View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by UNT Digital Library

SANDIA REPORT

SAND2007-xxxx7157
Unlimited Release
Printed October 2007

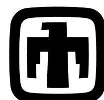
Simulation of Neutron Radiation Damage in Silicon Semiconductor Devices

Gary L. Hennigan, Robert J. Hoekstra, Joseph P. Castro, Deborah A. Fixel, John N. Shadid

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under Contract DE-AC04-94-AL85000.

Approved for public release; further dissemination unlimited.



Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

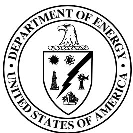
Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online ordering: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online



SAND2007-xxxx7157
Unlimited Release
Printed October 2007

Simulation of Neutron Radiation Damage in Silicon Semiconductor Devices

Gary L. Hennigan, Robert J. Hoekstra, Joseph P. Castro,
Deborah A. Fixel, John N. Shadid
Electrical and Microsystem Modeling Department

Sandia National Laboratories
Mail Stop 0316
P.O. Box 5800
Albuquerque, NM 87185-0316

email contact: glhenni@sandia.gov

Abstract

A code, **Charon**, is described which simulates the effects that neutron damage has on silicon semiconductor devices. The code uses a stabilized, finite-element discretization of the semiconductor drift-diffusion equations. The mathematical model used to simulate semiconductor devices in both normal and radiation environments will be described. Modeling of defect complexes is accomplished by adding an additional drift-diffusion equation for each of the defect species. Additionally, details are given describing how **Charon** can efficiently solve very large problems using modern parallel computers. Comparison between **Charon** and experiment will be given, as well as comparison with results from commercially-available *TCAD* codes.

Acknowledgment

The authors would like to thank the other *Charon* developers, including Roger Pawlowski and Larry Musson for their work on the infrastructure of the code, as well as Eric Keiter who helped in the initial implementation of *Charon*. In particular we are grateful to Roger Pawlowski for the excellent write-up in Appendix A on *L2* projection. Additionally we would like to acknowledge the valuable contributions to *Charon* that came from those actually using the code to solve problems, including Ken Kambour and Ralph Young. We would also like to thank the management and technical personnel involved in Sandia's *QASPR* (Qualification Alternatives to Sandia's Pulsed Reactor) project. We received valuable funding from this project as well as great technical contributions from folks such as Sam Myers and Bill Wampler. Lastly we would like to thank the *Nevada* developers, particularly Rich Drake, for listening to our concerns and acting on them so quickly.

Contents

Nomenclature 8
Symbols 9
1 Introduction 11
1.1 Enabling Technologies 11
2 Semiconductor Physics 13
2.1 Governing Equations 13
2.2 Empirical Models 14
2.3 Boundary Conditions 15
3 Discretization 17
3.1 Variational Multiscale Discretization 17
3.2 Scalar Electric Current Calculation 26
3.3 Results 26
4 Modeling of Neutron Radiation Damage 35
4.1 Drift Diffusion Transport Of Defects 35
4.2 Defect Reactions and Source Terms 37
4.3 Boundary Condition Implementation of Defect Species 43
4.4 Neutron Damage Examples 46
5 Parallel Performance of Charon 50
6 Future Work 52
7 Conclusions 54
References 55

Appendix

A L2 Projection 58
B Time Integration 60

Figures

1 A simple PN-junction diode used as a test problem in *Charon* and *Pisces*. . 27
2 Doping associated with test problem. 28
3 Comparison of a Galerkin discretization compared to the same problem with added *SUPG* stabilization. 29
4 Comparison of scalar electric current for FEM Galerkin, FEM SUPG and Pisces (Scharfetter-Gummel) solutions. 30
5 Self-convergence rate of *Pisces*. 31
6 Self-convergence of *Charon*. 32
7 A *Charon* mesh using automatic, adaptive *h*-refinement. 33
8 Geometry of a two-dimensional diode used as a *VMS* test problem. 34
9 *IV* plot of the currents versus the forward bias of a 2D diode using a partial *VMS* discretization. 34
10 A simple PN-junction diode and its geometrical configuration in the two-dimensional case 44
11 Neutron damage of a Si BJT biased at 0.22 mA within the cavity location of the SPR-III reactor. 48
12 Neutron damage of a Si BJT biased at 0.22 mA within the leakage location of the SPR-III reactor. 49
13 Weak scaling study comparing average iteration per Newton step for the different preconditioners 51
14 Weak scaling study comparing average time per Newton step for the different preconditioners 51

Tables

1 Scaling factors used to scale the semiconductor drift-diffusion equations . . 14
2 List of empirical models implemented within *Charon*. 15

Nomenclature

BJT Bipolar Junction Transistor

QASPR Qualification Alternatives to the Sandia Pulsed Reactor

SUPG Streamline Upwind Petrov Galerkin

VMS Variational Multiscale

TCAD Technology Computer Aided Design

Trilinos A package of packages for massively parallel numerical algorithms

NOX A nonlinear-solver package within Trilinos

Rythmos A time-integration package within Trilinos

SACADO A Trilinos package which implements automatic differentiation

Symbols

k_B Boltzmann's constant

q Electron charge

μ Carrier/species mobility

D Carrier/species diffusion coefficient

ε Permittivity

n Electron concentration

p Hole concentration

ψ Electric Potential

$\mathbf{E}$ Electric Field

R Recombination source-term

N_D Donor (electron) doping

N_A Acceptor (hole) doping

C Doping $(N_D - N_A)$

1 Introduction

In recent years there has been a concerted effort at the nuclear weapons laboratories to move away from experimental procedures and move toward computational methods, where feasible, to assist in the design, qualification and stewardship of weapons systems. Where possible, commercial applications are being used to model components, both mechanical and electrical. However, because of the unique modeling needs specific to weapons systems many commercial applications are insufficient to meet the modeling needs of the national laboratories. One such area is in semiconductor modeling in neutron-radiation environments.

Silicon semiconductor devices play an extremely important role in the operation of current stockpile weapons systems and in the design of possible future weapons systems. While there has been a small amount of experimental work done to empirically model the effects of neutron radiation damage [25, 28] and in modeling of various radiation sources in circuit, lumped-parameter modeling [9], there has been little work on modeling neutron radiation at the fluences that may be experienced by components in nuclear weapons and at the fidelity required to understand the physics occurring in sufficient detail.

To this end the computer code *Charon* has been developed at Sandia National Laboratories. Using the latest stabilized finite-element technology *Charon* is an attempt to model a silicon semiconductor device, at the carrier-concentration level, including the transport of the numerous defect complexes created by neutron radiation at the short time scales and high fluence levels.

Due to the device dimensions, the high level of fidelity and the addition of a significant number of unknowns for modeling the defect species *Charon* has been written to take advantage of massively-parallel computing platforms. This will allow for timely simulations that wouldn't be possible on the scalar platforms supported by commercial TCAD semiconductor analysis codes.

1.1 Enabling Technologies

As with most modern codes, *Charon* relies significantly on supporting technology to achieve its goals. The two most significant supporting technologies for *Charon* are *Trilinos* [13], *Nevada* and an automatic differentiation package referred to as *SACADO*.

Nevada is a software infrastructure designed to support development, testing, verification release of large scientific simulation applications [27]. *Nevada* is designed to facilitate development of new finite-element applications and provides much of the infrastructure necessary for such codes. Items like I/O, element libraries, material model interface, etc., are provided.

Trilinos is a collection of packages to support modern application development. *Charon*

specifically makes use of the following packages:

1. NOX [21] nonlinear solver,
2. Teuchos [24] basic tools,
3. AztecOO preconditioned Krylov Solver.

The other package used heavily by *Charon* is *SACADO*, which implements automatic differentiation [2]. Automatic differentiation makes generating the Jacobian necessary for the nonlinear solver almost trivial. The only requirement as far as *Charon* is concerned is that anything requiring differentiation be templated on the *SACADO* data type. Once that is accomplished there is no implementation of a Jacobian required from the developer, thus bypassing an extremely labor-intensive, and error-prone process which is generating Jacobians as well as generation of accurate sensitivities to input parameters allowing gradient-based algorithms such as sequential quadratic programming (SQP) techniques for optimization.

2 Semiconductor Physics

This section will describe the basic semiconductor physics, including the governing equations used for the drift-diffusion approximation and various empirical models used by *Charon*.

2.1 Governing Equations

The equations governing the transport of charge carriers within a semiconductor device can be approximated using the standard drift-diffusion equations given by:

$$\nabla \cdot (\varepsilon \mathbf{E}) - q\,(p - n + C) = 0, \tag{1}$$

$$\nabla \cdot \mathbf{J}_n - qG = q\frac{\partial n}{\partial t}, \tag{2}$$

$$-\nabla \cdot \mathbf{J}_p - qG = q\frac{\partial p}{\partial t}, \tag{3}$$

where

$$\mathbf{E} = -\nabla \psi, \tag{4}$$

$$\mathbf{J}_n = qn\mu_n \mathbf{E} + qD_n \nabla n, \tag{5}$$

$$\mathbf{J}_p = qp\mu_p \mathbf{E} - qD_p \nabla p. \tag{6}$$

By using (4)–(6) in (1)–(3) one obtains

$$-\nabla \cdot (\varepsilon \nabla \psi) - q\,(p - n + C) = 0, \tag{7}$$

$$\nabla \cdot (n\mu_n \nabla \psi) - \nabla \cdot (D_n \nabla n) + G + \frac{\partial n}{\partial t} = 0, \tag{8}$$

$$-\nabla \cdot (p\mu_p \nabla \psi) - \nabla \cdot (D_p \nabla p) + G + \frac{\partial p}{\partial t} = 0, \tag{9}$$

where the unknowns are: ψ, the electrostatic potential, n, the electron density, and p, the hole density, with ε, the permittivity of the semiconductor material, q, the fundamental electron charge, μ_n and μ_p, the electron and hole mobilities, respectively, D_n and D_p, the electron and hole diffusion coefficients, respectively, C the doping profile, and R the generation/recombination source term. C can be broken down further as

$$C = N_D^+ - N_A^-, \tag{10}$$

where N_D^+ and N_A^- represent the amount of ionized donor and acceptor dopant concentrations, respectively.

Typically (7)–(9) are scaled prior to discretization [22]. The most widely used scale

factors, and the ones used in *Charon*, are given in Table 1. After scaling (7)–(9) become

$$-\lambda^2 \nabla \cdot (\varepsilon_r \nabla \psi) - (p - n + C) = 0, \tag{11}$$

$$\nabla \cdot (\mu_n n \nabla \psi) - \nabla \cdot (D_n \nabla n) + G + \frac{\partial n}{\partial t} = 0, \tag{12}$$

$$-\nabla \cdot (\mu_p p \nabla \psi) - \nabla \cdot (D_p \nabla p) + G + \frac{\partial p}{\partial t} = 0, \tag{13}$$

where

$$\lambda^2 = \frac{V_0 \varepsilon_0}{q x_0^2 C_0}. \tag{14}$$

Quantity	Symbol	Value
$\mathbf{X}$	x_0	$\max(\mathbf{X}), \ \mathbf{X} \in \mathbf{D}$
ψ	V_0	$\frac{k_B T}{q}$
n, p, C	C_0	$\max(C(\mathbf{X})), \ \mathbf{X} \in \mathbf{D} \ \mid \ n_i$ Note 1
D_n, D_p	D_0	$\max(D_n(\mathbf{X}), D_p(\mathbf{X})), \ \mathbf{X} \in \mathbf{D}$
μ_n, μ_p	μ_0	$\frac{D_0}{V_0}$
R	R_0	$\frac{D_0 C_0}{x_0^2}$
t	t_0	$\frac{x_0^2}{D_0}$
E	E_0	$\frac{V_0}{x_0}$
J_n, J_p	J_0	$\frac{q D_0 C_0}{x_0}$

Table 1. Scaling factors used to scale the semiconductor drift-diffusion equations. Note 1: In parallel runs it is often convenient to have a global constant here to avoid doing a communication in order to find the maximum value of the doping.

2.2 Empirical Models

Commercial *TCAD* programs rely on numerous empirical models developed through the years by the semiconductor modeling community. Table 2 is a list of the empirical models currently implemented within Charon.

Model Name	Purpose
Arora [1]	Carrier Mobility
Philips [19, 20]	Carrier Mobility
Analytic [3]	Carrier Mobility
Carrier-Carrier [7]	Carrier Mobility
Lucent [5]	Carrier Mobility
Surface [23]	Carrier Mobility
Bandgap Narrowing [34]	Bandgap Variation with Doping Concentration
Concentration-Dependent Lifetimes	Recombination/Generation
Incomplete Ionization [17, 32, 30]	Fraction of Impurities Ionized

Table 2. List of empirical models implemented within *Charon*.

2.3 Boundary Conditions

The boundary conditions typically used for junction devices, such as the bipolar-junction transistor, are derived assuming the electrical contacts are far enough away from any junction that equilibrium conditions prevail. This allows Dirichlet conditions to be used. For the electric potential the boundary conditions are calculated such that the built-in potential across junctions is set at the associated contacts. For a contact at a donor-doped contact the electric potential is

$$\psi_{\text{n-contact}} = \frac{k_B T}{q} \ln\left(\frac{n}{n_i}\right). \tag{15}$$

In the case of 100% ionization of the dopants n in (15) can be approximated by $n \approx N_D$. In the case of a contact adjacent to an acceptor-doped contact the electric potential is

$$\psi_{\text{p-contact}} = -\frac{k_B T}{q} \ln\left(\frac{p}{n_i}\right). \tag{16}$$

Again, in the case of 100% ionization of the dopants, p in (16) can be approximated by $p \approx N_A$. The built-in potential can be calculated by subtracting (16) from (15). Assuming 100% ionization of the dopants this yields

$$V_{\text{bi}} = \psi_{\text{n-contact}} - \psi_{\text{p-contact}} = \frac{k_B T}{q} \ln\left(\frac{N_D N_A}{n_i^2}\right). \tag{17}$$

To calculate the concentration values at the contacts, at equilibrium

$$np = n_i^2. \tag{18}$$

Also, at equilibrium, charge neutrality must hold. Using (10), then at a contact

$$n|_{\text{contact}} - p|_{\text{contact}} - C = 0. \tag{19}$$

Solving (18) for p and substituting the result into (19) yields

$$(n|_{\text{contact}})^2 - Cn|_{\text{contact}} - n_i^2 = 0, \tag{20}$$

which is a quadratic with solution

$$n|_{\text{contact}} = \frac{1}{2}\left(C + \sqrt{C^2 + 4n_i^2}\right). \tag{21}$$

Similarly for p at a contact

$$p|_{\text{contact}} = \frac{1}{2}\left(C + \sqrt{C^2 + 4n_i^2}\right). \tag{22}$$

3 Discretization

In the discretization design process for *Charon* various discretization technologies were considered. These included finite volume methods, stabilized finite-element methods and mixed interpolation finite element methods. In that process it was decided that *Charon* would initially use the stabilized finite-element method (*FEM*) for its discretization and mixed interpolation methods, that would allow formulations closer to finite volume type methods, would be considered later in the R&D effort. One reason for the choice of finite-element methods is that it was demonstrated that the method commonly used to solve the drift-diffusion equations by commercial *TCAD* codes, the Scharfetter-Gummel method, suffers from excessive "numerical diffusion" when used for problems of more than one dimension [33]. In addition, the development of a variational type method for discretization would more naturally allow the development of adjoint based error estimators for spatial discretization adaptivity and error control. Finally, Sandia in general, and our organization in particular, has developed significant expertise in finite-element technology. Not only in the analysis of various partial-differential equations (PDEs) and in the iterative solution of these systems, but also in the generation of meshes for a wide variation of problem geometries and in the visualization of results from finite-element analysis.

3.1 Variational Multiscale Discretization

The first step in any finite-element discretization is to generate the weak form of the governing equations. For the semiconductor physics considered here the governing equations are given by (11)–(13). The associated Galerkin weak forms of (11)–(13) are given by, respectively

$$\lambda^2\varepsilon_r \int \nabla\psi \cdot \nabla w \, d\Omega - \lambda^2\varepsilon_r \int_\Gamma \nabla\psi \cdot \hat{\eta} w \, d\Gamma - \int p w \, d\Omega + \int n w \, d\Omega - \int C w \, d\Omega = 0, \quad (23)$$

$$\int \dot{n} w \, d\Omega + \int_\Gamma \mu_n n \nabla\psi \cdot \hat{\eta} w \, d\Gamma - \int \mu_n n \nabla\psi \cdot \nabla w \, d\Omega - \int_\Gamma D_n \nabla n \cdot \hat{\eta} w \, d\Gamma + \int D_n \nabla n \cdot \nabla w \, d\Omega + \int G w \, d\Omega = 0, \quad (24)$$

$$\int \dot{p} w \, d\Omega - \int_\Gamma \mu_p p \nabla\psi \cdot \hat{\eta} w \, d\Gamma + \int \mu_p p \nabla\psi \cdot \nabla w \, d\Omega - \int_\Gamma D_p \nabla p \cdot \hat{\eta} w \, d\Gamma + \int D_p \nabla p \cdot \nabla w \, d\Omega + \int G w \, d\Omega = 0, \quad (25)$$

where $\hat{\eta}$ is a unit normal to the boundary of the problem domain, w is the testing function and Γ is the boundary of the domain Ω.

For the Variational Multiscale (VMS) method we are going to split the problem domain into a coarse and a fine solution. The split of the solution space is represented by

$$S = \bar{S} \oplus S' \quad (26)$$

where

$$\bar{S} \rightarrow \text{Coarse solution subspace,}$$
$$S' \rightarrow \text{Fine solution subspace,}$$

and likewise for the weighting space

$$\bar{\nu} \rightarrow \text{Coarse weighting subspace,}$$
$$\nu' \rightarrow \text{Fine weighting subspace.}$$

This implies that

$$\psi = \bar{\psi} + \psi', \tag{27}$$
$$n = \bar{n} + n', \tag{28}$$
$$p = \bar{p} + p', \tag{29}$$
$$w = \bar{w} + w'. \tag{30}$$

We also require that on the boundary, Γ, the fine-scale solutions are identically zero [16].

3.1.1 Poisson Equation

Using (27)–(30) in (23), and assuming that either Dirichlet or Neumann conditions govern on the boundary Γ yields

$$\lambda^2 \varepsilon_r \int \left(\nabla\bar{\psi} \cdot \nabla\bar{w} + \nabla\bar{\psi} \cdot \nabla w' + \nabla\psi' \cdot \nabla\bar{w} + \nabla\psi' \cdot \nabla w' \right) d\Omega - \int \left(\bar{p}\bar{w} + p'\bar{w} + \bar{p}w' + p'w' \right) d\Omega + \int \left(\bar{n}\bar{w} + n'\bar{w} + \bar{n}w' + n'w' \right) d\Omega - \int \left(C\bar{w} + Cw' \right) d\Omega = 0. \tag{31}$$

We need to numerically model those terms associated with the coarse weighting space $\bar{\nu}$. These are the terms associated with the coarse weighting function $\bar{w}$. Gathering such terms from (31) one obtains:

$$\lambda^2 \varepsilon_r \int \nabla\bar{\psi} \cdot \nabla\bar{w} \, d\Omega + \lambda^2 \varepsilon_r \int \nabla\psi' \cdot \nabla\bar{w} \, d\Omega - \int \bar{p}\bar{w} \, d\Omega - \int p'\bar{w} \, d\Omega + \int \bar{n}\bar{w} \, d\Omega + \int n'\bar{w} \, d\Omega - \int C\bar{w} \, d\Omega = 0. \tag{32}$$

Now we need to remove any terms including the differentiation of any fine-scale variables. The single term in (32) involving such an operation is

$$\lambda^2 \varepsilon_r \int \nabla\psi' \cdot \nabla\bar{w} \, d\Omega. \tag{33}$$

To eliminate the derivative on ψ' we can make use of the vector identity

$$\nabla \cdot \left(a\vec{B} \right) = \nabla a \cdot \vec{B} + a \nabla \cdot \vec{B}, \tag{34}$$

where $a = \psi'$ and $\vec{B} = \nabla \bar{w}$ then (33) becomes

$$\lambda^2 \varepsilon_r \left[\int \nabla \cdot \left(\psi' \nabla \bar{w} \right) d\Omega - \int \psi' \nabla \cdot \nabla \bar{w} \, d\Omega \right]. \tag{35}$$

We can use the divergence theorem on the first term in (35) and it becomes

$$\lambda^2 \varepsilon_r \left[\int_\Gamma \psi' \nabla \bar{w} \cdot \hat{\eta} \, d\Gamma - \int \psi' \nabla \cdot \nabla \bar{w} \, d\Omega \right]. \tag{36}$$

Recognizing that the first integral in (36) contains a fine-scale solution on the boundary and that we earlier stipulated that all fine-scale solutions were zero at the boundary we can eliminate the first integral in (36), leaving us with

$$\lambda^2 \varepsilon_r \int \nabla \psi' \cdot \nabla \bar{w} \, d\Omega = -\lambda^2 \varepsilon_r \int \psi' \nabla \cdot \nabla \bar{w} \, d\Omega. \tag{37}$$

Now we can use (37) in (32) to obtain the coarse-scale equation

$$\begin{aligned} \textbf{\textit{Galerkin}} & \left\{ \lambda^2 \varepsilon_r \int \nabla \bar{\psi} \cdot \nabla \bar{w} \, d\Omega - \int_\Omega (p - n + C) \, \bar{w} \, d\Omega - \right. \\ \textbf{\textit{SUPG/GLS}} & \left\{ \lambda^2 \varepsilon_r \int \psi' \nabla \cdot \nabla \bar{w} \, d\Omega - \right. \\ \textbf{\textit{VMS}} & \left\{ \int p' \bar{w} \, d\Omega + \int n' \bar{w} \, d\Omega, \right. \end{aligned} \tag{38}$$

which is the final form of the equation to be numerically modeled. Note that if linear elements are used then the *SUPG/GLS* term in (38) will be zero because of the second derivative.

3.1.2 Electron Equation

Using (27)–(30) in (24) yields

$$\begin{aligned} \int \left(\dot{\bar{n}} + \dot{n}' \right) \left(\bar{w} + w' \right) d\Omega + \int_\Gamma \mu_n \left(\bar{n} + n' \right) \nabla \left(\bar{\psi} + \psi' \right) \cdot \hat{\eta} \left(\bar{w} + w' \right) d\Gamma - \\ \int \mu_n \left(\bar{n} + n' \right) \nabla \left(\bar{\psi} + \psi' \right) \cdot \nabla \left(\bar{w} + w' \right) d\Omega - \\ \int_\Gamma D_n \nabla \left(\bar{n} + n' \right) \cdot \hat{\eta} \left(\bar{w} + w' \right) d\Gamma + \int D_n \nabla \left(\bar{n} + n' \right) \cdot \nabla \left(\bar{w} + w' \right) d\Omega + \\ \int G \left(\bar{\psi} + \psi', \bar{n} + n', \bar{p} + p' \right) \left(\bar{w} + w' \right) d\Omega = 0. \end{aligned} \tag{39}$$

Before proceeding further with the electron equation it is necessary to represent the nonlinear source term in a way that facilitates the VMS analysis. Currently the nonlinear term in the electron equation is included by employing a Taylor series expansion about the

coarse solution $(\bar{\psi}, \bar{n}, \bar{p})$ that neglects the higher order terms (H.O.T) and the dependence on the other variables to obtain:

$$G(\psi, n, p) = \bar{G} + \frac{\partial \bar{G}}{\partial n}(n - \bar{n}) + \frac{\partial \bar{G}}{\partial p}(p - \bar{p}) + \frac{\partial \bar{G}}{\partial \psi}(\psi - \bar{\psi}) + H.O.T. \approx \bar{G} + \frac{\partial \bar{G}}{\partial n} n'. \quad (40)$$

This approximation produces a contribution that helps to stabilize the source contribution in the n equation. Using the notation of Donea and Huerta (see [6]) the term associated with the generation for the electron concentration now becomes

$$\int \bar{G} w \, d\Omega + \int \sigma_n n' w \, d\Omega \quad (41)$$

where

$$\sigma_n \equiv \frac{\partial \bar{G}}{\partial n} \quad (42)$$

for example.

Expanding out (39) and retaining only terms associated with the coarse space, $\bar{w}$, yields

$$\begin{aligned}
&\int \dot{\bar{n}} \bar{w} \, d\Omega + \int \dot{n}' \bar{w} \, d\Omega + \int_\Gamma \mu_n \bar{n} \nabla \bar{\psi} \cdot \hat{\eta} \bar{w} \, d\Gamma + \int_\Gamma \mu_n \bar{n} \nabla \psi' \cdot \hat{\eta} \bar{w} \, d\Gamma + \\
&\int_\Gamma \mu_n n' \nabla \bar{\psi} \cdot \hat{\eta} \bar{w} \, d\Gamma + \int_\Gamma \mu_n n' \nabla \psi' \cdot \hat{\eta} \bar{w} \, d\Gamma - \int \mu_n \bar{n} (\nabla \bar{\psi} \cdot \nabla \bar{w}) \, d\Omega - \\
&\int \mu_n \bar{n} \left(\nabla \psi' \cdot \nabla \bar{w}\right) d\Omega - \int \mu_n n' (\nabla \bar{\psi} \cdot \nabla \bar{w}) \, d\Omega - \\
&\int \mu_n n' \left(\nabla \psi' \cdot \nabla \bar{w}\right) d\Omega - \int_\Gamma D_n \nabla \bar{n} \cdot \hat{\eta} \bar{w} \, d\Gamma - \int_\Gamma D_n \nabla n' \cdot \hat{\eta} \bar{w} \, d\Gamma + \\
&\int D_n \nabla \bar{n} \cdot \bar{w} \, d\Omega + \int D_n \nabla n' \cdot \nabla \bar{w} \, d\Omega + \int \bar{G} \bar{w} \, d\Omega + \int \sigma_n n' \bar{w} = 0. \quad (43)
\end{aligned}$$

We can ignore the boundary terms if we assume that the unknowns are governed by either Dirichlet or Neumann conditions. Doing so and regrouping terms in (43) yields

$$\begin{aligned}
&\int \dot{\bar{n}} \bar{w} \, d\Omega - \int \mu_n \bar{n} (\nabla \bar{\psi} \cdot \nabla \bar{w}) \, d\Omega + \int D_n \nabla \bar{n} \cdot \nabla \bar{w} \, d\Omega + \int \sigma_n \bar{n} \bar{w} \, d\Omega + \\
&\int \sigma_n n' \bar{w} \, d\Omega + \int \dot{n}' \bar{w} \, d\Omega - \int \mu_n n' (\nabla \bar{\psi} \cdot \nabla \bar{w}) \, d\Omega - \\
&\int \mu_n \bar{n} \left(\nabla \psi' \cdot \nabla \bar{w}\right) d\Omega - \int \mu_n n' \left(\nabla \psi' \cdot \nabla \bar{w}\right) d\Omega + \int D_n \nabla n' \cdot \nabla \bar{w} \, d\Omega. \quad (44)
\end{aligned}$$

The derivatives on the fine-scale unknowns need to be removed. The first term to be considered is

$$\int \mu_n \bar{n} \left(\nabla \psi' \cdot \nabla \bar{w}\right) d\Omega. \quad (45)$$

We can again utilize (34) to obtain

$$\int \mu_n \bar{n} \left(\nabla \psi' \cdot \nabla \bar{w}\right) d\Omega = \int \mu_n \bar{n} \left[\nabla \cdot \left(\psi' \nabla \bar{w}\right)\right] d\Omega - \int \mu_n \bar{n} \psi' \nabla \cdot \nabla \bar{w} \, d\Omega. \quad (46)$$

We can use (34) again on the first term on the right-hand side of the equality in (46) to obtain

$$\begin{aligned}\int \mu_n \bar{n}\left[\nabla\cdot\left(\psi'\nabla\bar{w}\right)\right] d\Omega &= \int \nabla\cdot\left(\mu_n\bar{n}\psi'\nabla\bar{w}\right) d\Omega - \int \nabla\left(\mu_n\bar{n}\right)\cdot\left(\psi'\nabla\bar{w}\right) d\Omega \\ &= \int_\Gamma \mu_n\bar{n}\psi'\nabla\bar{w}\, d\Gamma - \int \nabla\left(\mu_n\bar{n}\right)\cdot\left(\psi'\nabla\bar{w}\right) d\Omega\end{aligned} \tag{47}$$

The first term on the right-hand side of the equation is a surface integral with the fine-scale variable ψ'. Earlier it was stipulated that such terms are zero. The second term can be expanded out as

$$\begin{aligned}\int \nabla\left(\mu_n\bar{n}\right)\cdot\left(\psi'\nabla\bar{w}\right) d\Omega &= \int \left(\mu_n\nabla\bar{n} + \bar{n}\nabla\mu_n\right)\cdot\left(\psi'\nabla\bar{w}\right) d\Omega \\ &= \int \mu_n\psi'\left(\nabla\bar{n}\cdot\nabla\bar{w}\right) d\Omega + \int \bar{n}\psi'\left(\nabla\mu_n\cdot\nabla\bar{w}\right) d\Omega.\end{aligned} \tag{48}$$

If we assume that the mobility, μ_n, a material property, is constant then the term involving the gradient of the mobility in (48) can be eliminated. This leaves

$$\int \mu_n\bar{n}\left(\nabla\psi'\cdot\nabla\bar{w}\right) d\Omega \approx -\int \mu_n\bar{n}\psi'\nabla\cdot\nabla\bar{w}\, d\Omega - \int \mu_n\psi'\left(\nabla\bar{n}\cdot\nabla\bar{w}\right) d\Omega. \tag{49}$$

The next term involving a differentiation of a fine-scale variable from (44) is

$$\int \mu_n n'\left(\nabla\psi'\cdot\nabla\bar{w}\right) d\Omega. \tag{50}$$

This term is effectively a quadratic term on the fine scale involving n' and ψ'. At present we are going to assume such terms are negligible.

The last term to consider in (44) is

$$\int D_n\left(\nabla n'\cdot\nabla\bar{w}\right) d\Omega. \tag{51}$$

We again employ (34) to obtain

$$\begin{aligned}\int D_n\nabla n'\cdot\nabla\bar{w}\, d\Omega &= \int \nabla\cdot\left(n' D_n\nabla\bar{w}\right) d\Omega - \int n'\nabla\cdot\left(D_n\nabla\bar{w}\right) d\Omega \\ &= \int_\Gamma n' D_n\left(\nabla\bar{w}\cdot\hat{\eta}\right) d\Gamma - \int n'\nabla\cdot\left(D_n\nabla\bar{w}\right) d\Omega.\end{aligned} \tag{52}$$

The surface term in (52) contains a fine-scale variable, n', which was taken to be zero at the surface leaving

$$\int D_n\left(\nabla n'\cdot\nabla\bar{w}\right) d\Omega = -\int n'\nabla\cdot\left(D_n\nabla\bar{w}\right) d\Omega. \tag{53}$$

Now we can use (49) and (53) in (44) which yields

$$\begin{aligned}&\int \dot{\bar{n}}\bar{w}\, d\Omega - \int \mu_n\bar{n}\left(\nabla\bar{\psi}\cdot\nabla\bar{w}\right) d\Omega + \int D_n\nabla\bar{n}\cdot\nabla\bar{w}\, d\Omega + \int \bar{G}\bar{w}\, d\Omega + \\ &\int \sigma_n n'\bar{w}\, d\Omega + \int \dot{n}'\bar{w}\, d\Omega - \int \mu_n n'\left(\nabla\bar{\psi}\cdot\nabla\bar{w}\right) d\Omega - \\ &\int n'\nabla\cdot\left(D_n\nabla\bar{w}\right) d\Omega + \int \mu_n\bar{n}\psi'\nabla\cdot\nabla\bar{w}\, d\Omega + \int \mu_n\psi'\left(\nabla\bar{n}\cdot\nabla\bar{w}\right) d\Omega.\end{aligned} \tag{54}$$

The equation above can be reordered to give some insight into the differences between Galerkin, SUPG and VMS. Doing so yields

$$
\begin{aligned}
&\textbf{\textit{Galerkin}}\left\{\begin{aligned}&\int \dot{\bar{n}}\bar{w}\,d\Omega - \int \mu_n \bar{n}\left(\nabla\bar{\psi}\cdot\nabla\bar{w}\right)d\Omega + \int D_n \nabla\bar{n}\cdot\nabla\bar{w}\,d\Omega + \\ &\qquad\qquad\qquad\qquad\qquad\qquad\quad \int \bar{G}\bar{w}\,d\Omega +\end{aligned}\right. \\
&\textbf{\textit{SUPG/GLS}}\left\{\begin{aligned}&\int \dot{n}'\bar{w}\,d\Omega - \int \mu_n n'\left(\nabla\bar{\psi}\cdot\nabla\bar{w}\right)d\Omega - \int n'\nabla\cdot\left(D_n\nabla\bar{w}\right)d\Omega + \\ &\qquad\qquad\qquad\qquad\qquad\qquad\quad \int \sigma_n n'\bar{w}\,d\Omega +\end{aligned}\right. \\
&\textbf{\textit{VMS}}\left\{\int \mu_n \psi'\left(\nabla\bar{n}\cdot\nabla\bar{w}\right)d\Omega + \int \mu_n \bar{n}\psi'\nabla\cdot\nabla\bar{w}\,d\Omega = 0\right.
\end{aligned} \tag{55}
$$

Again note that if linear elements are used then term above involving $\nabla\cdot\nabla\bar{w}$ will be zero.

3.1.3 Hole Equation

The derivation of VMS for the hole continuity equation follows closely that of the electron continuity in 3.1.2. First the source term represented as G in (25) must be approximated as above. It takes on the form of $\bar{G}+\sigma_p p'$. Using this relation in (25), along with (27) – (30), and ignoring surface terms, yields

$$
\begin{aligned}
&\int \left(\dot{\bar{p}}+\dot{p}'\right)\left(\bar{w}+w'\right)d\Omega + \int \mu_p\left(\bar{p}+p'\right)\nabla\left(\bar{\psi}+\psi'\right)\cdot\nabla\left(\bar{w}+w'\right)d\Omega + \\
&\quad \int D_p\nabla\left(\bar{p}+p'\right)\cdot\nabla\left(\bar{w}+w'\right)d\Omega + \int \bar{G}\left(\bar{w}+w'\right)d\Omega + \int \sigma_p p'\left(\bar{w}+w'\right)d\Omega = 0.
\end{aligned} \tag{56}
$$

Expanding out (56) and retaining only terms associated with the coarse space, $\bar{w}$, yields

$$
\begin{aligned}
&\int \dot{\bar{p}}\bar{w}\,d\Omega + \int \dot{p}'\bar{w}\,d\Omega + \int \mu_p\bar{p}\left(\nabla\bar{\psi}\cdot\nabla\bar{w}\right)d\Omega + \int \mu_p\bar{p}\left(\nabla\psi'\cdot\nabla\bar{w}\right)d\Omega + \\
&\qquad\qquad \int \mu_p p'\left(\nabla\bar{\psi}\cdot\nabla\bar{w}\right)d\Omega + \int \mu_p p'\left(\nabla\psi'\cdot\nabla\bar{w}\right)d\Omega + \\
&\quad \int D_p\left(\nabla\bar{p}\cdot\nabla\bar{w}\right)d\Omega + \int D_p\left(\nabla p'\cdot\nabla\bar{w}\right)d\Omega + \int \bar{G}\bar{w}\,d\Omega + \int \sigma_p p'\bar{w}\,d\Omega.
\end{aligned} \tag{57}
$$

We again need to eliminate the derivatives off of the fine-scale variables. The first term to be considered is

$$
\int \mu_p\bar{p}\left(\nabla\psi'\cdot\bar{w}\right)d\Omega \tag{58}
$$

This is the same form as (45) and the end result is

$$
\int \mu_p\bar{p}\left(\nabla\psi'\cdot\nabla\bar{w}\right)d\Omega \approx -\int \mu_p\bar{p}\psi'\nabla\cdot\nabla\bar{w}\,d\Omega - \int \mu_p\psi'\left(\nabla\bar{p}\cdot\nabla\bar{w}\right)d\Omega. \tag{59}
$$

We will again assume that the quadratic term

$$\int \mu_p p' \left(\nabla\psi' \cdot \nabla\bar{w}\right) = 0. \tag{60}$$

This leaves the term

$$\int D_p \left(\nabla p' \cdot \nabla\bar{w}\right) d\Omega, \tag{61}$$

which has the same form as (51) and results in

$$\int D_p \left(\nabla p' \cdot \nabla\bar{w}\right) d\Omega = -\int p'\nabla \cdot (D_p \nabla\bar{w})\, d\Omega. \tag{62}$$

Using (59) and (62) in (57) yields

$$\begin{aligned}\int \dot{\bar{p}}\bar{w}\, d\Omega + \int \dot{p}'\bar{w}\, d\Omega + \int \mu_p \bar{p}\,(\nabla\bar{\psi} \cdot \nabla\bar{w})\, d\Omega - \int \mu_p \bar{p}\psi'\nabla \cdot \nabla\bar{w}\, d\Omega - \\ \int \mu_p \psi'\,(\nabla\bar{p} \cdot \nabla\bar{w})\, d\Omega + \int \mu_p p'\,(\nabla\bar{\psi} \cdot \nabla\bar{w})\, d\Omega + \\ \int D_p\,(\nabla\bar{p} \cdot \nabla\bar{w})\, d\Omega + \int p'\nabla \cdot (D_p \nabla\bar{w})\, d\Omega + \int \bar{G}\bar{w}\, d\Omega + \int \sigma_p p'\bar{w}\, d\Omega.\end{aligned} \tag{63}$$

We can rearrange (63) to obtain more insight into the nature of the discretization compared to others. We obtain

$$\begin{aligned}&\textbf{\textit{Galerkin}}\begin{cases}\int \dot{\bar{p}}\bar{w}\, d\Omega + \int \mu_p \bar{p}\,(\nabla\bar{\psi} \cdot \nabla\bar{w})\, d\Omega + \int D_p \nabla\bar{p} \cdot \nabla\bar{w}\, d\Omega + \\ \qquad \int \bar{G}\bar{w}\, d\Omega + \end{cases} \\ &\textbf{\textit{SUPG/GLS}}\begin{cases}\int \dot{p}'\bar{w}\, d\Omega + \int \mu_p p'\,(\nabla\bar{\psi} \cdot \nabla\bar{w})\, d\Omega + \int p'\nabla \cdot (D_p \nabla\bar{w})\, d\Omega + \\ \qquad \int \sigma_p p'\bar{w}\, d\Omega - \end{cases} \\ &\textbf{\textit{VMS}}\left\{\int \mu_p \psi'\,(\nabla\bar{p} \cdot \nabla\bar{w})\, d\Omega - \int \mu_p \bar{p}\psi'\nabla \cdot \nabla\bar{w}\, d\Omega = 0\right.\end{aligned} \tag{64}$$

Again note that if linear elements are used then terms above involving $\nabla \cdot \nabla\bar{w}$ will be zero.

3.1.4 Fine-Scale Solutions

Approximations for the fine-scale variables in (38), (55) and (64) must be determined. In general they are specified as

$$\psi' = -\tau_\psi R_\psi, \tag{65}$$

$$n' = -\tau_n R_n, \tag{66}$$

$$p' = -\tau_p R_p, \tag{67}$$

where R_ψ, R_n and R_p are the residuals of PDEs in (11), (12) and (13) and τ_ψ, τ_n and τ_p are the stabilization parameters.

3.1.5 PDE Residuals

The residuals associated with the partial-differential equations (PDEs) are:

$$-\lambda^2 \nabla \cdot (\varepsilon_r \nabla \psi) - (p - n + C) = R_\psi. \tag{68}$$

$$\frac{\partial n}{\partial t} + \nabla \cdot (\mu_n n \nabla \psi) - \nabla \cdot (D_n \nabla n) + G = R_n \tag{69}$$

$$\frac{\partial p}{\partial t} - \nabla \cdot (\mu_p p \nabla \psi) - \nabla \cdot (D_p \nabla p) + G = R_p \tag{70}$$

For Poisson's equation, (68), ε_r is typically constant and (68) can be written

$$-\lambda^2 \varepsilon_r \nabla \cdot \nabla \psi - (p - n + C) = R_\psi. \tag{71}$$

In order to approximate the second-order Laplacian in (71), assuming first-order, linear elements, L2 projection must be used.

For (69) we can expand out the derivatives to obtain:

$$R_n = \frac{\partial n}{\partial t} + n \nabla \mu_n \cdot \nabla \psi + \mu_n \nabla n \cdot \nabla \psi + \mu_n n \nabla \cdot \nabla \psi - \nabla D_n \cdot \nabla n - D_n \nabla \cdot \nabla n + G. \tag{72}$$

Generally it is assumed that the material properties, μ_n and D_n, are constant and terms involving their derivative in (72) can thus be removed. Additionally if we assume steady-state (72) reduces to

$$R_n \approx \mu_n \nabla n \cdot \nabla \psi + \mu_n n \nabla \cdot \nabla \psi - D_n \nabla \cdot \nabla n + G. \tag{73}$$

In order to include the second-order terms in (73), assuming the use of first-order, linear elements, L2 projection must be used.

The residual for the hole carrier, (70), can be reduced in a fashion similar to that for the electron.

$$R_p = \frac{\partial p}{\partial t} - p \nabla \mu_p \cdot \nabla \psi - \mu_p \nabla p \cdot \nabla \psi - \mu_p p \nabla \cdot \nabla \psi - \nabla D_p \cdot \nabla p - D_p \nabla \cdot \nabla p + G. \tag{74}$$

Again assuming negligible variation in the material properties and steady state, μ_p and D_p, we obtain

$$R_p \approx -\mu_p \nabla p \cdot \nabla \psi - \mu_p p \nabla \cdot \nabla \psi - D_p \nabla \cdot \nabla p + G. \tag{75}$$

L2 projection must be used to obtain an approximation of the second-order terms in (75), assuming first-order, linear elements.

3.1.6 Stabilization Parameters

The stabilization parameters as implemented in the code are enumerated.

3.1.6.1 Poisson Stabilization parameter

For the Poisson equation, a diffusion-only equation, we have implemented the following

$$\tau_\psi = \left[\frac{3\left(\lambda^2 \varepsilon_r\right)^2}{\|\mathbf{g}_c\|^2} + \left(\frac{\partial S_\psi}{\partial \psi} \right)^2 \right]^{-\frac{1}{2}}. \tag{76}$$

where S_ψ is the source term for the Poisson equation. In general the value of S_ψ is $(p-n+C)$ and a derivative of this with respect to ψ is not available. To overcome this you can use an equilibrium approximation

$$S_\psi = n_i \left(e^{\frac{\psi}{V_T}} - e^{-\frac{\psi}{V_T}} \right). \tag{77}$$

where n_i is the intrinsic carrier concentration and V_T is the thermal voltage, both of which are constants.

3.1.6.2 Electron and Hole Stabilization Parameters

We have implemented several stabilization parameters typically used for convection-diffusion-reaction systems. The first one was

$$\tau = \left[\frac{2}{\Delta t} + \sqrt{\mathbf{u}\mathbf{g}_c\mathbf{u}^T} + \sqrt{3.0} D_{n|p} \|\mathbf{g}_c\| + \left| \frac{\partial G}{\partial n|p} \right| \right]^{-1} \tag{78}$$

where

$\mathbf{g}_c$ – The element contravariant tensor,

$D_{n|p}$ – The diffusion coefficient for the n or p carrier,

$\mathbf{u}$ – The velocity defined as:

for electrons (n): $\mu_n \nabla\psi$,

for holes (p): -$\mu_p \nabla\psi$.

Note that if the problem is steady-state then the Δt term will be absent. If we further assume the generation (or reaction) term is negligible then the $\partial G/\partial n|p$ term can also be neglected.

Another stabilization definition commonly used is

$$\tau = \left[\left(\frac{2}{\Delta t} \right)^2 + \mathbf{u}\mathbf{g}_c\mathbf{u}^T + 3\left(D_{n|p} \|\mathbf{g}_c\| \right)^2 + \left| \frac{\partial G}{\partial n|p} \right|^2 \right]^{-\frac{1}{2}}. \tag{79}$$

Finally the value of the stabilization parameter obtained when solving a one-dimensional convection-diffusion equation, with constant coefficients, is used. That is

$$\tau = \frac{1}{\sqrt{\mathbf{u}\mathbf{g}_c\mathbf{u}^T}} \left(\frac{1}{\tanh(\alpha)} - \frac{1}{\alpha} \right), \tag{80}$$

where

$$\alpha = \frac{\sqrt{\mathbf{u}\mathbf{g}_c\mathbf{u}^T}}{D_{n|p} \|\mathbf{g}_c\|}. \tag{81}$$

3.2 Scalar Electric Current Calculation

While calculation of the carrier densities is important, most engineers and analysts are more interested in the scalar electric current at the contacts. The electron and hole current densities are given by (5) and (6), respectively. One approach to calculating the scalar electric current at a contact is to take the solutions for ψ, n and p, using these in (5) and (6), integrating the resulting expressions for $\mathbf{J}_n$ and $\mathbf{J}_p$ over the surfaces associated with electrical contacts and finally summing these quantities to obtain the scalar electric current.

In his book Gresho (see [12], p. 853) argues that calculating fluxes with this approach is not as accurate, and in some sense is in fact inconsistent with the finite-element discretization. A consistent method of calculating fluxes incorporates the fluxes already contained in the weak equations. In this case the relevant terms come from (24) and (25). Gathering the surface terms in (24)

$$\int_{\Gamma} (\mu_n n \nabla \psi - D_n \nabla n) \cdot \hat{\eta} w d\Gamma \tag{82}$$

$$\int_{\Gamma} - (\mu_p p \nabla \psi + D_p \nabla p) \cdot \hat{\eta} w d\Gamma \tag{83}$$

Using the notation of (5) and (6), (82) and (83) can be rewritten as

$$- \int_{\Gamma} \mathbf{J}_n \cdot \hat{\eta} w d\Gamma \tag{84}$$

$$\int_{\Gamma} \mathbf{J}_p \cdot \hat{\eta} w d\Gamma \tag{85}$$

Using this notation the weak forms in (24) and (25) can be rewritten as

$$\int_{\Gamma} \mathbf{J}_n \cdot \hat{\eta} w d\Gamma = \int \dot{n} w d\Omega - \int \mu_n n \nabla \psi \cdot \nabla w d\Omega + \int D_n \nabla n \cdot \nabla w d\Omega + \int G w d\Omega, \tag{86}$$

$$- \int_{\Gamma} \mathbf{J}_p \cdot \hat{\eta} w d\Gamma = \int \dot{p} w d\Omega + \int \mu_p p \nabla \psi \cdot \nabla w d\Omega + \int D_p \nabla p \cdot \nabla w d\Omega + \int G w d\Omega. \tag{87}$$

Note that for the problems considered here the surface terms in the weak forms are typically not used because either there is a zero-flux condition at the boundaries, or there are specified Dirichlet conditions.

Once the solution is obtained for the primary degrees of freedom the right-hand sides of (86) and (87) contain known quantities, thus the fluxes can be solved for as a small linear problem. Since this is derived directly from the weak form it generally exhibits the same second-order convergence as the finite-element method and gives more accurate results.

3.3 Results

A simple test problem geometry is illustrated in Figure 1. While displaying a two-dimensional geometry, the physics of the problem is actually one-dimensional due to the invariance in

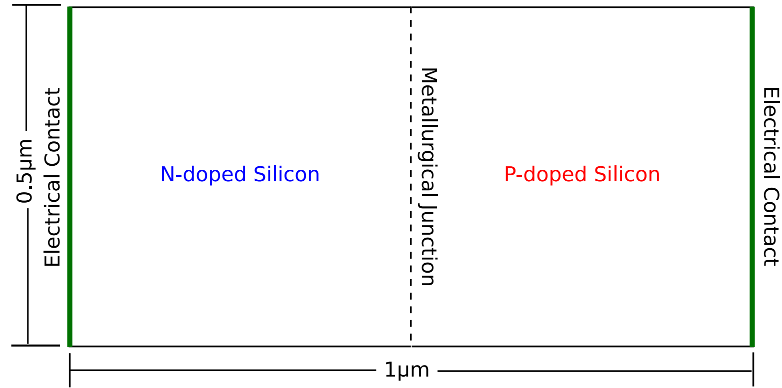


Figure 1. A simple PN-junction diode used as a test problem in *Charon* and *Pisces*.

the y direction, which makes it ideal for a test problem. The doping associated with the geometry shown in Figure 1 is shown in Figure 2.

Figure 3 shows the results of a simulation run using a pure Galerkin discretization compared to the same run using the additional *SUPG* terms in (55). It is obvious from that figure that the SUPG is very effective at eliminating the spurious oscillations associated with the Galerkin discretization of a convection-diffusion equation. It should be noted that similar results are obtained for the hole concentration. Figure 4 shows the scalar electric currents calculated using a Galerkin FEM formulation, a SUPG FEM formulation and the same problem solved using Pisces, a legacy code that utilizes the Scharfetter-Gummel method. All of the problems utilized a grid with 10 elements in the y-direction and 160 elements in the x-direction.

3.3.1 Comparison of Discretization

To compare the relative effectiveness of the finite-element discretization to the effectiveness of the more widely utilized Scharfetter-Gummel (*SG*), or "box", method the problem illustrated in Figure 1 was used whose doping profile is illustrated in Figure 2. An IV-sweep was performed in both *Charon* and *Pisces* from 0 volts to 1.0 volt in 0.05 volt increments. For *Pisces* the initial mesh density in x was 20 cells and this was doubled until the maximum density of 640 cells in x was obtained. For *Charon* the initial mesh density was 20 elements in x and this was doubled until the maximum number of cells in x was $10,240$.

Unless otherwise noted the *Charon* results are only utilizing the *SUPG* discretization. *VMS* results are very preliminary at this stage, and this preliminary result will be shown as a separate result.

The self-convergence of *Pisces* is shown in Figure 5. The value was calculated by

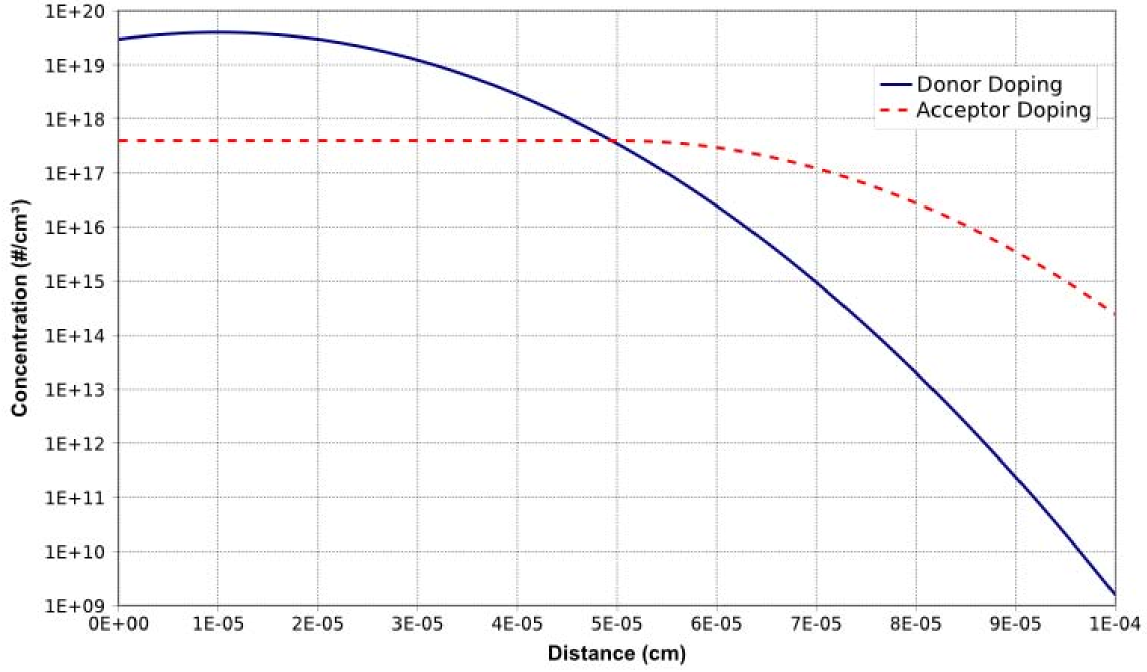


Figure 2. Doping associated with test problem.

taking the highest mesh density, 640 in the case of *Pisces*, and using that as a baseline. Then for each lower mesh density the percentage difference was calculated. The formula is given by

$$\text{Error} = \left.\frac{I_n - I_N}{I_N}\right|_{\text{bias}=V_b} \tag{88}$$

where, for *Pisces*,

$$\begin{aligned} n &= [20, 40, 80, 160, 320]\,, \\ N &= 640, \\ V_b &= [0.1, 0.3, 0.5, 0.8, 1.0]\,V, \end{aligned}$$

and for *Charon*

$$\begin{aligned} n &= [20, 40, 80, 160, 320, 640, 1280, 2560, 5120]\,, \\ N &= 10240, \\ V_b &= [0.1, 0.3, 0.5, 0.8, 1.0]\,V. \end{aligned}$$

Referring to Figures 5 and 6, in general *Charon* achieves a second-order rate of convergence more quickly than *Pisces*. However, *Pisces* has consistently lower errors. Both methods/codes display a rather disturbing convergence behavior at low bias levels. Another way to quantify this is to ask at what mesh density, for a given bias, is the result within one

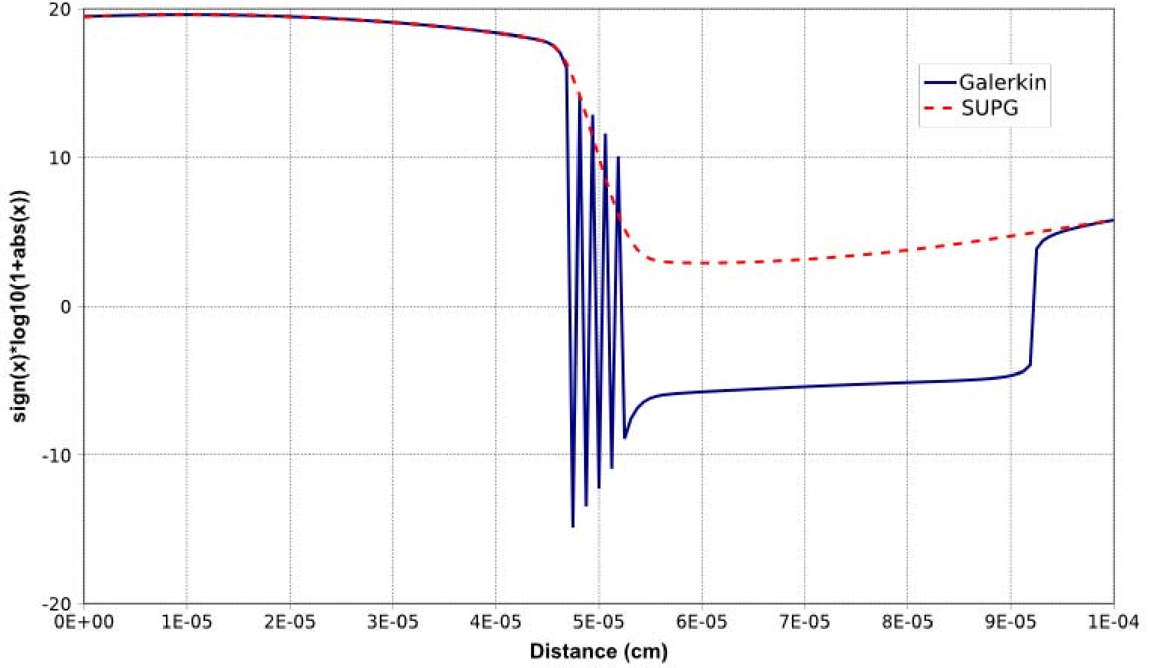


Figure 3. Comparison of a Galerkin discretization compared to the same problem with added *SUPG* stabilization.

percent of the "self-converged" result. Referring to Figure 5, and using $0.5V$ as the bias, *Pisces* would require approximately 70 cells in x to achieve the desired result. Referring to Figure 6, again using a bias of $0.5V$, *Charon* would require approximately 500 elements in x to achieve the same result.

For *Charon* there are technologies being investigated to reduce the gap between the *SG* method and the nodal finite-element method used by *Charon*. One method known to be very effective is stabilization combined with initial mesh refinement around the metallurgical junction of the device. Such refinement can be automatic and performed based on a simple calculation to determine the location of the metallurgical junction. An illustration of an *h*-refined mesh, using two levels of refinement, with an additional two layers surrounding the calculated junction location is shown for a coarse mesh in Figure 7 where the mesh is colored using the signed-log of the net doping, or

$$\text{Color} = \text{sign}(N_d - N_a)\log_{10}(1 + |N_d - N_a|)\,.$$

Referring to Figure 7 it should be noted that the algorithm utilized by *Charon* refines in both the x and y directions. In the pseudo-1D problem considered here the adaptation in y does not affect the accuracy of the results.

Using adaptive mesh refinement (*AMR*) and again requiring the current to match within one percent of the converged value at a forward bias of $0.5V$, the number of elements required by *Charon* in the x direction is approximately 110. This value was based on a starting

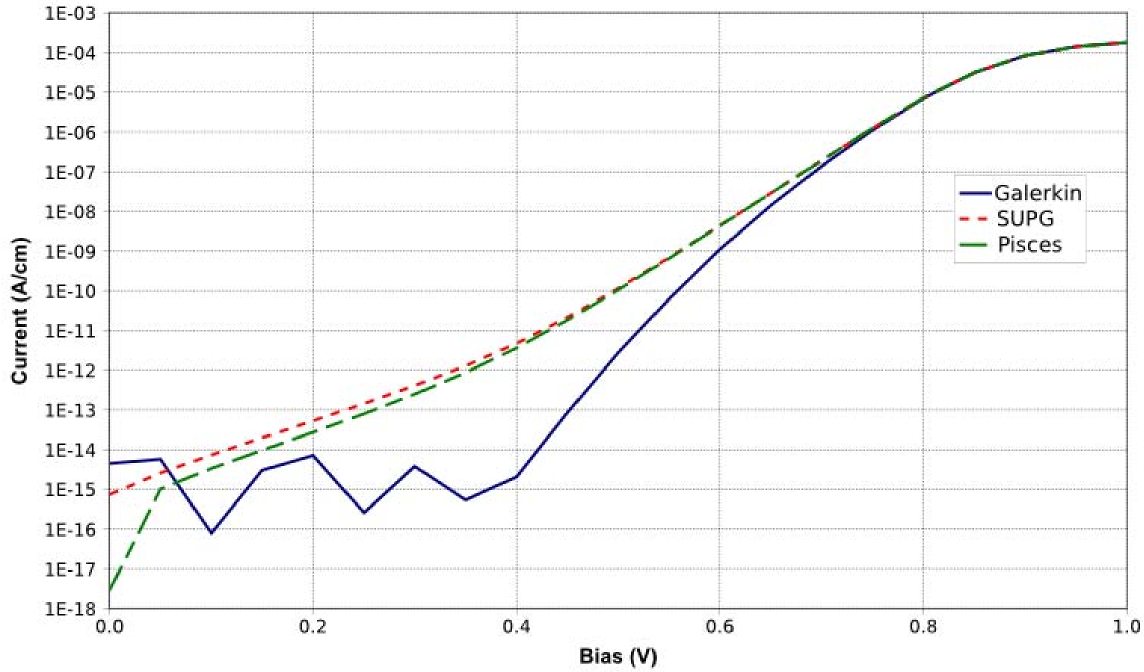


Figure 4. Comparison of scalar electric current for FEM Galerkin, FEM SUPG and Pisces (Scharfetter-Gummel) solutions.

mesh with 80 elements in the x direction and three levels of refinement at the metallurgical junction. Since neighboring elements are restricted to differ in the level of refinement by at most one this effectively causes layers of elements surrounding the junction to also be refined. Of course it should be noted that *AMR* can be used with most codes based on the *SG* method as well. A more complete comparison of these methods with *AMR* will be carried out in future work.

As derived in subsection 3.1, *VMS* is another possibility for reducing mesh density required by *Charon*. At present the only results are very preliminary and only for a true two-dimensional problem. The problem geometry is illustrated in Figure 8. Figure 9 shows the results of an *IV*-sweep for various mesh densities. Referring to that figure it is obvious that at the coarsest mesh density *VMS* offers improved accuracy. However, the improvement over *SUPG* seems to dissipate rapidly as the mesh is refined. These results were obtained with the *VMS* terms implemented in the Poisson equation only, (38). Work is ongoing to implement the required terms in the drift-diffusion equations for the electrons and holes.

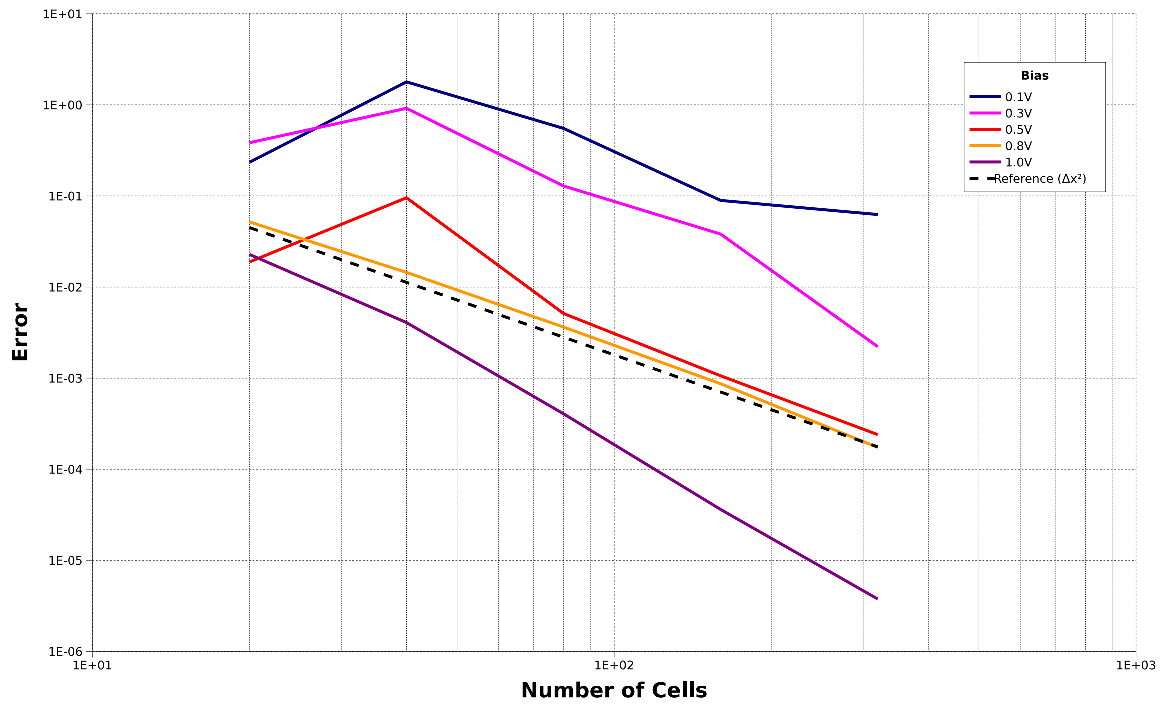


Figure 5. Self-convergence rate of *Pisces*.

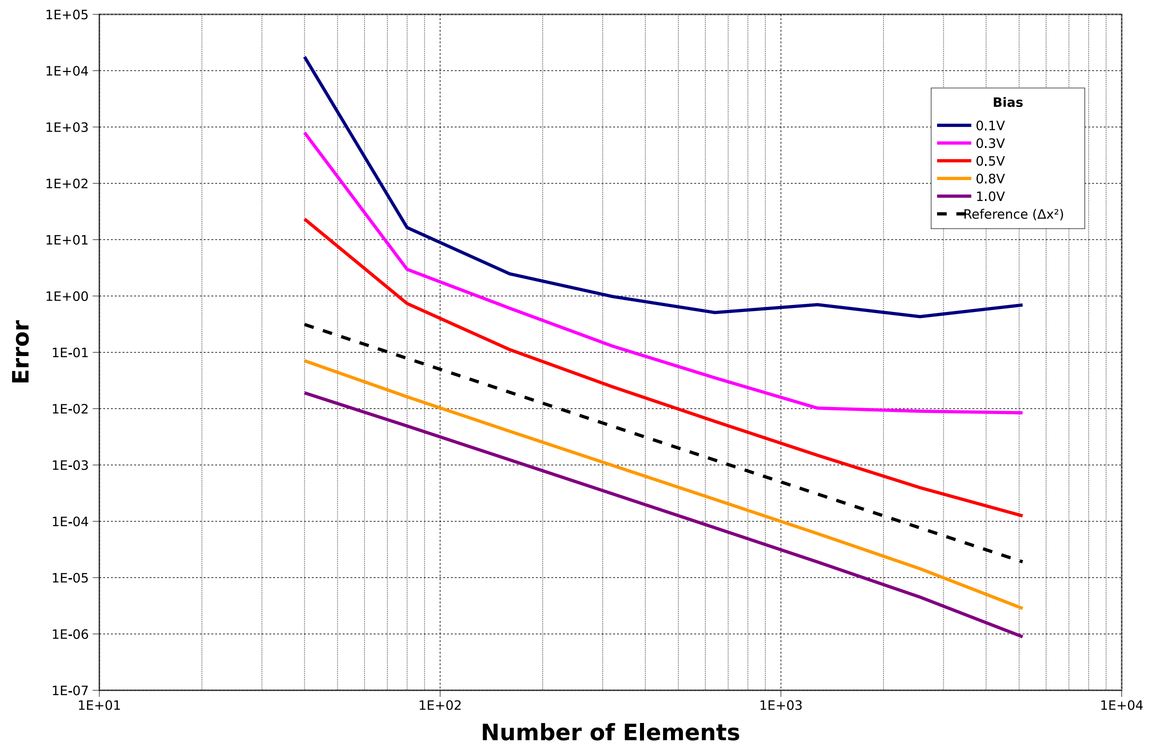


Figure 6. Self-convergence of *Charon*.

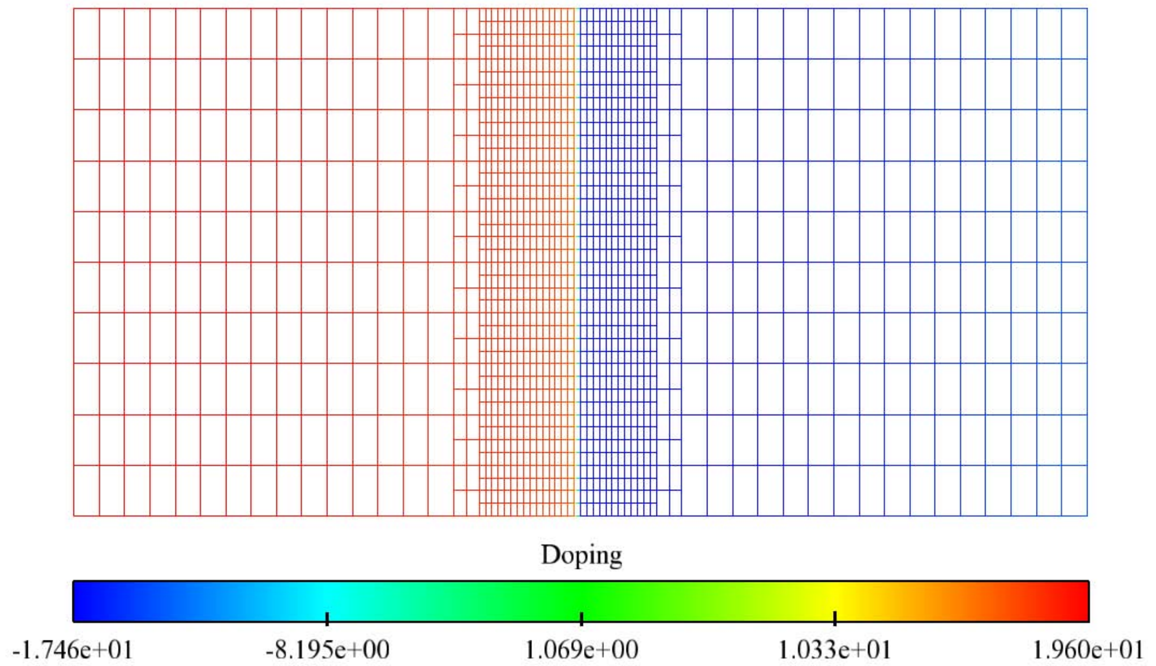


Figure 7. A *Charon* mesh using automatic, adaptive *h*-refinement. This mesh was refined by locating the elements containing the junction and refining that layer and two surrounding layers to a depth of two.

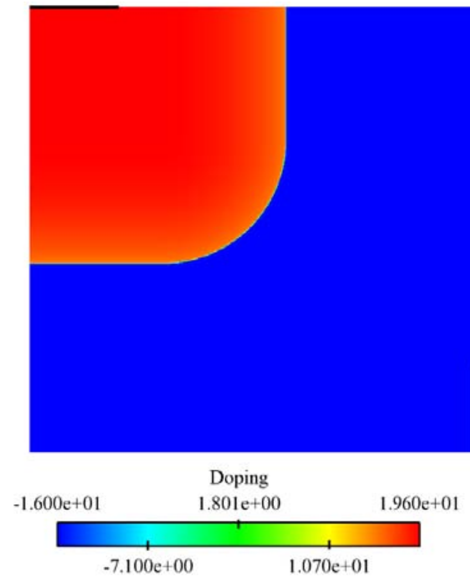


Figure 8. Geometry of a two-dimensional diode used as a *VMS* test problem. The diode measures 0.5μm on each side and the contacts are 0.2μm long. The coloring is based on the the signed log of the net doping, $N_d - N_a$.

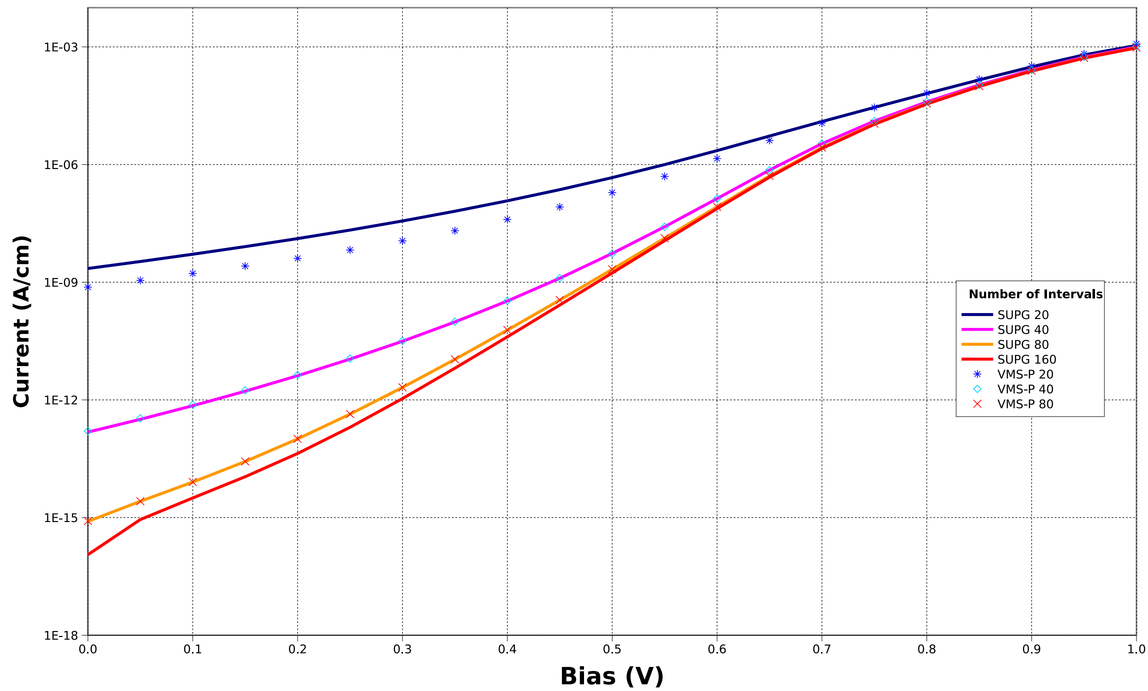


Figure 9. *IV* plot of the currents versus the forward bias of a 2D diode using a partial *VMS* discretization.

4 Modeling of Neutron Radiation Damage

This section describes the defect physics implementation necessary to model the neutron radiation damage of semiconductors within *Charon*. Neutron radiation causes displacement of silicon atoms within the semiconductor device. The displacement of the atoms in the lattice can be modeled as interstitials, vacancies and other defect complexes within the silicon lattice. Such defects can be modeled as additional "species", possibly electrically charged, within the material. These species can be transported either via diffusion, in the case of uncharged species, or via both diffusion and convection, by the electric field, in the case of charged species.

4.1 Drift Diffusion Transport Of Defects

The defect chemistry implemented within Charon is based upon the defect chemistry implemented within the 1D code written by Sam Myers. A detailed description of the 1D implementation is described within Bill Wampler's 1D theory document [36]. In general, the defect species have properties (mobility, diffusion coefficient, etc.) that are very similar to the carriers (electrons and holes), where the primary difference is in the magnitude of these properties. But the transport behavior of the defects can be significantly different than the carriers due to the following factors:

- Defects can be neutral and thus these particular defects will have zero mobility and also will not have a direct impact upon Poisson's equation.
- Defects can be immobile and thus have zero mobility and zero diffusivity.
- Primarily due to the bulleted items above and as stated earlier, the diffusion coefficients and mobilities can be many orders of magnitude smaller than then the carriers' properties; defect species transport can be source term dominated.

Note on the first bullet, that since some of the defect species are charged these particular species will directly contribute to Poisson's equation. Also, since the defect reactions may be non-reversible, we can no longer assume a common generation/recombination source term for all species. Thus the drift diffusion equations have the following form:

$$\nabla \cdot \left(\lambda^2 \mathbf{E}\right) = \left(p - n + C \sum_{i=1}^{N} Z_i Y_i\right) \text{ and } \mathbf{E} = -\nabla\psi \text{ in } \Omega \tag{89}$$

$$\nabla \cdot \mathbf{J}_n = \frac{\partial n}{\partial t} + R_n\left[\psi, n, p, Y_1, \ldots, Y_N\right] \text{ and } \mathbf{J}_n = \mu_n n \mathbf{E} + D_n \nabla n \text{ in } \Omega \tag{90}$$

$$-\nabla \cdot \mathbf{J}_p = \frac{\partial p}{\partial t} + R_p\left[\psi, n, p, Y_1, \ldots, Y_N\right] \text{ and } \mathbf{J}_p = \mu_p p \mathbf{E} - D_p \nabla p \text{ in } \Omega \tag{91}$$

$$-\nabla \cdot \mathbf{J}_{Y_i} = \frac{\partial Y_i}{\partial t} + R_{Y_i}\left[\psi, n, p, Y_1, \ldots, Y_N\right] \text{ and } \mathbf{J}_{Y_i} = \mu_{Y_i} Y_i \mathbf{E} - D_{Y_i} \nabla Y_i \text{ in } \Omega \tag{92}$$

Equations 89, 90, 91, 92 are a first order, coupled nonlinear system of PDE's in terms of the primal variables (Note: there are N number of (92) where N is the total number of defect species):

ψ: scalar electric potential;

n: electron concentration;

p: hole concentration;

Y_i: concentration of defect species i for $i = 1, ..., N$;

and the dual variables:

$\mathbf{E}$: electric field;

$\mathbf{J}_n$: electron current density;

$\mathbf{J}_p$: hole current density;

$\mathbf{J}_{Y_i}$ current density of defect species i for $i = 1, ..., N$.

The drift diffusion model involves two important types of functions; doping and source terms which are described below:

$C = N_D - N_a$: static doping profile, where N_D is the impurity donor density and N_a is the impurity acceptor density. Note that if the dopants are treated as a defect species that this parameter may be set to zero;

R_n: electron generation and recombination term;

R_p: hole generation and recombination term;

R_{Y_i}: generation and recombination term of defect species i.

There are also several material parameters that describe diffusivity and mobility of carrier or defect within the material:

λ: minimal Debye length of device;

D_n: electron diffusivity;

D_p: hole diffusivity;

D_{Y_i}: diffusivity of defect species i;

μ_n: electron mobility;

μ_p: hole mobility;

μ_{Y_i}: mobility of defect species i;

Z_i: integer charge number of defect species i.

The defect mobility and diffusivity are related by Einstein's relationship via

$$D_{Y_i} = \frac{kT}{q_{Y_i}} \mu_{Y_i} \tag{93}$$

Where

k: Boltzmann's constant;

T: lattice (material) temperature (typically 300 K);

q_{Y_i}: $\equiv Z_i q$, where q is the Coulumb charge.

The diffusivity is calculated with a model (derived using Fick's Law) that includes thermally activated diffusion of silicon vacancies, and silicon and boron interstitial atoms described by [35]

$$D_{Y_i} = D_{0_i} \exp\left(\frac{E_{A_i}}{kT}\right) \tag{94}$$

where,

D_{0_i}: diffusion pre-factor of defect species i;

E_{A_i}: activation energy of defect species i.

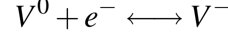
Within Charon, the diffusivity of a defect is calculated with (94) and then the mobility is calculated via Einstein's relation with (93).

4.2 Defect Reactions and Source Terms

Critical and sometimes dominant aspects of the defect density transport are the recombination/generation source terms. Thus, it will be of significant importance to verify the models that are used to model these terms. At the highest level there are two types of reactions:

- Carrier-Defect reactions
- Defect-Defect reactions

Within Charon, there is a need to break these primary reactions up into a larger set of reactions to appropriately identify, capture, and calculate the reactions. Within the 1D code all reversible reactions are captured within a single expression, but in order to use our input file structure it was easier to express all reactions into their non-reversible form. For example electron capture and electron emission may be expressed as a single reversible reaction:



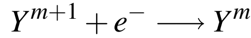
$$V^0 + e^- \longleftrightarrow V^-$$

or it may be expressed as two separate reactions:

$$V^0 + e^- \longrightarrow V^- \text{ and } V^- \longrightarrow V^0 + e^-.$$

The reaction types within Charon are listed below with the reaction form and the corresponding reaction equation. For a given species Y_i, the source/sink term, $R(Y_i)$, is expressed in terms of the recombination/generation source terms. The reactions of type charge-charge reactions only include reaction with species of opposing charge. Reactions of species with like charge are currently neglected (very small cross sections). Note that superscripts denote a given charge state and that holes are designated as h^+ within reactions rather than p so that they are not confused with protons.

- Electron Capture

  $$Y^{m+1} + e^- \longrightarrow Y^m$$

  $$\begin{aligned} R\left[Y^{m+1}, e^-\right] &= \sigma\left[Y^{m+1}, e^-\right] \nu_n n Y^{m+1} \\ R\left[Y^{m+1}\right] &= -R\left[Y^{m+1}, e^-\right] \\ R\left[e^-\right] &= -R\left[Y^{m+1}, e^-\right] \\ R\left[Y^m\right] &= R\left[Y^{m+1}, e^-\right] \end{aligned}$$

  where,

  - $\sigma\left[Y^{m+1}, e^-\right]$: electron capture cross-section for species Y^{m+1};
  - ν_n: electron thermal velocity within material (for Si = 2.3e7 cm/s @ 300K);

- Electron Emission

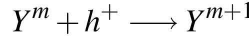
  $$Y^m \longrightarrow Y^{m+1} + e^-$$

  $$\begin{aligned} R\left[Y^m \mapsto e^-\right] &= \sigma\left[Y^{m+1}, e^-\right] \nu_n N_c Y^m \frac{\gamma\left(Y^{m+1}\right)}{\gamma\left(Y^m\right)} \exp\left(\frac{E_{Y^m} - E_c}{kT}\right) \\ R\left[Y^{m+1}\right] &= -R\left[Y^m \mapsto e^-\right] \\ R\left[e^-\right] &= R\left[Y^m \mapsto e^-\right] \\ R\left[Y^m\right] &= R\left[Y^m \mapsto e^-\right] \end{aligned}$$

  where,

- N_c: effective density of states in conduction band (for Si = 2.86e19 cm-3 @ 300K);
- $\gamma\left(Y^{m+1}\right), \gamma(Y^m)$: state degeneracies for Y^{m+1} and Y^m respectively (these values are typically 1.0 for the majority of defect species);
- E_{Y^m}: energy of the electron state associated with Y^m;
- E_c: conduction band minimum energy;
- $E_{Y^m} - E_c \equiv \Delta E_e^{Y^m}$: activation energy for electron emission of species Y^m;

- Hole Capture

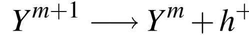


$$\begin{aligned} R\left[Y^{m+1}, h^+\right] &= \sigma\left[Y^m, h^+\right] \nu_p p Y^m \\ R\left[Y^m\right] &= -R\left[Y^m, h^+\right] \\ R\left[h^+\right] &= -R\left[Y^m, h^+\right] \\ R\left[Y^{m+1}\right] &= R\left[Y^m, h^+\right] \end{aligned}$$

where,

- $\sigma\left[Y^{m+1}, h^+\right]$: hole capture cross-section for species ;
- ν_p: hole thermal velocity within material (for Si = 1.9e7 cm/s @ 300K);

- Hole Emission



$$\begin{aligned} R\left[Y^{m+1} \mapsto h^+\right] &= \sigma\left[Y^m, h^+\right] \nu_p N_v Y^{m+1} \frac{\gamma(Y^m)}{\gamma(Y^{m+1})} \exp\left(\frac{E_v - E_{Y^{m+1}}}{kT}\right) \\ R\left[Y^{m+1}\right] &= -R\left[Y^{m+1} \mapsto h^+\right] \\ R\left[h^+\right] &= R\left[Y^{m+1} \mapsto h^+\right] \\ R\left[Y^m\right] &= R\left[Y^{m+1} \mapsto h^+\right] \end{aligned}$$

where,

- N_v: effective density of states in valence band (for Si = 2.66e19 cm-3 @ 300K);
- $\gamma\left(Y^{m+1}\right), \gamma(Y^m)$: state degeneracies for Y^{m+1} and Y^m respectively (these values are typically 1.0 for the majority of defect species);
- $E_{Y^{m+1}}$: energy of the hole state associated with Y^{m+1};
- E_v: valence band maximum energy;
- $E_v - E_{Y^{m+1}} \equiv \Delta E_h^{Y^{m+1}}$: activation energy for hole emission of species ;

- Complex Recombination Charge-Charge

$$X^m + Y^n \longrightarrow XY^{m+n} \text{ where } m,n \neq 0 \text{ and } m \times n < 0$$

$$\begin{aligned}
R\left[X^m, Y^n\right] &= 4\pi r_{Coul}\left\{D\left(X^m\right) + D\left(Y^n\right)\right\} X^m Y^n \\
R\left[X^m\right] &= -R\left[X^m, Y^n\right] \\
R\left[Y^n\right] &= -R\left[X^m, Y^n\right] \\
R\left[XY^{m+n}\right] &= R\left[X^m, Y^n\right]
\end{aligned}$$

where,

  - $D\left(X^m\right)$: diffusion coefficient of species X^m;
  - $D\left(Y^n\right)$: diffusion coefficient of species Y^n;
  - r_{Coul}: the reaction distance based on when the Coulumbic attraction is equivalent to kT and can be expressed as

    $$r_{Coul} = \frac{|m \times n|\left(1.40e - 4 \text{ cm} \cdot \text{K}\right)}{T}$$

    (Note: this only includes reactions where $m \times n < 0$, in other words species with opposite charge):

- Complex Dissociation Charge-Charge

$$XY^{m+n} \longrightarrow X^m + Y^n \text{ where } m,n \neq 0 \text{ and } m \times n < 0$$

$$\begin{aligned}
R\left[XY^{m+n} \mapsto\right] &= 4\pi r_{Coul}\left\{D\left(X^m\right) + D\left(Y^n\right)\right\}[Si] XY^{m+n} \times \\
&\quad \times \frac{\gamma\left(X^m\right)\gamma\left(Y^n\right)}{\gamma\left(XY^{m+n}\right)} \exp\left(-\frac{\Delta E\left(XY^{m+n}\right)}{kT}\right) \\
R\left[X^m\right] &= R\left[XY^{m+n} \mapsto\right] \\
R\left[Y^n\right] &= R\left[XY^{m+n} \mapsto\right] \\
R\left[XY^{m+n}\right] &= -R\left[XY^{m+n} \mapsto\right]
\end{aligned}$$

where,

  - $[Si]$: the number of Si atoms per unit volume;
  - $\gamma\left(X^m\right), \gamma\left(Y^n\right), \gamma\left(XY^{m+n}\right)$: state degeneracies for X^m, Y^n, and XY^{m+n} respectively (these values are typically 1.0 for the majority of defect species);
  - $\Delta E\left(XY^{m+n}\right)$: binding energy of species XY^{m+n};

- Complex Recombination Charge-Neutral

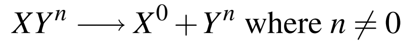
$$X^0 + Y^n \longrightarrow XY^n \text{ where } m,n \neq 0 \text{ and } m \times n < 0$$

$$\begin{aligned} R\left[X^0, Y^n\right] &= 4\pi r_{lattice} \left\{D\left(X^0\right) + D\left(Y^n\right)\right\} X^0 Y^n \\ R\left[X^0\right] &= -R\left[X^0, Y^n\right] \\ R\left[Y^n\right] &= -R\left[X^0, Y^n\right] \\ R\left[XY^n\right] &= R\left[X^0, Y^n\right] \end{aligned}$$

where,

  - $r_{lattice}$: approximately the order of the lattice parameter within Si (5e-8 cm);
  - X^0: is any neutral defect;

- Complex Dissociation Charge-Neutral



$$XY^n \longrightarrow X^0 + Y^n \text{ where } n \neq 0$$

$$\begin{aligned} R\left[XY^n \mapsto\right] &= 4\pi r_{Coul} \left\{D\left(X^0\right) + D\left(Y^n\right)\right\} [Si] XY^n \frac{\gamma\left(X^0\right)\gamma\left(Y^n\right)}{\gamma\left(XY^n\right)} \exp\left(\frac{-\Delta E\left(XY^n\right)}{kT}\right) \\ R\left[X^0\right] &= R\left[XY^n \mapsto\right] \\ R\left[Y^n\right] &= R\left[XY^n \mapsto\right] \\ R\left[XY^n\right] &= -R\left[XY^n \mapsto\right] \end{aligned}$$

where,

  - $\gamma\left(X^0\right), \gamma\left(XY^n\right)$: state degeneracies for X^0 and XY^n respectively (these values are typically 1.0 for the majority of defect species);
  - $\Delta E\left(XY^n\right)$: binding energy of species XY^n;

- Complex Recombination Neutral-Neutral - this is currently equivalent to Complex Recombination Charge-Neutral but may be handled as a separate case if need be in the future.

- Complex Dissociation Neutral-Neutral - this is currently equivalent to Complex Dissociation Neutral-Neutral but may be handled as a separate case if need be in the future.

- Annihilation Charge-Charge

$$X^m + V^n \longrightarrow [X] \text{ where } m = -n \text{ and } m \times n < 0$$

$$\begin{aligned} R[X^m, V^n] &= 4\pi r_{Coul} \{D(X^m) + D(V^n)\} X^m V^n \\ R[X^m] &= -R[X^m, V^n] \\ R[V^n] &= -R[X^m, V^n] \end{aligned}$$

where,

  - V^n: charged vacancy;
  - $[X]$: lattice material;

- Annihilation Neutral-Neutral

$$X^0 + V^0 \longrightarrow [X]$$

$$\begin{aligned} R\left[X^0, V^0\right] &= 4\pi r_{lattice} \left\{D\left(X^0\right) + D\left(V^0\right)\right\} X^0 V^0 \\ R[X^m] &= -R\left[X^0, V^0\right] \\ R\left[V^0\right] &= -R\left[X^0, V^0\right] \end{aligned}$$

where,

  - V^0: neutral vacancy;

- Annihilation Charge-Charge Create Electron

$$X^m + V^n \longrightarrow [X] + e^- \text{ where } m + n = -1 \text{ and } m \times n < 0$$

$$\begin{aligned} R[X^m, V^n] &= 4\pi r_{Coul} \{D(X^m) + D(V^n)\} X^m V^n \\ R[X^m] &= -R[X^m, V^n] \\ R[V^n] &= -R[X^m, V^n] \\ R\left[e^-\right] &= R[X^m, V^n] \end{aligned}$$

- Annihilation Charge-Neutral Create Electron

$$X^m + V^n \longrightarrow [X] + e^- \text{ where } m + n = -1 \text{ and } m \times n \quad = \quad 0$$

$$\begin{aligned} R[X^m, V^n] &= 4\pi r_{lattice} \{D(X^m) + D(V^n)\} X^m V^n \\ R[X^m] &= -R[X^m, V^n] \\ R[V^n] &= -R[X^m, V^n] \\ R\left[e^-\right] &= R[X^m, V^n] \end{aligned}$$

- Annihilation Charge-Charge Create Hole

$$X^m + V^n \longrightarrow [X] + h^+ \text{ where } m+n=1 \text{ and } m \times n < 0$$

$$\begin{aligned}
R[X^m, V^n] &= 4\pi r_{Coul}\left\{D(X^m) + D(V^n)\right\} X^m V^n \\
R[X^m] &= -R[X^m, V^n] \\
R[V^n] &= -R[X^m, V^n] \\
R\left[h^+\right] &= R[X^m, V^n]
\end{aligned}$$

- Annihilation Charge-Neutral Create Hole

$$X^m + V^n \longrightarrow [X] + h^+ \text{ where } m+n=1 \text{ and } m \times n = 0$$

$$\begin{aligned}
R[X^m, V^n] &= 4\pi r_{lattice}\left\{D(X^m) + D(V^n)\right\} X^m V^n \\
R[X^m] &= -R[X^m, V^n] \\
R[V^n] &= -R[X^m, V^n] \\
R\left[h^+\right] &= R[X^m, V^n]
\end{aligned}$$

4.3 Boundary Condition Implementation of Defect Species

With the defect species implementation, the boundary conditions must be specified for each species and charged species must be appropriately handled within the doping dependent boundary conditions. We can rewrite Equations 89, 90, 91, 92 as a second order boundary value problem in terms of only the primal variables:

$$\begin{aligned}
\nabla \cdot \left(\lambda^2 \nabla \psi\right) - \left(p - n + \sum_{i=1}^{N} q_{Y_i} Y_i\right) &= C \text{ in } \Omega \\
\nabla \cdot \left(-\mu_n n \nabla \psi + D_n \nabla n\right) - \frac{\partial n}{\partial t} - R_n\left[\psi, n, p, Y_1, \ldots, Y_N\right] &= 0 \text{ in } \Omega \\
\nabla \cdot \left(\mu_p p \nabla \psi + D_p \nabla p\right) - \frac{\partial p}{\partial t} - R_p\left[\psi, n, p, Y_1, \ldots, Y_N\right] &= 0 \text{ in } \Omega \\
\nabla \cdot \left(\mu_{Y_i} Y_i \nabla \psi + D_{Y_i} \nabla Y_i\right) - \frac{\partial Y_i}{\partial t} - R_{Y_i}\left[\psi, n, p, Y_1, \ldots, Y_N\right] &= 0 \text{ in } \Omega
\end{aligned}$$

Equivalently, in residual form:

$$\nabla \cdot \left(\lambda^2 \nabla \psi\right) - \left(p - n + \sum_{i=1}^{N} q_{Y_i} Y_i\right) = C \text{ in } \Omega \tag{95}$$

$$\frac{\partial n}{\partial t} + \nabla \cdot (\mu_n n \nabla \psi) - \nabla \cdot (D_n \nabla n) + R_n [\psi, n, p, Y_1, ..., Y_N] = 0 \text{ in } \Omega \tag{96}$$

$$\frac{\partial p}{\partial t} - \nabla \cdot (\mu_p p \nabla \psi) - \nabla \cdot (D_p \nabla p) + R_p [\psi, n, p, Y_1, ..., Y_N] = 0 \text{ in } \Omega \tag{97}$$

$$\frac{\partial Y_i}{\partial t} - \nabla \cdot (\mu_{Y_i} Y_i \nabla \psi) - \nabla \cdot (D_{Y_i} \nabla Y_i) + R_{Y_i} [\psi, n, p, Y_1, ..., Y_N] = 0 \text{ in } \Omega \tag{98}$$

In a complete mathematical model for a semiconductor device, boundary conditions reflect the interaction of the device with the circuit which it is imbedded in. In the following we shall be most interested in the dc operating conditions. This means that the ties between two switching events are long enough for quasi steady state to be reached. In this case, most of the important information is contained in the static voltage-current characteristics, i.e. the relation between contact voltages and current through the contacts under steady conditions.

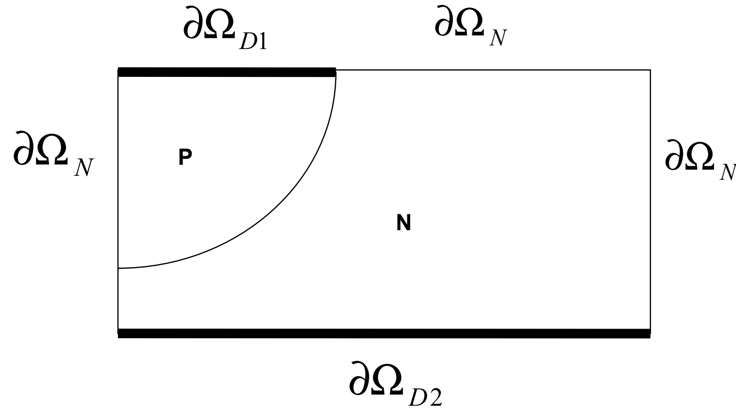


Figure 10. A simple PN-junction diode and its geometrical configuration in the two-dimensional case

Figure 10 shows a simple PN junction device; a PN diode and its geometrical configuration in the two dimensional case. The boundary conditions $\partial\Omega$ on the domain Ω are assumed to consist of a Dirichlet part $\partial\Omega_D$ and a Neumann part $\partial\Omega_N$ as follows:

$$\partial\Omega = \partial\Omega_D \cup \partial\Omega_N \, , \, \partial\Omega_D \cap \partial\Omega_N = \{\} \tag{99}$$

The Dirichlet part of the boundary corresponds to Ohmic contacts. There the potential ψ and the concentration n and p are prescribed. The boundary values are derived from the

following considerations. At Ohmic contacts the space charge, given by the right-hand side of Equation 95 vanishes. Enforcing quasi-neutrality results in

$$p - n + C + \sum_{i=1}^{N} q_{Y_i} Y_i = 0 \text{ for } x \in \partial\Omega_D. \tag{100}$$

Furthermore, the system is in thermal equilibrium there, which is express by the relation:

$$np = n_i^2 \text{ for } x \in \partial\Omega_D \tag{101}$$

where n_i is the intrinsic density ($\cong 10^{10}\ cm^{-3}$ in Si @ 300K). Plugging (101) into (100) and solving for n results in

$$n = \begin{cases} N_{quasi}, & N_{eff} > 0 \\ n_i^2/N_{quasi}, & N_{eff} < 0 \\ 0, & N_{eff} = 0 \end{cases} \quad \text{for } x \in \Omega_D \tag{102}$$

and

$$p = \begin{cases} n_i^2/N_{quasi}, & N_{eff} > 0 \\ N_{quasi}, & N_{eff} < 0 \\ 0, & N_{eff} = 0 \end{cases} \quad \text{for } x \in \Omega_D \tag{103}$$

where,

$$N_{quasi} = \left\{ N_{eff} + \left(N_{eff}^2 + 4n_i^2 \right)^{\frac{1}{2}} \right\} /2 \text{ and } N_{eff} = C + \sum_{i=1}^{N} q_{Y_i} Y_i \tag{104}$$

The potential at the contact is expressed as

$$\psi = \begin{cases} \ln\left(N_{quasi}/n_i\right) + \psi_{applied} & N_{eff} \geq 0 \\ & \\ -\ln\left(N_{quasi}/n_i\right) + \psi_{applied}, & N_{eff} < 0 \end{cases} \quad \text{for } x \in \Omega_D \tag{105}$$

where,

- $\psi_{applied}$: applied potential (bias) at the boundary;

When the doping is set statically (defined within C) the boundary condition within equations 102. 103, 104, and 105 need only be set once even for transient calculations. The reason for this is that the concentrations of charged defects at the contacts are typically set to zero via a Dirichlet boundary condition. When the dopants are treated as defect species and exhibit transient behavior at the boundary (this typically requires a reaction set with source/sinks of the dopant species), these boundary conditions must be treated transiently. We assume that the device geometry to be given by a domain of $\Omega \subseteq R^d$ with $d = 1, 2,$ or 3.

4.4 Neutron Damage Examples

To simulate the damage done to a device that is exposed to a radiation environment (including neutron, gamma, electron-beam, and ion-beam), a Frenkel pair source rate is used to simulate the damage done. When a neutron interacts with the device's lattice material it may "knock out" an atom within the lattice leaving a void (i.e., vacancy). A Frenkel pair is a vacancy defect within the lattice and the corresponding material interstitial (free material atom). A pulsed reactor environment will result in a transient and spatially dependent creation of Frenkel pairs due to radiation damage and the creation of electron-ion pairs due to ionization. How these source rates are quantified and generated is beyond the scope of this document and will not be discussed in detail here. The source rates are introduced within *Charon* as transient, spatially dependent creation source rates of the species and carriers, specifically vacancies, Si interstitials, electrons, and holes. The complete list of parameters that will vary for a given radiation environment and device are:

- Vacancy transient, spatially dependent source rate
  - this quantity is directly related to the Nickle Activity (neutron fluence dosimetry measurement)
- Si Interstitial transient, spatially dependent source rate
  - this quantity is directly related to the Nickle Activity (neutron fluence dosimetry measurement)
- electron transient, spatially dependent source rate
  - this quantity is directly related to the ionization dose
- hole transient, spatially dependent source rate
  - this quantity is directly related to the ionization dose
- Current at emitter contact (held constant within the experiments of interest)
- Temperature of device
- Base-Collector bias

The examples shown and described within this section are for an npn Si BJT device biased at 0.22 mA within the SPR-III (Sandia Pulse Reactor) reactor at Sandia National Laboratories. The reactor has two exposure locations: cavity and leakage. The cavity location provides a more direct exposure than the leakage location, thus the device is exposed to larger fluences. A simulation was done for devices at each location and the result was compared to the corresponding experimental result. The metric of comparison is the inverse gain where:

$$\text{Inverse Gain} = I_b/I_c$$

where,

- I_b: current at the base contact
- I_c: current at the collector contact

Although the gain better represents how functional the device is, the inverse gain is directly proportional to the amount of damage done to the device. Since there are large uncertainties within some of the physics parameters (e.g., cross-sections and diffusion prefactors), it was necessary to do a calibration of these physics parameters with the experimental data. Nominal values were calibrated for these physics parameters and then fixed within all the radiation damage simulations. Two simulation results and the corresponding experimental results are shown within Figure 11 and Figure 12 which represent the damage done to a npn Si BJT within the cavity and leakage position of SPR-III respectively. Below are the input values for each of the environments:

- SPR Cavity (shot #13245)
  - Ni Activity = 2.582×10^4 Bq/g
  - Total Dose = 1.082×10^5 *Rad Si*
  - Emitter Current = 2.296×10^{-4} A
  - Temperature of device = 18.5 ^0C
  - Collector Base Bias = 10.21 V
- SPR Cavity (shot #13361)
  - Ni Activity = 1.509×10^3 Bq/g
  - Total Dose = 4.395×10^3 *Rad Si*
  - Emitter Current = 2.210×10^{-4} A
  - Temperature of device = 24.3 ^0C
  - Collector Base Bias = 10.01 V

As can seen by Figure 11 and Figure 12 there is excellent agreement between the Charon simulations and the experimental results. Note that the oscillations at early times within the experimental data is due to thermal oscillations of the reactor which result in an oscillatory ionization response. This physics feature was deemed unimportant for our analysis and not reflected within the electron-ion creation source rates inputed into *Charon* and therefore not reflected within the response.

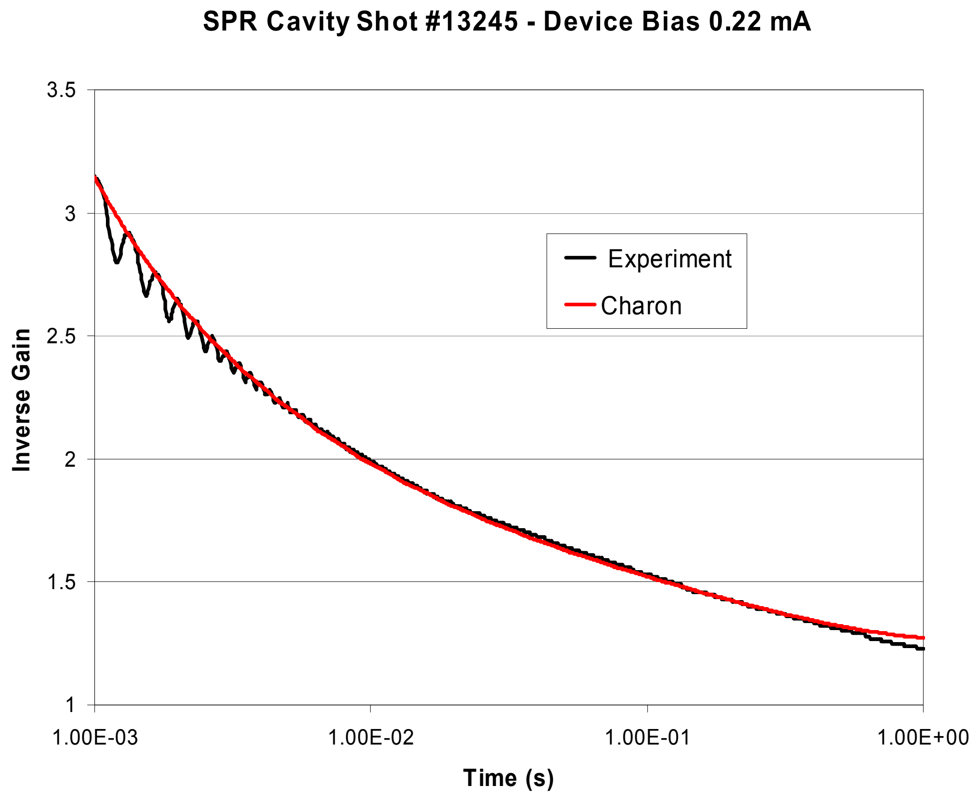


Figure 11. Neutron damage of a Si BJT biased at 0.22 mA within the cavity location of the SPR-III reactor.

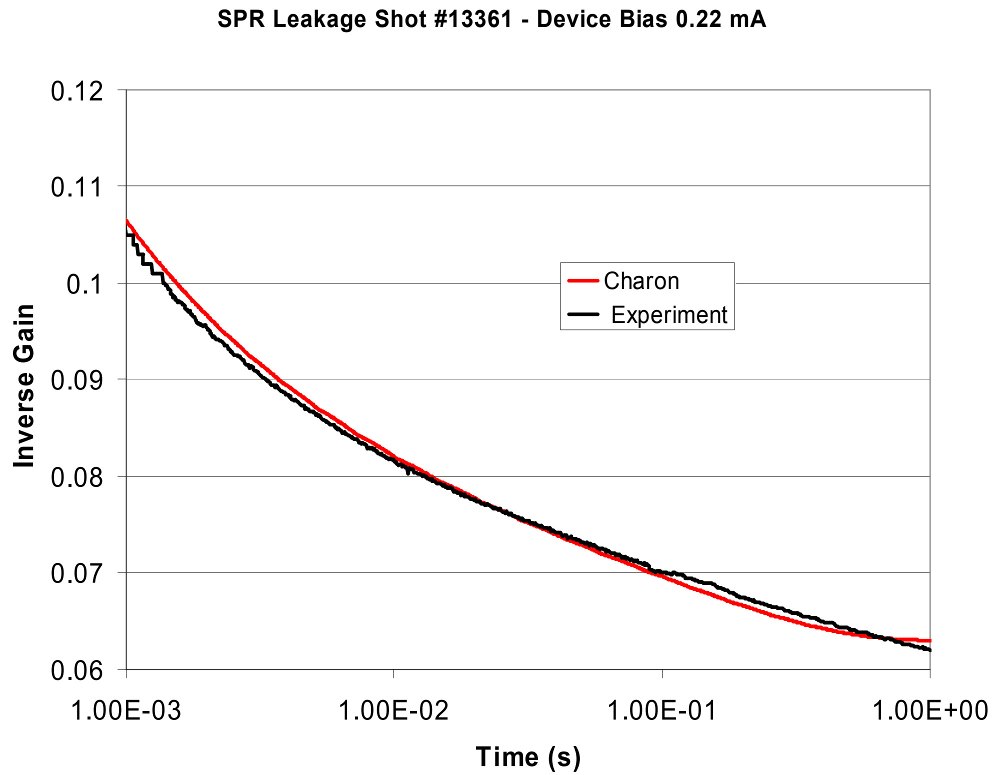


Figure 12. Neutron damage of a Si BJT biased at 0.22 mA within the leakage location of the SPR-III reactor.

5 Parallel Performance of Charon

It is well known that preconditioning can significantly reduce the solution times of large linear systems. The main approach used to date to improve the preconditioning of the linear systems in Charon is the use of multilevel preconditioners, specifically the multilevel preconditioners available in the ML library [10]. The ML library provides algebraic multigrid preconditioners, including an aggressive coarsening procedure. Recently, a smoothed aggregation preconditioner for nonsymmetric linear systems [29] has been implemented by the ML developers. For steady-state solutions, the use of this preconditioner reduces the solve time by roughly a factor of two compared with the baseline multilevel preconditioner. This new preconditioner performs restriction smoothing because the problem is nonsymmetric, and rather than using a single damping parameter, it calculates a local damping parameter for the smoothed aggregation.

The second approach to preconditioning that is being pursued is the use of physics-based preconditioners, for this case the use of preconditioners that are specifically tailored to the drift-diffusion equations. To date, various block preconditioners such as block Jacobi, block Gauss-Seidel, and block successive overrelaxation (SOR) have been implemented. More advanced physics-based preconditioners will be implemented in the future.

To demonstrate the parallel performance of Charon, figures 13 and 14 show a weak scaling study that compares the different preconditioners for a 2D $2 \times 1.5\mu m$ model npn BJT, with $0.1\mu m$ contacts. A steady-state drift-diffusion calculation was performed with voltage bias of 0.3V, with the solution of the nonlinear Poisson problem as an initial guess. The comparison concerns a 1-level domain decomposition ILU preconditioner, compared with two ML preconditioners. Both ML preconditioners use a 3-level aggressive coarsening method, and use ILU as smoother on the fine and medium mesh and a direct solver on the coarsest level. About 85 nodes were used to form each aggregate. The green line denotes the result for the baseline ML preconditioner (referred to as “no EMIN” in the plot), while the blue line denotes the result for the ML preconditioner for nonsymmetric linear systems (referred to as “with EMIN” in the plot). These calculations were performed on the Red Storm machine and were scaled up to 1024 nodes (one 2.4-Gz core per node was used). The time reported is the average time to construct the preconditioner and perform the linear solve for one Newton step. Note that for the calculation run on 1024 processors with 28 million unknowns, the ML preconditioner for nonsymmetric systems performs significantly better than the baseline ML preconditioner, and both ML preconditioners perform significantly better than the 1-level preconditioner.

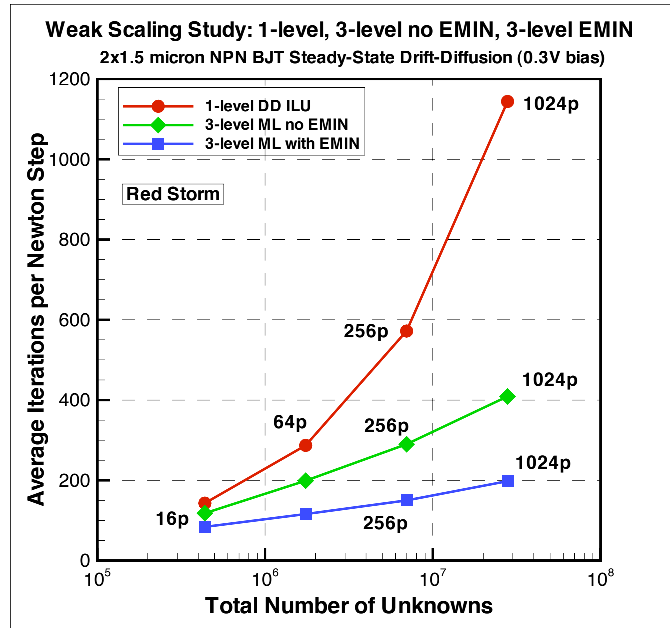


Figure 13. Weak scaling study comparing average iteration per Newton step for the different preconditioners

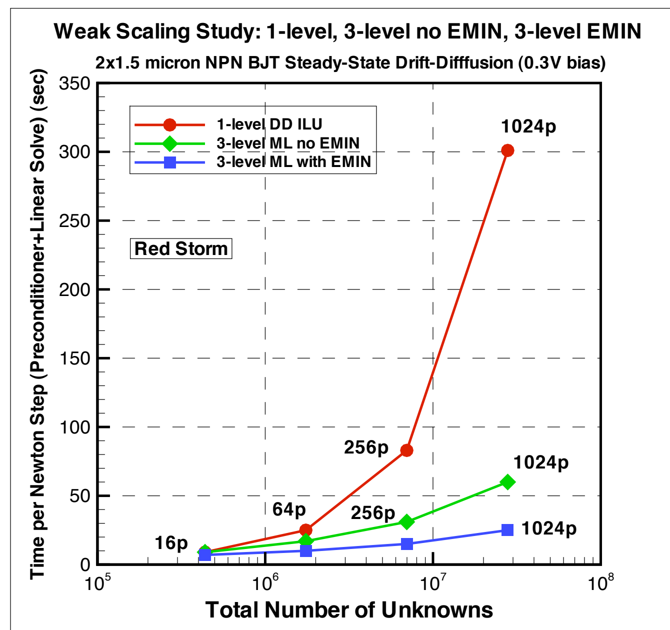


Figure 14. Weak scaling study comparing average iteration per Newton step for the different preconditioners

6 Future Work

The development of *Charon* has been driven primarily by the needs of the *QASPR* project. In the near future that will continue to be the case. The first phase of *QASPR* was to model a *BJT* under normal operating conditions and obtain the correct gain when compared to experiment. When this was accomplished the next step was to add the defects package and again match to experimental data. Much of this work was done with a pseudo-1D model of a *BJT*. At present we are attempting to do those same steps on a full two-dimensional model. This requires significantly more computational power and is thus being carried out on the largest computing systems available at Sandia.

The next major developmental step is to model oxide effects, which are believed to have a significant effect on the operation of junction devices [15, 8]. This may involve calibration of simple models, such as a constant E-field boundary condition on the surface, or surface traps, or a combination, or it may involve implementing a true multi-physics capability within *Charon* wherein different regions of the problem have different degrees of freedom. In this case one region would be a semiconductor material with mobile carriers and another would be an insulator with no mobile carriers.

The end goal of all the *QASPR* work is to satisfy the customers that simulation of a neutron environment can accurately capture the physics and be truly predictive and thus allow qualification of current weapons systems, as well as future weapons systems.

Many newer designs for weapons systems, and for replacement parts in existing weapon systems, are making use of Gallium-Arsenide (GaAs), and other III-V semiconductor devices. In the future *Charon* will likely need to model such devices. This presents a challenge in that doping concentrations are typically higher in such devices and a new set of material properties corresponding to GaAs will need to be added.

Another major step would be the capability of modeling MOSFETs within *Charon*. Most modern systems heavily utilize MOSFETs in their electronics. One application in this area would be space-based electronics in satellite systems, an area in which Sandia is heavily invested. Along this same line we are considering what steps would be necessary to model sub-micron devices within *Charon*. This would likely entail the implementation of more empirical models and perhaps utilize Boltzmann transport theory more closely than does a pure drift diffusion approach.

Modeling semiconductor devices with high fidelity in two-dimensions requires not only the largest computing systems available, but improvements in preconditioning for reducing the solve time for the large linear systems. The multilevel methods in the ML library have achieved a reasonable amount of success, but further work is needed in this area to handle semiconductor specific issues. Work is also underway to implement more advanced physics-based preconditioners to build on our earlier work with block preconditioners.

It should be noted that *Charon* is not limited to semiconductor physics. There is also

significant development of *Charon* being done for modeling reacting flows and plasma physics, both of which will continue to drive development of *Charon* forward.

Lastly, one of *Charon*'s primary goals was to act as a research platform for new discretization and algorithm technology. That is still an active goal and we have short-term plans to implement a hybrid finite-element, finite-volume scheme in *Charon* with some of the properties of the popular Scharfetter-Gummel scheme used in commercial *TCAD* codes. Additionally *Rythmos*, a time-integrator being implemented in *Trilinos*, is being tested within *Charon*. *Rythmos* offers arbitrary-accuracy in time integration and it is hoped will allow for a more robust solve of the stiff problem associated with the defect physics than does the current generalized-α method (see Appendix B).

7 Conclusions

In this paper we have described a new Sandia code called *Charon*. *Charon*, in this context, is being developed to model semiconductor devices within a neutron-radiation environment. The radiation environment of interest includes effects at short time scales and thus requires a thorough understanding of the physics involved and a dynamic code capable of accurately modeling the complex physics involved.

We have developed a stabilized finite-element method (SFEM) and a new variational multiscale (VMS) discretization for drift diffusion (DD) equations. These formulations have been demonstrated to stabilize the Galerkin FE discretization to control global oscillations, and the ill-conditioning of the resulting linear systems for iterative solution, due to convective effects. Results have demonstrated significant advantages of the formulations over the standard Galerkin formulations.

Extensive verification and validation has been carried out and we have demonstrated the methods converge at the expected rates. Comparison to the standard Scharfetter-Gummel (SG) method in Pisces has indicated that the formulation is converging to the same solution as an industry standard code as well.

Currently the *SFEM* requires higher resolutions than *SG* at the steep jumps developed at metallurgical junctions (at low to moderate bias levels) to accurately compute the device current. In the challenging, highly doped 2n2222 study presented in this report it was a factor of ≈ 7 times higher. This resolution was shown to be effectively provided with the use of adaptive mesh refinement (AMR) at the junction using ≈ 110 elements, rather than employing a uniform discretization in the entire domain of ≈ 500 elements. Additionally, preliminary work to evaluate the proposed variational multi-scale method (*VMS*) extensions to the *SFEM* discretization has been carried out with one preliminary result presented in this report. Further work on completing and evaluating the *VMS* formulation is recommend along with consideration of alternate discretization strategies.

In terms of the overall solution method the Newton-Krylov methodology as implemented in Charon has been demonstrated to be very robust and effective. The Newton iteration as formulated with automatic differentiation has been demonstrated to be robust for a complex set of drift diffusion and defect species formulations for radiation damage problems. The preconditioned parallel Krylov methods employing additive Schwarz domain decomposition coupled with multi-level strategies has been demonstrated to be reasonably robust and to perform well on large-scale problems.

Charon also serves as an advanced platform for testing of many of the latest algorithms being implemented within *Trilinos*, including SACADO, LOCA, NOX, ML and Rythmos.

References

[1] N. D. Arora, J.R. Hauser, and D.J. Roulston. Electron and hole mobilities in silicon as a function of concentration and temperature. *IEEE Transactions on Electron Devices*, 29(2):292–295, 1982.

[2] Roscoe A. Bartlett, David M. Gay, and Eric T. Phipps. Automatic differentiation of C++ codes for large-scale scientific computing. Technical Report SAND2006-0902C, Sandia National Laboratories, 2006.

[3] D.M. Caughey and R.E. Thomas. Carrier mobilities in silicon empirically related to doping and field. *Proceedings IEEE*, 7(11):2192–2193, 1967.

[4] J. Chung and G.M. Hulbert. A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-α method. *Journal of Applied Mechanics*, 60:371–375, June 1993.

[5] M.N. Darwish, J.L. Lentz, M.R. Pinto, P.M. Zeitzoff, T.J. Krutsick, and H.H. Vuong. An improved electron and hole mobility model for general purpose device simulation. *IEEE Transactions on Electron Devices*, 44(9):1529–1538, September 1997.

[6] Jean Donea and Antonio Huerta. *Finite Element Methods for Flow Problems*. John Wiley & Sons Ltd., 2003.

[7] J.M. Dorkel and P. Leturcq. Carrier mobilities in silicon semi-empirically related to temperature, doping and injection level. *Solid-State Electronics*, 24(9):821–825, September 1981.

[8] Arthur H. Edwards, Harold P. Hjalmarson, Peter A. Schultz, and Andrew C. Pineda. The oxygen vacancy in SiO2 and its role in radiation effects in the mos system. Technical Report SAND2004-4657C, Sandia National Laboratories, 2004.

[9] Tor A. Fjeldl, Yanqing Deng, Michael S. Shur, Harold P. Hjalmarson, Arnold Muyshondt, and Trond Ytterdal. Modeling of high-dose-rate transient ionizing radiation effects in bipolar devices. *IEEE Transactions on Nuclear Science*, 48(5):1721–1730, October 2001.

[10] Michael Gee, Christopher Siefert, Jonathan Hu, Ray Tuminaro, and Marzio Sala. ML 5.0 smoothed aggregation user's guide. Technical Report SAND2006-2649, Sandia National Laboratories, 2006.

[11] P.M. Gresho, R.L. Lee, and R.L. Sani. On the time dependent solution of the incompressible navier-stokes equations in two and three dimensions. In *Recent Advances in Numerical Methods in Fluids*, volume 1, pages 27–81, Swansea, U.K., 1980. Pineridge Press.

[12] P.M. Gresho and R.L. Sani. *Incompressible Flow and the Finite Element Method: Advection-Diffusion and Isothermal Laminar Flow*. John Wiley & Sons Ltd., 1998.

[13] Michael A. Heroux. An overview of trilinos. Technical Report SAND2004-0398P, Sandia National Laboratories, 2004.

[14] H.M. Hilber, T.J.R. Hughes, and R.L. Taylor. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering and structural dynamics*, 5:283–292, 1977.

[15] Harold P. Hjalmarson, Peter A. Schultz, and Arthur H. Edwards. Spontaneous charging of hydrogen atoms at and near the Si-SiO2 interface. Technical Report SAND2004-0583A, Sandia National Laboratories, 2004.

[16] Thomas J.R. Hughes, Gonzalo R. Feijóo, Luca Mazzei, and Jean-Baptiste Quincy. The variational multiscale method – a paradigm for computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 166(1–2):3–24, 1998.

[17] R.C. Jaeger and F.H. Gaensslen. Simulation of impurity freezeout through numerical solution of poisson's equation and application to mos device behavior. *IEEE Transactions on Electron Devices*, 27(5):914–920, May 1980.

[18] K.E. Jansen, C.H. Whiting, and G.M. Hulbert. A generalized-α method for integrating the filtered navier-stokes equations with a stabilized finite element method. *Computer Methods in Applied Mechanics and Engineering*, 190:305–319, 2000.

[19] D. B. M. Klaassen. A unified mobility model for device simulation - I. Model equations and concentration dependence. *Solid-State Electronics*, 35(7):953–959, 1992.

[20] D. B. M. Klaassen. A unified mobility model for device simulation - II. Temperature dependence of carrier mobility and lifetime. *Solid-State Electronics*, 35(7):961–967, 1992.

[21] Tamara G. Kolda and Roger P. Pawlowski. Nox: A C++ nonlinear solver package. Technical Report SAND2003-8492P, Sandia National Laboratories, 2003.

[22] Kevin M. Kramer and W. Nicholas G. Hitchon. *Semiconductor Devices, A Simulation Approach*. Prentice Hall PTR, 1997.

[23] C. Lombardi, S. Manzini, A. Saporito, and M. Vanzi. A physically based mobility model for numerical simulation of nonplanar devices. *IEEE Transactions on Computer-Aided Design*, 7(11):1164–1170, November 1988.

[24] Kevin R. Long, Heidi K. Thornquist, Michael A. Heroux, Kristopher W. Kamshoff, Paul M. Sexton, and Roscoe A. Bartlett. Teuchos from a developer's perspective. Technical Report SAND2005-7196P, Sandia National Laboratories, 2005.

[25] George C. Messenger. A summary review of displacement damage from high energy radiation in silicon semiconductors and semiconductor devices. *IEEE Transactions on Nuclear Science*, 39(3):468–473, June 1992.

[26] N.M. Newmark. A method of computation for structural dynamics. *Journal of the Engineering Mechanics Division ASCE 85*, EM3:67–94, 1959.

[27] Allen C. Robinson. Internet URL `https://sourceforge-web.sandia.gov/sf/docman/do/downloadDocument/projects.alegranevada/docman.root.nevada_framework_for_application/doc7375`, 2006.

[28] A. Ruzin, G. Casse, M. Glaser, A. Zanet, F. Lemeilleur, and S. Watts. Comparison of radiation damage in silicon induced by proton and neutron irradiation. *IEEE Transactions on Nuclear Science*, 46(5):1310–1313, October 1999.

[29] M. Sala and R. Tuminaro. A new Petrov-Galerkin smoothed aggregation preconditioner for nonsymmetric linear systems. *Submitted to SIAM Journal on Scientific Computing*, 2006.

[30] S. Selberherr. Mos device modeling at 77k. *IEEE Transactions on Electron Devices*, 36(8):1464–1474, August 1989.

[31] J.N. Shadid, A.G. Salinger, R.C. Schmidt, T.M. Smith, S.A. Hutchinson, G.L. Hennigan, K.D. Devine, and H.K. Moffat. A finite element computer program for reacting flow problems part 1: theoretical development. Technical Report SAND98-2864, Sandia National Laboratories, Jan 1999.

[32] M.R. Shaheed and C.M. Maziar. A physically based model for carrier freeze-out in si- and sige-base bipolar transistors suitable for implementation in device simulators. In *1994 Bipolar/BiCMOS Circuits and Technology Meeting*, pages 191–194, 1994.

[33] Mahesh Sharma and Graham F. Carey. Semiconductor device simulation using adaptive refinement and flux upwinding. *IEEE Transactions on Computer-Aided Design*, 8(6):590–598, June 1989.

[34] S. M. Sze. *Physics of Semiconductor Devices*. John Wiley & Sons, 2nd edition, 1981.

[35] S. M. Sze. *Semiconductor Devices, Physics and Technology*. John Wiley & Sons, 2nd edition, 2002.

[36] W.R. Wampler. Parameters used in one dimensional drift diffusion reaction transistor model. Technical Report White Paper, Web File Share WFS334254, Sandia National Laboratories, 2006.

[37] W.L. Wood, M. Bossak, and O.C. Zienkiewicz. An alpha modification of newmark's method. *International Journal for Numerical Methods in Engineering*, 15:1562–1566, 1981.

A L2 Projection

In the stabilization term, $\langle -\tau\mathcal{L}\phi, \hat{R}\rangle$, we need a flux that is continuous at the nodes to be able to compute the second order term. The fluxes are piecewise constant across first order elements, and thus are not continuous. We use an L2 projection to estimate the derivatives at the nodes.

We wish to compute a variable $\tilde{\psi}$ at the nodes based on a least-squares projection from known values at the quadrature points, ψ, of the elements that own the node.

This forms the minimization problem:

$$\min_{\forall\tilde{\psi} \mathrm{in} V} \int_{\Omega} [\tilde{\psi} - \psi]^2 \, d\Omega \tag{106}$$

We note that ψ is a discontinuous quantity at the nodes while we need $\tilde{\psi}$ to be continuous at the nodes.

Since $\tilde{\psi}$ is a variable in the finite element space, we use the definition:

$$\tilde{\psi} = \sum_j \tilde{\psi}_j \phi_j \tag{107}$$

where ϕ are the basis functions in the discretization. The minimization problem can then be solved by taking the derivative wrt to $\tilde{\psi}$ and setting the equation equal to zero:

$$2\int_{\Omega} [\tilde{\psi} - \psi] \, \phi_i d\Omega = 0 \tag{108}$$

This is the residual equation for the unknown, $\tilde{\psi}$, at the node i. Rearranging, we get the following:

$$\int_{\Omega} \tilde{\psi}\phi_i d\Omega = \int_{\Omega} \psi\phi_i d\Omega \tag{109}$$

Using equation 107 and rearranging we get:

$$\sum_j^N \int_{\Omega} \phi_j \phi_i d\Omega \tilde{\psi}_j = \int_{\Omega} \psi\phi_i d\Omega \tag{110}$$

This results in a linear system of the form $\mathbf{M}\tilde{\psi} = R$ where:

$$\mathbf{M} = \sum_j^N \int_{\Omega} \phi_j \phi_i d\Omega \tag{111}$$

$$R = \int_{\Omega} \psi\phi_i d\Omega \tag{112}$$

M is known as the mass matrix. The true L2 projection (called a consistent projection) would require a linear solve of this system. We approximately solve this system by lumping

the mass matrix. In lumping, we make the assumption that over an element, the value of the variable $\tilde{\psi}$ at node j is equal to the value at node i (over the element, $\tilde{\psi}$ is constant):

$$\tilde{\psi}_j = \tilde{\psi}_i. \tag{113}$$

This allows us to bring the summation over j into the integral for the right hand side:

$$\int_\Omega \sum_j^N \phi_j \phi_i d\Omega \tilde{\psi}_i. \tag{114}$$

The sum of the basis functions always sum to one, so we get the following systems of equations:

$$\int_\Omega \phi_i d\Omega \tilde{\psi}_i = \int_\Omega \psi \phi_i d\Omega. \tag{115}$$

The left and right hand side are vectors, so solving for $\tilde{\psi}$ becomes a vector divide. The left hand side can be solved once and stored. The right hand side will have to be reevaluated for each residual fill.

We note that lumping results in a projected value that is essentially the weighted value of the quarter of the element closest to the node.

B Time Integration

In this section a brief description of the Generalized-α method is presented. This time integration algorithm is useful for allowing the parameterized selection of time integration schemes which include both first and second order methods. Through this generalized algorithm dissipative first order methods (e.g. Backward Euler), neutrally stable second order methods (e.g. midpoint rule) and second order methods with high wave number damping (BDF2) can be obtained. The traditional approach of Trapezoidal rule is still obtainable by setting three coefficients discussed below: α_f, α_m, and γ. The Generalized-α method can be written [4, 18]

$$\mathbf{M}^{n+\alpha_m}\mathbf{a}^{n+\alpha_m} + \mathbf{C}^{n+\alpha_f}\mathbf{v}^{n+\alpha_f} + \mathbf{K}^{n+\alpha_f}\mathbf{d}^{n+\alpha_f} = \mathbf{F}^{n+\alpha_f}$$

Using vector notation (*i.e.*, $\mathbf{x}$ and $\dot{\mathbf{x}}$) and noting that $\mathbf{d} = 0$ for our fluids problems, we can rewrite the above equation as

$$\mathbf{M}^{n+\alpha_m}\dot{\mathbf{x}}^{n+\alpha_m} + \mathbf{C}^{n+\alpha_f}\mathbf{x}^{n+\alpha_f} = \mathbf{F}^{n+\alpha_f} \tag{116}$$

where the matrices and vectors are evaluated by

$$\begin{aligned} \psi^{n+\alpha_f} &= (1-\alpha_f)\psi^n + \alpha_f\psi^{n+1} \\ \psi^{n+\alpha_m} &= (1-\alpha_m)\psi^n + \alpha_m\psi^{n+1} \end{aligned}$$

and the time steps are found by

$$\begin{aligned} t^{n+\alpha_f} &= (1-\alpha_f)t^n + \alpha_f t^{n+1} \\ t^{n+\alpha_m} &= (1-\alpha_m)t^n + \alpha_m t^{n+1} \end{aligned}$$

Thus $t^{n+\alpha_f}$ and $t^{n+\alpha_m}$ are the times where the forcing function (and damping matrix) and the mass matrix are evaluated. The third coefficient, γ, relates $\mathbf{x}$ and $\dot{\mathbf{x}}$.

$$\mathbf{x}^{n+1} = \mathbf{x}^n + \Delta t\left[(1-\gamma)\dot{\mathbf{x}}^n + \gamma\dot{\mathbf{x}}^{n+1}\right] \tag{117}$$

These coefficients can be manipulated to produce a variety of time integration methods as shown in Table B.1.

To fit this within Charon, there are essentially three parts: generate initial predictions for $\mathbf{x}^{n+\alpha_f}$ and $\dot{\mathbf{x}}^{n+\alpha_m}$, feed these vectors into the Newton-Krylov solver, and back out the solution vectors $\mathbf{x}^{n+1}$ and $\dot{\mathbf{x}}^{n+1}$. The second step is easily completed by feeding the solutions into the current Newton-Krylov solver, and only requires explanation on the iterative update.

To get the initial predictions for $\mathbf{x}^{n+\alpha_f}$ and $\dot{\mathbf{x}}^{n+\alpha_m}$ (*i.e.*, $\mathbf{x}_{(0)}^{n+\alpha_f}$ and $\dot{\mathbf{x}}_{(0)}^{n+\alpha_m}$), we first need a prediction for $\mathbf{x}^{n+1}$ (*i.e.*, $\mathbf{x}_{(0)}^{n+1}$). Using first-order Euler evaluation, we can obtain

$$\mathbf{x}_{(0)}^{n+1} = \mathbf{x}^n + \Delta t^n\,\dot{\mathbf{x}}^n$$

Integration Methods	α_f	α_m	γ
Newmark [26]	1	1	$0 < \gamma \leq 1$
HHT-$\alpha(1/2 \leq \rho_\infty \leq 1)$ [14]	$\frac{2\rho_\infty}{1+\rho_\infty}$	1	$\frac{1}{2}\left(\frac{3-\rho_\infty}{1+\rho_\infty}\right)$
WBZ-$\alpha(0 \leq \rho_\infty \leq 1)$ [37]	1	$\frac{2}{1+\rho_\infty}$	$\frac{1}{2}\left(\frac{3-\rho_\infty}{1+\rho_\infty}\right)$
Explicit Euler	0	0	0
Implicit Euler	1	1	1
Midpoint rule	1/2	1/2	1/2
Gear's Two-step	1	3/2	1
Trapezoidal Rule [31, 11]	1	1	1/2
Generalized-α $(0 \leq \rho_\infty \leq 1)$ [4]	$\frac{1}{1+\rho_\infty}$	$\frac{1}{2}\left(\frac{3-\rho_\infty}{1+\rho_\infty}\right)$	$\frac{1}{1+\rho_\infty}$

Table B.1. Various time integration algorithms derivable from the Generalized-α method coefficients.

Using Adams-Bashforth second-order evaluation, we have

$$\mathbf{x}_{(0)}^{n+1} = \mathbf{x}^n + \frac{\Delta t^n}{2}\left(2 + \frac{\Delta t^n}{\Delta t^{n-1}}\right)\dot{\mathbf{x}}^n - \frac{(\Delta t^n)^2}{2\,\Delta t^{n-1}}\dot{\mathbf{x}}^{n-1}$$

This is easily interpolated to $t^{n+\alpha_f}$

$$\mathbf{x}_{(0)}^{n+\alpha_f} = (1-\alpha_f)\,\mathbf{x}^n + \alpha_f\,\mathbf{x}_{(0)}^{n+1}$$

and the time derivative is found by

$$\dot{\mathbf{x}}_{(0)}^{n+\alpha_m} = \left(1 - \frac{\alpha_m}{\gamma}\right)\dot{\mathbf{x}}^n + \frac{\alpha_m}{\alpha_f \Delta t \gamma}\left(\mathbf{x}_{(0)}^{n+\alpha_f} - \mathbf{x}^n\right)$$

During each Newton iteration, k, a delta change to the solution, $\Delta\mathbf{x}_{(k)}^{n+\alpha_f}$, is determined, and the solution is updated

$$\mathbf{x}_{(k+1)}^{n+\alpha_f} = \mathbf{x}_{(k)}^{n+\alpha_f} + \Delta\mathbf{x}_{(k)}^{n+\alpha_f}. \tag{118}$$

The time derivative can be updated by

$$\dot{\mathbf{x}}_{(k+1)}^{n+\alpha_m} = \left(1 - \frac{\alpha_m}{\gamma}\right)\dot{\mathbf{x}}^n + \frac{\alpha_m}{\alpha_f \Delta t \gamma}\left(\mathbf{x}_{(k+1)}^{n+\alpha_f} - \mathbf{x}^n\right). \tag{119}$$

If we write the above equation for the kth iteration and subtract it from the $(k+1)^{th}$ iteration, we find the time derivative can be updated by a simple expression involving $\Delta\mathbf{x}_{(k)}^{n+\alpha_f}$,

$$\dot{\mathbf{x}}_{(k+1)}^{n+\alpha_m} - \dot{\mathbf{x}}_{(k)}^{n+\alpha_m} = \frac{\alpha_m}{\alpha_f \Delta t \gamma}\left(\mathbf{x}_{(k+1)}^{n+\alpha_f} - \mathbf{x}_{(k)}^{n+\alpha_f}\right)$$

or

$$\dot{\mathbf{x}}^{n+\alpha_m}_{(k+1)} = \dot{\mathbf{x}}^{n+\alpha_m}_{(k)} + \frac{\alpha_m}{\alpha_f \Delta t \gamma} \Delta \mathbf{x}^{n+\alpha_f}_{(k)}. \tag{120}$$

If Eq, (120) is used to update the time derivative, it is possible, through error or other means, that the solution and the time derivative can become "unsynchronized" such that they do not satisfy Eq. (119), but because of the Newton iterations they will satisfy Eq. (116). This is a very difficult error to find, and thus for safety the time derivative should be updated using Eq. (119).

Once the Newton-Krylov solver determines the solution vectors, $\mathbf{x}^{n+\alpha_f}$ and $\dot{\mathbf{x}}^{n+\alpha_m}$, we need to back out the desired solutions, $\mathbf{x}^{n+1}$ and $\dot{\mathbf{x}}^{n+1}$.

$$\mathbf{x}^{n+1} = \left(1 - \frac{1}{\alpha_f}\right) \mathbf{x}^n + \frac{1}{\alpha_f} \mathbf{x}^{n+\alpha_f}$$

$$\dot{\mathbf{x}}^{n+1} = \left(1 - \frac{1}{\alpha_m}\right) \dot{\mathbf{x}}^n + \frac{1}{\alpha_m} \dot{\mathbf{x}}^{n+\alpha_m}$$

Note: $\dot{\mathbf{x}}$ should be for physics which you are integrating (*i.e.*, diffusion physics $\dot{\mathbf{x}}$ should be for diffusion $\dot{\mathbf{x}}_{diff}$).

DISTRIBUTION:

1	MS 0321	James S. Peery, 01400
1	MS 0321	Jennifer E. Nelson, 01430
1	MS 0427	Robert A. Paulsen, 02118
1	MS 0110	Charles J. Barbour, 12930
1	MS 1056	Samuel M. Myers, 01110
1	MS 1056	William R. Wampler, 01111
1	MS 0316	Scott A. Hutchinson, 01437
1	MS 1083	Charles E. Hembree, 017311
1	MS 1146	Patrick J. Griffin, 01384
1	MS 1146	Donald B. King, 06771
1	MS 1084	Ralph W. Young, 01748
1	MS 1083	Kenneth E. Kambour, 017311
1	MS 1137	Arlo L. Ames, 06322
5	MS 0316	Gary L. Hennigan, 01437
5	MS 0316	Robert J. Hoekstra, 01437
1	MS 0316	Joseph P. Castro, 01437
1	MS 0316	Deborah A. Fixel, 01437
1	MS 0316	John N. Shadid, 01437
1	MS 0316	Eric R. Keiter, 01437
1	MS 1318	Roger P. Pawlowski, 01414
1	MS 1318	Andrew G. Salinger, 01414
1	MS 1318	Eric T. Phipps, 01411
1	MS 1318	Roscoe A. Bartlett, 01411
1	MS 1318	S. Scott Collis, 01416
1	MS 1320	Todd S. Coffey, 01414
1	MS 0316	Thomas M. Smith, 01433

1	MS 0828	Antony A. Giunta, 01544
1	MS 0828	Vincente J. Romero, 01544
1	MS 0829	Brian M. Rutherford, 12377
1	MS 1318	Robert Heaphy, 01416
1	MS 1081	Biliana S. Paskaleva, 01437
1	MS 0123	D. L. Chavez, 01011
1	MS 0899	Technical Library, 9536

Sandia National Laboratories