our simulator was directly computing the steady-state solutions and the previous simulations were using time-accurate methods to reach steady-state, we were able to identify unstable steady-state branches while the time-accurate simulation missed these. These unstable steady-state branches were able to explain the hysteretic effects found on this grid. If the applied voltage is started at zero and is increased, the current stays on the higher stable branch until the voltage is 0.318. The current then drops to the lower stable branch and continues on. If the applied voltage is started at 0.48 volts and is decreased, the current will stay on the lower stable branch until the voltage is 0.25, and then jumps up to the higher stable branch.

Table 1 shows that the preconditioner we use is scalable. The number of GMRES iterations for each Newton step and the number of Newton iterations for each continuation step are essentially independent of the mesh.

N_x	N_k	Avg. Newton Its. Per Continuation Step	Avg. Krylov Its. Per Newton Step
86	72	2.24	156
172	144	2.51	167
344	288	2.41	180
688	576	2.42	196

Table 1: Krylovs/Newton as mesh is refined

As we refine the grids, the number of Newton iterations per continuation step and the number of Krylov iterations per Newton iteration are remaining relatively constant which we expect of a scalable preconditioner.

Table 2 reports on the parallel efficiency of the entire application. The results show that roughly 40% of the code is running in scalar mode.

# of Procs.	Linear Solve Time (sec.)	Speedup Factor	Efficiency (%)
1	431.21	---------	--------
2	263.69	1.64	82.0
4	115.71	3.73	93.3
8	75.23	5.73	71.6
16	45.83	9.50	59.4

Table 2: Parallel efficiency

The grid used in this table is N_x = 688, N_k = 576. As we increase the number of processors used for this job, the efficiency stays above 70% up to 8 processors.

Table 3 presents scalability results of the parallel simulator.

N_x	N_k	# of Procs.	Avg. *W(f)* Evaluation Time (sec.)
172	144	1	0.1209
344	288	4	0.2814
688	576	16	0.5505

Table 3: Scalability

As we quadruple both the number of unknowns and the processors, the function evaluation time should stay flat if the simulator is scaling perfectly. From the table, we see the function evaluation time is doubling. The scaling is consistent with the speedup, telling us the code is 40% serial.

9. CONCLUSIONS

The results from coupling the RTD simulator with LOCA look very promising. We are able to duplicate previously published results at lower computational cost and able to tackle finer grids that before were computationally infeasible. The results from these finer grids seem to indicate that important physics was not resolved in the grid which has been most widely used. We are currently working on getting Fast Fourier Transforms into the simulator to handle the two *x* convolutions in Eq. (4) and the *k* convolution in Eq. (3) that are apart of evaluating the potential energy term *P(f)*. This is the most computationally intensive term, and we anticipate further speedup once the FFTs are incorporated, and we hope they will also improve the efficiency and scalability of our parallel simulator.

10. Acknowledgements

The research of MSL, CTK, and PZ was partially supported by a United States Army Research Office DURINT grant. The research of CTK was also partially supported by United States National Science Foundation grants DMS-0209695 and DMS-0070641.

REFERENCES

[1] P. Zhao, H. L. Cui, and D. L. Woolard. Dynamical Instabilities and I-V Characteristics in Resonant Tunneling Systems. Phys. Rev. B, 63:75302, 2001.
[2] P. Zhao, H. L. Cui, D. L. Woolard, K. L. Jensen, and F. A. Bout. Simulation of Resonant Tunneling Structures: Origin of I-V Hysteresis and Plateau-Like Structure. Journal of Applied Physics, 87:1337-1349, 2000.
[3] F. A. Bout. and K. L. Jensen. Lattice Weyl-Wigner Formulation of Exact Many-Body Quantum-Transport Theory and Applications to Novel Solid-State Quantum-Based Devices. Phys. Rev. B, 42:9429-9438, 1990.
[4] H. B. Keller. Lecture on Numerical Methods in Bifurcation Problems. Springer-Velag, New York, 1987.
[5] Andrew G. Salinger, Nawaf M. Bou-Rabee, Roger P. Pawlowski, Edward D. Wilkes, Elizabeth A. Burroughs, Richard B. Lehoucq, and Louis A. Romero. LOCA 1.0 Library of Continuation Algorithms: Theory and Implementation Manual. Technical Report SAND2002-0396, Sandia National Laboratory, March 2002.
[6] C. T. Kelley. Iterative Methods for Linear and Nonlinear Equations, Volume 16 of Frontiers in Applied Mathematics. SIAM, Philadelphia, PA. 1995