Fall 2019

A Multicarrier Technique for Monte Carlo Simulation of Electrothermal Transport in Nanoelectronics

Tyler J. Spence
Louisiana Tech University

A MULTICARRIER TECHNIQUE FOR MONTE CARLO SIMULATION OF ELECTROTHERMAL TRANSPORT IN NANOELECTRONICS

by

Tyler J. Spence, M.S.E., B.S.E.E., B.S.

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

COLLEGE OF ENGINEERING AND SCIENCE
LOUISIANA TECH UNIVERSITY

November 2019

LOUISIANA TECH UNIVERSITY

GRADUATE SCHOOL

August 29, 2019
Date of dissertation defense

We hereby recommend that the dissertation prepared by

Tyler J. Spence, M.S.E., B.S.E.E., B.S.

entitled **A Multicarrier Technique for Monte Carlo**

Simulation of Electrothermal Transport in Nanoelectronics

be accepted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Engineering, Micro & Nanoscale Systems Conc.

Dr. Arden Moore, Supervisor of Dissertation Research

Dr. Shengian Wang,
Head of Engineering

Members of the Doctoral Committee:
Dr. Heath Berry
Dr. Lee Sawyer
Dr. Sandra Zivanovic
Dr. Hisham Hegab

Approved:

Hisham Hegab
Dean of Engineering & Science

Approved:

Ramu Ramachandran
Dean of the Graduate School

GS Form 13a
(10/18)

ABSTRACT

The field of microelectronics plays an important role in many areas of engineering and science, being ubiquitous in aerospace, industrial manufacturing, biotechnology, and many other fields. Today, many micro- and nanoscale electronic devices are integrated into one package. The capacity to simulate new devices accurately is critical to the engineering design process, as device engineers use simulations to predict performance characteristics and identify potential issues before fabrication. A problem of particular interest is the simulation of devices which exhibit exotic behaviors due to non-equilibrium thermodynamics and thermal effects such as self-heating. Frequently, it is desirable to predict the level of heat generation, the maximum temperature and its location, and the impact of these thermal effects on the current-voltage (IV) characteristic of a device. This problem is furthermore complicated by nanoscale device dimensions. As the ratio of surface area to volume increases, boundary effects tend to dominate the transfer of energy through a device. Effects such as quantum confinement begin to play a role for nanoscale devices as geometric feature sizes approach the wavelength of the particles involved. Classical approaches to charge transport and heat transfer simulation such as the drift-diffusion approach and Fourier's law, respectively, do not provide accurate results at these length scales. Instead, the transport processes are governed by the semi-classical Boltzmann transport equation (BTE) with quantum corrections derived from the Schrödinger equation (SE). In this work, a technique is presented for coupling a 3D phonon Monte Carlo (MC)

simulation to an electron multi-subband Monte Carlo (MSBMC) simulation. Both carrier species are first examined separately. An electron MC simulation of bulk silicon, a silicon n-i-n diode, and an intrinsic-channel fin-field effect transistor (FinFET) structure are also presented. The 3D phonon MC algorithm is demonstrated in bulk silicon, a silicon thin film, and a silicon nanoconstriction. These tests verify the correctness of the MC framewprk. Finally, a novel carrier scattering system which directly accounts for the interaction between the two particle populations inside a nanoscale device is shown. The tool developed supports quantum size effects and is shown to be capable of modeling the exchange of energy between thermal and electronic particle systems in a silicon FinFET.

APPROVAL FOR SCHOLARLY DISSEMINATION

The author grants to the Prescott Memorial Library of Louisiana Tech University the right to reproduce, by appropriate methods, upon request, any or all portions of this Dissertation. It is understood that "proper request" consists of the agreement, on the part of the requesting party, that said reproduction is for his personal use and that subsequent reproduction will not occur without written approval of the author of this Dissertation. Further, any portions of the Dissertation used in books, papers, and other works must be appropriately referenced to this Dissertation.

Finally, the author of this Dissertation reserves the right to publish freely, in the literature, at any time, any or all portions of this Dissertation.

Author _____

Date _____

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENTS

This dissertation was completed with the help, technical and otherwise, of many people.

I would like to thank my advisor, Dr. Arden Moore, for his advice, assistance, and patience. I would also like to thank my committee members Dr. Heath Berry, Dr. Sandra Zivanovic, Dr. Lee Sawyer, and Dr. Hisham Hegab, all of whom have influenced me strongly during my education.

I also could not have completed this work without the encouragement and support of my family and countless friends, who have cheered me on as they observed, sometimes with concern, the trials and challenges I experienced throughout the whole process.

CHAPTER 1

INTRODUCTION

Electronic circuits sometimes integrate many microscopic transistors and other electronic devices onto the surface of a silicon (Si) chip in an arrangement known as an integrated circuit (IC). The modern form of this engineering design methodology and its associated manufacturing processes are together referred to as very large scale integration (VLSI). These VLSI circuits can contain billions of transistors with characteristic length scales on the order of 10 nm or smaller. This large scale of integration allows for lower voltages, lower power levels, and high throughput due to parallelization and co-location of subcomponents. These circuits find uses in an extremely broad array of application areas and support many other industries. ICs are used in virtually all electronic and computing systems as microprocessors, graphics processors, voltage regulators, timers, analog-to-digital and digital-to-analog converters, and many other circuits.

Important to the success of VLSI is the concept of electronic design automation (EDA). EDA is an engineering workflow and supporting set of computer aided design (CAD) tools which streamline and organize the process of creating an IC, beginning with functional descriptions of the design and ending with the generation of drawings and files used by the semiconductor foundry to produce the chip. A critical step of EDA is the simulation of circuits or the simulation of individual devices before fabrication. Circuit simulation

allows device engineers to predict the performance of a circuit and identify potential issues in advance of ordering an expensive fabrication run, while device simulation is used to extract important information about a new transistor design. Information such as the current-voltage (IV) characteristic of the device, for example, is important to the circuit designer as it governs decisions regarding the use of the device in a circuit, such as the proper bias voltage. The likely current, voltage, and temperature limits of a device can also be identified by device simulation and utilized by an engineer in the design phase to mitigate risk of operational failure in a finalized circuit. The branch of EDA concerned with simulation of fabrication processes and device behavior is called technology CAD (TCAD) and the term Device TCAD is used to refer specifically to the simulation of device behavior. The methods presented in this work comprise a Device TCAD system for electrothermal simulation of nanoscale multigate transistor devices. The novel, underlying technique proven in these simulations can be readily extended to many other device types and materials. This work also applies the method to the analysis of hotspot generation in a kind of multigate transistor called a fin-field effect transistor (FinFET). In FinFETs, the transistor channel is formed by a 3D fin-like structure with the gate electrode contacting all three fin surfaces. FinFETs use the higher gate contact surface area to establish greater electrostatic control over the transistor channel than an ordinary planar metal oxide semiconductor field-effect transistor (MOSFET).

In solid state devices, the transfer of thermal energy and the transport of electrical charge carriers are the most commonly tracked physical processes. Examples of common boundary conditions include applied voltages, net surface charge, electric field flux, and temperature as they occur on the various external faces of the device. Device simulation

is the applied technique of describing the device behavior by numerical calculation of desired properties of these two processes with specified device geometry, composition, and boundary conditions.

In this work, an object-oriented, semi-classical Monte Carlo (MC) simulation framework is developed that can simulate multiple simultaneous interacting populations of particles within a given semiconductor device during operation. Each population follows their respective statistical mechanics in terms of number of carriers and their energy state occupation. Particle scattering events are implemented as rules-based changes of propagation direction and dynamic state as dictated by established models of each scattering type: point defect, electron-ion, electron-phonon, three- and four-phonon, etc. Validation of the scattering algorithms employed is accomplished prior to full device simulation by comparing the charge carrier mobility and phonon thermal conductivity values returned by the simulation to existing experimental data. All simulation development was performed with the resulting code being highly parallelized and compatible with established UNIX-based high performance computing (HPC) practices. Due to the scale and complexity of the calculations involved, the use of HPC is necessary to reduce runtimes from days or weeks to hours. The following sections describe the challenges, applications, and research need associated with nanoscale device simulation along with an overview of the approach taken in this work.

1.1 Research Challenges and Applications

Emerging and advanced electronic devices, such as FinFETs, nanowire transistors (NWTs), high electron mobility transistors (HEMTs), monolothic microwave integrated

circuits (MMICs), Micro-electromechanical systems (MEMS), and other high performance devices play an integral role in advanced research for military and civilian technologies. Many of these devices possess complex geometries, exhibit quantum size effects, or require advanced non-Si materials. Notably, the level of physics fidelity required to accurately model single- and multi-species transport processes (using electrons, holes, phonons, etc.) within the relevant operating regimes, length scales, and materials extends beyond what is necessary for more traditional semiconductor devices. The high electric fields, speed of operation, sub-micron dimensions, dynamic multicarrier interactions, and quantum effects involved mean that device-level electrothermal simulations must account for the resulting non-equilibrium carrier populations and quasi-ballistic transport in order to realistically predict important device parameters such as drain current, switching speed, self-heating levels, and hot electron temperatures within the channel region [3].

Simulation of electrothermal transport in nanodevices is lacking in its ability to address the technological challenges presented. Semi-classical MC methods are attractive for simulating carrier transport within such devices due to their ability to model realistic device-level geometries and sub-continuum transport while also incorporating more sophisticated quantum corrections and carrier scattering models than drift-diffusion or even hydrodynamic approaches. Few frameworks for electronic device simulation directly incorporate quasi-particle phonon transport modeling, instead often relying on some form of diffusion-based equation where the lattice thermal conductivity is treated simply as a model input parameter rather than determined by the framework from information on spatially-varying defect concentration, defect location, temperature, or other relevant physical attributes of the device itself. This detail is especially important to precise modeling

of electrothermal behavior. Ashok *et al.* [4] and Padmanabhan *et al.* [5] have developed electrothermal MC simulations of gallium nitride (GaN) HEMTs in which thermal effects were instead accounted for by solving the Boltzmann moment expansion equations for acoustic and optical phonons, respectively. Other well-known studies have also utilized MC methods to simulate electron transport in Si devices while incorporating electron-phonon scattering and lattice temperature considerations via a variety of means [6, 7, 8, 9, 10, 11, 12]. Where MC is used for the thermal portion of the above studies, they do not necessarily integrate it directly with the electron MC simulation.

Another perceived challenge to modeling charge carriers and phonons interacting within the same domain is the large difference in time scales between the charge carrier and phonon transport processes. For example, electron transients and carrier scattering times in Si are on the order of picoseconds or even tenths of picoseconds, whereas the thermal transients in a device may be on the order of nanoseconds or longer [3, 13, 14]. This difference between timescales is a major consideration in the simulation of processes involving both particle species, as it implies the simulation of thousands of electron events per phonon event. Various attempts at electrothermal simulation have been made which try to side-step this issue and arrive at a self-consistent solution of two independent electron MC and phonon MC simulations through techniques similar to iterative solutions for systems of nonlinear equations, where thermal information from one simulation is used to as inputs to an independent electronic simulation and vice-versa. To the best of the author's knowledge, no simulation studies exist where the two populations are simulated together and individual accounting of energy and momentum exchanges between the populations during scattering is performed. Additionally, quantum confinement effects

limit the applicability of the standard electron MC approach and require either quantum potential corrections or a hybrid approach such as the multi-subband Monte Carlo (MSBMC) algorithm.

This work seeks to directly address these existing deficiencies in the current state of the art with a novel, high physics fidelity multicarrier MC simulation framework suitable for predicting and understanding hotspot location and severity within advanced electronic devices. Besides its value as an engineering tool, the framework will facilitate new understanding into the complex multicarrier transport phenomena within these devices. Examples of new insights include identifying the dominant carrier-carrier interactions, hotspot root cause, and internal charge distribution inhomogeneities which are not readily observable via experiment. Besides failure prediction, the validated framework developed here can also be used as a design tool to develop and evaluate novel device designs, compositions, and operational parameters, virtually, at little or no cost during the pre-fabrication phase. In this way, scientists and engineers can explore new approaches to extending semiconductor device life in applications such as elevated temperatures prior to beginning expensive prototyping, testing, and qualification activities.

The need to develop high temperature-compatible electronics is explicitly cited on p. 126 and in Chapter 11: *The Role of Technology Development in Planetary Exploration* of the NASA Decadal Survey as a necessary technological advancement in order to facilitate long-term *in situ* scientific studies on Venus and Mercury [15]. Correspondingly, this work has direct applicability to many of the missions of scientific discovery outlined in the survey. The techniques presented here can be utilized for lifetime projection and failure analysis for devices under study that will be subjected to elevated environmental

temperatures such as those which would be encountered on missions to Venus, Mercury, or the deep atmosphere of Gas Giants, at 500 °C or higher. The resulting insights would enable superior reliability projections compared to the current state-of-the-art as well as facilitate informed design and operation decisions regarding onboard electronics. These predictive capabilities will greatly reduce the risk of electronics failure during *in situ* missions to locations of interest with high temperature environments.

Also, the simulation framework described herein and its resulting insights could be leveraged to enhance the performance and reliability of high power electronic devices. Such devices are used not only for space applications, as described above, but also in other industries and institutions as well. HEMTs based on wide-bandgap semiconductors have risen to the forefront of modern power amplification technology and play an integral role in aerospace radar, communication, imaging, and radio astronomy electronics [16]. However, these devices suffer from large amounts of localized heat generation. Thus, an in-depth understanding of carrier interactions and accurate prediction of hotspot formation represents a pressing need in developing high performance HEMTs and maximizing their reliability. In this application area as well, the identification of root cause failure, mitigation of lifetime-shortening heating effects, and optimization of future designs are just some possible benefits. Further complications arise in more standard high-power devices within elevated temperature environments, where the additional thermal energy can make often-neglected minority carriers and scattering channels have non-negligible impact on device operation characteristics.

Modern integrated microprocessors also operate using more and more complex semiconductor device types. Replacing the older planar MOSFET, 32 nm and smaller

FinFET technology allows unprecedented device density. The high density and fast operating speeds create a unique thermal environment that can have profound effects on the performance of such devices.

High physics fidelity simulations of the kind presented in this work also find use in the design of high sensitivity amplifiers. Circuits such as low noise amplifiers (LNAs) in extreme low temperature applications can exhibit self-heating in the absence of external temperatures, which can affect the behavior of the device. Quantum computing is an emerging technology that makes use of very low temperature, microwave electronics operating inside cryostats [17]. The circuits and electronic devices in quantum computers are extremely sensitive to thermal noise and require highly detailed, high performance designs to operate properly [18].

1.2 Solution Approach

The custom-developed MC simulation framework presented here is designed for configurability and extensibility. The MC program's general structure is discussed in Chapter 3. The tool is coded in C++, taking advantage of the object-oriented features of the language to create a simulation "kernel" which can handle multiple different particle types. Object-oriented programming (OOP) is a programming paradigm that encapsulates data and code into semantic units known as objects. Objects are variable-like structures that are grouped into classes, which can be thought of as complex datatypes. Objects can be instances of different classes the same way ordinary variables have an associated datatype, such as *int*, *float*, or *double*. An important feature of OOP is the concept of inheritance, where classes are given an explicit hierarchy. For example, a *Carrier* class may exist with

two subclasses: *Electron* and *Phonon*. The C++ class inheritance rules ensure that any *Electron* or *Phonon* instances have at least the internal data and functions common to all *Carrier* objects, while allowing the subclasses to exhibit unique additional behaviors. Major functions in the MC tool are coded to operate on collections of *Carrier* objects, and object inheritance is used to handle special features of more specific particles on a per-particle basis, internal to the object. This implementation allows easily configurable simulation of multiple kinds of particles, either separately or simultaneously. It is this feature of the program in particular that supports the ultimate goal of electrothermal simulation. For this work, a collection of *Carrier* subclasses with different dispersion relations, electric charge, effective mass, etc, have been generated. An MC simulation of an electronic device can be set up by electing to initialize *Electron* objects, or a different MC simulation of heat transfer processes can be set up similarly by populating the simulation domain with *Phonon* objects instead. This type of implementation is critical to direct multicarrier simulation, as the freedom in simulation input parameters allows the simulation of multicarrier interaction to arise automatically as merely a specially chosen simulation configuration.

The tool developed also supports parallelization through the message passing interface (MPI) library [19]. The MPI application programming interface (API) handles distributing tasks among the different available computing resources and also provides facilities for communication and synchronization between task processes. The simulations are designed to run on a large distributed HPC platform such as the Louisiana Optical Network Initiative (LONI) [20]. LONI was used to generate all datasets in this work.

Additional features of the MC tool were required to fully support the goal of verifying the simulator with an electrothermal simulation of a FinFET device. The FinFET

geometry and scale place additional requirements on both the electron and phonon simulation approaches. Quantum confinement in the fin plane breaks the assumptions associated with the semi-classical MC approach, and so the MSBMC algorithm was adopted. MSBMC directly solves the Schrödinger equation (SE) in the confinement directions, while still utilizing the normal MC algorithm in 1D along the transport axis. SE solutions are obtained for cross-sectional slices of the device channel, and each slice is coupled to the 1D MC simulation via the solution of a 3D Poisson's equation (PE) over the entire domain. Due to the constricted aperture shape formed by the larger size of the source and drain reservoirs and the thinner width of the fin structure, the phonon simulation needed to properly reproduce the phenomenon of thermal constriction resistance (TCR). The primary challenge of correctly simulating TCR is the proper handling of particle reflection and scattering at surfaces with moderately complex geometry. The MC tool therefore has a generic approach for properly handling surface interactions for a variety of geometries.

Finally, a strategy for allowing the electron and phonon populations to interact was devised based on the idea of fast phonon recycling, where thermal energy leaves the electron population in the form of optical phonons that decay into acoustic modes. Physically, optical phonons do not contribute much to heat transfer due to their low velocity, while the acoustic phonons they decay into are capable of thermal transport. Energy and momentum exchanges from inelastic scattering of electrons is used to create and destroy optical phonon objects. Optical phonon objects subscribe to a scattering process in the MC Kernel which can allow them to decay into two acoustic phonons, which can travel through the device. Acoustic phonon density affects some elastic scattering processes for electrons, completing a feedback loop. The optical phonon population acts

as a reservoir of energy that can be accessed by both the slow-moving phonon population and the fast-moving electron population. The particle-based conceptualization of the process allows the MC algorithm to perform proper accounting of energy and momentum exchange between the three systems. This conceptualization also facilitates an immediate physical understanding of the resulting dataset. The OOP design of the MC tool allows the fast phonon recycling process to be easily installed in the MSBMC-enabled simulation via the creation of *PhononSiOptical* objects.

The code associated with this work is organized into the five modules listed in Table 1.1. The relevant physics and programming background, along with implementation

Table 1.1: Summary of code modules

Module Name	**Description**
MC Kernel	Calculates particle trajectories and handles scattering events and dispersion tables
Simulation	Reads the simulation inputs, coordinates the other modules' functions, calculates results and distributions, and prints or stores outputs
Poisson	Sets up Poisson mesh, calculates fields, potential, and charge distribution at mesh sites
Thermal	Sets up thermal mesh, calculates thermal energy and temperature distribution at mesh sites
Quantum	Contains functions assocaited with the MSBMC implementation for confined populations

details of the simulation package are provided in the following chapters. In Chapter 2, the basics physics and governing equations are presented. Chapter 3 explains details of the MC algorithm common to electron and phonon simulation. Chapters 4 and 5 describe electron- and phonon-specific features, respectively, along with several studies verifying

the capabilities of the simulation framework. In Chapter 6, the electrothermal coupling mechanism is shown, and the results of the final simulation are presented. Chapter 7 concludes the work with an interpretation of the results and a discussion of potential future work.

CHAPTER 2

BACKGROUND

The simulation of modern electronic devices is complicated by the extremely small length scales of new devices. Devices such as FinFETs, HEMTs, and NWTs can have nanometer-sized features. The consequences of these features are two-fold: the introduction of quantum mechanical behaviors in the transport processes invalidates many classical approaches which are traditionally used to simulate larger devices, and the high ratio of surface area to volume means that boundary effects become prominent in nanodevices as compared to their larger counterparts. In order to properly simulate the transport of charge and heat in a nanoscale device, the material makeup of the device must also be studied. A transistor is usually constructed from a solid crystalline, semiconducting material. Such a material is recognized as having two constituent parts: the delocalized electron gas, also called the Fermi gas, and the atomic lattice. The individual elements of these two parts, electrons and lattice atoms respectively, are in continuous motion. The assumption that the wavefunction of the entire material can be characterized by separate electronic and atomic wavefunctions is called the Born-Oppenheimer approximation [21]. Based on this assumption, the electron gas and lattice atoms are treated as separate, independent populations of particles with explicit corrections made in the form of electron-phonon scattering where the assumption is invalid.

The fundamental behavior of electrons in a crystal and the interpretation of lattice vibrations as particles called phonons are presented in this section, as well as the primary equation governing transport in nanoscale devices.

2.1 Quantum Treatment of Charge Transport

The transport of charge through a device corresponds to the motion of electrons between device regions. As the lattice ion potential varies quickly in space with nearly the same order of magnitude as a typical electron wavelength, the electron must be described by quantum mechanics (QM). The electron is not considered pointlike with a well-defined position and momentum, and its properties are instead determined by its wavefunction. The wavefunction is complex-valued and denoted by $\psi(\mathbf{r})$ in the position basis. The square modulus of the wavefunction, $\psi^*(\mathbf{r})\,\psi(\mathbf{r})$, is interpreted as the probability density associated with finding the particle to be in position $\mathbf{r}$ at time t when a measurement is made. The wavefunction is so called because it is a solution to the Schrödinger wave equation, or simply the SE [22]. The time-dependent form of the SE for electrons is

$$\left[\frac{-\hbar^2}{2m_e}\nabla^2 + qV(\mathbf{r})\right]\psi(\mathbf{r}) = i\hbar\frac{\partial\psi(\mathbf{r})}{\partial t}, \tag{2.1}$$

where i is the imaginary unit, $\hbar$ is the reduced Planck's constant, m_e is the mass of an electron, q is the elementary charge, and $V(\mathbf{r})$ is the classical electrostatic potential affecting the particle. Free electrons correspond to a family of solutions of Equation 2.1 with $V(\mathbf{r}) = 0$ and their wavefunctions take the form

$$\psi_{\mathbf{k}}(\mathbf{r}) = Ae^{i\mathbf{k}\cdot\mathbf{r}}e^{-iEt/\hbar}, \tag{2.2}$$

where A is a normalization coefficient, $\mathbf{k}$ is a quantity called the wavevector, and E is the particle energy. The angular frequency of the time-varying component of the wavefunction can be identified as $\omega = E/\hbar$. Substituting Equation 2.2 into Equation 2.1 allows one to find the relationship $E(\mathbf{k})$ between energy and wavevector for a free electron,

$$E(\mathbf{k}) = \frac{\hbar^2 |\mathbf{k}|^2}{2m_e}. \tag{2.3}$$

It can be seen by inspection that the quantity $\hbar\mathbf{k}$ corresponds to the particle momentum $\mathbf{p}$ by analogy to the classical case $E = \frac{|\mathbf{p}|^2}{2m_e}$. In this case, particle velocity can be found from the particle momentum by $\mathbf{v} = \frac{\mathbf{p}}{m}$ as one would expect from classical physics. The previous result of $E = \hbar\omega$ implies that the dispersion relation for free electron wavefunctions is

$$\omega(\mathbf{k}) = \frac{\hbar |\mathbf{k}|^2}{2m_e}. \tag{2.4}$$

The dispersion relation is used to derive the equation of motion for an electron wavepacket. In what is called *semi-classical* dynamics, the particle wavefunction is assumed to take the form of a localized pulse of some non-zero width centered around $\mathbf{p}$ in momentum space and $\mathbf{r}$ in position space. The wavepacket is often taken to be Gaussian when the wave envelope is important [23]. The momentum and position centers evolve in time according to classical mechanics. This model is valid when the potential influencing particle motion is slowly varying in space, with changes in potential occuring over distances larger than the average wavepacket width.

As the lattice ions present a quickly-varying potential, the semi-classical approach is not directly applicable to the electron population. The approach can still be applied, however, by taking advantage of the lattice regularity. This procedure will result in slightly different electron behavior. Incorporation of the lattice ions alters the above solution to

the SE by providing a new potential function $V(\mathbf{r})$ with the same periodicity as the lattice. Solutions in a periodic potential exhibit discrete energy eigenstates instead of a continuous spectrum of possible energies [24]. Additionally, the new potential will result in a change to the particle dispersion relation. A new dispersion relation in turn implies there will be different particle dynamics. Bloch's theorem states that the spatial component of solutions to the one-electron SE for a periodic potential have the form

$$\psi_{n,\mathbf{k}}(\mathbf{r}) = e^{i\mathbf{k}\cdot\mathbf{r}} u_{n,\mathbf{k}}(\mathbf{r}), \tag{2.5}$$

where the index n corresponds to a discrete energy eigenstate, and $u_{n,\mathbf{k}}$ is a function with the same periodicity $\mathbf{k}$ of the lattice [25]. The particles modeled in this way may be called electron quasi-particles to distinguish them from the true electrons in the equivalent many-body problem. Electron quasi-particles with such wavefunctions are known as Bloch electrons and do not carry the above direct proportionality of momentum and velocity seen in free electrons. The expression $\mathbf{p} = \hbar\mathbf{k}$ is now termed the crystal momentum, and velocity is computed with

$$\mathbf{v} = \nabla_{\mathbf{p}} E\left(\frac{\mathbf{p}}{\hbar}\right). \tag{2.6}$$

In many cases, the behavior of Bloch electrons can be approximated by otherwise free electrons with an effective mass different from m_e. Particle dynamics of Bloch electrons are discussed in greater detail in Section 3.8. Using these concepts, it is possible to treat motion of electron wavepackets in a crystal lattice semi-classically without having to explicitly consider the Coulomb interaction between the electrons and the lattice ions. Quantum corrections can sometimes be applied where additional sharp potentials exist, such as narrow channels in MOSFETs or NWTs, to extend the semi-classical model further.

2.2 Quantum Treatment of Heat Transfer

In semiconductors, thermal energy is typically identified with the collective vibration of the atoms around their equilibrium positions. This behavior is in contrast with materials like metals with extremely high free electron density, where most of the thermal energy is accounted for by the electron population [26]. The interatomic bonds holding together the lattice can be conceptualized as springs with elastic constant C connected between pairs of atoms, each with mass m, as shown in Figure 2.1. Accordingly, atoms

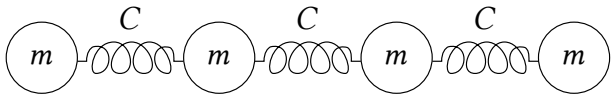


Figure 2.1: Harmonic representation of interatomic forces

have equilibrium positions to which they are drawn if perturbed. The interatomic potential can be expanded as a Taylor series. The constant term of the Taylor series is unrelated to lattice dynamics since atomic motion is related only to the derivative of interatomic potential. All odd terms are zero due to symmetry as the Taylor series is expanded about the equilibrium position. Of the remaining terms, the quadratic term has the largest influence on lattice dynamics by a great measure. Considering the interatomic potential to be quadratic is known as the harmonic approximation [27]. The resulting Hamiltonian of a one-dimensional monatomic lattice of N atoms is

$$H = \sum_{n=1}^{N} \left\{ \frac{1}{2m} p_n^2 + \frac{1}{2} C (q_{n+1} - q_n)^2 \right\} \tag{2.7}$$

where C is the elastic constant, p_n is the atomic momentum, q_n is the displacement of atom n, and m is the atomic mass. By the above reasoning the interatomic potential is modeled quadratically, like a linear spring obeying Hooke's law. The solutions are expected to be

wavelike, and the discretization of the wave modes into particles is described in several solid-state physics texts [28, 26]. By enforcing periodic boundary conditions, all possible wave modes can be enumerated by wavenumbers k given by

$$k = \frac{2\pi n}{Na}, \tag{2.8}$$

where N is the number of atoms in the chain and n are the integers in the interval $[1, N]$. A change of coordinates will be made using

$$Q_k = N^{\frac{-1}{2}} \sum_{n=1}^{N} q_n e^{-ikna} \tag{2.9}$$

and

$$P_k = N^{\frac{-1}{2}} \sum_{n=1}^{N} p_n e^{ikna}, \tag{2.10}$$

where k is the wavenumber, and a is the lattice constant. This change of variables can be regarded as a discrete Fourier transform on the atomic displacements. The Hamiltonian can then be rewritten in terms of Q_k and P_k to give

$$H = \sum_k \left\{ \frac{1}{2m} P_k P_{-k} + C Q_k Q_{-k} \big(1 - \cos(ka)\big) \right\}, \tag{2.11}$$

where the sum is taken over all possible wavenumbers. The purpose of this change is to remove the coupling between consecutive terms of the sum in the Hamiltonian. Defining the dispersion relation

$$\omega(k) = \sqrt{\frac{2C}{m}\big(1 - \cos(ka)\big)}, \tag{2.12}$$

it is possible to rewrite Eq. 2.11 as

$$H = \sum_k \left\{ \frac{1}{2m} P_k P_{-k} + \frac{1}{2} m\omega(k)^2 Q_k Q_{-k} \right\}. \tag{2.13}$$

Equation 2.13 is the sum of the Hamiltonians for N independent harmonic oscillators. Taking the quantities P, Q, p, and q to be operator-valued instead of scalar, and requiring the usual canonical commutation relationships, the system becomes a collection of N independent quantum harmonic oscillators. Thus, each mode k has the familiar quantized energy eigenvalues of the form

$$E(k) = \left(n + \frac{1}{2}\right)\hbar\omega(k), \tag{2.14}$$

where n is interpreted as the mode's occupation number. This quantization allows the description of lattice displacements as particle-like entities called phonons [26]. For convenience, phonons are typically simply referred to as "particles" despite their true nature as a collective excitation. Each elementary excitation of the lattice modes is associated with a phonon of a given momentum and energy. Similarly to electrons, phonons are quantum particles described by their wavefunctions. The energy and momentum of a phonon are given by $E = \hbar\omega$ and $\mathbf{p} = \hbar\mathbf{k}$, where ω and $\mathbf{k}$ are the frequency and wavevector of the phonon wavefunction, respectively. Phonons do not carry physical momentum, and the quantity $\hbar\mathbf{k}$ for phonons is termed the crystal momentum as it is for electron quasi-particles. The distribution of phonons in a region of a material corresponds to distribution of vibrational energy in that region and therefore also the distribution of thermal energy. Transport of phonons through a device similarly corresponds to the transfer of thermal energy between device regions.

Now, the exchange of energy between electrons and the crystal lattice can be described as a Bloch electron scattering against, absorbing, or emitting phonons. The interaction of the two types of particles couples the transfer of heat and the transfer of

charge in a device. At the macroscopic scale, the impact of this linkage is seen in Joule's law where electric current is described as a source of heat [23].

2.3 The Boltzmann Transport Equation

The central problem of device simulation is the calculation of the single-particle distribution function $f(\mathbf{r}, \mathbf{p}, t)$. The function f answers the question "what is the probability density associated with finding a particle with crystal momentum $\mathbf{p}$ and position $\mathbf{r}$ at time t?" The interpretation of f is therefore similar to the interpretation of the square modulus of the particle wavefunction ψ. The quantum mechanical wavefunction itself is completely deterministic, and there is uncertainty associated with the value position or momentum will take after measurement. The distribution function f presumes an ensemble of classical particles with deterministic, but unknown, position and momentum. It is often difficult or impossible to find an analytical form of the distribution function for a given device during operation. The utility of the distribution function for determining device behavior can be easily demonstrated. Consider the expression for electric current density given the distribution function for electrons,

$$\mathbf{J}(\mathbf{r}, t) = \frac{-q}{\Omega} \sum_{\mathbf{p}} \frac{\mathbf{p}}{m_e} f(\mathbf{r}, \mathbf{p}, t), \tag{2.15}$$

where q is the charge of an electron and Ω is the volume used for the normalization of f [23]. Equation 2.15 gives the flow of charge directly as a simple operation involving only the distribution and a few fundamental constants.

To find f for a particular species of energy carrier, one must solve an equation constraining f. Consider the function f for one dimensional system and the area in phase space covered by $(r, r + dr) \times (p, p + dp)$ as shown in Figure 2.2. The change in time of

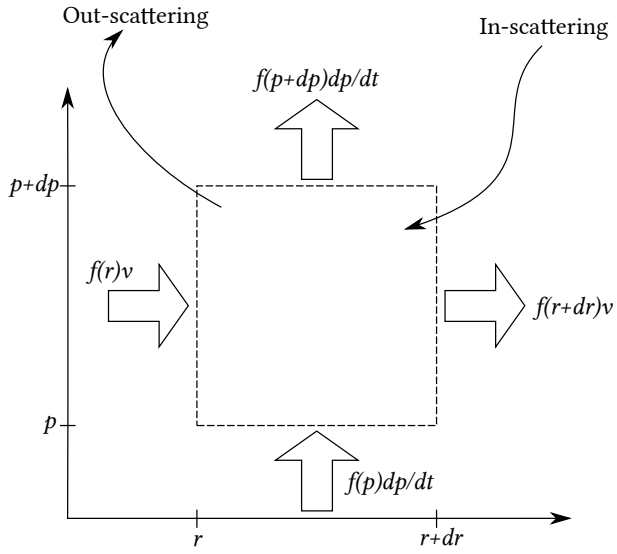


Figure 2.2: Carrier motion in a region of position-momentum space

the probability of finding a particle in this area is given by the product $\frac{df}{dt} dr\, dp$. As time proceedes, particles may enter or leave the area on the position axis through their velocity. Particles may also enter or leave the area on the momentum axis through applied forces $F = \frac{dp}{dt}$. Finally, particles may scatter into or out of the area during a given time interval. Conservation of particle number requires that the change in occupation of the area be equal to the difference in in-flow and out-flow, or

$$\begin{aligned} \frac{df}{dt} dr\, dp = \quad & [f(r) - f(r + dr)]v\, dp \quad + \\ & [f(p) - f(p + dp)]F\, dr \quad + \\ & \left(\frac{df}{dt}\right)_{coll} dr\, dp. \end{aligned} \tag{2.16}$$

If we rearrange the terms and let the differentials approach zero, Equation 2.16 becomes

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial r} + F\frac{\partial f}{\partial p} = \left(\frac{\partial f}{\partial t}\right)_{coll}. \tag{2.17}$$

The Boltzmann transport equation (BTE) for 3D devices is then given by the relationship described in Equation 2.17 with scalar quantities generalized as appropriate. The final result is

$$\frac{\partial f}{\partial t} + \frac{\mathbf{p}}{m} \cdot \nabla f + \mathbf{F} \cdot \frac{\partial f}{\partial \mathbf{p}} = \left(\frac{\partial f}{\partial t} \right)_{coll}, \tag{2.18}$$

where $\mathbf{p}$ is the crystal momentum, m is the particle mass, and $\mathbf{F}$ is any force field acting on the population, such as one due to an electric field [23]. Electric fields used with the BTE are typically computed classically and may incorporate one of several quantum corrections to the electric potential. The notation $\frac{\partial}{\partial \mathbf{p}}$ indicates the gradient of momentum, and the term on the right hand side is called the collision operator. The collision operator is an expression of the change in distribution due to particle scattering. All scattering processes, including boundary surface scattering, are captured in the collision operator. Scattering rates are computed quantum mechanically and incorporated into the solution as described in Section 3.4.

Using the particle representations of both Bloch electron wavefunctions and quantized lattice vibrations, the BTE can be formulated to describe the flow of charge or heat in a nanoscale device. The equation is a partial differential equation (PDE) in seven dimensions, and the collision operator often has no tractable analytical form. The scattering processes include electron-phonon scattering processes and must be treated self-consistently when solving the BTEs for both species simultaneously. These properties make it difficult to find the simultaneous distribution functions f_e and f_p for both electrons and phonons in a generic transistor device. The novelty of this work is the adjustment to independent MC solution approaches to support simultaneous calculation of both particle distributions.

CHAPTER 3

THE MONTE CARLO METHOD

In the previous chapter, the BTE was derived, and its relevance was explained as the principle equation to solve in device simulation. This equation is a linear first order PDE of seven dimensions: three position coordinates, three momentum coordinates, and one time coordinate. The equations can be solved using a deterministic numerical method or a stochastic one. While both types of methods have limits of accuracy and precision, deterministic methods have the benefit of giving the same answer for identical simulation conditions where stochastic methods will not, but stochastic methods can in many cases be more efficient than their deterministic counterparts [29]. Deterministic methods frequently require iterating over the different dimensions of the simulation domain, which can be inefficient given high the dimensionality of the problem. The stochastic approach to the solution of the BTE actually comes in the form of a direct physical simulation of the semi-classical dynamics of the involved particles. This approach is an MC method, as it implements carrier scattering as a pseudorandom process. This use of random numbers and trials in simulation makes the numerical method similar to an experimental technique.

In the 1950s, shortly after the invention of transistors, the complexity of the problem of charge transport was already beginning to be understood. It was found that field strengths and other conditions inside of semiconductor devices placed the constituent

materials well outside of the linear, or Ohmic, response region [30]. Nonlinear transport was not unheard-of at that time, as it was studied as early as the 1930s by Landau and Kompanejez [31] and Davydov [32, 33], but progress in the field rapidly increased with the motivation of understanding and improving transistors.

In the process of studying nonlinear and high-field problems, new physical processes were discovered [29]. These processes were studied as useful effects that could be leveraged in new kinds of transistors, which, of course, led for the need to more closely study the physics present in those designs. Additionally, the tendency towards miniaturization in semiconductor technology has contributed to the importance of studying the charge transport problem, which only becomes more complex as feature sizes decrease.

Analytical approaches for tackling the charge transport problem were already regarded as extremely difficult and limited. In 1966, Kurosawa [34] introduced a numerical approach based on an MC technique, which was a very well received development by scientists in the field. An MC method relies on supplying random inputs, drawn from a particular probability distribution, to a deterministic algorithm. The distribution of results from the deterministic steps are thus also randomly distributed, and the statistical properties of the resulting distribution are interpreted to provide the solution to the mathematical problem being studied. MC approaches had been used for solving problems before, with widespread use in many scientific fields [35, 36, 37].

The MC method for charge transport simulation was popular because its results were easily and directly interpretable in regards to the physics being studied. Kurosawa's approach, combined with fast, then-modern computers, which were only expected to increase in power, provided a path to seriously studying these transport processes. The

technique was further refined by Rees and Fawcett in the latter portion of the 1960s with the introduction of the self-scattering scheme [38, 39] and consideration of band structure nonparabolicity and anisotropy [40, 41]. Important improvements from many authors (Lebwohl, Ruch, Baccanari, Price, and Hauser) on the method came in the 1970s to handle many-particle interactions [42], transient behavior [43, 44], harmonic behavior [45], and semiconductor alloys [46].

The method was used to study many materials, such as gallium arsenide (GaAs) [47] and germanium (Ge) [48], and soon a common, standard physical model of Si arose which explained many transport processes in the material [49, 50, 51]. Studies focusing on the need for simulating higher energy carriers arrived at a method for employing numerical band structures where the analytic, nonparabolic approximation of band structures failed to reproduce proper results [52, 53, 54, 55, 56]. The computational expense of numerically handling anisotropic band structures is extremely high compared to having an analytic alternative. This method was found necessary in an important work by Abramo *et al.*, after it was shown that even analytical models with fitting parameters chosen to create certain levels of agreement with the density of states function cannot properly recreate the high energy tails of the distribution function [57]. Scattering in lattice interactions is treated perturbatively, and scattering mechanisms due to thermal lattice vibrations, ionized impurities, and carrier-carrier scattering have been developed that reproduce the Si mobility over a large range of temperatures [58, 59, 60]. Additional complex scattering can account for plasmonic interactions and the Pauli exclusion principle in degenerate materials [61, 62, 63, 64].

Many of these advanced features are well-studied in academic contexts, but the use of MC in industry is limited. Some highly proprietary and internal codes exist within companies which implement several of the advanced techniques as required for their application area [65].

In this work, the ultimate goal is to allow the combined simulation, via the MC method, of electron- and phonon-based transport processes. To that end, all features common to electron and phonon MC were identified and abstracted. An object-oriented coding approach was taken, and all MC functions operate on *Carrier* objects. See Table 3.1 for a summary of *Carrier* object features. The algorithms are agnostic to exactly which

Table 3.1: *Carrier* object functions and member variables

Function or Variable Name	**Description**
Carrier::Carrier	initializes object memory (function)
Carrier::~Carrier	Safely deallocates object memory (function)
Carrier::E	Returns or assigns carrier energy (function)
Carrier::k	Returns or assigns carrier wavevector (function)
Carrier::P	Returns or assigns carrier momentum (function)
Carrier::v	Returns or assigns carrier velocity (function)
Carrier::q	Particle charge (variable)
Carrier::tl	Time lived since beginning of simulation iteration (variable)
Carrier::ts	Time until next scattering event (variable)
Carrier::tls	Time lived since last scattering event (variable)

species of carrier is being simulated and process arrays of electrons, holes, and phonons together. In this chapter, the basic features of the MC methods for carrier transport are described and the novel object-oriented coding implementation is presented.

3.1 Overview of the Monte Carlo Algorithm

In an MC method, particles, referred to here as carriers, are treated semi-classically. Carriers are initialized according to some equilibrium distribution which nominally describes the material or device being studied. Initialization is carried out by generating random numbers according to the chosen distribution and using them to assign initial dynamic properties of the carriers, such as energy, velocity, frequency, and position. These properties are constrained to be consistent with the carrier dispersion relation and other characteristics of the carrier type and device being simulated.

As described in Sections 3.4 and 3.8, the particles are then processed individually to simulate "free flight" under ordinary classical mechanics. The classical electrostatic potential is incorporated at this point, providing a field of force vectors that influence the carrier motion. The duration of a carrier's free flight is also chosen randomly from a distribution with characteristics calculated using Fermi's golden rule (FGR). FGR-based scattering times are consistent with the behavior of the carrier as it interacts quantum mechanically with perturbing potentials. After calculating the carrier position and velocity for the duration of the free flight, a scattering event is triggered. Scattering centers are considered to be localized in space and randomly distributed throughout the simulation domain, and scattering events are considered to be instantaneous. Additional random numbers are generated to select a new, after-scattering state for the carrier. The state selection is again constrained to be consistent with the carrier dispersion relation, but this time also taking into account the kinetics of the scattering event. During processing, the carrier population will evolve in time, tending towards the proper statistics. These

statistical properties and the underlying carrier distribution function are the desired end-results of the simulation. In particular, the distribution function found will solve the BTE for the carrier type being simulated.

The flowchart for a general MC simulator, as implemented in this work, is found in Figure 3.1. The main MC Kernel function computes the new locations, momenta, and

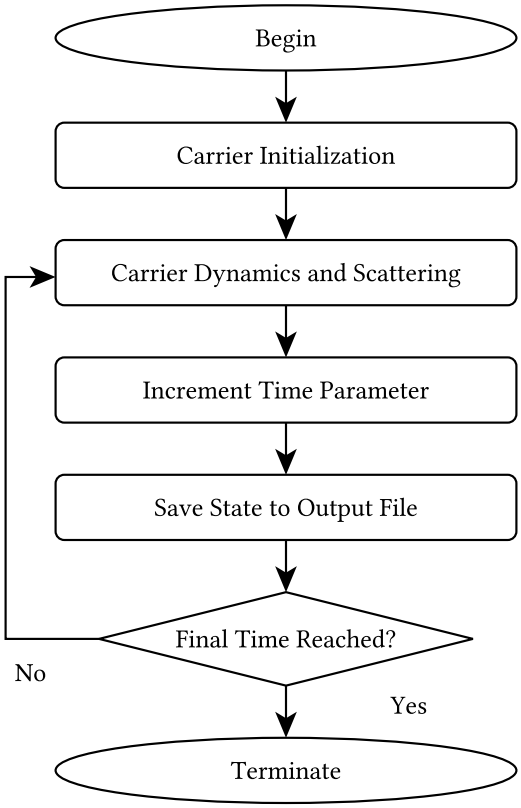


Figure 3.1: Generic MC method flowchart

flight times for each carrier in the device after a specified timestep. The function is given a timestep and *Carrier* instance pointer as an input parameter. Each instance is stepped through a sequence of free-flight and scattering events until its state is brought up to the correct value for the next timestep. A single carrier at a time or an ensemble of carriers

can be simulated, depending on the study performed. The method is often viewed as a direct simulation of the physical transport processes occurring in the material or device rather than a numerical method. Table 3.2 describes the different functions that comprise the code implementation of the MC method in this work.

Table 3.2: MC Kernel functions

Function Name	**Description**
processCarrier	Calculates flight times for *updateMomentum* and handles scattering events
updateMomentum	Computes new momentum and position for carrier after given flight time, calls *boundaryScatter*
boundaryScatter	Checks new carrier location against domain boundary list and corrects trajectories
BandStructure::BandStructure	Loads and initialized stored dispersion relation table from file
BandStructure::˜BandStructure	Safely deallocates stored dispersion relation table
BandStructure::E	Computes $E(\mathbf{k})$ when using a stored dispersion relation table
BandStructure::k	Computes $\mathbf{k}(E)$ when using a stored dispersion relation table
BandStructure::v	Computes $\mathbf{v} = \nabla_{\mathbf{p}} E(\mathbf{p})$ when using a stored dispersion relation table

3.2 Random Number Generation

An MC method relies on generation of random numbers according to a specified probability density function (PDF). There are two complicated aspects to this process. First, computers are generally only capable of generating pseudorandom numbers, which is considered acceptable for numerical applications. Henceforth, the distinction between truly random and pseudorandom numbers will not be made, and all references to "random"

numbers should be interpreted as pseudorandom. Second, most standard programming libraries only generate uniformly or normally distributed random numbers. Thus, a central component to an MC software program is an algorithm for transforming a uniformly distributed set of numbers into a set of numbers with a prescribed distribution.

MC methods make use of various techniques for performing these transformations. In this work, the two major techniques used are rejection sampling and inverse transform sampling. The first method, rejection sampling, is simply generating uniformly random data points while only storing those which contribute to the desired probability density. The method takes advantage of the fact that sampling according to a PDF can be done by uniformly sampling 2D points and keeping samples which fall into the region below the graph of that PDF. The second method is inverse transform sampling. Inverse transform sampling makes use of the quantile function of the PDF, which is the inverse of the corresponding cumulative distribution function (CDF). The CDF is the integral of the PDF for a given distribution, and so it is a mapping which takes numbers from the domain of the distribution to a probability between zero and one. The quantile function therefore maps a set of uniformly distributed samples from the interval $[0, 1]$ into the distribution used. Due to the fact that many distributions have PDFs which cannot be easily integrated, the quantile function may or may not have an analytical form. Inverse transform sampling can be used with a numerical solution giving the quantile function in those cases, but otherwise it may be necessary to resort to rejection sampling. If the quantile function is mathematically complex, it may be still computationally less expensive to perform rejection sampling. There were no dedicated random number generation functions created for the

MC Kernel module, as random number generation occurs in various places throughout the module with the appropriate technique used in each case.

3.3 Initial Conditions

Using the above techniques for random number generation, a population of carriers is created to match a specified distribution. The energies must be distributed according to a distribution derived from the density of states. The density of states is generally found from the dispersion relation and corresponds to the inclusion of material properties in the simulation. Depending on the spin of the particles comprising the population, the occupation pattern of the available states may change. Boson and fermion initialization is therefore handled differently. Energies are picked first from the proper distribution, and the dispersion relation is used to compute wavevectors. Other properties are either derived from the energy and/or wavevector, such as velocity; or they are derived from simulation initialization parameters, such as position from doping regions. Population statistics generated in this way are clearly of a population in thermodynamic equilibrium. The desired solution, which represents a transport process, must necessarily be a non-equilibrium system. The MC method will converge from this equilibrium starting point towards the desired non-equilibrium solution. Theoretically, almost any distribution could be used to initialize the system, but equilibrium solutions are favored as they tend to have analytical or well-known properties. They are typically selected due to the expected similarity to the proper solution, reducing convergence time [66]. While the initialization of carriers is generically part of the MC method, the code for initializing individual *Carrier*

objects or *Carrier* object populations lies in the corresponding initialization functions within the Simulation module.

3.4 Calculation of Scattering Rates

Within the simulation, scattering serves to relax the population to the correct distribution. The dynamics of carriers as derived from the potentials mentioned in Chapter 2 do not take into account certain higher order effects. For phonons, higher order terms in the atomic force potential are neglected, but the statistics associated with those terms are restored by the inclusion of phonon-phonon scattering. For electrons, the ionized impurities, acoustic deformation potentials, and other sources of perturbing potentials were also not directly included. Scattering is also the mechanism used to restore the electron statistics to match the physical system.

Scattering rates are commonly dependent on carrier density, position, energy, and other properties. These rates are calculated quantum mechanically using a result known as FGR. FGR can be derived from perturbation theory, as shown in [67] and [23]. Let H' be the perturbing Hamiltonian, such as those due to the previously mentioned higher order effects. The unperturbed problem, $H|\psi\rangle = E|\psi\rangle$ from Equation 2.1, has a complete set of orthonormal solutions $\psi_{\mathbf{k}}(\mathbf{r})e^{-iE(\mathbf{k})t/\hbar}$. A general time-varying solution $\Psi(\mathbf{r}, t)$ can be composed as the complex superposition

$$\Psi(z, t) = \sum_{\mathbf{k}} c_{\mathbf{k}}(t)\psi_{\mathbf{k}}(\mathbf{r})e^{-iE(\mathbf{k})t/\hbar}. \tag{3.1}$$

The family of coefficients $c_{\mathbf{k}}(t)$ describes the probability amplitude associated with measuring the particle to be in a state with momentum pulse centered around $\hbar\mathbf{k}$. Let

the before-scattering momentum be represented by the symbol $\hbar\mathbf{k}_0$. Then, the before-scattering state is represented by $c_{\mathbf{k}_0}(t \leq 0) = 1$ and all other $c_{\mathbf{k}}(t \leq 0) = 0$. The perturbing potential "turns on" at $t = 0$ and the coefficients change to reflect the scattering event. The perturbation is assumed to be "weak." This assumption means that after $t = 0$, $c_{\mathbf{k}_0}$ is still nearly equal to 1 and other coefficients are small, but may now be nonzero. The probability of a state $\mathbf{k}_n$ being the final scattering state is

$$P(\mathbf{k} = \mathbf{k_0'}) = \lim_{t\to\infty} \left|c_{\mathbf{k}_0'}(t)\right|^2, \tag{3.2}$$

which gives the scattering rate from $\mathbf{k_0}$ to $\mathbf{k_0'}$ as

$$S(\mathbf{k_0}, \mathbf{k_0'}) = \lim_{t\to\infty} \frac{\left|c_{\mathbf{k}_0'}(t)\right|^2}{t}. \tag{3.3}$$

To find scattering rates, one must solve for $c_{\mathbf{k}_0'}(t)$ and evaluate the limit in Equation 3.3. Doing so involves substituting the wavefunction expression in Equation 3.1 and the total Hamiltonian into the time-dependent SE

$$\left(H_0 + H' - i\hbar\frac{\partial}{\partial t}\right)\Psi(z,t) = 0. \tag{3.4}$$

This substitution results in the equation

$$H' \sum_{\mathbf{k}} c_{\mathbf{k}}(t)\psi_{\mathbf{k}}(\mathbf{r})e^{-iE(\mathbf{k})t/\hbar} = i\hbar \sum_{\mathbf{k}} \frac{\partial c_{\mathbf{k}}(t)}{\partial t}\psi_{\mathbf{k}}(\mathbf{r})e^{-iE(\mathbf{k})t/\hbar} \tag{3.5}$$

in terms of the probability amplitude coefficients. Taking the inner product of both sides of Equation 3.5 against $\psi^*_{\mathbf{k}_0'}e^{iE(\mathbf{k}_0')t/\hbar}$ yields

$$i\hbar\frac{\partial c_{\mathbf{k}_0'}}{\partial t} = \sum_k H_{\mathbf{k}_0'\mathbf{k}_0} c_{\mathbf{k}}(t)e^{i(E(\mathbf{k}_0')-E(\mathbf{k}_0))t/\hbar}, \tag{3.6}$$

where $H_{\mathbf{k}_0'\mathbf{k}_0}$ is called the matrix element and is given by

$$H_{\mathbf{k}_0',\mathbf{k}_0}(t) = \int_{-\frac{L}{2}}^{+\frac{L}{2}} \psi_{\mathbf{k}_0'}(z)U_S(z,t)\psi_{\mathbf{k}_0}dz. \tag{3.7}$$

Using the weak perturbation assumption and integrating Equation 3.6, the expression

$$c_{\mathbf{k}_0'}(t) = \frac{1}{i\hbar} H_{\mathbf{k}_0'\mathbf{k}_0} e^{i\Lambda t/2} \frac{\sin(\Lambda t/2)}{\Lambda t/2} t \tag{3.8}$$

is found where

$$\Lambda = [E(\mathbf{k_0'}) - E(\mathbf{k_0}) \mp \hbar\omega]/\hbar. \tag{3.9}$$

Summing this expression and taking the limit in Equation 3.3 gives the transition probability, per unit time, as

$$\begin{aligned} S(\mathbf{k}_0, \mathbf{k}_0') = \quad & \frac{2\pi}{\hbar} \left| H^a_{\mathbf{k}_0', \mathbf{k}_0} \right|^2 \delta\left(E(\mathbf{k}_0') - E(\mathbf{k}_0) - \hbar\omega\right) \\ & + \frac{2\pi}{\hbar} \left| H^a_{\mathbf{k}_0', \mathbf{k}_0} \right|^2 \delta\left(E(\mathbf{k}_0') - E(\mathbf{k}_0) + \hbar\omega\right) \end{aligned} \tag{3.10}$$

which is the final statement of Fermi's Golden Rule. To use FGR to obtain a rate of scattering out of a specific $\mathbf{k_0}$ state, one must integrate over the range of possible $\mathbf{k_0'}$ vectors.

These separate, independent scattering processes derived from each perturbing potential are characterized by their scattering rates Γ_i and their scattering time τ_i related by

$$\tau_i = \frac{1}{\Gamma_i}. \tag{3.11}$$

The net rate of scattering of all combined processes is simply the sum

$$\Gamma_T = \sum_{i \in S} \Gamma_i, \tag{3.12}$$

where the subscripts T and i indicate the total rate and the rates associated with individual processes, respectively.

3.5 Self-scattering and After-scattering State Selection

This work's MC method implementation relies on generating random flight times such that the resulting average carrier scattering rates are consistent with the results given by FGR. Expressions for computing these flight times and scattering probabilities depend on the aforementioned parameters in complicated, nonlinear ways. A key advancement in the MC method by Rees was a way to account for these parameter-dependent scattering rates [38, 39]. This approach works by imagining an additional, fictitious scattering processes called "self-scattering." The rate of self-scattering is defined to be such that the total scattering rate is constant. That is, the parameter dependence of self-scattering exactly opposes the parameter dependence of the sum of the real scattering processes' rates. The after-scattering state of a self-scattering event is chosen to be equal to the before-scattering state, so particles undergoing self-scattering to not alter the statistics of the population. By incorporating self-scattering, MC codes can compute scattering lifetimes and other data assuming a constant total rate.

The sum of all scattering processes, including self-scattering is demonstrated in Figure 3.2. This figure was generated using this work's MC module scattering functions for electrons described in Section 4.3. Scattering in the simulation proceeds according to the process described in Section 3.7 using the total scattering rate, meaning the total scattering rate is used to compute all carrier lifetimes.

Once a scattering event is reached by the code, the program must make a determination assigning that scattering event to a particular process, potentially including self-scattering. This determination is made using a uniformly distributed random number from the interval $(1, 0)$. The interval between 0 and 1 is partitioned, as shown in Figure 3.3,

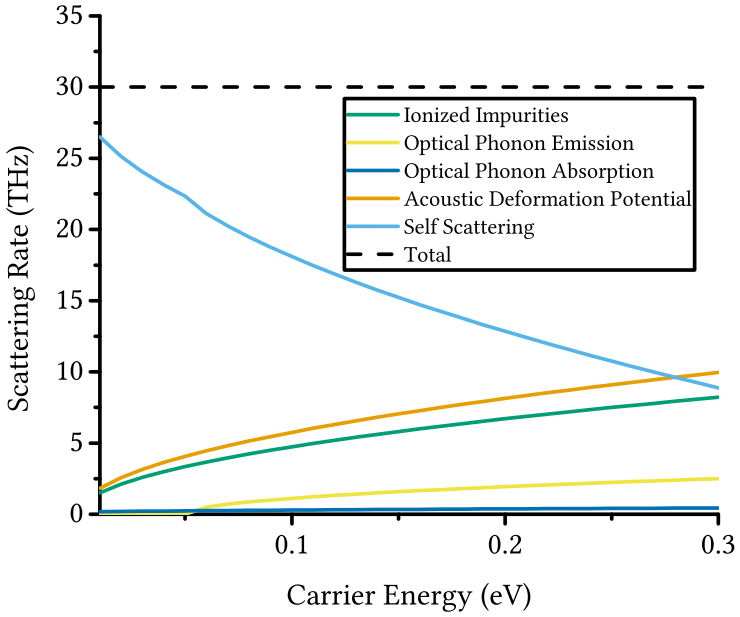


Figure 3.2: Scattering rates as a function of energy

into regions such that there is one region per scattering process, including self-scattering. The region containing the generated random number selects the scattering process. It

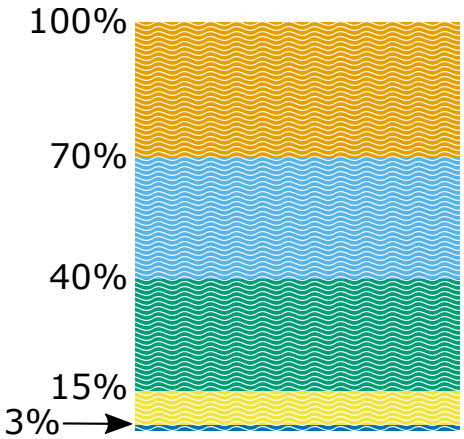


Figure 3.3: Example scattering process selection partitions

is only at this point that the actual individual, parameter-dependent scattering rates are computed using physics-derived formulas previously discussed. The self-scattering is

computed last, picked by subtracting the other process rates from the "global scattering rate" specified at the start of the simulation. The partition endpoints are chosen such that the the size of the process's region compared to the unit interval matches the ratio of that process's scattering rate and the total scattering rate. For this to work, the global scattering rate must be picked such that the sum of real scattering rates is not expected to exceed it during simulation. Before simulation, the operator provides the global scattering rate as a constant input to the simulation.

Once a scattering process is determined, additional random numbers are used with process-specific formulas to calculate the after-scattering state. These formulas are physics-derived and may vary in complexity. Velocities, or momenta, may be chosen by simply resetting the velocity vector direction or may be derived from more complicated physical constraints. The new carrier state is drawn from random distributions matching the physics of the scattering process. As more and more scattering events occur, those distributions affect the total population statistics. The impact of scattering on the total population is proportional to the relative rates and strengths of the scattering processes selected for representation in the code.

3.6 Carrier Lifetime Generation

In the MC method, carrier scattering is assumed to occur at a constant rate, because the likelihood of scattering is always the same in any given period of time. In terms of a single carrier's lifetime, this means it will on average scatter the same number of times in any period of the same length. Considering an ensemble of carriers, an equivalent interpretation of the constant scattering rate is the average proportion of the ensemble

that will have scattered after the period passes. In both cases, the average behavior can be summed up in the expression

$$\Delta n_{CF} = -\Gamma_0 n_{CF} \Delta t, \tag{3.13}$$

where n_{CF} is the concentration of unscattered carriers still undergoing free flight, Δn_{CF} is the change in that concentration during the free-flight period, Γ_0 is the average number of scattering events per unit time, and Δt is the time interval of flight. Allowing the differences to go to zero, the differential equation

$$\frac{dn_{CF}}{dt} = -\Gamma_0 n_{CF} \tag{3.14}$$

is obtained. The solution to this equation gives the probability density of scattering,

$$\mathcal{P}(t) = \Gamma_0 e^{-\Gamma_0 t}. \tag{3.15}$$

This function is the probability density of an exponential distribution, or the probability per unit time of a particle scattering in the interval $(t, t + dt)$. The quantile function of an exponential distribution is known to be

$$P^{-1}(p; \Gamma_0) = -\Gamma_0 ln(1 - p), \quad 0 \leq p < 1 \tag{3.16}$$

As $1 - p$ is uniformly distributed in the interval $[0, 1]$, it can be replaced by p in the above equation. The final result is the standard formula for generating free flight times for carriers,

$$\tau = -\Gamma_0 ln(r) \tag{3.17}$$

where τ is the scattering time, Γ_0 is the process scattering rate, and r is the output of a uniform pseudorandom number generator.

3.7 Implementation of Scattering in Code

There are two equivalent ways to implement constant-rate scattering in code, based on the above equations. In the first approach, a carrier or ensemble of carriers are all simulated through a fixed timestep Δt. For each carrier, a uniform pseudorandom number r is generated. Inverting Equation 3.16 gives

$$r_0 = 1 - e^{\left(\frac{-\Delta t}{\Gamma_0}\right)}. \tag{3.18}$$

A comparison can be done, and if $r < r_0$ then the carrier is scattered. The net effect of processing the carriers this way is that at the end every timestep Δt, on average a number of carriers will scatter equal to the number of carriers that should have scattered in that timestep. This implementation is simple, but has the drawback of "snapping" scattering events to intervals of Δt.

The second approach, depicted in Figure 3.4, is computationally more expensive but allows arbitrary free flight times. A timestep Δt is still utilized, but carrier trajectories are not immediately computed through the entire timestep duration. Instead, carrier lifetimes are generated using Equation 3.17. Lifetime calculation is done during carrier initialization or immediately after scattering. If a carrier's lifetime is greater than the timestep length, then that carrier position and momentum is merely updated using classical equations of motion and the local electric field vector through to the end of the timestep. The carrier lifetime is then reduced by a value of Δt and the carrier is ready for the next timestep. However, if the particle is due to scatter before the end of the timestep, the position and momentum are updated only to the time of the scattering event. At the end of that lifetime, the carrier is scattered and a new lifetime is generated. The particle is assigned a new state

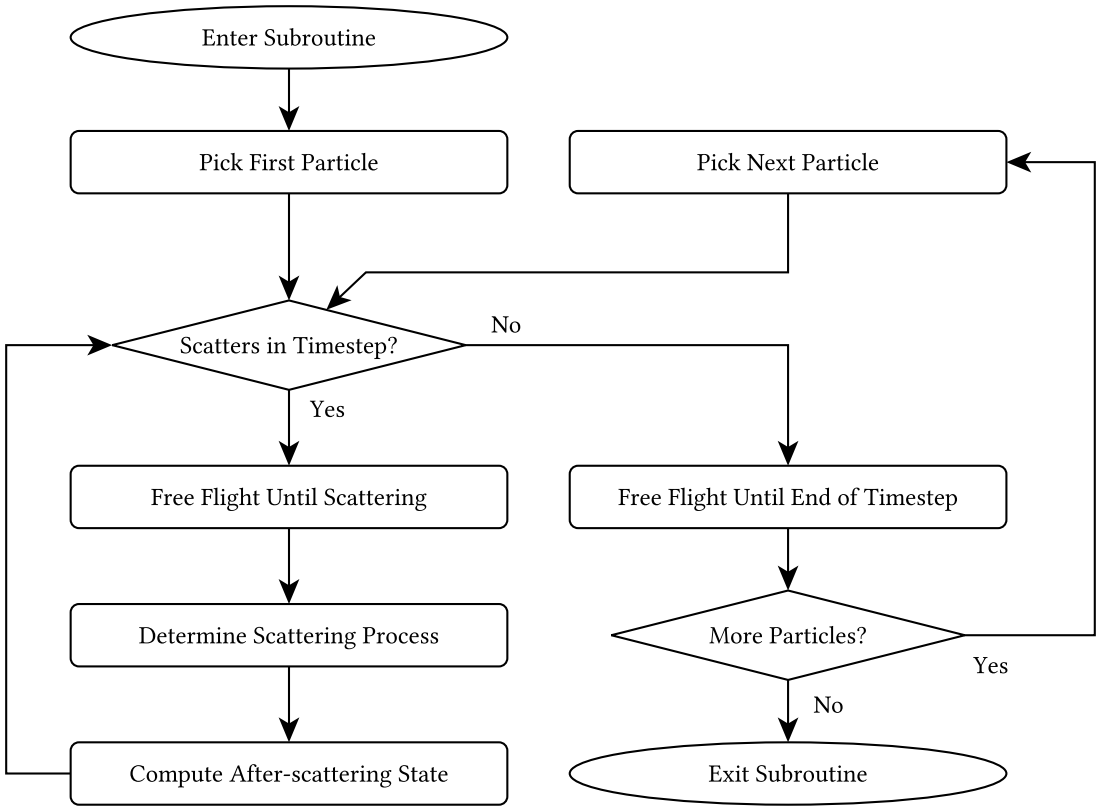


Figure 3.4: Carrier dynamics and scattering flowchart

using the band structure and scattering process data. The carrier and its new lifetime are recursively considered according to the preceding logic, until the carrier reaches the end of the timestep interval.

3.8 Treatment of Particle Dispersion

The dispersion relation is an important input parameter for a given carrier species, as it defines major behavioral properties of the carrier as it moves through the material medium. In code, the dispersion relation can be treated two ways: analytically or numerically. Both approaches are used in this work. In the analytical approach, the dispersion relation and its derived expressions are directly used in the code as closed-form

functions. The inverse of the dispersion relation is used during initialization when selecting wavevector magnitudes from randomly sampled energies. The derivative of the dispersion relation is used to select carrier velocity components from the wavevector components. So, the general requirements of analytical dispersion relations are the existence of a reasonable analytic derivative and a reasonable analytic inverse. The analytic approach is sometimes taken for low-energy electrons and holes where a few analytic approximations with simple derivatives and inverses are available. The analytic approach is taken in this work for some electron subclasses, and the code corresponding to the analytic function is described in Chapter 4.

If the dispersion relation is complex and cannot be used directly in an analytic form, or if the analytic forms involve slow math API functions, then the numerical approach must be used. In the numerical approach, a table is created in memory. The dispersion relation table stores wavevectors indexed by energy value. Generally, the dispersion relation is a many-to-many relationship and fetching entries from the table is complicated. A C++ support class has been created for this purpose called *BandStructure*. The class constructor reads a list of (E, k_x, k_y, k_z) coordinates for a number of degenerate bands of the dispersion relation from a binary file. Note that the file IO operations are conducted only in one node, and MPI functions are used to synchronize the loaded band structures between other allocated notes. The process of fetching an energy value from the table is provided by the function *Bandstructure::E*. This function merely returns the energy corresponding to the closest stored $\mathbf{k}$ in the table.

The wavevectors all lie within a region of reciprocal space called the 1st Brillouin Zone. This zone typically has a high level of symmetry. A sub-region called the irreducible

wedge may exist that can be rotated and positioned to cover the entire space. The function *BandStructure::wedge* returns the vector in the irreducible wedge that is symmetric to a given input vector. This function and its sister function *BandStructure::dewedge*, are used to exploit the symmetry of the space and reduce memory requirements. When producing wavevectors from the table, all degenerate energy states are considered equally as likely. This includes symmetric partners of wavevectors in the irreducible wedge. The *BandStructure::dewedge* function automatically picks from any such wavevector, distributed uniformly randomly.

The function *BandStructure::k* is responsible for fetching wavevectors corresponding to a particular energy. The code in *BandStructure::k* searches for all stored wavevectors corresponding to an energy within an interval around the argument energy. Each band is searched and the results from all bands are collected into a shortlist. A vector is uniformly randomly selected and passed through *BandStructure::dewedge*. The resulting vector and its corresponding band index are returned to the calling function.

The final *BandStructure* support function is *BandStructure::grad* which computes the gradient of energy with respect to wavevector for a given band and argument wavevector. This function is used to compute velocities from the table when the *BandStructure* object acts as a carrier's dispersion relation. The *BandStructure::grad* function recognizes the symmetry of the 1st Brillouin Zone as well.

3.9 Carrier Dynamics

As described in Sections 2.1 and 2.2, the dynamics of the semi-classical particles are derived from their dispersion relations. The following familiar results from classical

physics apply:

$$\mathbf{F} = \frac{d\,\mathbf{p}}{dt} = -\nabla_{\mathbf{r}} H, \tag{3.19}$$

and

$$\mathbf{v} = \nabla_{\mathbf{p}} H, \tag{3.20}$$

where H is the Hamiltonian of the particle species being discussed, $\mathbf{F}$ is the dynamic force acting on the particle, $\mathbf{r}$ is the position of the particle, and $\mathbf{v}$ is the velocity of the particle. Note that for electrons and phonons, the Hamiltonian has a useful property: the only the contribution to the Hamiltonian that depends on position is the potential energy, while the only contribution to the Hamiltonian that depends on momentum is the kinetic energy. Thus, Equations 3.21 and 3.22 can be rewritten as

$$\mathbf{F} = \frac{d\,\mathbf{p}}{dt} = -\nabla_{\mathbf{r}} U, \tag{3.21}$$

and

$$\mathbf{v} = \frac{d\,\mathbf{r}}{dt} = \nabla_{\mathbf{p}} E(\mathbf{p}), \tag{3.22}$$

where U is the potential energy and $E(\mathbf{p})$ is the band structure or dispersion relation, in terms of $\mathbf{p}$. Now the force on an electron quasi-particle due to an applied electrostatic potential, aside from the lattice ion potential, can be calculated from its gradient. The velocity of electrons and phonons can also be calculated from the gradient of the band structure or dispersion relation. Velocity computed this way also corresponds to the group velocity of the carrier wavefunction, and therefore to the velocity of the envelope of the carrier wavepacket. These equations are the primary principles used to simulate the dynamic trajectories of semi-classical carriers. The gradient of electrostatic potential is

used in the PE solver to generate the electric force field. The force field is referenced in *updateMomentum* to incrementally adjust the momentum vector. The carrier object uses its adjusted momentum vector to compute its own velocity from its band structure. The resulting velocity is obtained via the *Carrier::v* function and used to incrementally adjust the position vector.

3.10 Calculating the Distribution Function

MC simulations can be either single-particle or ensemble simulations. The main characteristics of the approach are similar. A single carrier can be simulated, replacing it with a randomly initialized particle as it encounters boundary conditions which may remove it from the simulation domain. The amount of time the carrier spends, on average, in different regions of the phase space will contribute to the magnitude of the distribution function of that region. The carrier must travel to every region of the phase space within the simulation domain's constraints many times to build up a reasonable approximation of the distribution function for the entire domain. The single-particle mode is therefore more appropriate for bulk tests, such as the mobility experiments in Ch. 4, where a specific cross section of a highly symmetric phase space is being examined. In the ensemble mode, the distribution function is computed by comparing the proportion of carriers occupying a specific region of the phase space to the total number of simulated carriers. Due to this difference, coarse data is available immediately for the entire device. In particular, the ensemble mode will converge faster in device simulations where field quantities are used as an input to self-consistently solve a simultaneous equation such as the PE. Using the PE

as an example, the entire device's distribution function is required to compute the charge density as an input to the PE solver.

CHAPTER 4

ELECTRON MONTE CARLO IN SILICON

This chapter describes characteristics specific to the modeling of electrons with the MC method introduced in Chapter 3, and presents the results of several electron MC simulations. Electrons inside a Si lattice acquire a unique band structure different to the dispersion relation of the other particles simulated in this work. Electrons also require special treatment due to their nonzero electric charge and status as fermions.

The different models of the electronic Si band structure will be discussed and compared, along with the details of their representation within the code. The initialization of electron individuals and electron populations is also discussed, and this chapter also will elaborate on the relevant scattering processes seen by electrons in Si and the experiments that were performed to verify the correctness of the approach.

The simulations presented serve to verify the correctness of the MC simulation. A simulation of an electric field applied bulk material was performed to verify that the tool can properly reproduce the temperature dependence of electron mobility in Si. Two devices, an n-i-n diode and a nanoscale FinFET, were simulated to test the ensemble simulation mode, various features of device boundary conditions, and the quantum confinement code. While the n-i-n device structure is based on earlier work by another author, the specific

form of the diode and the presented simulation results are original to this work. The FinFET simulation results are also presented for the first time here.

4.1 Initialization of Fermion Populations

Electrons can theoretically be initialized within the simulation to a range of physical and unphysical populations. Some populations may lead to numerical instability that causes the simulation to halt, but the likelihood of this is expected to decrease if the population statistics resemble a solution of the PE, SE, and BTE. In order to reach acceptable convergence times, it is also prudent to initialize the population to have statistics near in some sense to the expected solution found by the simulation. Simply giving the electrons the final solution's statistics should of course also be avoided, but this is generally not possible anyway due to the solution being unknown *a priori*.

The established method, which follows the above guidance, is to initialize the electron population to the equilibrium population from which the final solution is expected to depart. For electrons, this means initializing the particles so that their energy follows the Fermi-Dirac distribution

$$f_{FD} = \frac{1}{e^{\frac{E}{k_B T}} + 1}, \tag{4.1}$$

where E is the energy of the particle above the Fermi energy, k_B is Boltzmann's constant, and T is the population temperature. The generation of velocities can then follow from the band structure chosen, and the particle positions can be initialized according to the doping profile of the device being simulated. The Fermi-Dirac distribution has no trivial inverse due to the form of the denominator. The distribution must therefore be generated by the code through the use of rejection sampling instead of inverse transform sampling. However, it is notable

that the Fermi-Dirac distribution will approach the Maxwell-Boltzmann distribution in the high temperature or low density limits. Thus, the Maxwell-Boltzmann distribution can be used for low concentrations where electron degeneracy can be neglected. The Maxwell-Boltzmann distribution takes the form

$$f_{MB}\frac{1}{e^{\frac{E}{k_BT}}}, \tag{4.2}$$

where all variables have the same interpretation as in Equation 4.1. Physically, this can be interpreted as the spins of the electrons minimally interacting due to their large average separation distance. For these low concentrations, the inverse transform sampling of the Maxwell-Boltzmann distribution may be used to approximate the proper energies, and this approximation is utilized for generation of electron populations in all simulations associated with this work. Random energies are generated using the quantile function for the Maxwell-Boltzmann distribution,

$$E = \frac{3}{2}k_BT\ \ln(r), \tag{4.3}$$

where r is a uniform random number in the interval $[0, 1]$. The units of E depend on the units k_B is recorded in. In this work, all quantities are stored internally with base SI units, so E has units of joules.

4.2 Models of Silicon Band Structure

The band structure of Si is anisotropic due to the anisotropy of its diamond lattice structure. The diamond lattice is a kind of cubic lattice, so the 1st Brillouin Zone appears as a truncated dodecahedron as shown in Figure 4.1. The overall band structure inherits the 48-fold symmetry of the 1st Brillouin Zone. The band structure also departs markedly

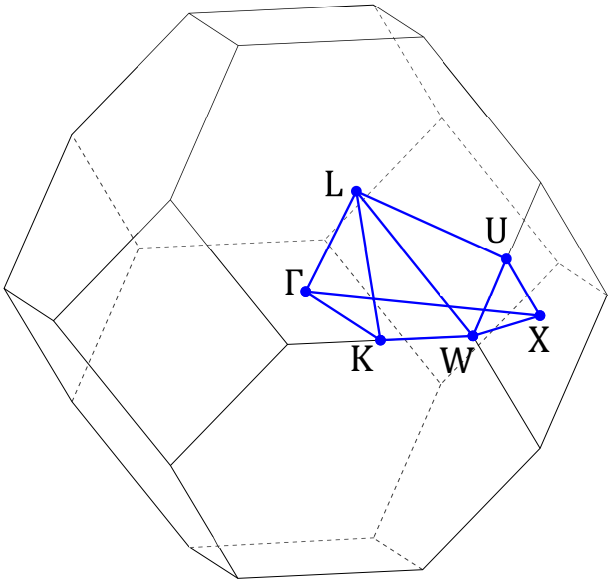


Figure 4.1: 1st Brillouin Zone for a diamond lattice

at high energies from the simplified parabolic effective mass approximation. The full band structure was calculated as part of this work, and is presented in Figure 4.2. The figure

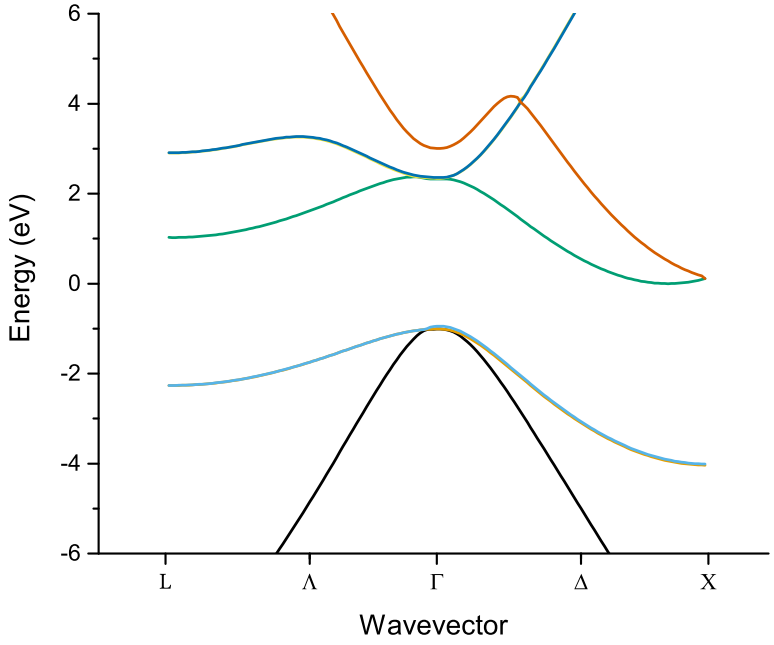


Figure 4.2: Full band structure of Si

shows the Si band structure along high symmetry lines, displaying a hallmark feature of the Si band structure: the appearance of lowest energy minimum near the edge of the 1st Brillouin Zone, along the $\Gamma - X$ line of symmetry. The relationship $E(k)$ is approximately parabolic as k is varied away from these valley minima in any direction. In Fig. 4.2, the parabolic shape of the low energy region of the X-valley is evident.

The fullband structure was calculated using the empirical pseudopotential method as described in [68]. The empirical pseudopotential method uses a plane wave expansion of the crystal Hamiltonian in Fourier space to calculate the eigenvalues of the Hamiltonian at each specific site in the 1st Brillouin Zone. The method uses pseudopotentials, which means the electrostatic potential of the lattice cell is approximated using only the net charge of the lattice ion cores. This method is applied in MATLAB [69] to a set of inputs which includes the desired **k** vectors and a list of form factors and pseudopotentials determined by experiment. The MATLAB data is saved as a band table in the format required by the band structure object. In this work, three eigenvalues above the band gap are saved as three bands in the band table file. The **k** vectors chosen are vectors in a regular, rectangular mesh where such points lie within the irreducible wedge. The mesh spacing was chosen so that 100 mesh points exist along one axis in the reciprocal lattice unit cell.

Fig. 4.3 highlights the six equivalent X-valleys, symmetrically placed around the 1st Brillouin zone. The anisotropy of the system manifests as ellipsoidal, rather than spherical, surfaces of constant energy around the valley minima. As the effective mass is dependent on the slope of the energy as a particle moves out of the ellipsoid, away from the minimum, the direction of increasing k will select different effective masses. This distinction is the origin of the colors in Figure 4.3, showing the valleys corresponding to transverse effective

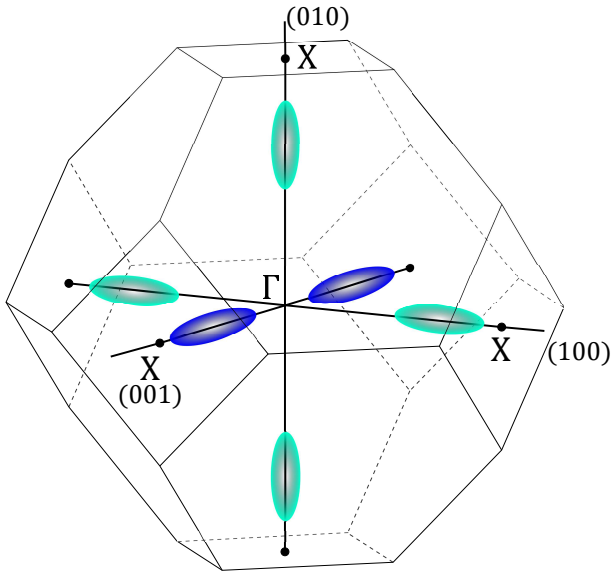


Figure 4.3: Six valleys of lowest energy

masses in light blue and longitudinal effective masses in dark blue. In any given population where the transport is mostly in the ⟨100⟩ direction, about one third of the electrons will be in a valley aligned with the ellipsoid's semi-major axis while two thirds will be in a valley arranged perpendicularly. The occupation of the different valleys leads to a mean effective mass which can be used to create an isotropic model to reproduce many transport metrics, namely conductivity. This mean is referred to as the conductivity effective mass.

The isotropic parabolic model for the 6 valleys is

$$E(\mathbf{k}) = \frac{\hbar^2(\mathbf{k} - \mathbf{k}_v)^2}{2m_c}, \tag{4.4}$$

where $\mathbf{k}_v$ is the wavevector of the minimum energy point in valley v and m_c is the conductivity effective mass. Distinguishing which valley the electron resides in is unnecessary in the isotropic case. In literature and in code, the displacement of the valley minimum is

typically suppressed:

$$E(\mathbf{k}) = \frac{\hbar^2 \mathbf{k}^2}{2m_c}. \tag{4.5}$$

In both cases, the constant energy surfaces can be made ellipsoidal by replacing m_c with the reciprocal effective mass tensor

$$\left(\frac{1}{m}\right)_{uv} = \begin{bmatrix} \frac{1}{m_{xx}} & 0 & 0 \\ 0 & \frac{1}{m_{yy}} & 0 \\ 0 & 0 & \frac{1}{m_{zz}} \end{bmatrix}, \tag{4.6}$$

where m_{xx}, m_{yy}, and m_{zz} are the effective masses of the corresponding coordinate axes.

The effective masses in Equation 4.6 are selected from the transverse effective mass m_t or the longitudinal effective mass m_l depending on the orientation of the valley. To better match the higher energy tails of the distribution, a nonparabolicity factor α can be added to the nonparabolic band's dispersion relation. This modification takes Equation 4.5 into the form

$$E(\mathbf{k})[1 + \alpha E(\mathbf{k})] = \gamma = \frac{\hbar^2 \mathbf{k}^2}{2m_c}, \tag{4.7}$$

which can also be modified to account for anisotropic valleys by similarly replacing the effective mass on the right hand side.

As previously mentioned in Chapter 2, velocity for parabolic bands is found from momentum via $\mathbf{v} = \mathbf{p}/m_c$. For nonparabolic bands, the gradient of the band function with respect to $\mathbf{p}$ must be taken to produce the following velocity function:

$$\mathbf{v} = \frac{\mathbf{p}}{m\sqrt{1 + 4\alpha\gamma}}. \tag{4.8}$$

The above approximations lead to the three well-accepted ways in the literature to describe the band structure of simulated electrons in Si: a single isotropic, parabolic band; a "model band structure" consisting of six anisotropic, nonparabolic valleys; and, a table with a fully detailed, numerically computed band structure. For convenience, these band structures are referred to throughout this work as the parabolic band structure, the nonparabolic band structure, and the full band structure, respectively. Of these three models, only the full band structure does not possess an invertible closed form expression. As the function $E(\mathbf{k})$ must be inverted several times during the simulation, there are clear advantages to using such expressions. However, only the full band structure is capable of reproducing the correct velocity distribution at high energies even when compared to using nonparabolicity factors.

The object-oriented nature of the code associated with this work allows the creation of a natural set of *Carrier* objects corresponding to the different band structures. The subclass-specific functions for *ElectronSiParabolic*, *ElectronSiNonparabolic*, and *ElectronSiFullband* utilize the proper formulas as discussed above for the generation of energy, wavevector, and velocity. The class *ElectronSiFullBand* utilizes the *BandStructure* class described in Chapter 2.

4.3 Scattering of Electrons in Silicon

Scattering of electrons in Si crystals happen as the electrons interact with and are scattered from various perturbing potentials. Perturbing potentials are generally of two classes: irregularities in the lattice and dopant ion charges. Irregularities in the lattice scatter electrons due to the fact that the repeating, perfect crystalline structure is already

"baked-in" to the flight dynamics via the band structure. Lattice ions that are missing or present when it was assumed otherwise will therefore affect the dynamics of the electron quasi-particle. Irregularities in the lattice due to actual physical imperfections such as crystal defects will cause scattering, as well as distortions of the lattice due to thermal vibrations. The latter case is interpreted as electrons scattering against phonons.

The carrier scattering rates used in the simulation were computed using the procedure outlined in [23]. The model employed includes four scattering processes: acoustic deformation potential scattering, intervalley phonon absorption, intervalley phonon emission, and ionized impurity scattering. These processes were chosen to reproduce the correct bulk mobility dependence on temperature and doping concentration. The ionized impurity scattering introduces variation of mobility with doping concentration, while the acoustic deformation potential scattering introduces strong dependence on temperature. These four scattering processes are used for both this work's bulk mobility verification and the device simulations. The ionized impurity scattering rate is the Conwell-Weisskompf formulation [70]. The functional form of the phonon scattering formulas are taken from [23], with constant parameters chosen to reproduce the correct temperature dependence of mobility. All of these rate formulas are derived from FGR in the cited sources, and thus have a similar form. The following equations summarize the scattering rate expressions used in this work:

$$\frac{1}{\tau_{ADP}} = A_1 T_L^{\eta} \sqrt{E(k)} \tag{4.9a}$$

$$\frac{1}{\tau_{abs}} = A_2 \sqrt{E(k) + \hbar\omega_0} \tag{4.9b}$$

$$\frac{1}{\tau_{emi}} = A_3 \sqrt{E(k) - \hbar\omega_0} \tag{4.9c}$$

$$\frac{1}{\tau_{ion}} = \quad A_4 N_1^{\frac{1}{3}} \sqrt{E(k)} \tag{4.9d}$$

where τ_n is the characteristic scattering time for process n in seconds, T_L is the lattice temperature with exponential factor η, N_1 is the semiconductor dopant concentration in cm^{-3}, $E(k)$ is the energy band structure in eV, and $\hbar\omega_0$ is the optical phonon energy for intervalley phonon scattering in eV. More complex expressions that explicitly take into account band structure anisotropy or other material properties can be derived as well, but in practice a simple tuning process of the coefficients is enough to reproduce the bulk electron conductivity, as described in the following section.

Selecting the state after scattering can be complicated. For electrons interacting electromagnetically with ionized impurities, the final state is typically a path whose direction is only deflected by a small angle. Dead-on strikes perturb the path much more dramatically, but they are far less common than glancing approaches to the perturbing potential. This can be implemented directly in the code if the angular distribution of electron velocities is expected to be a significant quantity. For replicating the bulk mobility, an alternative approach is to effect the same net change in momentum by scattering with uniform angles but much less frequently. This approach is taken in the code, and it affords an advantage in computation time and makes the process for selecting after-scattering velocities identical for ionized impurity scattering and phonon scattering. Phonon scattering comes in two flavors: acoustic phonon and optical phonon scattering. These processes are elastic and inelastic, respectively. Elastic scattering conserves energy and selects a new wavevector uniformly randomly that corresponds to the same energy the particle had before scattering. Inelastic scattering does not conserve energy, which

corresponds to energy entering the lattice system and leaving the electron gas. A new wavevector is selected that corresponds to the lower or higher energy of the scattering state.

4.4 Verification of Bulk Electron Mobility

Since the MC tool was custom-created, its accuracy had to be assessed before it could reliably be used to study more complex devices or materials. This mobility simulation was conducted to verify the appropriateness of the selected scattering processes by reproducing the temperature dependence of the electron mobility of bulk Si. Another goal of this simulation is to pin down the correct scattering rate coefficients that set the relative strengths of the processes. The ability to reproduce the temperature trend is taken as support that the scattering processes chosen, their functional forms, and the coefficients are correct and sufficiently descriptive of the internal device physics for simulations involving conductivity, mobility, and temperature. Thus, it is assumed that scattering coefficients which reproduce the mobility are reflective of physically correct proportions of contribution to the mobility from each process. This assumption applies exactly given full treatment of after-scattering state selection and approximately given the uniform selection of angles for impurity scattering. This section describes the overall simulation approach, the input parameters used, and the simulation results.

The tuning process started with bulk mobility computed from a single constant scattering rate. Mobility μ was computed by letting electrons move under the influence of constant electric field $\mathbf{E}_{appled}$, computing their average velocity $\mathbf{v}_{avg}$, and using the formula

$$\mu = \frac{|\mathbf{v}_{avg}|}{|\mathbf{E}_{applied}|}. \tag{4.10}$$

When performing the bulk simulation, care must be taken to select domain sizes and simulation timesteps that allow for the scattering processes to occur without being limited. For instance, a low scattering rate's corresponding mobility will appear artificially higher if the mean free path is longer than the simulation domain. An identical aliasing effect occurs when the mean time between scattering is much longer than the simulation time. This phenomenon was observed early on in debugging.

Table 4.1 was prepared to assist in selecting simulation settings that were appropriate for different mobilities. Ideally, the scattering rate is related to the mobility through

Table 4.1: Limits for simulation settings

Γ (Hz)	τ (s)	μ (cm^2/V s)	$\lvert\mathbf{E}_{crit}\rvert$ (V/m)	$\lvert\mathbf{E}_{app}\rvert$ (V/m)	$\lvert\mathbf{v}_{avg}\rvert$ (m/s)	MFP (m)
10^{11}	10^{-11}	67717.67	42095.88	5000	33858.83	3.4×10^{-7}
10^{12}	10^{-12}	6771.77	420958.84	5000	3385.88	3.4×10^{-9}
10^{13}	10^{-13}	677.18	4209588.36	5000	338.59	3.4×10^{-11}

the relationship

$$\mu = \frac{q}{m_e}\tau, \tag{4.11}$$

where τ is the combined scattering time, q is the elementary charge, m_e is the electron mass. Equation 4.11 can be derived under the assumption that scattering events are elastic and completely randomize the carrier velocity. Carriers accelerate in the electric field, scatter randomly, and re-accelerate, which results in a finite average velocity. The relationship

between average velocity and average energy is found through the dispersion relation. For the purposes of Table 4.1, the parabolic dispersion relation was used. These formulas make it possible to determine mean free path, mean free flight time, and electric field given a target mobility for simulation.

Additionally, due to the presence of optical phonon emission as a selected scattering process, the mobility simulations must all take place below the critical field strength for velocity saturation. When the average carrier energy approaches the optical phonon energy, many carriers will become eligible for the optical phonon emission process. The electric field at which this occurs is related to the optical phonon energy by the formula

$$|\mathbf{E}_{crit}| = \mu^{-1}\sqrt{\frac{(2\hbar\omega_0)^2}{m_c}} \tag{4.12}$$

This relationship was used to identify the maximum applied electric field for each test. Mimicking the constant-rate mobility also serves to confirm the dynamics code in *processCarrier* and *updateMomentum* operate properly, and it verifies some of the basic scattering and state selection code as well.

Once it was verified that the constant-rate simulation followed the accepted formula for ideal scattering, the rate dependence was introduced. The different scattering processes each have different domains where they are to a good approximation the sole scattering process contributing to the mobility. Acoustic phonon scattering dominates in high temperatures, masking other processes. Ionized impurity scattering dominates in low temperatures with high doping concentration. Optical phonon scattering does not depend on any applied conditions, only electron energy, and is thus dominant in cold, intrinsic Si simulations. The scattering rate is linearly proportional to the mobility in all elastic

scattering processes, so selecting the rates for those processes in their dominance regime requires few iterations. Turning on inelastic scattering alters the average energy of the population in a complex way, which affects the other scattering processess' coefficients due to the formulas' energy dependence. This results in a nonlinear change in each coefficient being required to iteratively approach the final system of scattering rates.

The values for the scattering rate constants for Equations 4.9a through 4.9d can be found in Table 4.2.

Table 4.2: Scattering rate constants

Constant	Value
A_1	2.96×10^7
A_2	5.00×10^{12}
A_3	0.75×10^{12}
A_4	3.50×10^9
η	1.50
$\hbar\omega$	0.05 eV

Figure 4.4 shows the mobility vs temperature dependence extracted from the simulator compared to experimental values.

Good agreement is seen throughout the relevant temperature range. Of the four scattering types, the dominant scattering process is acoustic deformation potential scattering in both doped and intrinsic regions at high temperatures. There is no hard rule for how long to run the simulation. As more iterations are conducted, the sample average approaches the presumed true solution. Between 700 and 1000 core hours were used per

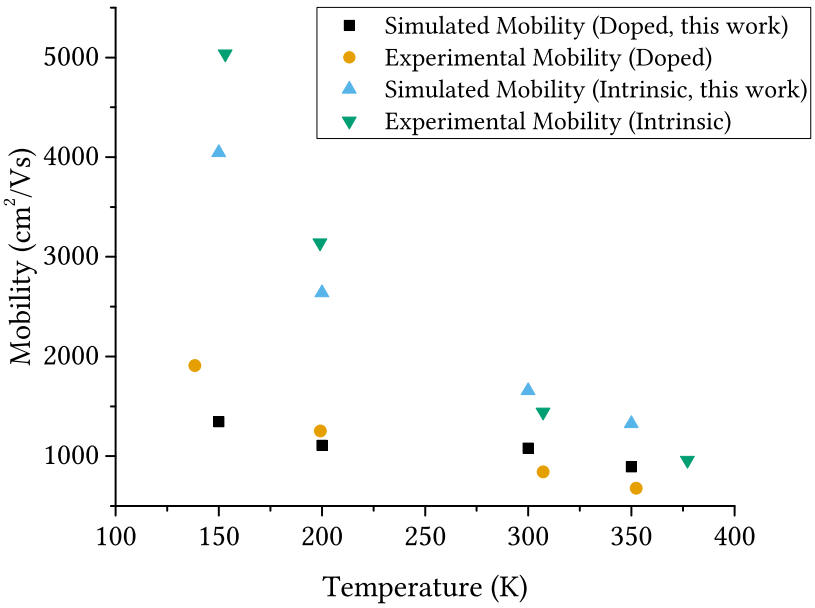


Figure 4.4: Comparison of simulated mobilities to experimental data from [1].

temperature point in Figure 4.4, so that the change in low order digits in the average at the end of each timestep was on the order of 10 cm^2/Vs.

4.5 Geometric Boundary Conditions

One of the primary advantages of the MC method is the support for arbitrary device geometries. In terms of code implementation, the boundary reflection calculation can become complicated, as many reflections and scattering events occur. When the carriers are processed in free flight, the position vector is incrementally updated. Using only the boundary geometry and the available carrier properties, boundary-crossing must be detected and corrected. In this section, the code for computing the boundary interactions is described. For the current implementation, all faces are rectangles with arbitrary size, position, and 90° orientation. Thus, any shape that can be composed of rectangular faces is supported.

The particle position is checked against the boundary geometry information for boundary violations immediately after any incremental update to its position vector. There are six possible arrangements of the particle and face: particle inside or outside of the face, for each coordinate axis. When a particle is checked against a face, the interaction is first classified as one of these scenarios. Then, the particle and face are transformed into a new coordinate system so that the interaction is always a particle on the inside of a face on the YZ-plane. This allows the same function to be used for all six scenarios.

Particles are then checked to see if they are outside the face plane *and* have an outward-moving velocity component. The particle velocity, current position, and face position are used to compute the time at which the particle must have intersected with the face. This time is compared against the last recorded time of scattering, which is a stored property of the particle. If the time of last scattering is more recent than the apparent collision time, then the face could not have been crossed. In that case, the function exits. Otherwise, it is assumed the particle passed through the face. The particle position at the time of intersection is computed. A new velocity vector is computed based on whether or not the face is flagged for diffuse or specular scattering. Specular scattering events are computed by reversing only the velocity component normal to the face. Diffuse scattering is performed by selecting a new random velocity that moves away from the face. The carrier travels forward again on the new trajectory for the time interval calculated earlier. The final, scattered particle position and velocity are transformed back into the original coordinate system, and the function exits. This algorithm for comparing a particle position to a single boundary face can be seen in the Figure 4.5. Multiple faces are handled by calling the boundary checking function on each face in the set of faces comprising the device. If

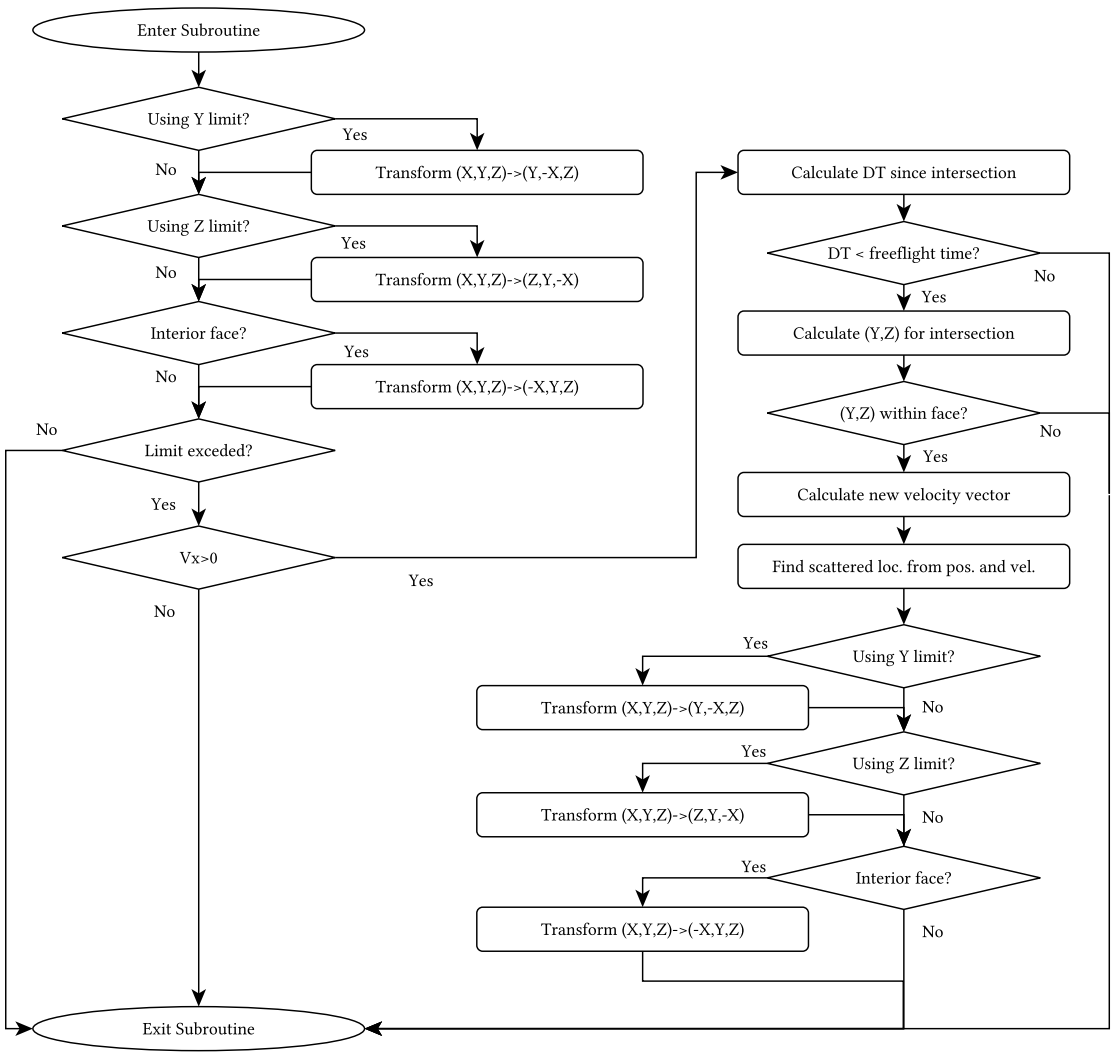


Figure 4.5: Carrier boundary checking flowchart

any face causes a scattering event, all of the faces are checked again. The re-checking of faces allows for correct scattering in situations such as tight corners.

4.6 Simulation of Electron Ensembles

Electrons are treated one-at-a-time in the bulk mobility simulations. Instead, a large population of particles can be simulated together. This mode of simulation is referred to as an ensemble simulation. Ensemble simulation must be used when data for every mesh site

in the device is required at once, upfront. This situation exactly occurs when simulating an electronic device, as the electric field must be calculated from local electron density throughout the device. A charge mesh must be created and the instantaneous amount of charge in the mesh cell at a given time is used to calculate the cell's charge density. Using the single-particle method, the required charge density distribution function could only be generated once the single particle had spent some amount of time in every region of the device. Initial data such as doping region, doping concentration, and material-specific information are provided to the MC initialization function via a device input file. This data tells the MC initialization function from which *Carrier* subclass to create *Carrier* instances, how many of each type there should be, and where they should be randomly placed during initialization. The MC initialization function also uses the scattering functions to compute an initial carrier free-flight time for each instance.

In device simulation, there must also be a new kind of boundary condition corresponding to Ohmic contacts. The Ohmic boundary condition is created by maintaining a neutral region of a few points deep in the charge mesh.

The MC module does so by adding or removing charge as needed to maintain the neutrality of the region and by allowing electrons to freely leave the device through those boundaries. This step is performed after new electron positions have been calculated by the MC dynamics and boundary code. First, the C++ erase-remove idiom is used on the data structure containing the electron population, using the *out_of_bounds_e* function. Second, an account is made of all of the electrons in the neutral region. The proper number for neutrality is calculated during initialization and stored as a global variable. If the number of electrons to add or remove has an integer part, then that many are added or removed

from the corresponding region. If the number to add or remove has a fractional part, then this fractional part is interpreted as a probability of adding or removing a single particle. Uniform random numbers are drawn from the interval [0, 1] and compared against the fractional part to determine if another particle will be added or removed. On average, this process tends to maintain the boundary cells that lie against the designated Ohmic contacts at net zero charge while allowing electrons to leave the device. Electrons are added according to the equilibrium distribution from whence the population was originally initialized. Current injection occurs in the simulation due to the applied electric field causing a shift in the center of the electron population. This shift causes a deficit in the cathode charge density that triggers the action of this portion of the simulation function to add carriers.

4.7 Poisson's Equation

In the event that the particles being simulated carry an electric charge, the electric field in the device must be computed by solving Poisson's equation before each iteration of the main MC Kernel function. Poisson's equation is a simple relation involving electric charge density ρ and electrostatic potential V:

$$\nabla_{\mathbf{r}}^2 V = \frac{\rho}{\varepsilon}, \tag{4.13}$$

where ε is the electric permittivity of the material. Here, V and ρ are both functions of $\mathbf{r}$ only, not t or $\mathbf{p}$. In this case, $\varepsilon{=}\varepsilon_{Si}\varepsilon_0$ where ε_{Si} and ε_0 are the relative permittivity of Si and the permittivity of free space, respectively. The Poisson code is organized into the functions described in Table 4.3. The quantity of interest is actually the electric field $\mathbf{E}$,

Table 4.3: Poisson module functions

Function Name	Description
setupPoisson	Called during initialization, prepares memory space for storing field, potential, and charge density
prepPoisson	Called before *solvePoisson*, assigns mobile charges to mesh points
solvePoisson	Solves Poisson system and computes electric field from potential in parallel
freePoisson	Frees memory resources associated with Poisson system after simulation is complete
savePoisson	Saves charge density, electric field, and potential data to files

which can be found from V through the relationship

$$\mathbf{E} = -\nabla_{\mathbf{r}} V \tag{4.14}$$

after the potential is calculated.

In the current version of the solver, external faces can be either Dirichlet and Neumann boundary conditions. Dirichlet boundary conditions specify the value of the potential on the face, while Neumann boundary conditions specify the electric field value on the face. A rectangular mesh is chosen to cover the region of the device for which the electric field is required. The mesh is a rectangular lattice of real-space points for which the equation will be approximately solved. While this lattice is different from the crystal lattice, each mesh point lies in the center of a rectangular mesh cell, defined exactly as the Wigner-Seitz cell is for crystal lattices. Different simulations may have different mesh cell dimensions, as well as different numbers of mesh points N in each direction. The basic appearance of this mesh used in all the device simulations is depicted in Figure 4.6, with

$(N_x, N_y, N_z) = (5, 10, 5)$. Rows and columns with the index 0 or $N + 1$ are used to store the

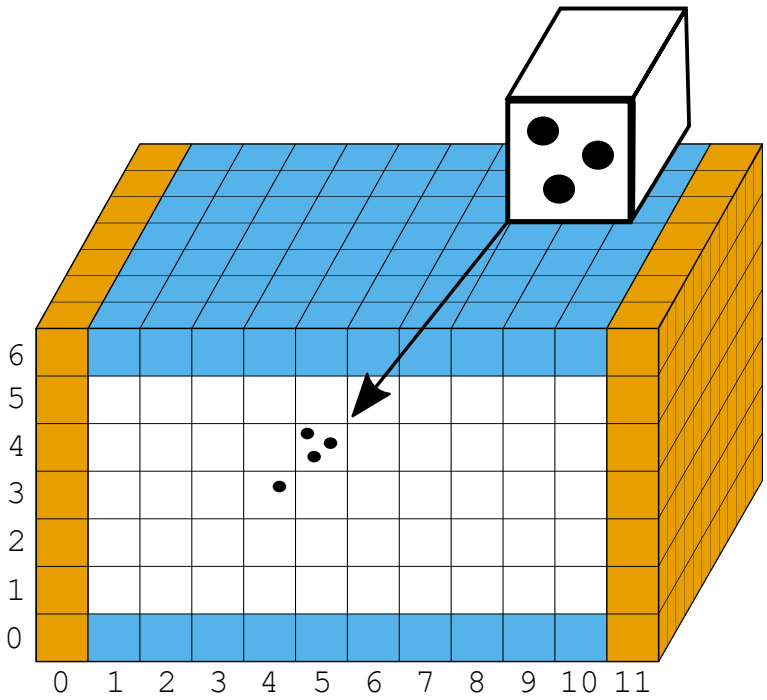


Figure 4.6: Example mesh, showing charge assignment algorithm

values associated with boundary conditions. The mesh size and the time interval between solutions of the PE are related to the numerical stability of the solver. The time interval must be lower than the reciprocal of the plasma frequency ω_p given by

$$\omega_p = \sqrt{\frac{q^2 N_D}{\varepsilon m^*}}, \tag{4.15}$$

where q is the charge of an electron, N_D is the highest dopant concentration in the simulation, ε is the electric permittivity of the material, and m^* is the lowest encountered effective mass in the simulation [71]. The smallest mesh spacing Δx is constrained to be on the same order of mangitude as the Debye length λ_D given by

$$\lambda_D = \sqrt{\frac{\varepsilon_{Si} k_B T}{q^2 N_D}}, \tag{4.16}$$

where k_B is Boltzmann's constant, T is the population average temperature, and q and N_D are defined as in Equation 4.15. It is noted in [72] that a $\Delta x/\lambda_D$ ratio of 2.5 leads to stable MC simulations, while higher ratios greater than $4-6$ lead to runaway energy increases in the population. The simulation parameters in this work are chosen to satisfy both of these time and mesh constraints while maintaining a convenient and manageable number of mesh points. These cells are shown colored in Figure 4.6. Rectangular subregions of colored faces can individually be manipulated to adhere to different boundary conditions. In all device simulations conducted in this work, the device has "source" and "drain" regions with Dirichlet conditions specifying the applied voltage. For the diode, all other faces are given Neumann conditions of $\mathbf{E} = 0$, corresponding to the idea that no current density will cause charges to exit the device laterally. Transistor simulations have an additional "gate" region assigned Dirichlet boundary conditions.

The discrete form of 4.13 is

$$\frac{\delta_x^2 V_{ijk}}{\Delta x^2} + \frac{\delta_y^2 V_{ijk}}{\Delta y^2} + \frac{\delta_z^2 V_{ijk}}{\Delta z^2} = \frac{\rho ijk}{\varepsilon}. \tag{4.17}$$

In this notation, V_{ijk} refers to the approximated value of V solved at the coordinates corresponding to mesh point (i, j, k) and the δ_x^2 operator is the centered finite difference approximation of the second derivative with respect to x, given by

$$\delta_x^2 h_{ijk} = h_{i+1\,j\,k} - 2h_{ijk} + h_{i-1\,j\,k} \tag{4.18}$$

where h_{ijk} is a generic discrete function. The other operators δ_y^2 and δ_z^2 are defined similarly for their respective axes. Equation 4.17 can be solved for V_{ijk}, giving

$$
\begin{aligned}
V_{ijk} = \quad & \left(\frac{\Delta y^2 \Delta z^2}{\Delta^{-1}}\right)[V_{i+1jk} + V_{i-1jk}] \quad + \\
& \left(\frac{\Delta x^2 \Delta z^2}{\Delta^{-1}}\right)[V_{ij+1k} + V_{ij-1k}] \quad + \\
& \left(\frac{\Delta x^2 \Delta y^2}{\Delta^{-1}}\right)[V_{ijk+1} + V_{ijk-1}] \quad + \\
& \left(\frac{\Delta x^2 \Delta y^2 \Delta z^2}{\Delta^{-1}}\right)\left(\frac{\rho_{ijk}}{\varepsilon}\right)
\end{aligned}
\tag{4.19}
$$

where $\Delta = 2(\Delta y^2 \Delta z^2 + \Delta x^2 \Delta z^2 + \Delta x^2 \Delta y^2)$. The system to be solved contains an instance of Equation 4.19 for each internal mesh point. Boundary 0 and $N + 1$ mesh points are not given equations like the above, and are assigned special values depending on the associated face's prescribed boundary conditions. For Dirichlet conditions, the mesh points are simply set equal to the control voltage from the simulation input file. For Neumann conditions, the boundary cell value is set equal to the immediately neighboring internal cell in the direction normal to the boundary face. This assignment is the same as setting a low order finite difference derivative equal to zero at the boundary.

To solve the system, a discrete charge density vector ρ_{ijk} must be computed for Equation 4.17. Also, the boundary points must be assigned values so that they are available for use during the solution process. In doped regions, a uniform charge density corresponding to the dopant ions is assigned to each of the Poisson solver's mesh points. Then, during the *prepPoisson* function, the mobile charges are assigned to appropriate mesh points. While there are various strategies for assigning charge to mesh points, this work takes the simple approach of assigning any charge within the mesh cell to the corresponding mesh point with no interpolation, shown in the callout in Figure 4.6.

To start with, a potential equal to the anode potential is assigned to every mesh point. Iteratively, the relationship in Equation 4.19 is used to compute new discrete potential values. The new potential values are sometimes interpolated between the result of this calculation and the previous iteration's result, which is known as successive overrelaxation. The interpolation parameter can be altered to give the system faster convergence at the expense of numerical stability. Between iterations, the Neumann boundary conditions must be re-calculated. The iterations continue until the relative error is below some threshold of satisfactory accuracy, included as a simulation setting. Future solutions of Poisson's equation use the previous solution as a starting point, to increase speed performance. This is the same self-consistent electric field simulation technique presented in [71], updated to solve in three dimensions.

4.8 Simulation of Electrical Characteristics of an NiN Diode

Using the modules and procedures built up in preceding discussions, a device level simulation can be constructed. The device simulation makes use of the Poisson and MC modules, including the bulk and boundary scattering and Ohmic boundary condition features. A flowchart describing the coordination of these features in the device simulation is shown in Figure 4.7.

To verify the implementation of the device-level features of the MC tool, an n-i-n diode device was simulated. This device is intended to resemble the 2D MC study of GaAs and Si diodes presented by Tomizawa in [73]. Whereas Tomizawa's device was only simulated in two dimensions, this device is simulated in 3D. The n-i-n structure is also a useful building block for constructing other devices, which further motivates its selection

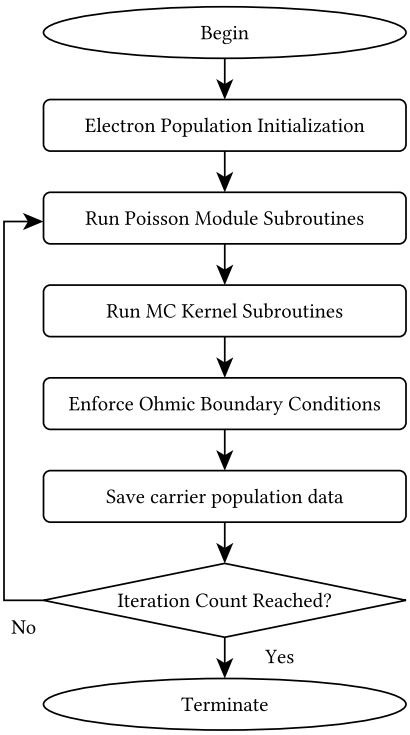


Figure 4.7: Basic electronic device simulation flowchart

as an initial device simulation test. The device, its dimensions, and its Poisson boundary conditions are depicted in Figure 4.8.

In Figure 4.8, yellow faces correspond to Dirichlet conditions and blue faces correspond to the no-flux Neumann condition. The n-type regions are doped to $2\times10^{17}\text{cm}^{-3}$. The structure is 550nm in length with a cross section of 500nm$\times$500nm. The doping regions extend 150nm into the device from the source and drain edges. The axis drawn by tracing from the source to the drain will be referred to in subsequent discussions as the "transport axis." The first and last interior Poisson mesh cells along the transport axis are used to implement the Ohmic boundary conditions described in Section 4.6. The lateral faces, in blue in Figure 4.8 use specular reflection for boundary scattering. The device is simulated under 750mV forward bias at 300K.

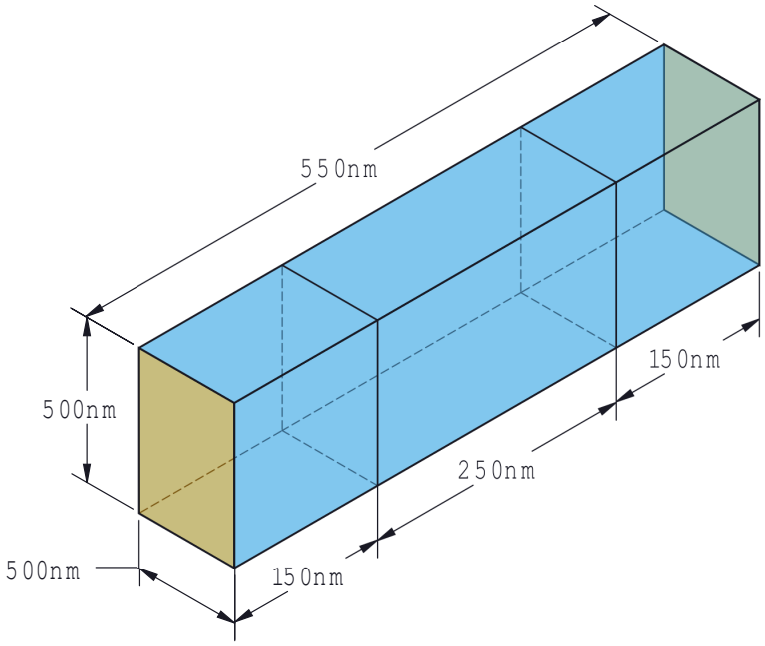


Figure 4.8: Dimensions, Dirichlet (yellow/orange), and Neumann (blue) boundary conditions of simulated n-i-n diode (not to scale)

Figure 4.9 shows the resulting simulated electric potential along the transport axis as compared to the Si diode simulated by Tomizawa. Note that the simulation solves for potential at discrete points, typically warranting a scatter plot instead of a line plot. However, the mesh resolution for potential in the transport axis varies between only 1*nm* or 2*nm* in all simulations, resulting in a quasi-continuous impression of the potential at the scales plotted. The potential remains fairly constant in the doped region and transitions from ground to the bias voltage through the intrinsic region. There is a slight downward dip before the potential begins to rise, at the n-i junction on the cathode side of the device. These features are all expected results by comparison to Tomizawa's work.

A theoretical examination of the expected graph can explain the downward dip. Consider Figure 4.10. In Figure 4.10a the separate band diagrams show the position of the Fermi level, conduction band edge, and valence band edge in each separate region of the

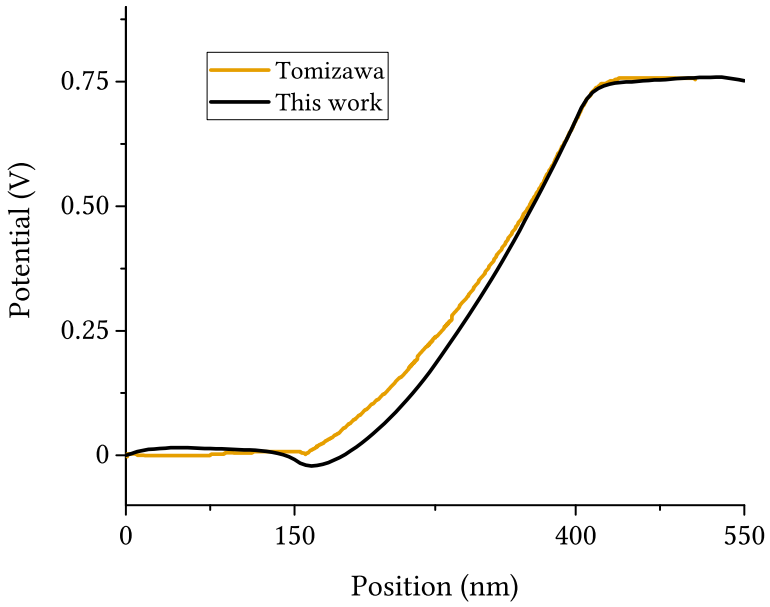
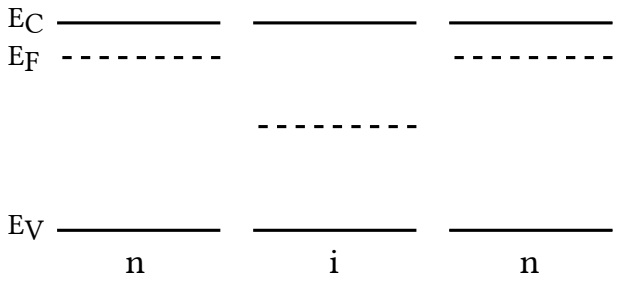
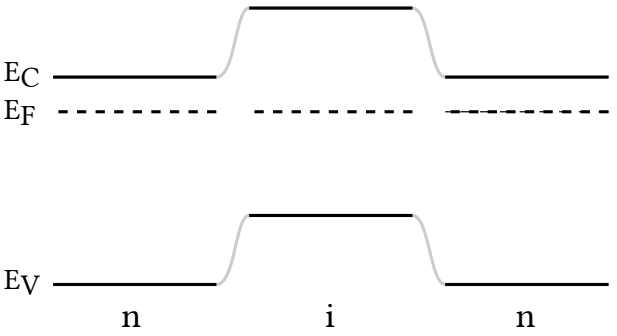


Figure 4.9: Potential comparison for simulated n-i-n junction diode

device. In equilibrium, the Fermi levels throughout the device will be constant, forcing the conduction and valence band profile to assume the shape seen in Figure 4.10b. Since electrons are negative, the upward potential energy "bump" corresponds to a barrier region of *lower* built-in potential. Therefore, plots of potential energy have the form of Figure 4.11, which shows the potential energy profile for the equilibrium device for two different applied drain voltages. At the n-i junction there is a negative, then positive, curvature in the potential due to the positive, then negative, charge densities formed at that junction. In the i-n junction, the opposite occurs with positive, then negative curvature. These patterns of curvature will persist in all bias points. Hence, when the drain potential is increased, the i-n junction curvature naturally fits the "low to high" trend, and a recognizable inflection feature is developed at the location of the n-i junction where the "low to high" trend runs against the required curvature pattern. The curvature and band-bending diagrams



(a) Separate band diagrams showing relative position of Fermi level and band edges



(b) Combined band diagram showing constant Fermi energy in equilibrium

Figure 4.10: Reference band diagrams for discussions on n-i-n devices

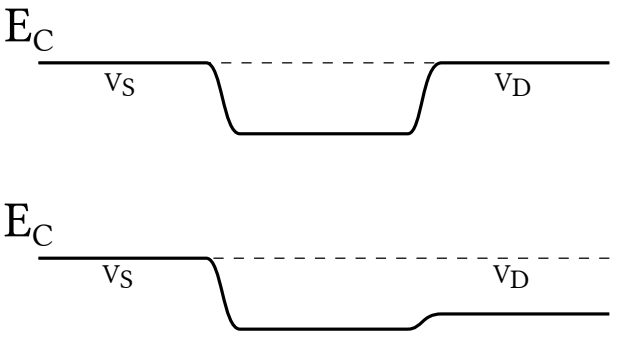


Figure 4.11: Plot of the ideal built-in potential of an n-i-n structure in equilibrium

presented here will appear again in descriptions of the FinFET simulation, as the same underlying n-i-n structure is used.

A further set of three simulations was conducted to test the framework's ability to simulate different band structures and, in particular, test the stability of the full band code. The results can be see in the plot of carrier velocity distribution, separated by band structure, in Figure 4.12. Due to the relatively low energies encountered, there is not a very

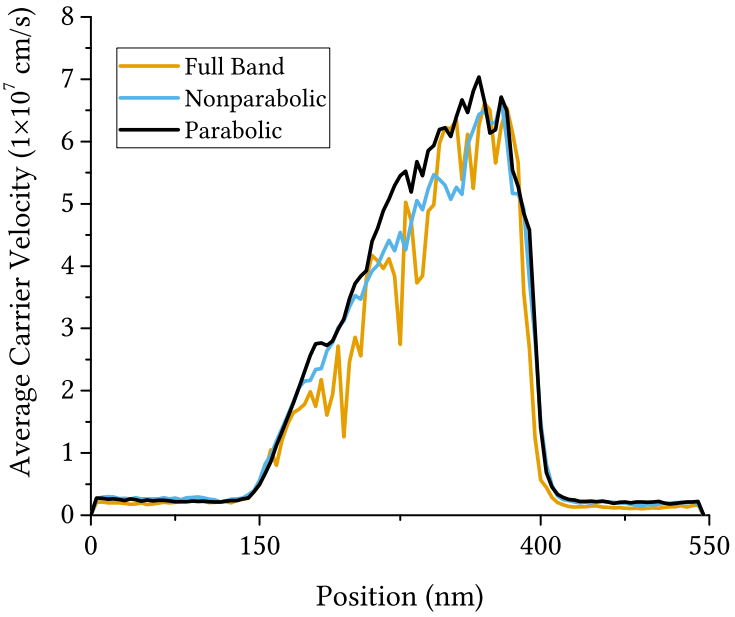


Figure 4.12: Comparison of simulated velocity profile for different bandstructures

large difference in the velocity distribution. Higher energies are expected to have slightly lower velocities in the full band simulation, due to the shallower slope of the density of states. The nonparabolic velocities should follow more closely with the full band velocities, while the parabolic simulation should show velocities consistently higher than both. The emerging trend in the plot appears to follow roughly this expectation.

These figures demonstrate the correctness of the implemented MC and Poisson solvers through their similarity to Tomizawa's device simulations, and also show the power of the MC simulation technique itself. With this verified PE and MC framework, statistics

on carrier populations as a function of location within devices can be computed easily, even in scenarios with more complex geometry or in materials with more complicated band structures or scattering phenomena.

4.9 Quantum Confinement of Charge Carriers

The next simulation requires special handling of the electron population. The MC module needs assistance in simulations where the size of the simulation domain approaches the same order of magnitude as the average electron wavelength. This scenario, called quantum confinement, breaks the assumption of slowly varying potential that is critical to viewing electrons as semi-classical objects. In potential wells of such small sizes, it no longer makes sense to speak of the position of a particular electron in the cross section. Conceptually, the simulation domain is split into semi-classical transport directions and confinement directions. In the later case of a FinFET, there are two dimensions of confinement and one dimension remains semi-classical. The four lateral sidewalls of the fin channel are assumed to be impassible potential barriers. Electrons that are in the fin have wavefunctions that do not penetrate into the gate oxide or deeper into the substrate. This scenario imposes vanishing Dirichlet boundary conditions in the confinement direction on the solutions to the SE. The semi-classical picture of an electron partially localized in a Gaussian wavepacket violates these boundary conditions. Instead, the 2D SE with these boundary conditions must be satisfied at every position in the fin along the transport axis. The boundary conditions cause these solutions to have an array of orbital-like shapes corresponding to discrete wavelengths and energies. Harsher confinement results in wider intervals between discrete energy eigenstates. However, *along the transport axis*, electron

wavepackets can still maintain their semi-classical wavepacket profiles and transport properties with quasi-continuous energy spectra. While the discrete energy eigenstates replace the kinetic energy previously calculated in the confinement plane, the kinetic energy component associated with the transport axis can still take on continuous values that follow the normal dispersion relation. The energies are added together, and can span a new array of values as a new band structure. The new band structure looks like the 1D cross-section of the previous 3D band structure, duplicated and shifted by the discrete energies. The resultant energy situation is given the name "sub-bands" and the discrete energies are called "sub-band minima."

The state-of-the-art methods for handling these quantum confinement scenarios are the MSBMC approach described below and approaches utilizing "quantum corrections" to the ordinary 3D potential, such as those found in [74, 75, 76]. The MSBMC directly addresses the decomposition of the Hamiltonian, with the general procedure for solving confined "slices" introduced [77]. The method has appeared in many studies, specifically involving FinFET and NWT simulations [78, 79, 80, 81, 82].

The MSBMC algorithm can be seen in Figure 4.13. The main loop of the previous setup, where the MC module and Poisson modules are run alternately, is modified to include the Quantum module. The Quantum module provides functions that support calculating solutions to the SE eigenproblem for each "slice" of the fin along the transport axis. Table 4.4 summarizes the Quantum module functions. In the first step, the Quantum module must run first to calculate the subband structure. For convenience, the transport axis slices are defined by the Y-axis of the Poisson mesh. The 2D mesh used to solve the SE is simply the cross section at a given Y-axis mesh position. The SE is solved at every slice

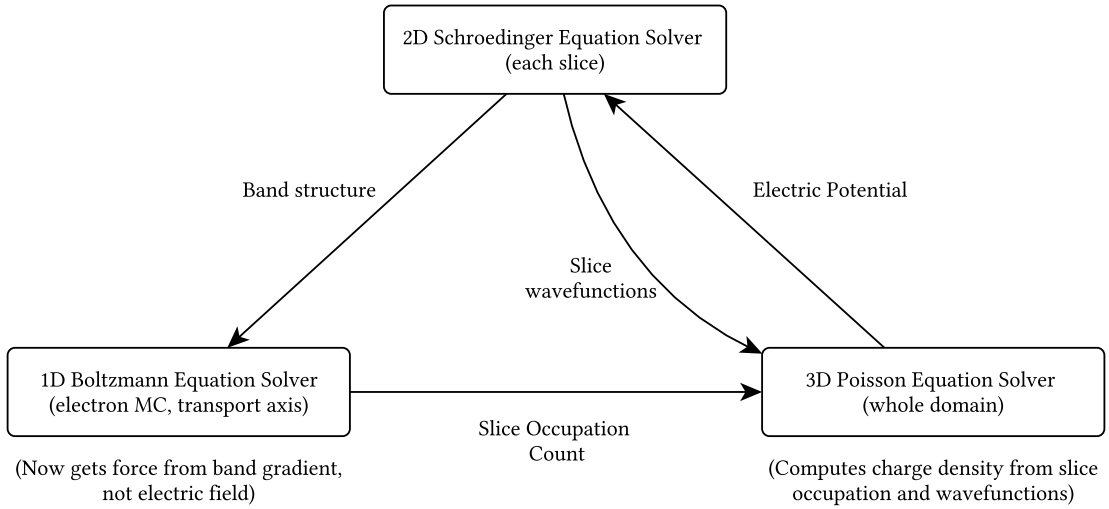


Figure 4.13: MSBMC flowchart

with the vanishing wavefunction boundary condition. For each slice, the Hamiltonian is constructed using a finite difference approach, similar to the Poisson solver in Section 4.7, and a potential energy term taken from the output of the Poisson module. The numerical library Eigen is used to find the eigenfunctions and eigenvalues of the Hamiltonian [83]. The eigenvalues are sorted and saved for each slice in a 2D array that can be accessed by the MC module. The indices into the 2D array are subband (eigenvalue) index and Y-axis mesh position. The eigenfunctions are used to compute the particle probability densities in the plane of each slice. The densities are made available to the Poisson solver for later charge distribution calculations.

In the second step, the MC solver is run. The solver is adjusted when running in this mode, such that the applied field for calculating particle dynamics is calculated according to

$$F_y = \frac{dE_n(y)}{dy}, \tag{4.20}$$

Table 4.4: Quantum module functions

Function Name	Description
setupQuantum	Called during initialization, allocates memory for subbands and eigenstate probability densities
solveSchroedinger	Solves the SE for each Poisson mesh slice along the transport axis, sorts and saves eigenvalues and eigenstate magnitudes for each slice
projectCarrier	Turns 3D electrons into confined electrons by assigning it to a subband with energy corresponding to the carrier XZ wavevector components
freeQuantum	Frees memory resources associated with Poisson system after simulation is complete

where $E_n(y)$ is the aforementioned subband energy array, n is the subband index, and y is a position in the transport axis. The electric potential from the Poisson solver does not need to be incorporated at this point as another force, as it is included in the Hamiltonian for the SE calculation. Carrier trajectories are only calculated in the Y-axis, and positions and momenta for the confinement directions are not tracked.

When a carrier object is made by the MC Kernel functions, it is a 3D carrier with all positions and momenta. The function *projectCarrier* calculates the kinetic energy that would be associated with the X- and Z- axes, and picks a subband such that the total energy of the carrier is the same. Once the carrier subband is assigned in this way, the *updateMomentum* function in the MC module can access the proper subband for computing the derivative in Equation 4.20.

In the third step, the Poisson module calculates a new potential. At this point MC module has computed new positions for each carrier, and therefore the carriers' distributions among the slices may have changed. The total charge in a slice is combined with the electron density solutions from the Quantum module to compute an expected charge distribution for each slice. The charge distributions and the standard Poisson boundary conditions are all taken together in the same 3D PE solver used in the previous simulation. The 3D PE thereby self-consistently couples the 1D MC solution to the array of 2D SE solutions.

The process repeats, with other features such as the Ohmic boundary conditions, in the modified device simulation loop described in Figure 4.14.

4.10 Simulation of FinFET IV and Confinement Characteristics

The MSBMC module was tested in a simulation of a nanoscale multigate field-effect transistor (FET) device with an intrinsic channel. The basic material structure of the device is n-i-n, like the diode simulations, so that this study can build on the previous. The cross section of the device was chosen so that there was a noticeable separation of subbands due to quantum confinement, but the device was not so small that a thermal simulation of the same device would no longer be possible with the bulk phonon dispersion. The dimensions were also selected so that the device had an aspect ratio more similar to that of a typical FinFET than of a NWT, and is therefore identified in this work as a FinFET. The fin is 2 0nm wide by 30 nm tall, compared to the IBM 14 nm FinFET process with 10 nm width and 25 nm height [84]. This IBM process was first presented in 2014, and commercial processors utilizing the process were first sold in mid-2017. The simulated FinFET also

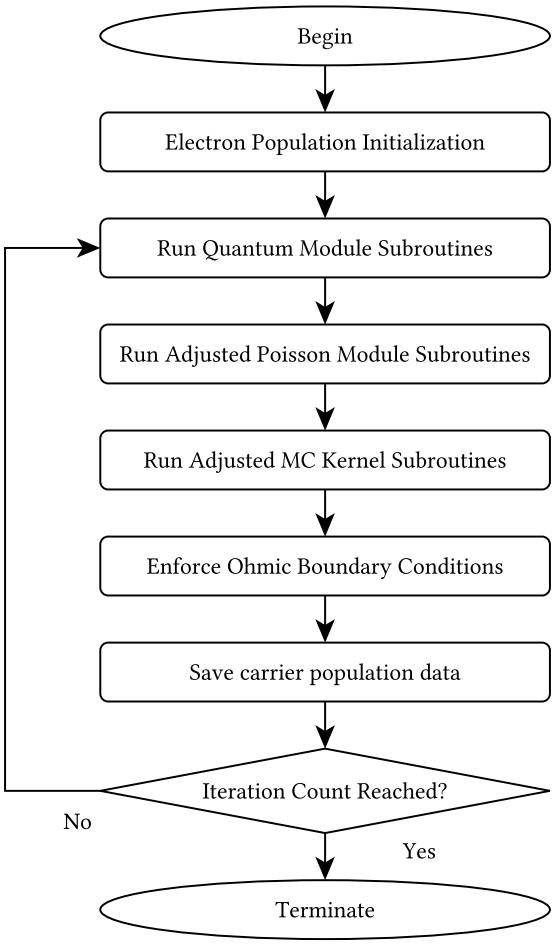


Figure 4.14: Electronic device simulation flowchart with Quantum module enabled

has a similar channel length at 40 nm compared to the 30 nm channel in the cited IBM process. Thus, this device is only about twice as wide as a recent, representative transistor technology.

The electrostatic boundary conditions of the simulation were upgraded to allow control through a gate electrode and can be adjusted to support multiple gate shapes. This change in boundary conditions, as well as the quantum confinement, are the primary differences between the FinFET simulations and the n-i-n diode simulations. Available

types are planar gate, Π gate, and gate all around (GAA) as pictured in Figure 4.15 and Figure 4.16. The Π-gate style was selected for this device, as it is a FinFET.

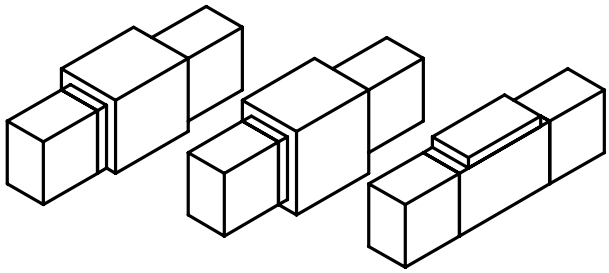


Figure 4.15: 3D view of the three possible gate shapes: (left to right) Π-Gate, GAA, and Planar MOSFET

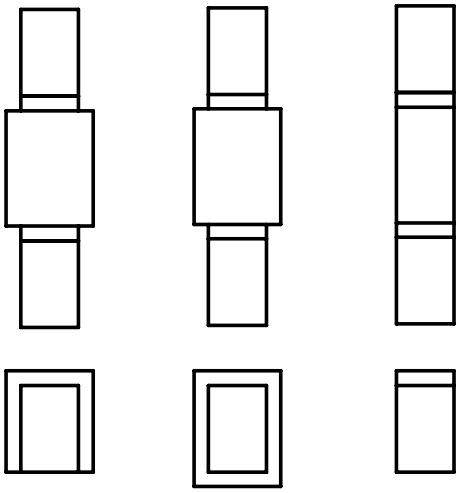


Figure 4.16: Top and front view of the three possible gate shapes: (left to right) Π-Gate, GAA, and Planar MOSFET

A summary of the device dimensions and Poisson boundary conditions is found in Figure 4.17. It should be noted that the "gate voltage" applied to this device is directly

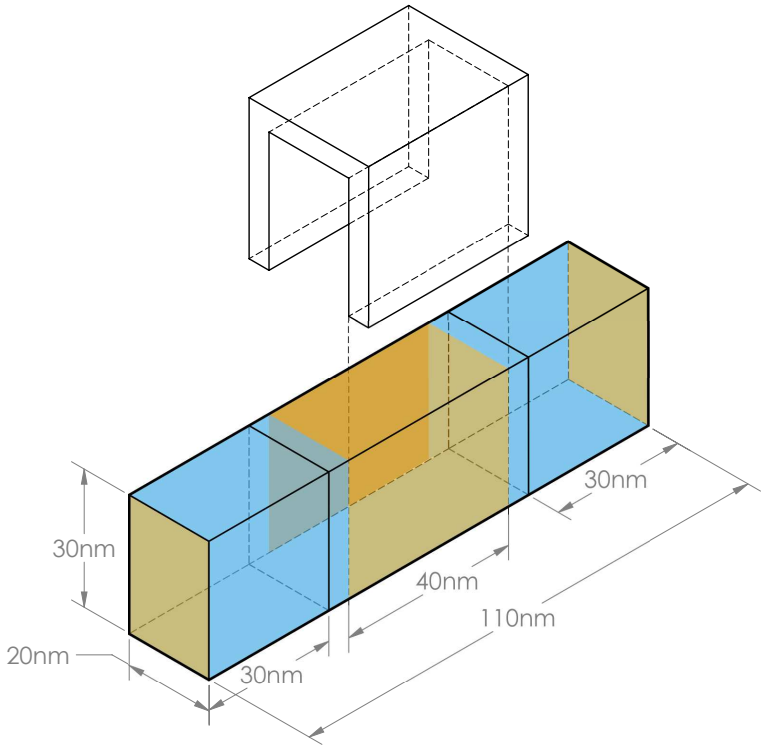


Figure 4.17: Dimensions, Dirichlet (yellow/orange), and Neumann (blue) boundary conditions for the simulated FinFET (not to scale)

imposed at the oxide surface. This is an approximation of a real device, where there will be some loss in potential across the gate oxide, as well as a drifting or sagging effect in the interface potential as the channel is traversed towards the drain. The approximation serves to over-estimate the electrostatic control the gate voltage will have over the channel. The direct application of the gate voltage to the surface should also be considered when examining the IV curves and potential profiles presented later.

The doping level was chosen to provide a high enough density of electrons in the Ohmic boundary layer of the simulation for the boundary neutrality condition to be

maintained accurately by the code. The above parameters are typical of these types of devices, with many similar devices simulated in other works [85, 86, 87, 88]. In particular, this device is similar to the device treated analytically in [87] in terms of dimensions, gate geometry, and doping concentration.

The goal of this study is to further verify the simulator's accuracy in qualitative and quantitative terms in these situations which will show up in the final, electrothermal case. In particular, key observations that must be made are:

- the electrostatic potential throughout the device resembles the classic FET device profile
- a recognizable "transistor-like" IV plot can be obtained showing distinct regions of triode and saturation behavior
- the currents and internal potentials plotted are within reasonable ranges for similar devices
- the subband structure has the appropriate shape and varies correctly with confinement parameters
- the charge density must vanish at the fin boundaries

To begin, the simulated electrostatic potential is examined. Refer to Figure 4.18 which shows the familiar n-i-n band diagram. Theoretically, an applied gate voltage will adjust the Fermi level in the intrinsic region. This phenomenon is called the field-effect, from which FET derives its name. Thermal populations of electrons reside in the source and drain wells, separated by the potential barrier. Some high energy electrons in both wells escape, in about equal number, and travel to the other well. This process is named

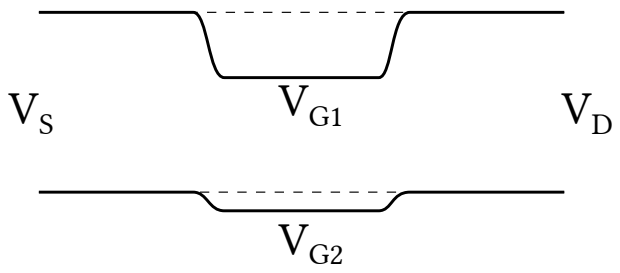


Figure 4.18: Band diagram showing a field-effect device in off (above) and on (below) states

thermionic emission, and occurs exactly in this scenario as in early vacuum tubes. The result is a net zero electron flux.

Raising the Fermi level, or lowering the conduction band edge, allows more of the high energy electrons in both wells to escape. The increase in carrier occupation in the channel region is related to the conception of a channel forming in classical MOSFET analysis. When a bias is applied, by increasing the drain potential, there will be an imbalance in thermionic current. The net result is a nonzero electron flux. Additional features of the device behavior may occur due to other fields or potentials present in different devices due to their construction, while this process is what gives both the vacuum tube triode and the MOSFET many of their recognizable IV characteristics.

In some cases, due to short channels, FET devices may see an additional barrier lowering due to the tugging-down of the channel potential by the drain boundary condition. Figure 4.19 demonstrates several of these concepts. One can observe the phenomenon of drain-induced barrier lowering (DIBL) as well as the general shape that the potential profile should have under non-equilibrium conditions. As previously mentioned in the n-i-n diode discussion, the n-i junction curvature results in a kink or blip. This feature of

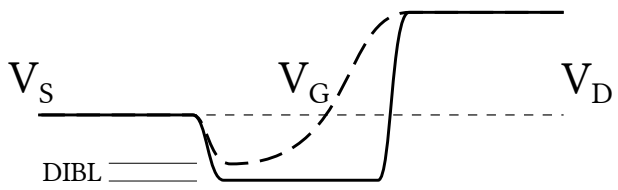
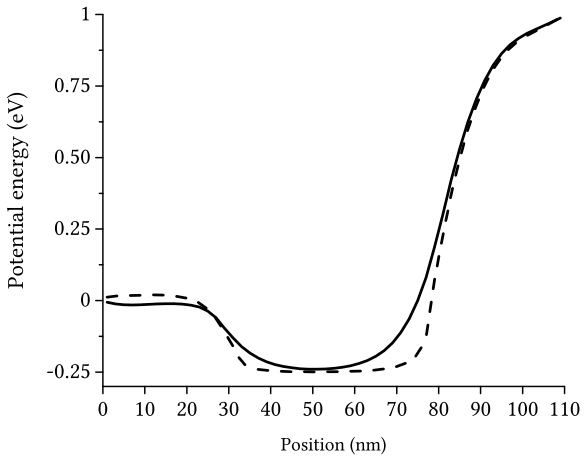


Figure 4.19: Schematic depiction of the DIBL phenomenon

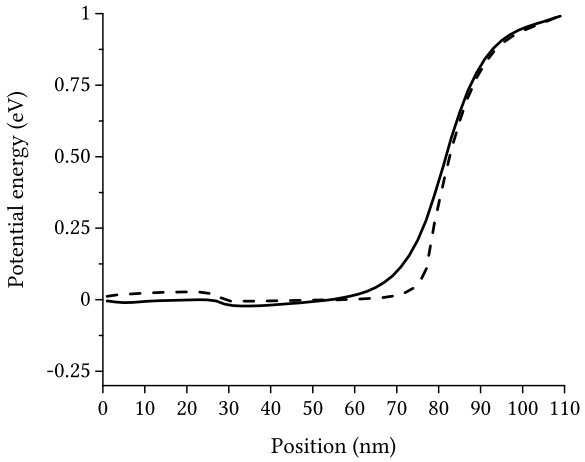
the graph is the barrier associated with thermionic emission of carriers into the channel. It can also be seen that a thermal population of carriers in the drain well have very little chance of climbing to the height of the n-i junction's thermionic barrier.

Turning to the results of the FinFET simulation, the above behavior is clearly mimicked by the simulated device in Figure 4.20. The figure shows the potential in each slice, averaged among the 2D slice mesh cells, through the device as a solid line. The surface potential, averaged among the upper surface mesh cells, is shown as a dotted line.

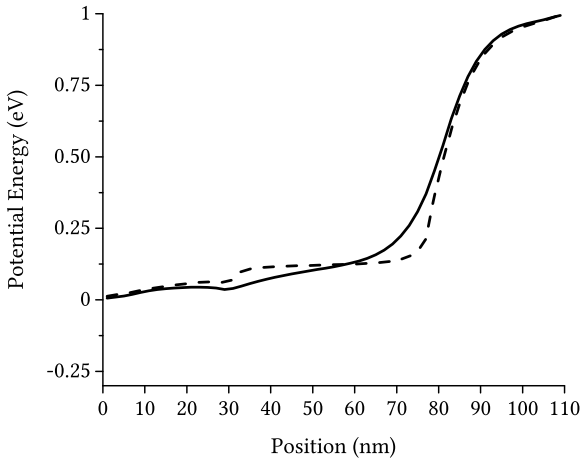
In the simulated device, the sagging of the potential as it approaches the drain well is still evident despite not simulating the oxide field and potential. The gating of the channel by the barrier height for the applied surface potential ϕ_s is demonstrated, as well. Notably, there is a sagging of the potential in the source well ahead of the n-i junction. This does not occur in many depictions of a MOSFET band diagram. In this case, it is due to the lack of electrostatic control over the source region due to the selected boundary conditions: the potential is only enforced at the end faces of the domain. In a planar MOSFET device, the well potential is better control, since the source electrode spreads over the surface up to a point very near the gate oxide. This may result in a contribution to the device current that results from ordinary field acceleration of carriers in the source, on top of the thermionic current. The potential profile is otherwise identical to the expected form.



(a) $\phi_s = -250$ mV, $V_{DS} = 1000$ mV



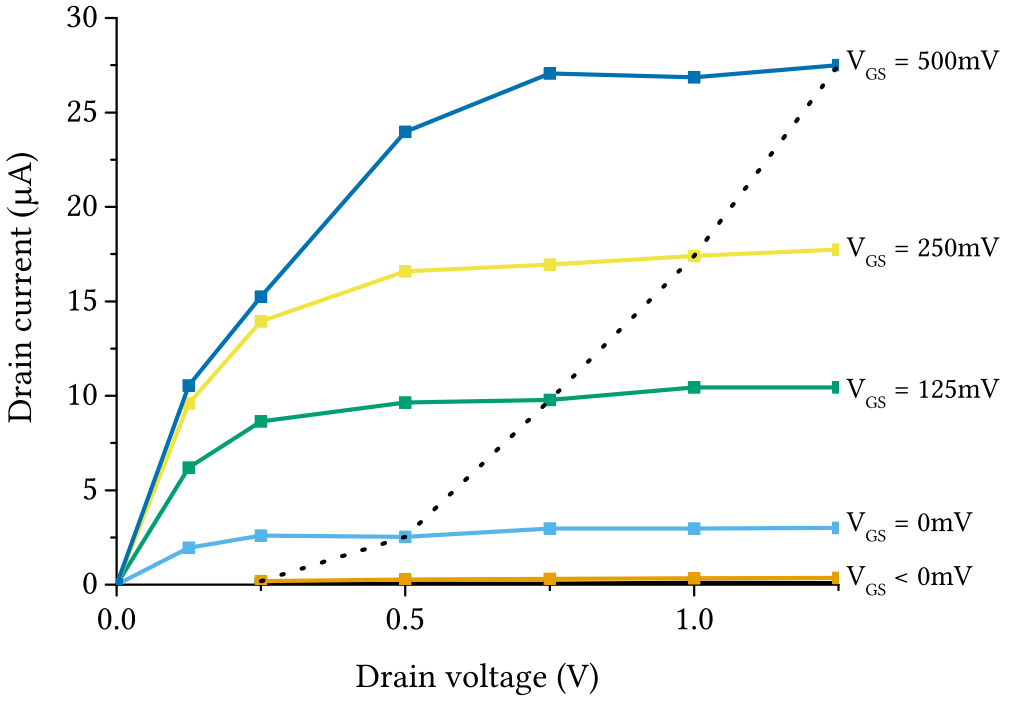
(b) $\phi_s = 0$ mV, $V_{DS} = 1000$ mV



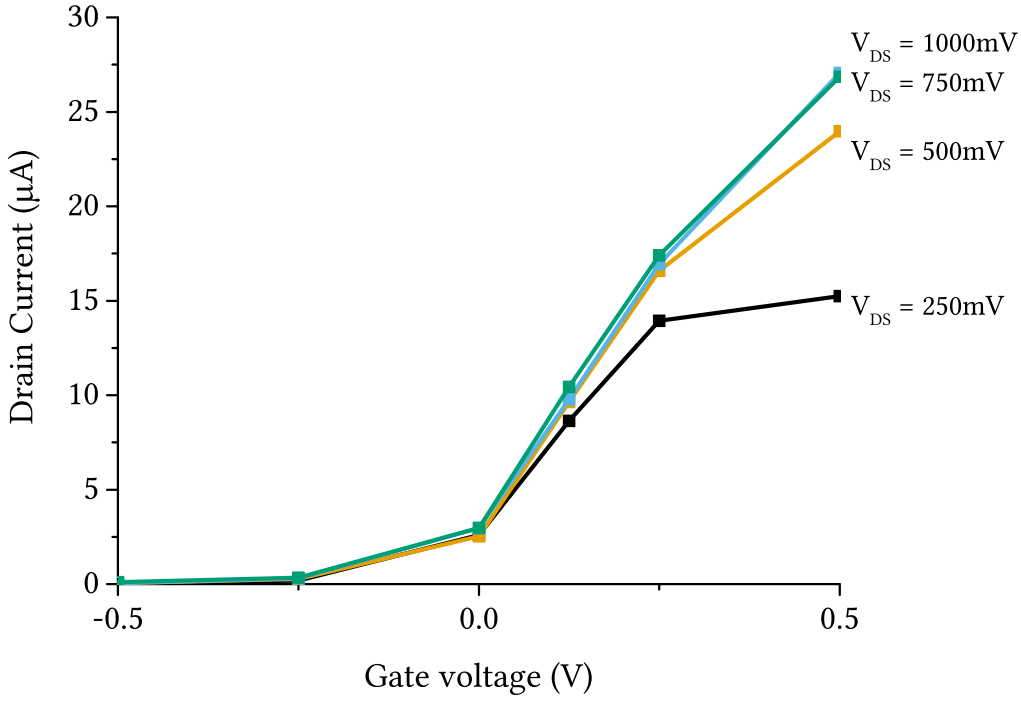
(c) $\phi_s = 125$ mV, $V_{DS} = 1000$ mV

Figure 4.20: Simulated potential profiles through FinFET device

The IV characteristic is plotted in Figure 4.21. The transistor-like quality of these figures is evident immediately. There are two clearly defined regions of operation. In Figure 4.21a the dotted line separates the triode region (to the left) and the saturation region (to the right). Thresholding behavior can be observed in Figure 4.21b, where the current appears to "turn on" suddenly near a gate voltage corresponding to $\phi_s = 0$. The currents for high bias points reach into the 20 μA to 30 μA region, which is very close to the values of 30 μA to 50 μA given for the similar device in [87].



(a) Drain current vs. drain-source voltage



(b) Drain current vs. gate-source voltage

Figure 4.21: IV characteristics for the simulated FinFET device

In Figure 4.22, the subband structure of the simulated device is plotted against position. The conduction minima are evenly spaced and the band edge follows the potential

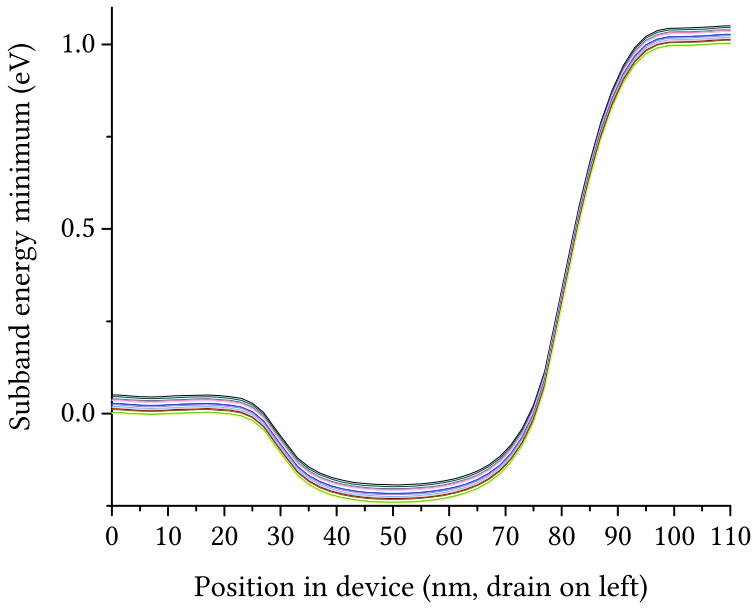


Figure 4.22: Full subband diagram of simulated FinFET

profile closely. This is expected due to the way the electrostatic potential is incorporated into the slice Hamiltonians. A clearer picture of the simulated subband separation is shown in Figure 4.23. The device was re-simulated with the larger 50nm fin height to verify the relationship between average subband separation and level of confinement. The trend is the correct one, with more confinement causing larger separation intervals.

Finally, the spatial extent of the confinement wavefunction is examined in Figure 4.24. Consistent with expectations, the electrons are spatially distributed according to the orbital-like patterns typical of solutions to the SE. Interestingly, the higher energy drain population appears to occupy states on average that have lower wavelengths. This is also

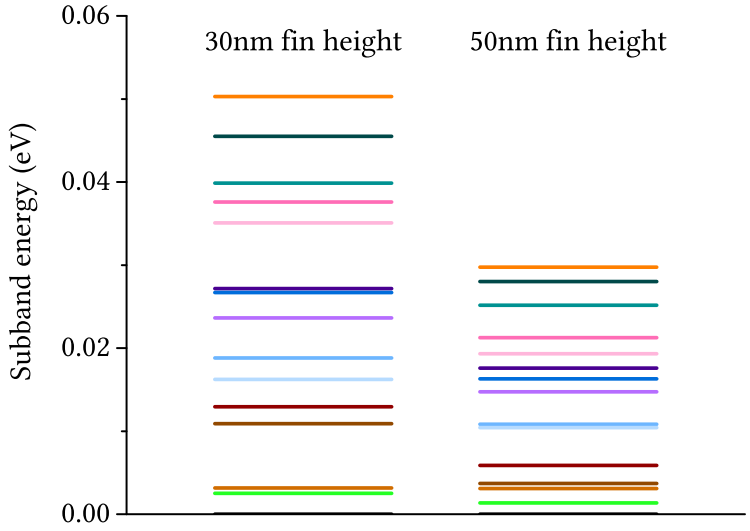


Figure 4.23: Simulated subband separation at different levels of confinement

consistent, as those higher energy electrons will have higher wavevector magnitudes and therefore lower wavelengths.

The FinFET device shows strong positive characteristics, indicating that the electron simulation capabilities of the MC framework is sufficient for its use in the Chapter 6 simulations. Simulations were run until the average device current and the average potential were stable at less than 1% relative error between iterations. The electronic device simulations generally consume between 100 to 500 core hours, depending on input parameters. The IV extraction, consisting of many independent bias point simulations, consumes 5000 to 8000 core hours depending on the number of points included in the plot to form a recognizable IV curve.

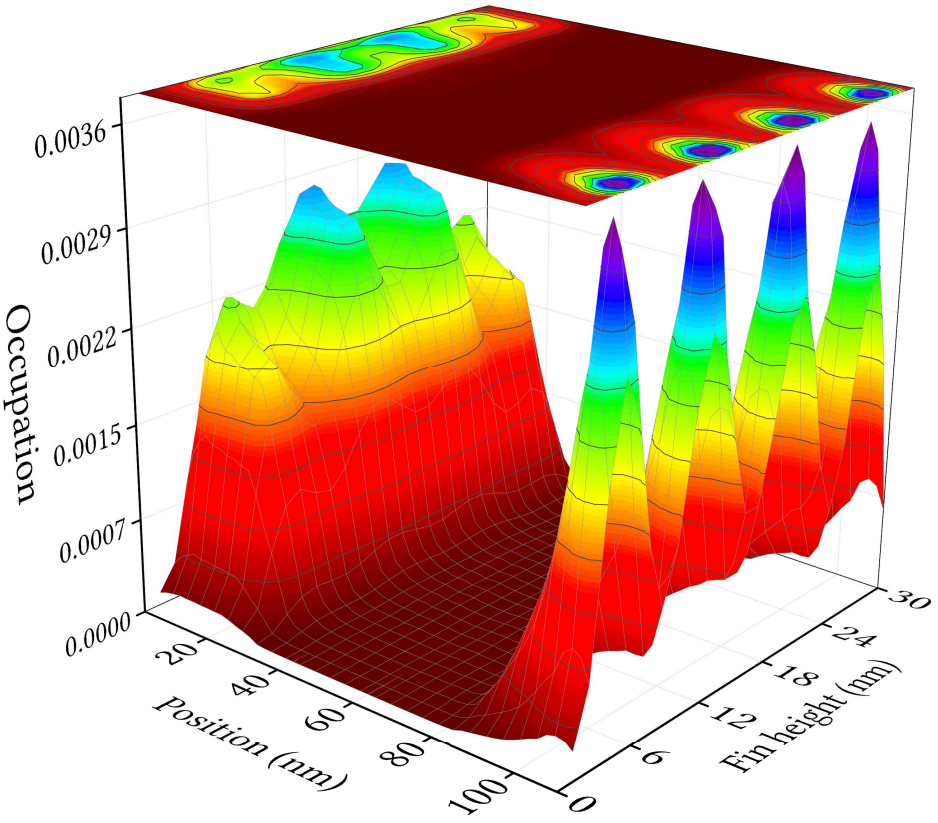


Figure 4.24: Simulated charge density

CHAPTER 5

PHONON MONTE CARLO IN SILICON

This chapter describes the MC method for phonons in cases where the approach differs from the technique discussed so far. The bosonic nature of phonons, their dispersion relation, and the different scattering processes distinguish phonon MC simulation from electron MC simulation. The differences are noted, and several studies are presented to demonstrate the verification of the MC phonon simulator produced in this work.

The inability of Fourier's law to handle simulations in nanodevices is specifically noted as a motivation for phonon-based analysis methods in [89], which states that temperature errors as high as ±10% can be observed between phonon- and Fourier-based analysis, which is ±30 K for devices operating at 300 K. To simulate phonon-based heat transfer processes, researchers began to adopt the MC method used to simulate electrons. An MC scheme for solving the phonon BTE was developed in several works in the early-to-mid 2000s [90, 91]. These MC methods are described as holding the advantage over other approaches for pratical, engineering applications due to their time efficiency [66].

Phonon-based transport becomes important when the length scales of the device approach the mean free path of the material being analyzed. In this case, phonons may carry energy to different parts of the device with minimal scattering. This phenomenon is called ballistic conduction, and is not modeled by Fourier's law. The phonon mean

free path in Si is approximately 300 nm in the undoped case [92], which is above the the majority of device dimensions used for the simulations in this work. Additionally, the length scales discussed typically correspond to devices with high surface area to volume ratios. Such ratios imply an increase in the amount of boundary surface scattering that compound the previously mentioned decrease in bulk scattering. For example, thermal conductivity of thin-film devices and related nanostructures can be much lower than in bulk Si specifically due to boundary scattering effects [13, 93]. In fact, the conductivity can be reduced by an order of magnitude in nanostructures with dimensions as low as 35nm [14]. The exact impact of these effects is strongly dependent on device geometry [13]. The phonon MC technique is utilized in the study of phononic membranes specifically due to their suitability to geometrically complex devices [94].

Advanced phonon MC techniques exist for various cases. Some phonon MC methods utilize a perturbation approach, simulating only deviations from the equilibrium population [95]. This technique can lead to improved convergence times for observing low-magnitude departures from equilibrium systems. Phonon MC simulations of extremely small devices may require special treatment of the dispersion relation to support flexural and torsional modes. This special dispersion handling is analogous to the use of the MSBMC method for electrons, being used when the phonons experience confinement in potential wells similar in size to their wavelengths. Relevant phonon wavelengths for transport in Si are around 5 nm to 10 nm [14]. This effect can be ignored in devices above 10 nm to 30 nm [66], placing several of this work's simulations near that point. However, according to [96], phonon wavelengths below 10 nm contribute to 90% of the heat transfer,

which specifically highlights that confinement effects need not be treated until below such lengths.

A key difference between the phonon and electron studies in this work is the lack of a single particle phonon simulation mode. As for electrons, phonon scattering rates depend on the spatially varying temperature. However, phonon energy density is actually used to compute the spatially varying temperature profile. Thus, a device temperature mesh must be constructed and used even for bulk simulations. Bulk simulations can still be achieved using special boundary conditions or large domain sizes.

5.1 Initialization of Boson Populations

The initialization of the bosonic phonon populations is conducted similarly to the electron initialization. The major difference is the use of the phonon dispersion and the use of Bose-Einstein statistics as opposed to Fermi-Dirac statistics. The boson energies must follow the Bose distribution:

$$f_{BE} = \frac{1}{e^{\frac{E}{k_B T}} - 1}, \tag{5.1}$$

which differs only from the Fermi-Dirac case in the sign of the 1 in the denominator.

Like the Fermi distribution, there is no analytic inverse for the associated PDF and CDF. This work handles generation of phonon energies through a technique like inverse transform sampling, but using a tabular form of the quantile function. The phonon dispersion relation is evenly discretized on the energy or frequency axis. The phonon density of states is used with the thermal mesh cell volume and temperature to compute the expected number of phonons in an energy interval. The result is a table of phonon number densities indexed by energy interval index. The phonon number density N for a

frequency bin of width $\Delta\omega$ centered on ω_n is calculated with

$$N = f_{BE}(\hbar\omega)\,\Delta\omega\,g(\hbar\omega)\,\Omega, \tag{5.2}$$

where g is the phonon density of states and Ω is the cell volume. The function g can be computed from the dispersion relation, with care taken to account for the longitudinal and degenerate transverse branches. The CDF of this is easily calculated with a loop, and is used to generate correctly distributed energies given uniform random numbers. During the generation of the phonon number density table, the relative proportion of longitudinal to transverse phonon count is saved in a separate table using the same indexing. This table is used after energy selection with another random number to select the appropriate polarization state. The resulting population of phonons is has physically realstic energy and polarization state distributions, up to discretization error.

Additional initialization parameters include particle location and velocity. Carrier location utilizes uniform random numbers to place phonons evenly within the temperature mesh cell. Carrier velocity components v_x, v_y, and v_z are computed using the following formulas:

$$v_y = V_G\,\cos(\theta) \tag{5.3a}$$

$$v_z = V_G\,\sin(\theta)\,\cos(\phi) \tag{5.3b}$$

$$v_x = V_G\,\sin(\theta)\,\sin(\phi) \tag{5.3c}$$

In the above, V_G is the group velocity and the terms $cos(\theta)$, $sin(\theta)$, and ϕ are computed using the random numbers r_1 and r_2 as in

$$\cos(\theta) = 2r_1 - 1 \tag{5.4a}$$

$$\sin(\theta) = \sqrt{1 - \cos^2(\theta)} \tag{5.4b}$$

$$\phi = 2\pi r_2 \tag{5.4c}$$

Much of this initialization procedure, including the energy and velocity selection is informed by the work done by Mazumder and Majumdar [90]. This particle frequency binning is also described in [66].

5.2 The Silicon Phonon Dispersion Relation

The dispersion relation of phonons in Si is far less complex than the bandstructure. In this work, the dispersion relation is approximated as isotropic. The relationships between energy, wavevector magnitude, and group velocity magnitude are represented with band tables. The bands are loaded with the piecewise linear curve shown in Figure 5.1. The

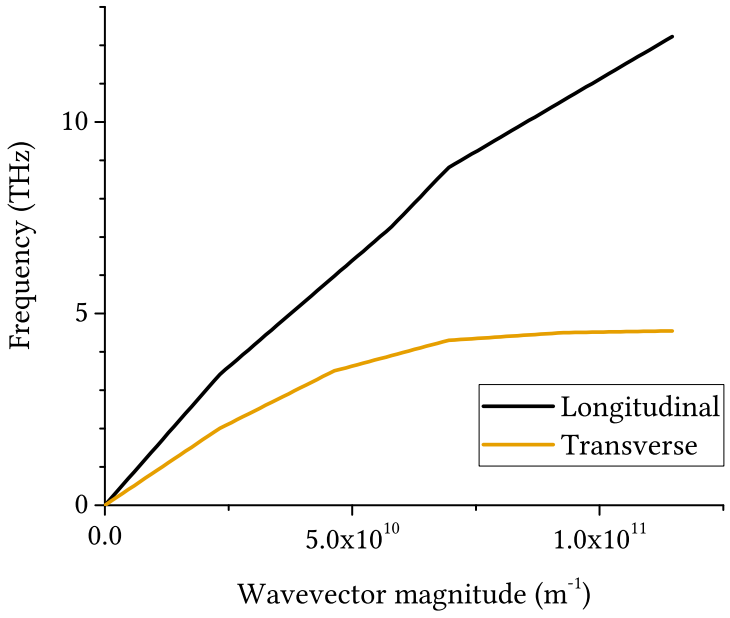


Figure 5.1: Piecewise linear interpolation of Si acoustic phonon dispersion

values for the Si phonon dispersion table were sourced from [97]. Looking at the dispersion

relation, it can be seen that there are different longitudinal and transverse velocities. Also, no transverse phonon states exist above a critical energy. Furthermore, lower energy phonons actually have higher velocities. The inclusion of these features is therefore taken into account by the MC Kernel through their inclusion in the band table.

The optical phonon branches are not depicted in Figure 5.1. The optical phonons in this work are simulated using constant, isotropic energy corresponding to the approximation of the phonon branches as essentially flat.

5.3 Scattering of Phonons in Silicon

The scattering rate formulas for phonons are computed using FGR and perturbing potentials, just as they are for electrons. As mentioned previously, scattering of phonons can occur due to the neglected anharmonic components of the interatomic potential. Scattering events derived from those perturbing potentials are the phonon-phonon scattering events. Phonon-phonon scattering events can a be normal process (N-process) or Umklapp process (U-process) scattering. In both N-processes and U-processes, two phonons scatter against each other resulting in altered wavevectors for both particles. If the resultant wavevectors lie within the 1st Brillouin Zone, then this event is labeled a N-process. Momentum is conserved in such cases. If the resultant wavevectors lie outside the 1st Brillouin Zone, then the wavevector, and momentum, are “folded” back into the zone from the other side, causing the phonon to appear to have reversed direction.

The folding is show schematically in 5.2. The dashed gray lines indicate the

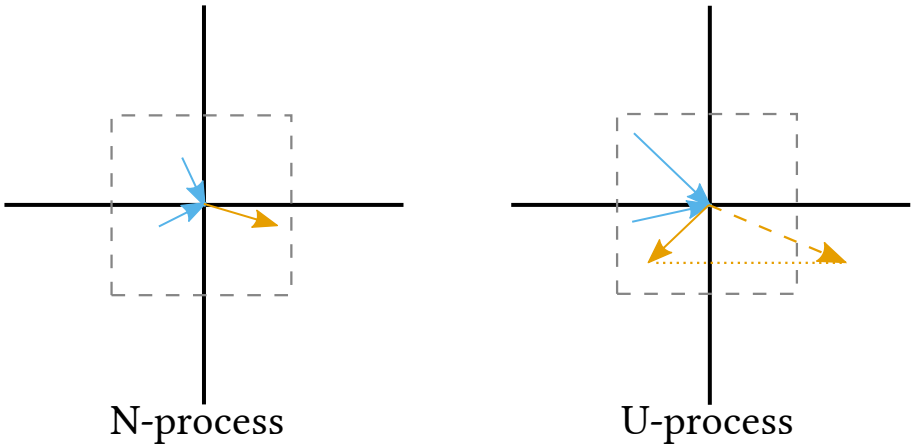


Figure 5.2: Comparison of N- and U-processes

boundary of the 1st Brillouin Zone in a simplified crystal. The blue arrows represent the momenta of phonons before collision. The lefthand scenario depicts a resultant momentum that is within the 1st Brillouin Zone. The righthand scenario depicts a resultant momentum that is outside the Zone, which is aliased to a location within the zone. Events with this folding action are labeled U-processes. The folding occurs due to the discrete lattice restricting the possible unique values of wavelength. This process is identical to frequency aliasing when digitally sampling an analog electronic signal.

Momentum is not generally conserved in U-processes. In this work, several approximations are made for phonon scattering events. First, contributions due to N-processes are not considered directly. N-process scattering contributes little to resistance, but it can bias the population wavevectors. The biased population may be such that U-process scattering causes more thermal resistance than it would taken alone. Instead of carriers subscribing to N-process scattering events in the MC module, the U-process scattering

coefficients are tuned up to account for the indirect N-process contribution to resistance. Additionally, U-process events are treated as totally elastic, and therefore momentum conserving. However, U-process events do still randomize particle velocity direction. The U-process scattering coefficients can be chosen such that the total elastic scattering rate has the same effect on resistance that the correct inelastic implementation would. This approximation was seen for ionized impurity scattering in electron populations earlier. Since the inelastic and elastic U-process scattering variants have the same temperature dependence, this is an acceptable simplification that will not effect the trend of thermal resistance with temperature.

Scattering can also occur on lattice imperfections, resulting in what is known as impurity scattering. Impurity scattering is treated elastically, assigning isotropic random velocity directions, just as U-process scattering is. The three scattering rates τ_i used are given by the following formulas:

$$\frac{1}{\tau_I} = B_I \omega^4 \tag{5.5a}$$

$$\frac{1}{\tau_{LU}} = B_L \omega^2 T^3 \tag{5.5b}$$

$$\frac{1}{\tau_{TU}} = \frac{B_{TU} \omega^2}{\sinh\left(\frac{\hbar\omega}{k_B T}\right)}, \quad \omega \geq \omega_{cutoff} \tag{5.5c}$$

where ω is the phonon frequency, B_i are the scattering rate coefficients, T is the lattice temperature, k_B is Boltzmann's constant, and ω_{cutoff} is the minimum frequency at which transverse U-process scattering can occur. The formulas included in this work are adopted from a common set used in several works [90, 66, 96, 91]. The reassignment of velocity is done by selecting another set of $\cos(\theta)$ and ϕ as done in the initialization procedure. The scattering model is similar to the one used in [66].

5.4 Boundary Conditions in Thermal Simulations

There are two primary kinds of phonon boundary scattering conditions: specular scattering and diffusive scattering. As described in Section 4.5, specular scattering simply inverts the velocity component normal to the face, while diffusive scattering selects a new random direction for the velocity. Specular surfaces perpendicular to the transport axis cannot contribute to resistance, while diffusive scattering will play a major role in the conductivity simulations.

An additional flavor of boundary condition for the thermal simulations relates to generation of carriers. A face or region can be fixed at a certain temperature. This kind of fixing is an isothermal boundary condition. For the volumetric region contained in boundary mesh cells on isothermal faces, this can be accomplished by destroying the phonon population and re-creating one at the correct temperature at the start of every iteration. Faces can be made isothermal by calculating the the average phonon flux that would be passing through the face at thermal equilibrium, and initializing the appropriate number of phonons. The flux calculation is a modified version of the calculation that computes volume density during initialization, and requires an additional table to be generated. Care must be taken during isothermal phonon injection, and during diffusive boundary scattering, to inject phonons with random *hemispherical* velocity distributions. The equations governing the hemispherical velocity generation are given by Equations 5.3a through 5.3c as above, but using

$$\cos(\theta) = \sqrt{r_1} \tag{5.6a}$$

$$\sin(\theta) = \sqrt{1 - \cos^2(\theta)} \tag{5.6b}$$

$$\phi = \quad 2\pi r_2 \tag{5.6c}$$

instead for random angle assignment. These algorithms are described in [95, 66, 90].

5.5 Temperature Calculation

The thermal mesh is managed by the functions in the Thermal module. A summary of Thermal module functions is found in Table 5.1. Temperature calculation is starts with

Table 5.1: Thermal module functions

Function Name	Description
setupThermal	Called during initialization, prepares memory space for storing cell energies, cell temperatures, and the energy-temperature lookup table
Efind	Called during initialization, computes the energy-temperature lookup table for the mesh cell volume
calculateEnergies	Called before *calculateTemperatures* to sum the phonon energies in each cell
calculateTemprature	Utilizes the *findT* function to assign temperature mesh values from energy mesh values
FindT	Given an energy, returns temperature values from the energy-temperature lookup table
saveThermal	Saves the energy and temperature mesh for a given timestep to a file
freeThermal	Frees the resources allocated during *setupThermal*

calculateEnergies by counting the phonon energy in each thermal mesh cell. The phonon frequency and energy is known, as that information is carried within all *Carrier* classes. The conversion to temperature from energy is done in *calculateTemperatures* using the same tabular method used for the quantile function during initialization. The average

temperature of a certain volume of cell can be used to compute the expected value of the number of phonons and their energy distributions. The temperature and the total phonon energy calculated in this way is saved into a table for use by the Thermal module by the function *Efind*.

Other phonon simulation settings that impact temperature calculation are the number of simulated particles and the thermal mesh size. The thermal mesh size obviously is linked to the spatial resolution of the temperature result. Less obviously, the number of simulated particles impacts the temperature resolution due to the use of superparticles. Superparticles are particles representative of an ensemble of other particles. Typically the superparticle is treated as an ordinary particle for for all dynamics calculations. However, superparticles have special properties considered during other calculations, such as how large of an ensemble it represents. A single particle's momentum and energy are constrained to each other by the dispersion relation. The ensemble average of momentum and energy are not constrained in this way, and do not follow the relationship in general. Therefore, only one can be a totally accurate summary of the ensemble, while the other is an approximation. The approximation is valid if the ensemble represented is composed of a large number of already-similar particles. Due to the large number of phonons in even a nanodevice, superphonons that exactly conserve energy are utilized. During initialization, an initial superphonon-to-phonon ratio is supplied. The number of particles created is reduced by this factor, and superparticles are used for each. The phonons initialized in this way can be conceptualized as ensembles of identical phonons with the correct energy and momentum distribution for a thermal population. However, the energy and momentum resolution is reduced, as now energy must be counted in packs of superphonon energies.

This impacts the resolution of the temperature calculation directly. The temperature calculation yields best results when there are around 2000 simulated particles in each thermal mesh cell. The superphonon ratio is therefore selected such that the phonon density at the expected average temperature, divided by the ratio, is approximately equal to 2000.

5.6 Verification of Bulk Thermal Conductivity

This work primarily studies transport phenomena that are a function of temperature. The first phonon verification test is selected to establish that the selected dispersion relation and scattering settings could replicate the temperature dependence of thermal conductivity in bulk Si. The logic of this step is similar to the electron mobility study conducted in Chapter 4. If the proper bulk conductivity trend can be reproduced, then band structure and the relative strengths of the scattering processes are taken to be physically accurate.

The same approach was taken to tuning the scattering coefficients was in the electron study. Theoretical values from literature were used as a starting point, and different scattering rates were isolated by careful selection of simulation settings. For example, impurity scattering was tuned in a low-temperature simulation where virtually all of the resistivity is attributed to that process.

The simulation itself is of a simple rectangular prism, with specular boundary conditions on all lateral faces. The specularity simulates an infinite crystal, as scattering events on specular boundaries do not modify momentum or velocity components associated with the transport axis. The end faces are maintained at constant temperature as described in Section 5.4. For all tests, the endpoint temperatures were selected to be +5 K and −5 K

from the desired average temperature. The temperature profile of an ideal bulk material should assume a linear gradient between the endpoint temperatures, crossing the average at the midpoint.

The simulation proceeds according to the flowchart in Figure 5.3. The simulation

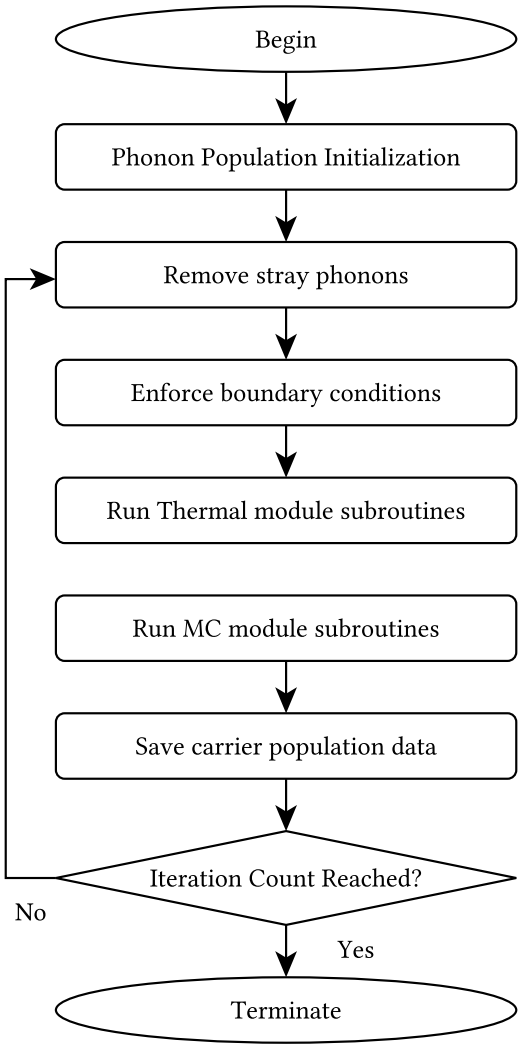


Figure 5.3: Bulk conductivity simulation flowchart

begins by initializing the phonon population and setting up the Thermal module. The phonons in end face cells are initialized to a thermal population at the temperature

associated with those boundary conditions. The middle cell phonons are initialized to the "cold end" temperature. During simulation, the middle cells will rise to meet the expected linear temperature gradient.

The loop begins, with out-of-bounds checking to precede the temperature field calculation. Phonons that have moved outside of the domain through the end faces will corrupt the temperature calculation, so this order is fixed. The positions of every carrier on the transport axis are saved. The temperature calculation is done next, followed by the MC calculation of trajectories and scattering. The new positions are compared to the old positions to determine the net energy flux across the midplane of the simulation domain. This flux is added to a running average. The desired output statistics are calculated and printed or saved. In this case, the conductivity and temperature profile are the desired outputs. The temperature profile is saved to the filesystem in *saveThermal*. The running average energy flux is printed at this stage. For convenience, the energy flux is converted into a conductivity and conductance, which are also printed.

The simulation runs until the temperature profile is linear, and the average conductivity is recorded. The results of this simulation at critical temperatures is displayed in Figure 5.4. In the graph, resulting datapoints from the simulation are compared to conductivity measurements conducted in [2]. The results of the simulation indicate that the phonon bulk properties are sufficient to replicate the temperature trend.

5.7 Conductivity Studies in Thin Films

Once the bulk scattering is correctly set up, the next step is to verify the influence of boundary scattering on thermal conductivity. This verification is done in a similar

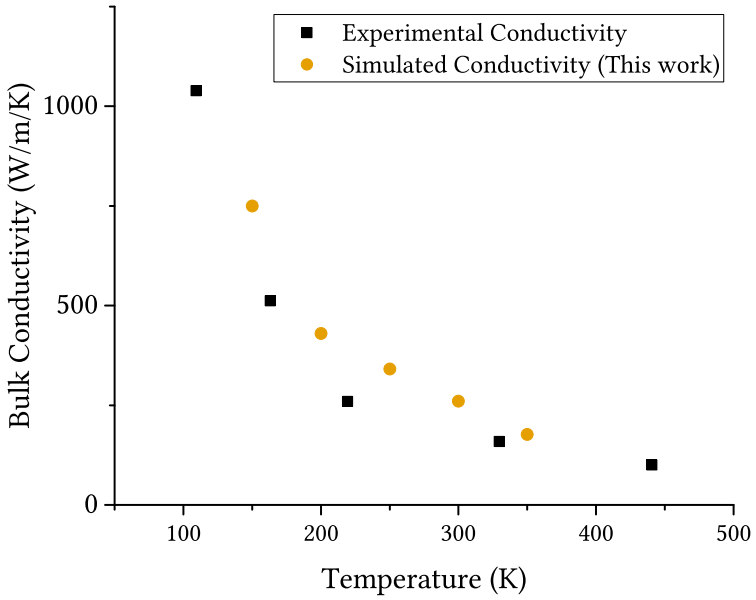


Figure 5.4: Comparison of simulated bulk conductivities with experimental data from [2]

conductivity study, but on Si thin films instead of bulk Si. The simulation is, in fact, identical in terms of code. The boundary scattering input setting of the upper and lower faces is altered from specular to diffuse, so that collisions with the top and bottom faces now do impact transport axis velocity and momentum. There is a corresponding decrease in phonon mean free path as the boundary scattering becomes dominant over bulk scattering. Instead of a sequence of simulations being performed with changing average temperature, these simulations test the conductivity as the thin film thickness changes. The thin film simulations were all performed with the hot and cold ends of the domain set to 310 K and 290 K, respectively. The conductivity results, plotted vs thin film thickness, can be seen in Figure 5.5. The expected trend is to see decreasing conductivity as the thin film height is lowered. As separation between the diffuse top and bottom layers decreases, a larger percentage of the perimeter of the cross section has a diffusive boundary. This

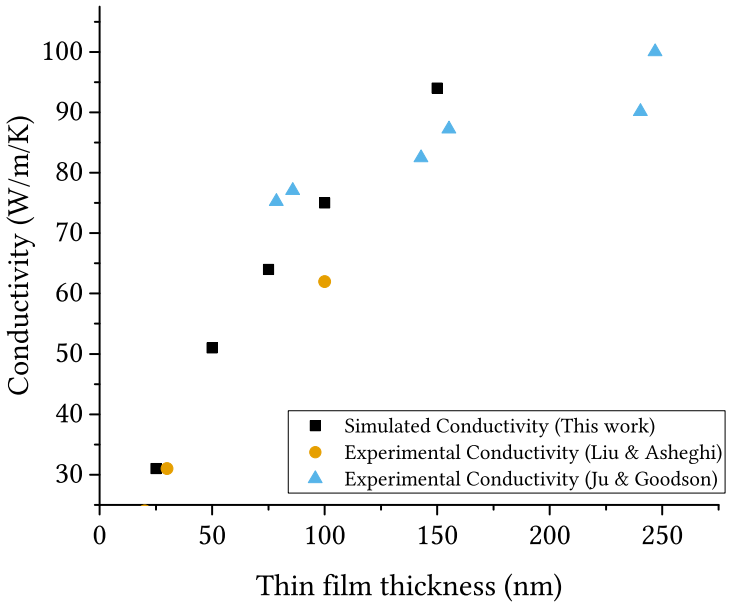


Figure 5.5: Comparison of simulated thin film conductivities

trend is observed and is consistent with the data presented in [98, 92]. The results of this verification study represent the last such verification of the total MC framework. The bulk and thin film simulations were run until the average conductivities were stable in their last decimal place in units of $W/(m\,K)$, consuming around 1000 core hours per data point.

5.8 Thermal Constriction Resistance

The final phonon-only study is a simulation to reproduce a phenomenon known as TCR, which arises with certain nanostructure geometries. The results of this simulation are an original result of this work. This study is not done purely to test the accuracy of the system, but it stands on its own as an investigation. The motivation for this study relates to the geometry of the thermal constriction, which appears in many modern electronic devices. The narrow channels in devices like FinFETs restrict the transfer of heat away from the generated hotspots [99]. This geometry also appears in many MEMS devices.

The “thermal dogbone” structure depicted in Figure 5.6 is the geometric type examined in this study. The two regions of larger cross-sectional area are dubbed the “reservoirs,”

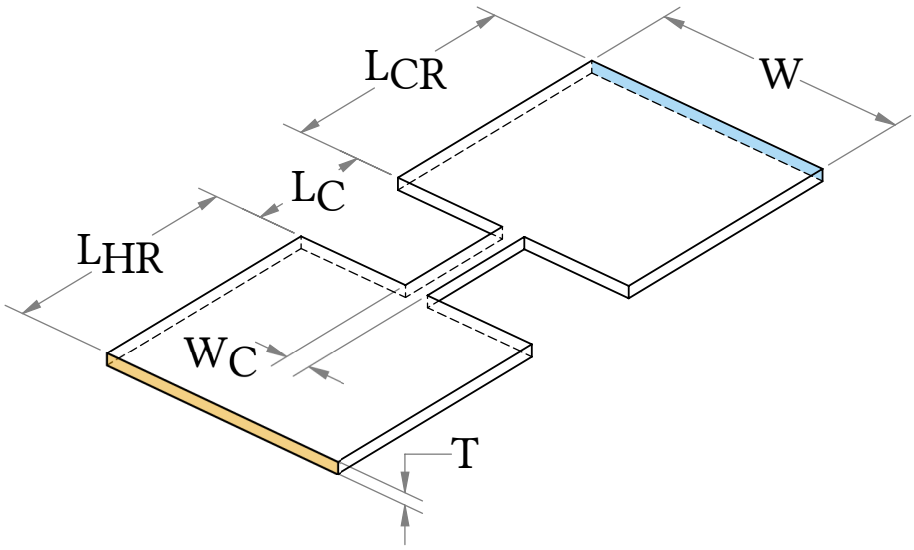


Figure 5.6: 3D view of the “thermal dogbone”

while the connecting bridge is referred to as the “constriction.” Table 5.2 summarizes the dimensions of the two structures simulated in this work. The TCR arises when the phonon

Table 5.2: Dogbone dimensions

Simulation	T	W	W_C	L_{HR}	L_{CR}	L_C	T_{HR}	T_{CR}
TCR	50 nm	1050 nm	50 nm	1000 nm	1000 nm	varies	310 K	290 K
FinFET	30 nm	300 nm	20 nm	300 nm	300 nm	120 nm	300 K	300 K

flux meets a sudden change in cross section at a nanoconstriction, both passing into and out of the region of smaller cross section. There is a resistance associated with both the constriction and expansion aperture. Some portion of the total thermal resistance of the conducting region is derived from the effect of these apertures, and cannot be attributed to

the resistance in the reservoirs or constriction themselves. Figure 5.7 shows the classical thermal circuit compared to the equivalent circuit accounting for thermal constriction and thermal expansion resistance. where R_{HR}, R_C, R_{CR}, R_{const}, and R_{exp} are the resistances

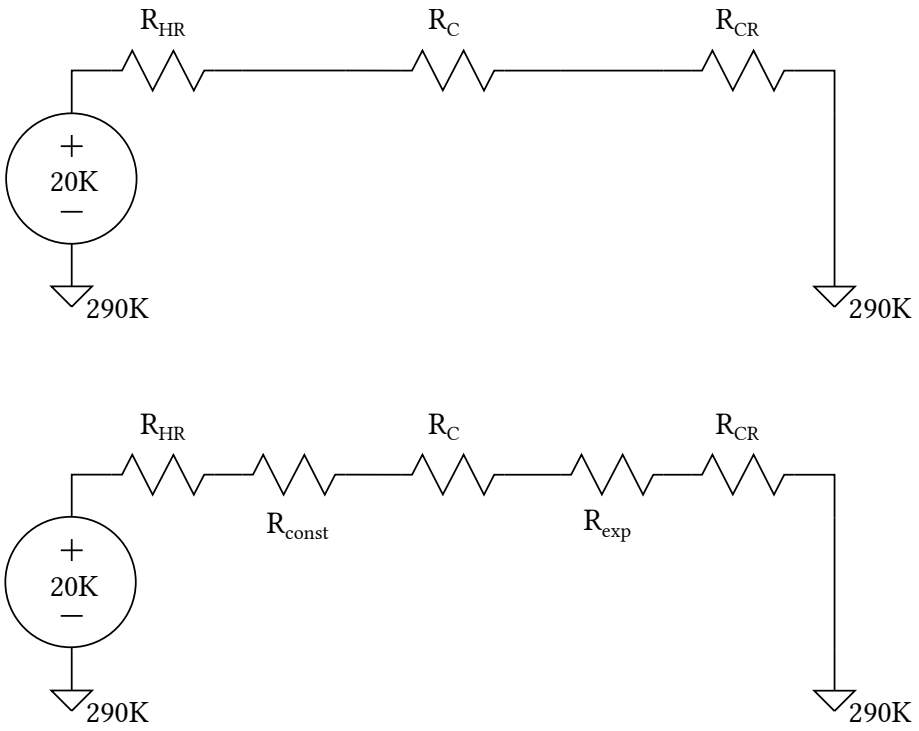


Figure 5.7: Thermal circuit for dogbone structure

of the hot reservoir, the cold reservoir, the constriction beam, the constriction aperture, and the expansion aperture, respectively. Note that the thermal resistances other than the aperture resistances are constructed from material properties using the formula

$$R = \frac{L}{\kappa A} \tag{5.7}$$

where R, L, κ, and A are the thermal resistance, region length, thermal conductivity, and region cross-sectional area, respectively. Using specific dimensions from Figure 5.6, these

other resistances are given by

$$R_{HR} = \frac{L_{HR}}{\kappa_{TF} W T} \tag{5.8a}$$

$$R_{C} = \frac{L_{C}}{\kappa_{NW} W_{C} T} \tag{5.8b}$$

$$R_{CR} = \frac{L_{CR}}{\kappa_{TF} W T} \tag{5.8c}$$

where the quantities k_{TF} and k_{NW} refer to the thin film and nanowire conductivities, respectively. This is a non-classical heat transfer effect modeled by phonon-based methods. The effect is only detectable when the phonon mean free path is significantly larger than the aperture dimensions, so that the multiple geometric reflections involved at the aperture are not washed out by other scattering events. The flexible geometric boundary conditions and easily interpretable output data of the MC approach facilitate simple calculation of the TCR through simulation. The actual simulation code is, again, largely the same as the previous phonon simulation. Inputs to the simulation that change are completely due to the difference in boundary faces and their specularity. The TCR simulation has the 32 boundary faces depicted in Figure 5.6, and all faces are considered diffusive. The energy and conductance are still printed as they were for the thin film simulation. In Figure 5.8, the resulting resistances are plotted vs constriction length for a constant constriction aperture of 50 nm $\times$ 50 nm.

As the constriction length approaches zero, the thermal resistance reaches a nonzero intercept with the resistance axis. The intercept represents the remaining resistance which is not due to the constriction's nanowire conductivity, length, and area. In the absence of TCR effects, this intercept should be completely explained by the thin film conductances of the reservoirs. Using the previously calculated 50nm film conductivity and the reservoir

dimensions, the individual reservoir's resistance can be calculated to be

$$\begin{aligned} R_{res} &= \left(\frac{1}{\kappa_{50\text{ nm}}}\right)\left(\frac{L}{A}\right) \\ &= \left(\frac{1}{50\ W/m\ K}\right)\left(\frac{1000\text{ nm}}{1050\text{ nm} \times 50\text{ nm}}\right) \\ &= 3.81 \times 10^5\ \text{K/W}. \end{aligned} \tag{5.9}$$

Linear regression of the data in Figure 5.8 reveals a resistance intercept of 22.5×10^5 K/W. If the only resistances in the system are due purely to the thin film and nanowire conductivities, then this would be an error of approximately 295%. Attributing the residual resistance equally to the constriction and expansion apertures, the 50 nm × 50 nm TCR is found to be 7.44×10^5 K/W. In the absence of reservoir resistance, transport is still not

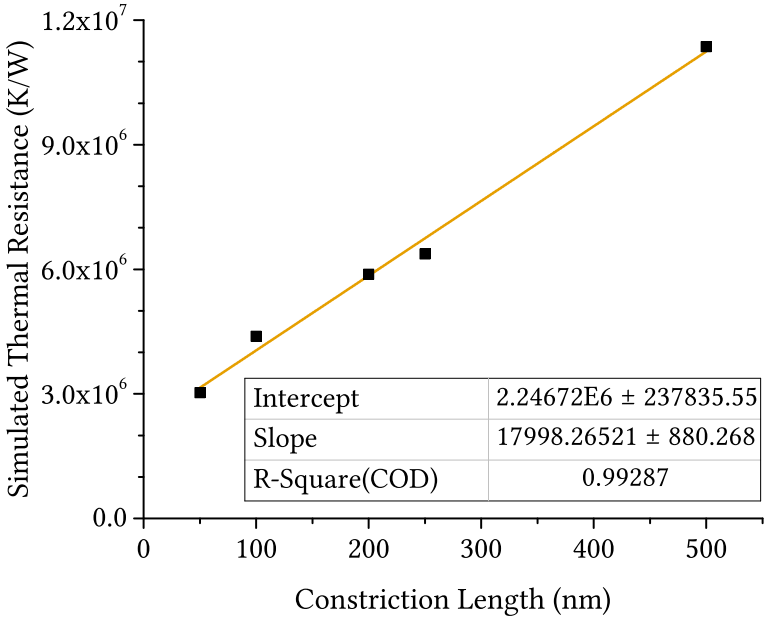


Figure 5.8: Comparison of simulated resistances

appropriately characterized by a fixed, effective nanowire conductivity when the length of the constriction is short. Such an effective nanowire conductivity could be used if the total nanowire resistance were much larger than the resistance of the constriction and

expansion apertures. Using the linear regression in Figure 5.8, it can be seen that a length of over 1200 nm is required before the intercept resistance is less than 10% of the resistance of the structure.

5.9 Thermal Simulations of FinFET Geometry

In preparation for the electrothermal FinFET simulation in Chapter 6, the adjusted "dogbone" dimensions listed in Table 5.2 are used in another phonon-only simulation. The dimensions of the constriction have been altered to match the dimensions of the FinFET simulated in Chapter 4, and the reservoir sizes have been reduced to decrease computation time. The constriction has a aperture of 20 nm × 30 nm and the constriction length is 120 nm, slightly longer than the FinFET domain size. The simulation demonstrated a total resistance of 10.6×10^6 K/W. A surface plot of the temperature mesh with interpolated isothermal contours and the corresponding heatmap are seen in Figures 5.9 and 5.10.

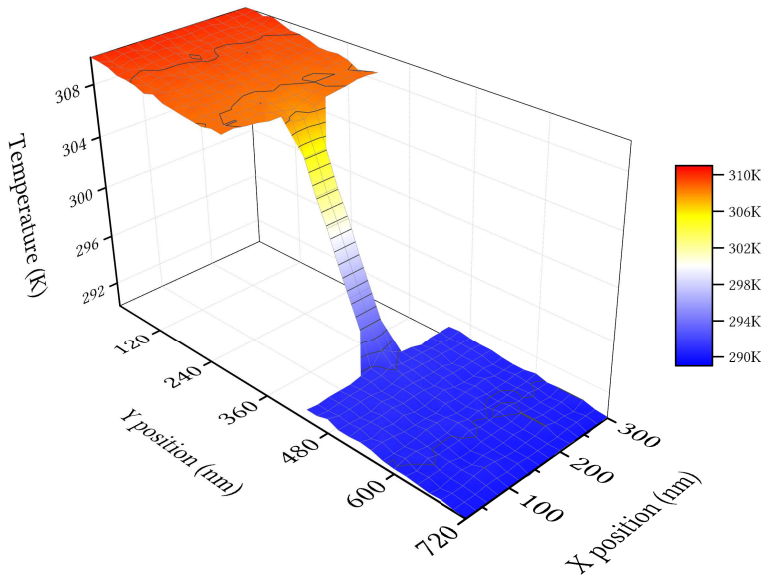


Figure 5.9: Surface plot of temperature distribution in thermal simulation of FinFET geometry

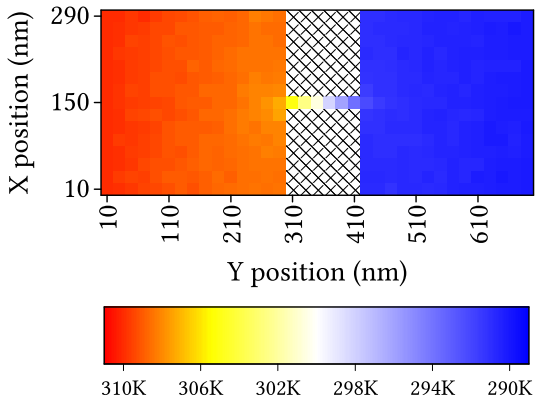


Figure 5.10: Heatmap plot of temperature distribution in thermal simulation of FinFET geometry

The increased slope in the constriction region is of note and consistent with the expectation that the nanowire resistance is much larger than the reservoir thin film resistance. Notable features include the slight "fan in" and "fan out" shape of the surface. The fan out trend is particularly evident in the expansion reservoir, as shown in Figure 5.11. These features can all be observed clearly in the trace depicting the temperature profile down the centerline of the structure is given in Figure 5.12. This simulation, as

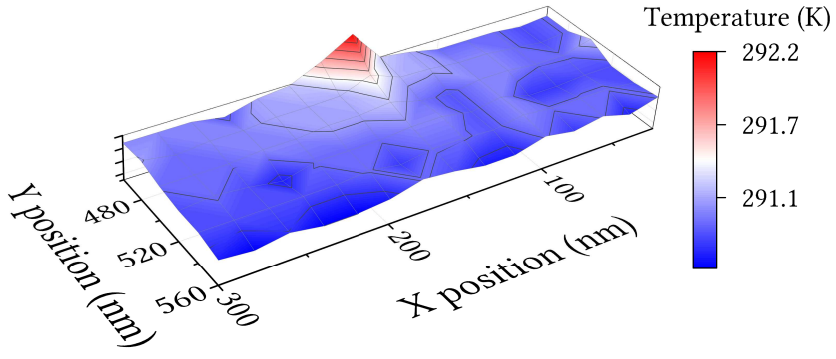


Figure 5.11: Detail of temperature trend in FinFET geometry expansion reservoir

well as the simulations done in Section 5.8, is considered finished when this temperature profile was approximately symmetric, after around 6000 core hours. The symmetry is

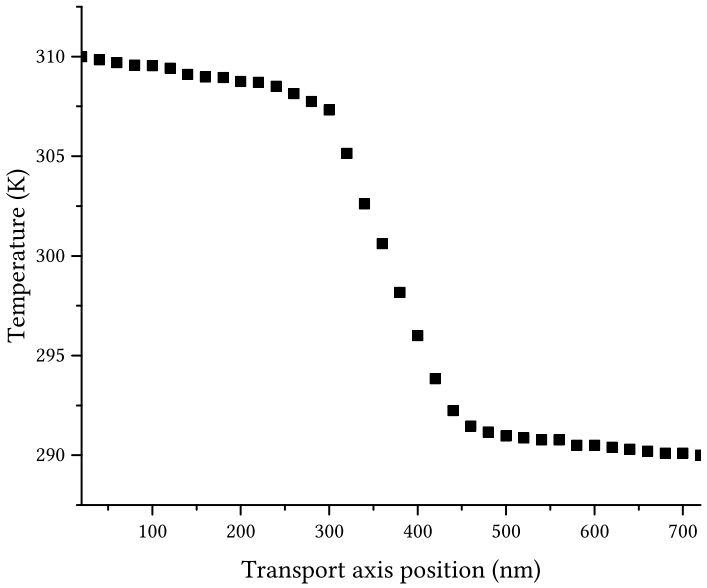


Figure 5.12: Temperature along FinFET transport axis

achieved well after the relative error in the conductance average is below 1%. The extended runtime of this simulation compared to the other simulations in this work so far is due to the propagation of heat from the hot reservoir boundary face. The middle domain cells are initialized to the cold reservoir's boundary temperature. Due to the constriction, it takes a large number of iterations for each cell to rise to its proper value as phonons gradually enter and pass through the constriction.

CHAPTER 6

ELECTROTHERMAL SIMULATION

The primary goal of this work is to create and test a solution for numerical simulation of simultaneous, interacting heat transfer and charge transport processes in nanoscale devices where classical approaches to either problem cannot be applied. The previous chapters have discussed the motivation, related concepts, and challenges related to this goal and documented the supporting work building up to it. That supporting work primarily consists of custom-built MC framework, designed with a forward-looking plan for its architecture that allows it to natively supports multicarrier simulation. The previous chapters have also described the verifications and testing that established the validity of the MC framework.

In this chapter, the theory and details of the electrothermal simulation system are discussed, and the results of an electrothermal simulation are presented. The first sections discuss both the physical process by which energy is transferred between electrons and phonons and also the code mechanism which supports the coupling of the populations. Competing strategies are compared and contrasted with this work. The final section includes graphs and numerical results of the electrothermal study.

6.1 Fast Phonon Recycling

Optical phonons, like acoustic phonons, are quantized lattice vibrations. While acoustic phonons correspond to the oscillation of whole unit cells with respect to one another, optical phonons are those vibrational patterns in which different atoms within a unit cell oscillate with respect to one another. Thus, optical phonons are only possible in material systems that have multiple atoms per lattice site, such as Si. Optical phonons are named so because of the tendency for electromagnetic radiation to excite these vibrational modes in ionic crystals: the positive and negative sublattices move in opposite directions in the same applied field. Si optical phonons have their own branch in the dispersion relation, above the acoustic branches, and come in longitudinal and transverse modes. Of course, these modes do not respond to electromagnetic excitation in covalent materials like Si, but the vibrations themselves can still occur if the lattice supports it.

A MC study by Pop shows that the majority of heat generation is due to optical phonon modes, with longitudinal optical phonons alone responsible for approximately 60% [13]. These optical phonons are generated by interactions with the lattice by high energy electrons. The optical branches of the dispersion relation do not pass through the origin of the E vs k graph, and they are relatively flat compared to acoustic phonons. These characteristics mean that there is a minimum energy investment required to create an optical phonon and, once created, they have very low velocities. This minimum is referred to as the optical phonon energy, $\hbar\omega_0$. Electrons accelerated to high velocities in an electric field will reach a point where their kinetic energy meets or exceeds $\hbar\omega_0$. At this point, electron scattering against lattice particles has the possibility of being an inelastic “optical

phonon emission" event. The emission rate continues to increase with excess energy above $\hbar\omega_0$.

The large impact of optical phonons on heating partly is attributed to the low optical phonon velocity. Significant optical phonon population will build up in localized areas as they are generated and linger [14]. This optical phonon "bottleneck" is responsible for a localized increase in resistance near the drain, where the electric field is expected to peak [13]. The high electric field causes optical phonon generation, while optical phonon absorption scattering also occurs in the electron population at increased rates when the local optical phonon concentration is increased. This back-and-forth exchange of energy during emission and absorption events is called fast phonon recycling [100].

Optical phonons have very low group velocity, and contribute very little to thermal energy *transport* [13]; however, optical phonons decay into more mobile acoustic modes after some time. This decay is due to the anharmonic terms of the interatomic potential, which couples modes which are ideally linearly independent. The phonon quantization was performed using linear springs with only harmonic potentials, and hence this decay must be manually reinserted as a scattering process. The acoustic modes, with their higher velocities, carry energy away from the generation center and into lower-field parts of the device.

A sought-after feature of combined electrothermal simulation is the modeling of microscopic nonlocality, where electrons reach the optical phonon emission energy at a certain point in the channel, but may deposit their energy with the lattice over many scattering events as they travel, which could be on the order of 5 nm [13, 7]. This is a

separate effect from secondary generation of acoustic phonons, which contribute to a similar smearing of the apparent point of heat generation.

6.2 The Optical Phonon Object and Electrothermal Coupling by Scattering

Optical phonons are instances of the class *PhononSiOptical*. The optical dispersion relation is approximated in this work as a constant nonzero energy, and accordingly *PhononSiOptical* objects return zero when their velocity is queried by the system. The energy returned is equal to the minimum of the real Si optical phonon dispersion branch. *PhononSiOptical* objects can participate in property-dependent scattering and subscribe to scattering processes in the MC module just like any other *Carrier* object, but their zero velocity will prevent the MC module from updating their positions during processing of the optical population.

Optical phonons are created in the code using the already-implemented optical phonon emission scattering processes corresponding to Equation 4.9c. This process is an electron scattering process, implemented with a constant rate dependent only on electron energy. With electrothermal coupling disabled, the emission scattering event will simply decrease the electrons energy. No account is kept of the total energy leaked in this fashion, and neither optical absorption nor emission processes are aware of any optical phonon density in the device. When electrothermal coupling is enabled, the usual after-scattering state selection code will also add an optical phonon instance to the optical phonon population. The new phonon's position is set to be the location of the electron at the time of scattering. The difference in the electron's momentum due to the energy change is recorded as the phonon's momentum.

The behavior of the optical absorption scattering process, Equation 4.9b, is also modified when electrothermal coupling is enabled. The scattering process is simply not allowed to occur if there is no nearby optical phonon for absorption. This change effectively alters the scattering rate and introduces a dependence on the spatially-varying optical phonon population density.

Both [13] and [100] ascribe a 1 ps to 2 ps timescale for acoustic decay, with the result being two acoustic phonons. The *PhononSiOptical* particles subscribe to a single scattering process in the MC module with this scattering time. The scattering process is implemented by the code responsible for after-scattering state selection. The optical phonon object's position is updated to a location outside the simulation domain. From that point on, the object will not interact with any other particles within the simulation domain through any mechanism. The particle will be safely deallocated at the beginning of the next timestep when out-of-bounds particles are identified and removed. After updating position, the code will produce two acoustic phonons. The acoustic phonons are selected with $E = \hbar\omega_0/2$ so that energy is conserved, but the current implementation does not make an attempt to conserve momentum. Instead, the acoustic phonons are both given random velocities as if they were being created during initialization. While the optical phonon momentum is recorded in the *PhononSiOptical* object, conserving both energy and momentum is involved. A search would have to be conducted for a pair of locations in the acoustic dispersion table with energies and momenta that summed to the optical phonon energy and momentum. One or the other must be picked to simplify the calculation. As the primary goal is modeling the conversion of electrical energy to thermal energy

and calculating the spatial extent of a temperature increase, the preferred quantity for conservation is total energy.

6.3 Timescale Resolution

As mentioned in the introduction, electron populations are fast, with events occurring the order of picoseconds or tenths of picoseconds apart, whereas the phonon populations are slow with potentially nanoseconds or longer between events of interest [3, 13, 14]. This is a challenge of electrothermal simulations. The number of events separated by one picosecond, on average, occurring in one nanosecond's time is, of course, 1000. Therefore MC calculations alone for electrons will involve around 1000 times as much computation as phonons do, when the same time interval is simulated. Additionally, the electron MC algorithm involves solving the PE and SE systems for the potential mesh and band structure. These two calculations are very slow compared to the bare MC, as they are deterministic finite difference solvers operating on a 3D domain. Timing profiles of the completed simulations have shown that around 80% of the real-world time taken to simulate the electron population is spent solving these finite difference problems.

This problem is addressed in various ways by different authors. In many cases, the solution is to solve the phonon BTE with an analytic, approximate method and couple this solution to the MC solver in a way similar to the coupling of the SE and PE equations. Clearly this approach sacrifices accuracy. The MC method is used to solve the phonon BTE because of the intractability of an accurate analytical expression for the phonon-phonon scattering, not to mention the coupled electron-phonon scattering. Often the electron simulation is used to statistically characterize the phonon generation, which is used as an

input for either the previously mentioned type of approximate solver or used as an input for a phonon MC study. The latter does have the accuracy of the MC method, but will fail to reproduce transport processes where the behavior of the electron population is influenced by the phonons. The next kind of solution involves alternating decoupled electron and phonon MC simulations, feeding generation statistics into the latter while feeding lattice temperatures into the former [101, 102, 103]. The alternating solution takes advantage of the fact that the electron population reaches steady state early on and does not need to run past that point, thereby avoiding the problem of running the electron MC for the entire phonon timestep. The shortcoming of that approach is that the re-evaluation time for computing a new electron population must be carefully chosen in a case-by-case basis so that *too much* time does not pass and the phonon generation statistics still appropriate through the duration of the phonon timestep. Another shortcoming is the lack of a direct accounting of the energy and momentum exchange between the populations, which is only satisfied in an average sense.

As already discussed, the approach taken in this work differs by actually implementing the physics of the electrothermal population coupling. However, a solution is still required for the timestep problem. In the current framework, the different populations are being simulated together in the same process, and all properties of the various populations can be read by the main simulation function at any moment. This capability leads to a unique kind of timestep handler. The electron phase of the calculation is accelerated by allowing the band structure and potential calculations to be skipped if there is no significant change in a given monitor condition. The monitor condition is configurable and can relate to any particle population. Candidates are a temperature change, a change in

applied boundary condition, or the end of a fixed time interval. The underlying reasoning is that a steady-state electron population will result in identical solutions from the PE and SE solvers each time they are invoked. Much like a variable timestep algorithm for integration, this allows the framework to invoke the heavier solvers sparsely or frequently, depending on the requirement. Since the MC module has "live" access to the temperature data as it evolves during phonon sims, the sleep/wake period for the PE and SE solvers can be adjusted on the fly. The electron population is still treated by the MC module during every timestep, and scattering events still occur without being skipped. Similarly, early evaluation of the acoustic phonon population can be triggered by a monitor condition, such as a certain number of additional acoustic phonons being emitted. This approach is used successfully in this approach to dynamically adjust temperature-dependent scattering rates as the phonon temperature changes.

6.4 Ensemble Repackaging

A further complication arose during the development of this framework, directly due to the use of the MC module scattering system to couple the populations. Acoustic phonons in this work are simulated as superphonons, while electrons are directly simulated $1:1$. All acoustic events, including creation, are conceptually representative of *thousands* of such events happening simultaneously in an ensemble of similar particles. Hence, naively connecting the optical emission scattering event of an *ElectronSiNonparabolic* to the creation of a single *SiAcousticPhonon* object would overestimate the heat generation by a factor equal to the superphonon-to-phonon ratio. That ratio is on the order of $20,000$ in many of the simulations associated with this work. This issue is not encountered in

methods which do not individually account for momentum and energy exchange through scattering.

The solution was to change the superparticle-to-particle ratio from a global property of all *SiAcousticPhonons* to a member property, so that each superphonon had an individual ratio. Now, instead of the scattering code creating a new acoustic phonon object, the scattering code modifies the ensemble of an appropriate nearby phonon. The properties of two acoustic phonons are still generated according to the logic of the previous section, but the resulting velocity, position, and energy are used to calculate new ensemble values. First, the MC Kernel looks for an acoustic superphonon in the current mesh cell that has a similar velocity to the phonon to be added. As discussed in Section 5.5, a superparticle cannot exactly conserve energy and momentum. The other methods used so far focus on conserving energy, so the new ensemble properties are calculated to preserve exactly the total energy:

$$E_{new} = \frac{E_{add} + nE_{old}}{n+1}, \tag{6.1}$$

where E_{new} and E_{old} are the superphonon's new and old energies, n is the superphonon's ensemble count, and E_{add} is the energy of the acoustic phonon being added. The superphonon n is incremented. When the energy sum is taken for the purposes of calculating temperature, the total energy nE_{new} will be correct.

The new phonon energy requires that a new group velocity be selected from the table. This will be an approximate velocity summarizing the ensemble's general direction of travel. Since the repackaged superphonon selection favors those with similar velocities to the phonon being injected, the error is minimized. The new phonon group velocity is

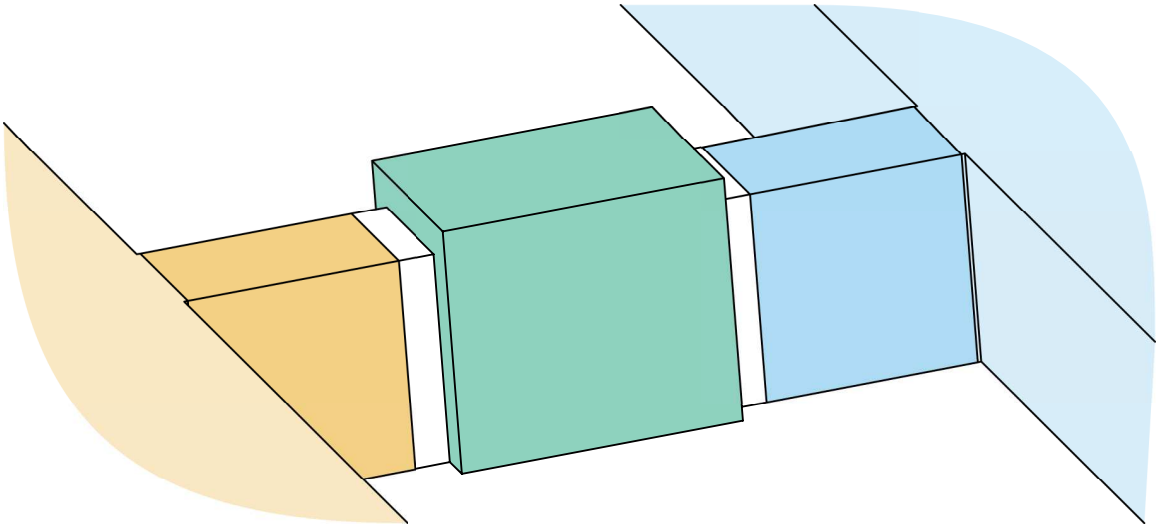
used with Equations 5.3a to 5.3c to compute the new cartesian velocity components. The required angles θ and ϕ are chosen by interpolating the the superphonon and injected phonon's angles based on their relative contribution of energy.

The final result is that more energy is added to a specific location in the phonon population, and the velocity of the system is tweaked slightly. This mechanism is tested in the following simulation, and does confer energy from the optical emission process into the phonon population at the correct location.

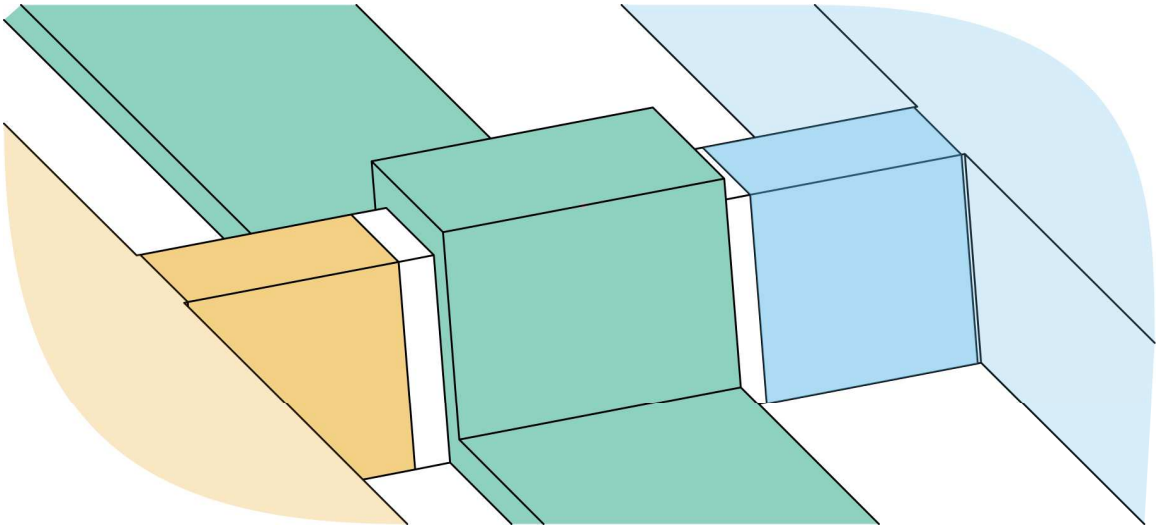
6.5 Simulation of Electrothermal Characteristics of a FinFET

Here, the results of an electrothermal simulation of the previously presented electron-only FinFET is discussed. The geometry, doping, and boundary conditions of the electrical domain of the simulation are exactly as described in Chapter 4. The electrothermal properties of FinFET devices similar to this one are studied in several recent works [104, 105, 106, 107]. The FinFET itself now lies inside the constriction of the dogbone simulated at the end of Chapter 5. The phonon simulation domain and thermal mesh are larger than the electron simulation domain and Schrödinger-Poisson mesh. The constriction is 120nm long, and the phonon simulation includes the reservoirs. The electrons are simulated only in the interior 110nm of the constriction. The interpretation of the electron MC's Ohmic boundary conditions is that the reservoirs connected to the source and the drain are, in fact, doped and contacted, being a homogenous extension of the electron population in the boundary cells of the Schrödinger-Poisson mesh. The "hot" and "cold" boundaries of the dogbone are both fixed at 300 K. This boundary condition represents the thermal environment to which the transistor must dump its excess thermal

energy. Figure 6.1 shows this arrangement, highlighting the gate, drain, and source of the new system.



(a) FinFET placement, with gate contact hidden



(b) FinFET placement, with gate contact shown

Figure 6.1: Location of FinFET within constriction of dogbone structure, showing source (yellow/orange), gate (green), and drain (blue) regions

The temperature inside the device at any moment is, classically, a function of the generation profile, the material *bulk* conductivity, and the material geometry. The boundary scattering in both the constriction comprising the transistor's fin, and in the thin films comprising the reservoirs, contribute to the resistance on top of the bulk properties. This extra resistance will increase the size of the hotspot that forms in the transistor, as heat cannot escape the device as efficiently. Additionally, the heat flux generated by the phonon repackaging will have to travel through the expansion resistance before reaching the 300 K boundary, which will increase the hotspot temperature further. These factors are why FinFET and NWT performance is suspected to be strongly related to self-heating [13].

For electrothermal coupling, the code is modified as discussed above for optical phonon emission and absorption in the electron MC. Optical phonons are simulated $1:1$ with electrons, while acoustic phonons are simulated as superparticles. Optical-to-acoustic decay therefore proceeds with the repackaging procedure outlined above. The electron MC monitor conditions are configured to solve the SE-PE system every iteration for the first 100 iterations, to ensure the simulation's stability through startup transients as the equilibrium population rapidly relaxes into to the transport statistics. After that, the SE-PE system is only solved in the iteration immediately following when the Thermal module recalculates the temperature field. Acoustic phonons are simulated through a longer timestep than electrons, after a configurable integer number of timesteps is computed for electrons. Acoustic phonon timestep evaluations are not set to be triggered by phonon injection. An additional point of coupling is the electron acoustic deformation potential (ADP) scattering, which is proportional to the local acoustic phonon density. ADP scattering is treated as elastic in this work, so the only possible impact is a net decrease in average velocity as

higher temperatures cause more scattering. The combined simulation sequence can be seen in Figure 6.2.

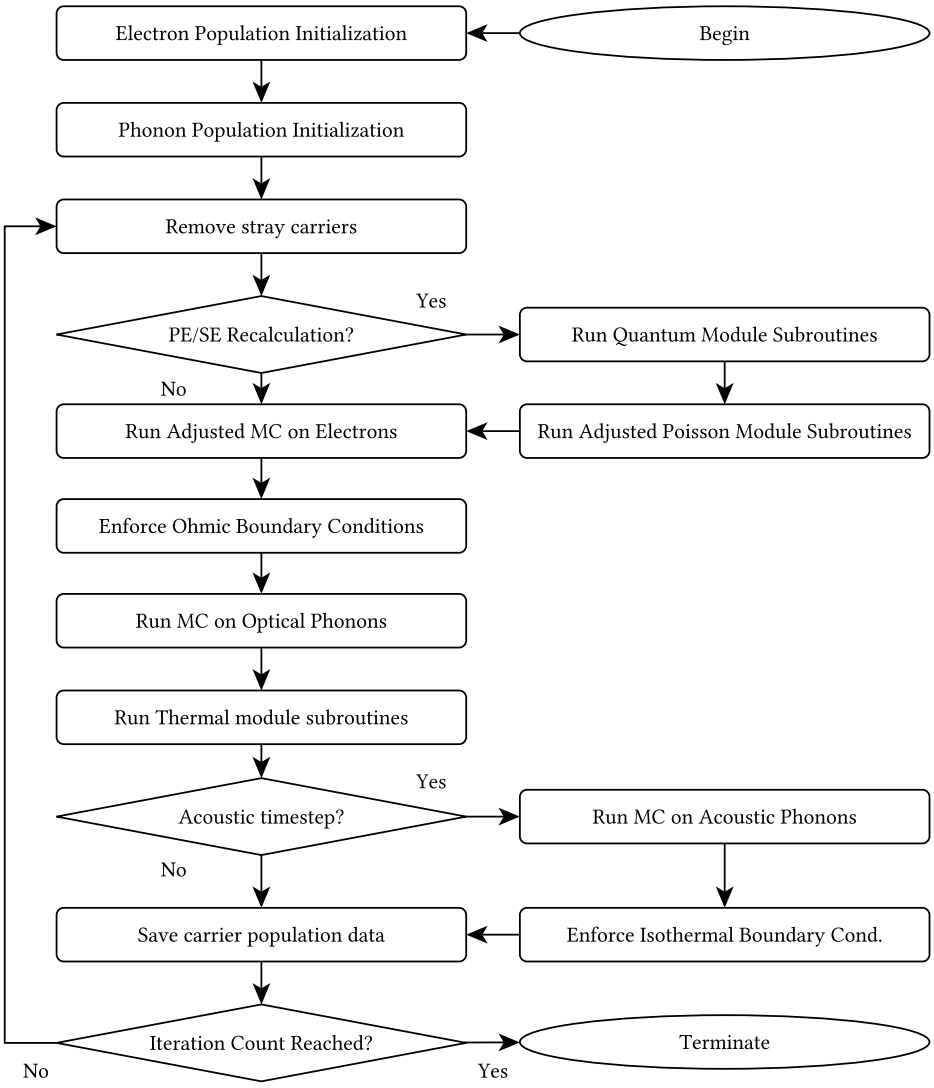
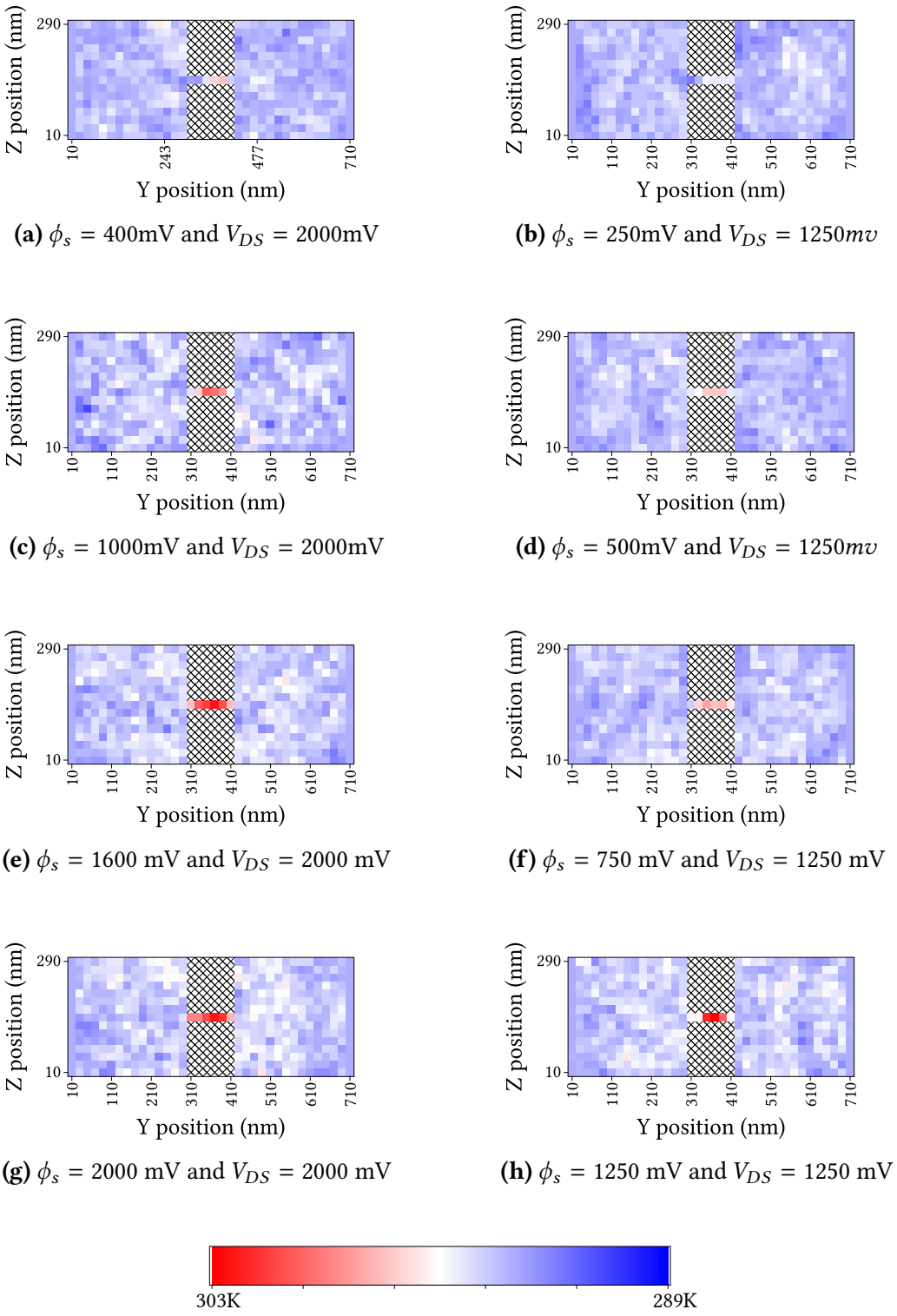


Figure 6.2: Combined simulation algorithm flowchart

The simulation was conducted at several different bias points to discern if the surface potential under the gate oxide would control the shape and intensity of the hotspot formed. Four different gate potentials were simulated for V_{DS} = 2000 mV: ϕ_s = 200 mV, 1000 mV, 1600 mV, and 2000 mV. Another four simulations were done at V_{DS} = 1250 mV using ϕ_s = 250 mV, 500 mV, 750 mV, and 1250 mV. The results are summarized in the heatmaps displayed in Figure 6.3. Indeed, it seems that gate potential does alter the shape of the hotspot, with heating being more localized towards the drain when the gate potential is lower compared to the drain potential. The generation of heat near the drain is expected because the drain region has the steepest slope in potential, and therefore the highest electric field. The higher gate potentials result in an increased electric field early on, nearer to the source, which increases the proportion of electrons near the source that have enough energy to emit optical phonons. The highest temperature in any channel hotspot was found to be 303.0 K, for the bias point $\phi_s = V_{DS}$ = 1250 mV. The hottest temperature observed in a reservoir during any iteration, across all bias points, was 300.8 K, with the mean reservoir cell temperature being 300.0 K. These simulations are very computationally expensive. The convergence time of electronic properties is far shorter than the thermal properties, and so the simulation time is essentially the same as for the TCR simulations in Sections 5.8 and 5.9.



(a) ϕ_s = 400mV and V_{DS} = 2000mV

(b) ϕ_s = 250mV and V_{DS} = 1250mv

(c) ϕ_s = 1000mV and V_{DS} = 2000mV

(d) ϕ_s = 500mV and V_{DS} = 1250mv

(e) ϕ_s = 1600 mV and V_{DS} = 2000 mV

(f) ϕ_s = 750 mV and V_{DS} = 1250 mV

(g) ϕ_s = 2000 mV and V_{DS} = 2000 mV

(h) ϕ_s = 1250 mV and V_{DS} = 1250 mV

Figure 6.3: Heatmap plots of temperature distribution in FinFET device for various bias points

The $V_{GS} = V_{DS} = 1250$ mV bias point was singled out for a special simulation using a 10× increase in dopant concentration in the drain and source regions. The result was a higher current density and therefore higher maximum hotspot temperature of 310.3 K. The corresponding heat map and surface plot are given in Figures 6.4 and 6.5. Even given the temperature rise observed, the I_D at this bias point was reduced by only 1%. While a

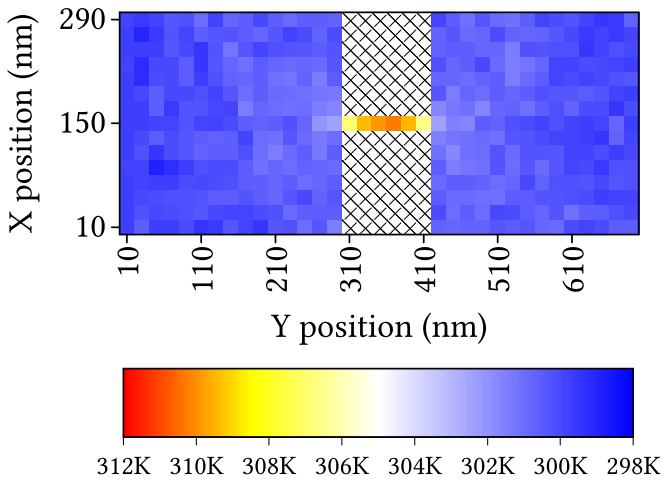


Figure 6.4: Surface plot of temperature distribution in FinFET device with increased dopant concentration

decrease in drain current is expected due to the increased scattering and resistivity, the low magnitude of the drop is not surprising. ADP scattering is the only temperature-dependent scattering process used for electrons in this simulation, and it is not the only active scattering process near 300 K. The impact on total mobility, and therefore resistance and drain current, would be more evident with a more detailed scattering model. However, evidence of feedback from the phonon MC to the electron MC was observed as an increase in ADP scattering rate for channel carriers from 7.64×10^{13} Hz to 8.15×10^{13} Hz, an increase of 6.6%. This near the expected rise of 5.1%. The difference could be explained by

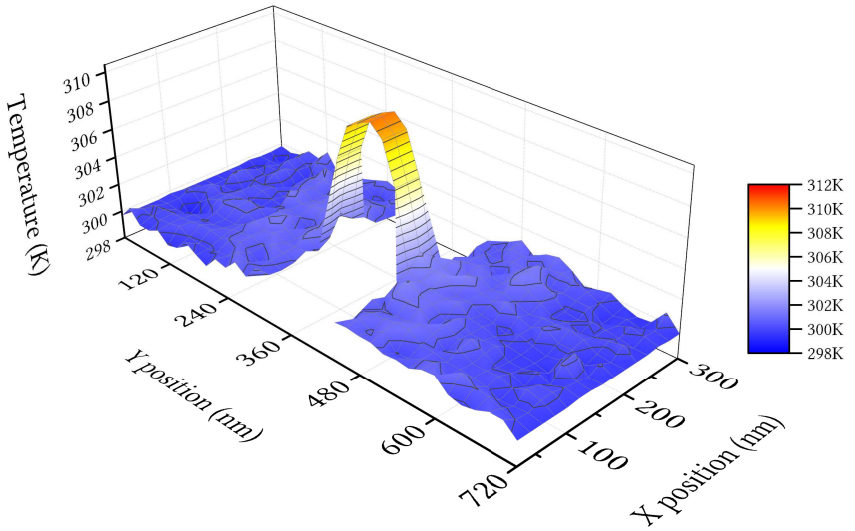


Figure 6.5: Heatmap plot of temperature distribution in FinFET device with increased dopant concentration

subtle changes in average channel carrier energy, which is also a factor in computing ADP scattering rate.

Even though the heating at this bias point is not associated with a significant electrical impact, a temperature rise of 10 K is significant and cannot be accounted for using classical thermal analysis. For example, take a straightforward application of Fourier's Law to a bar of Si with the same dimensions as the constriction containing the FinFET. The heat generation is assumed to be uniform throughout the volume, and temperature change through the reservoir is ignored, with both ends of the constriction taken to be 300 K. The temperature variation is parabolic, given by

$$T = -\frac{\dot{q}L_C^2}{2\kappa}\left(1 - \frac{r^2}{L_C^2}\right) + T_s \tag{6.2}$$

where $\dot{q}$ is the volumetric heat generation, L_C is, again, the constriction length, κ is the thermal conductivity, r is the position along the transport axis, and T_s is the boundary temperature [108]. Even using the MC tool created in this work to find the values $\dot{q}$ =

2.78×10^{17} W/m^3 and $k_{nw} = 21.4$ W/(m K), the resulting peak temperature is 323.4 K. Using a bulk conductivity value of 130 W/(m K), the peak temperature is seen to be 303.8 K. The results lie on either side of the simulation's prediction, which is to be expected. Using the nanowire conductivity neglects the short length of the channel, as Fourier's Law does not predict that some phonons are escaping ballistically to be thermalized in the reservoir. Using the bulk conductivity neglects the boundary scattering that *does* occur. A comparison of the different predictions is shown in Figure 6.6. While not directly applicable

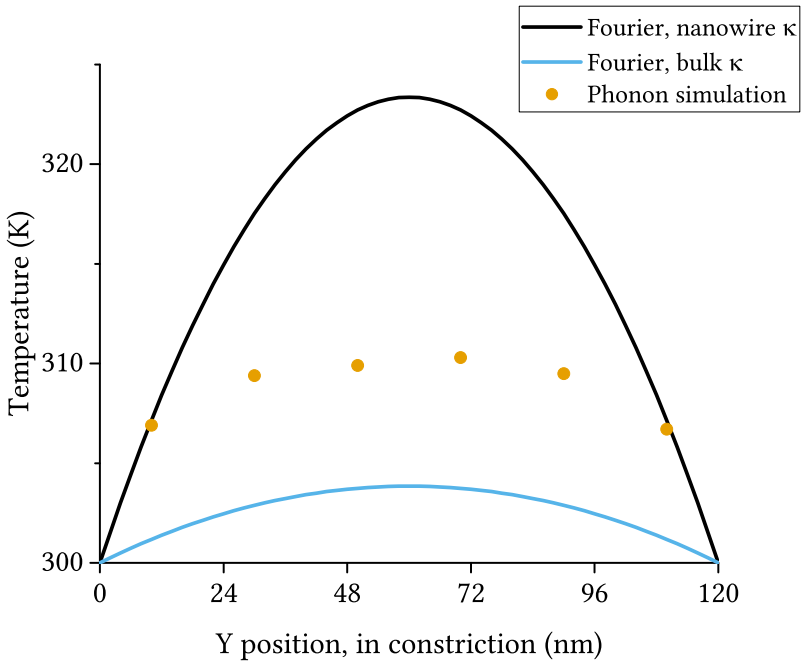


Figure 6.6: Comparison of classical calculations to simulation result

to a single transistor, a ballpark estimate of the impact of this error can be found using reliability formulas used to estimate lifetimes of microprocessors. Thermal derating curves for commercial embedded microprocessors utilize a modified form of the Arrhenius rate equation [109] to compute a quantity called acceleration factor (AF). The AF is the ratio

of the expected lifetime of a transistor operating at T_{use} to the lifetime found in thermal testing at T_{test}, given by

$$AF = \exp\left(\frac{E_a}{k_b}\left(\frac{1}{T_{use}} - \frac{1}{T_{test}}\right)\right) \tag{6.3}$$

where E_a is a parameter called the "activation energy" and k_b is Boltzmann's constant. Assuming a value of E_a as done in [109], the formula above estimates that the lifetime of a part used at 310.3 K will be only 57% the lifetime seen at the low estimate of 303.8 K. Thus, a reduction in mean lifetime of 43% would not be accounted for with the standard classical temperature estimate, even given an accurate number for heat generation.

CHAPTER 7

CONCLUSION

In this work, an object-oriented MC framework for studying quantum transport phenomena in nanoscale semiconductor devices was developed. The framework was developed with the ultimate goal of improving the state-of-the-art in electrothermal analysis for micro- and nanodevices. To that end, the object-oriented features of the C++ language were leveraged to create a generic, configurable kind of MC simulator that can solve the BTE for particle systems characterized by a dispersion relation and one or more scattering processes that consist of scattering rate calculations and scattering state selection. The code is extensible enough that a library of different particle types, with dispersion relations and scattering models of varying levels of accuracy and efficiency, can be easily constructed. The unified simulation framework is simultaneously aware of simulation data and population parameters belonging to any particular particle system being simulated. This flexibility allows for the coupled simulation of electrons and phonons critical to electrothermal simulation, despite their different timescales. Flexibility in describing scattering processes in code is a generic tool for analyzing systems perturbatively, and is utilized to directly account for the energy and momentum exchange between different particle systems.

Additionally, the development of the new MC framework was conducted with several verification studies to confirm it's accuracy and validity as a scientific tool. The results of said studies were presented, each showing that the framework is capable of handling the relevant challenges. First, the electronic simulation capabilities were verified through three increasingly more complex studies: a bulk mobility simulation, an n-i-n device simulation, and a FinFET simulation that accounts for quantum confinement in the fin plane. The bulk mobility simulation showed that the framework can reproduce the temperature dependence of the mobility of bulk Si, confirming the scattering model and band structures used in this work to define the behavior of electrons Si. As the bulk mobility simulation was the first MC simulation, it also served to verify that the basic simulator could properly replicate the dynamics of accelerating particles in constant force fields and that the scattering functions were being executed correctly. The n-i-n diode simulation confirmed the tool's ability to handle internally generated electric fields, bypassing the need to directly simulate point-to-point Coulombic interactions between every charged body in the device. The implementation of boundary scattering and the specification of device geometry was also first incorporated in this simulation, and seen to operate properly. The final electron-only simulation was the FinFET simulation, showing that the simultaneous SE-PE system can be solved self-consistently with the BTE for electrons. The FinFET is also a highly representative example of a modern nanoscale electronic device, like those actually studied studied by peers in the field. The second suite of simulations tested the expanded the functionality for supporting phonon-based heat transfer simulations. Like the bulk mobility test, a bulk conductivity test was done to verify the scattering models for Si phonons were appropriate for thermal simulations.

The subsequent tests showed the ability of the simulator to handle features of ballistic conduction and surface-to-volume effects common in nanoscale heat transfer scenarios, including a simulation of a nanoconstriction structure commonly seen in modern devices. The final simulation was an electrothermal study of a FinFET device, proving that the framework correctly allows energy from one particle simulation to enter and affect other simultaneous particle systems. The electrothermal study showed differences between the simulation results and the classical expectation for Joule heating and conduction in the device, in line with expectations based on theory.

Several approximations were intentionally made in this work, in support of the primary goal of demonstrating the framework is sufficiently capable of simulating the electrothermal interactions described above. The scattering models, in several places, can be improved. Generally, state selection only approximately conserved momentum in favor of exactly conserving energy. This was done in order to avoid an expensive search of the band and dispersion tables for matching states which conserve both properties, an open problem in MC simulation. Energy conservation was selected due to this work's focus on predicting temperature values. The handling of ionized impurity scattering and U-process scattering for electrons and phonons, respectively, can be enhanced by more realistic after-scattering state selection. This change would produce more accurate momentum distributions as the velocities for those scattering events are not necessarily uniformly distributed, as implemented in this work. Additional scattering processes could be added as needed, depending on the operating regime of the device, that capture physical processes not relevant at the temperature range or in the material system studied here.

While the band structure for Si was treated in detail, spin-orbit coupling was neglected in the hole bands. No treatment was given of hole transport, due to the complexity of the hole band structure. The coexistence of mobile positive and negative charges causes stability difficulties in the PE solver, which can be resolved by instead solving the nonlinear PE equation incorporating quasi-fermi levels. This stability comes at the expense of code complexity and time. The dispersion relation for phonons in Si was given a simplified treatment as well, although the code does support loading arbitrarily complex dispersions. Especially optical phonons, which were treated as particles obeying a flat, constant dispersion.

Device geometry can be fairly complex as it stands, supporting anything that can be composed of orthogonally arranged rectangular prisms. A highly-beneficial future task is the improvement of the boundary scattering code to support triangular faces of arbitrary orientation.

Moving forward, there are many areas where more detailed models can be added. The framework was designed with modularity in mind, which not only helped during the creation and implementation of the strategies for supporting electrothermal simulation, but also will allow the platform to be built upon in the future. New material systems can be studied by developing out the appropriate particle classes, with dispersion relations and band structures imported from other programs. In closing, the framework developed here and the general strategy for simulating complex multi-particle systems stand as an exciting new tool to use for diving into studies of more complex devices, allowing for the analysis, design, or optimization of important semiconductor device technologies.

REFERENCES

[1] F. J. Morin and J. P. Maita, "Electrical properties of silicon containing arsenic and boron," *Phys. Rev.*, vol. 96, no. 1, pp. 28–35, Oct. 1954.

[2] C. J. Glassbrenner and G. A. Slack, "Thermal conductivity of silicon and germanium from 3K to the melting point," *Phys. Rev.*, vol. 134, pp. A1058–A1069, May 1964. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRev.134.A1058

[3] O. Muscato and V. Di Stefano, "Heat generation and transport in nanoscale semiconductor devices via Monte Carlo and hydrodynamic simulations," *COMPEL*, vol. 30, no. 2, pp. 519–537, 2011.

[4] A. Ashok, D. Vasileska, O. L. Hartin, and S. M. Goodnick, "Electrothermal Monte Carlo simulation of GaN HEMTs including electron-electron interactions," *IEEE Trans. Electron Devices*, 2010.

[5] B. Padmanabhan, D. Vasileska, and S. Goodnick, "Reliability concerns due to self-heating effects in GaN HEMTs," *J. of Integrated Circuits and Systems*, vol. 8, no. 2, pp. 78–82, Sep. 2013.

[6] E. Pop, R. W. Dutton, and K. E. Goodson, "Analytic band Monte Carlo model for electron transport in Si including acoustic and optical phonon dispersion," *J. Appl. Phys.*, vol. 96, no. 9, pp. 4998–5005, 2004.

[7] ——, "Monte Carlo simulation of Joule heating in bulk and strained silicon," *Appl. Phys. Lett.*, vol. 86, no. 8, p. 082101, 2005.

[8] K. Raleva, D. Vasileska, S. Goodnick, and M. Nedjalkov, "Modeling thermal effects in nanodevices," *IEEE Trans. Electron Devices*, vol. 55, no. 6, pp. 1306–1316, Jun. 2008.

[9] J. A. Rowlette and K. E. Goodson, "Fully coupled nonequilibrium electron-phonon transport in nanometer-scale silicon FETs," *IEEE Trans. Electron Devices*, vol. 55, no. 1, pp. 220–232, Jan. 2008.

[10] T. Sadi and R. W. Kelsall, "Monte Carlo study of the electrothermal phenomenon in silicon-on-insulator and silicon-germanium-on-insulator metal-oxide field-effect transistors," *J. Appl. Phys.*, vol. 107, no. 6, p. 064506, 2010.

[11] S. Sinha and K. E. Goodson, "Review: Multiscale thermal modeling in nanoelectronics," *Int. J. Multiscale Comput. Eng.*, vol. 3, no. 1, pp. 107–133, 2005.

[12] S. Sinha, E. Pop, R. W. Dutton, and K. E. Goodson, "Non-equilibrium phonon distributions in sub-100nm silicon transistors," *J. Heat Transfer*, vol. 128, no. 7, pp. 638–647, 2006.

[13] E. Pop, S. Sinha, and K. E. Goodson, "Heat generation and transport in nanometer-scale transistors," *Proc. IEEE*, vol. 94, no. 8, pp. 1587–1601, Aug. 2006.

[14] E. Pop, "Energy dissipation and transport in nanoscale devices," *Nano Res.*, vol. 3, no. 3, pp. 147–169, Mar. 2010.

[15] "Vision and voyages for planetary science in the decade 2013-2022," Tech. Rep., 2011.

[16] A. Bar-Cohen, J. D. Albrecht, and J. J. Maurer, "Near-junction thermal management for wide bandgap devices," in *2011 IEEE Compound Semiconductor Integrated Circuit Symposium*. Defence Advanced Research Projects Agency, 2011.

[17] R. M. Incandela, L. Song, H. Homulle, E. Charbon, A. Vladimirescu, and F. Sebastiano, "Characterization and compact modeling of nanometer CMOS transistors at deep-cryogenic temperatures," *IEEE Journal of the Electron Devices Society*, vol. 6, pp. 996–1006, 2018.

[18] B. Patra, R. M. Incandela, J. P. G. van Dijk, H. A. R. Homulle, L. Song, M. Shahmohammadi, R. B. Staszewski, A. Vladimirescu, M. Babaie, F. Sebastiano, and E. Charbon, "Cryo-CMOS circuits and systems for quantum computing applications," *IEEE Journal of Solid-State Circuits*, vol. 53, no. 1, pp. 309–321, Jan. 2018.

[19] "MPICH high performance portable MPI," https://www.mpich.org/, 2019, accessed: 2019-08-09.

[20] "LONI Louisiana Optical Network Initiative," https://loni.org/, 2019, accessed: 2019-08-09.

[21] M. Born and R. Oppenheimer, "Zur quantentheorie der molekeln," *Ann. Phys.*, vol. 389, no. 20, pp. 457–484, 1927.

[22] E. Schrödinger, "An undulatory theory of the mechanics of atoms and molecules," *Phys. Rev.*, vol. 28, pp. 1049–1070, Dec. 1926.

[23] M. Lundstrom, *Fundamentals of Carrier Transport*. Cambridge University Press, 2009.

[24] D. Neamen, *Semiconductor Physics And Devices*, 3rd ed. New York, NY, USA: McGraw-Hill, Inc., 2003.

[25] F. Bloch, "Über die quantenmechanik der elektronen in kristallgittern," *Zeitschrift für Physik*, vol. 52, no. 7, pp. 555–600, Jul. 1929.

[26] C. Kittel, *Solid State Physics*, 8th ed. John Wiley and Sons, Inc., 2004.

[27] G. P. Srivastava, *The Physics of Phonons*. CRC Press, 1990.

[28] N. W. Ashcroft and N. D. Mermin, *Solid-State Physics*. Saunders College, 1976.

[29] C. Jacoboni and L. Reggiani, "The Monte Carlo method for the solution of charge transport in semiconductors with applications to covalent materials," *Rev. Mod. Phys.*, vol. 55, no. 3, pp. 645–705, Jul. 1983.

[30] W. Shockley, "Hot electrons in germanium and Ohm's law," *Bell System Technical Journal*, vol. 30, no. 4, pp. 990–1034, 1951. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/j.1538-7305.1951.tb03692.x

[31] L. Landau and A. Kompanejez, *Phys. Z. Sowjetunion*, 1934.

[32] B. Davydov, "Über die Geschwindigkeitsverteilung der sich im Elektron," *Phys. Z. Sowjetunion*, 1936.

[33] ——, *Phys. Z. Sowjetunion*, 1937.

[34] T. Kurosawa, in *Proceedings of the International Conference on the Physics of Semiconductors*, 1966.

[35] N. P. Buslenko, D. I. Golenko, Y. A. Shreider, I. M. Sobol, and V. G. Sragovich, *The Monte Carlo Method: The Method of Statistical Trials*, 1966.

[36] H. A. Meyer, "Symposium on Monte Carlo methods," 1956.

[37] G. I. Marchuk, G. A. Mikhailov, M. A. Nazaraliev, R. A. Darbinjan, B. A. Kargin, and B. S. Elepov, *The Monte Carlo Methods in Atmospheric Optics*, 1980, ch. 11.

[38] H. D. Rees, "Calculation of steady state distribution functions by exploiting stability," *Phys. Lett. A*, vol. 26, no. 9, pp. 416 – 417, 1968. [Online]. Available: http://www.sciencedirect.com/science/article/pii/037596016890251X

[39] ——, "Calculation of distribution functions by exploiting the stability of the steady state," *J. Phys. Chem. Solids*, vol. 30, no. 3, pp. 643 – 655, 1969. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0022369769900183

[40] W. Fawcett, C. Hilsum, and H. D. Rees, "Effects of non-parabolicity on non-Ohmic transport in InAs," *Solid State Commun.*, vol. 7, no. 17, pp. 1257 – 1259, 1969. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0038109869901896

[41] W. Fawcett and H. D. Rees, "Electron population inversion in GaAs induced by high electric fields," *Phys. Lett. A*, vol. 28, no. 11, pp. 731 – 732, 1969. [Online]. Available: http://www.sciencedirect.com/science/article/pii/037596016990588X

[42] P. A. Lebwohl and P. J. Price, "Hybrid method for hot electron calculations," *Solid State Commun.*, vol. 9, no. 14, pp. 1221 – 1224, 1971. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0038109871900159

[43] J. G. Ruch, "Electron dynamics in short channel field-effect transistors," *IEEE Trans. Electron Devices*, vol. 19, no. 5, pp. 652–654, May 1972.

[44] G. Baccarani, C. Jacoboni, and A. M. Mazzone, "Current transport in narrow-base transistors," *Solid-State Electron.*, vol. 20, no. 1, pp. 5 – 10, 1977. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0038110177900260

[45] P. J. Price, "Transport properties of the semiconductor superlattice," *IBM J. Res. Dev.*, vol. 17, no. 1, pp. 39–46, Jan. 1973.

[46] J. R. Hauser, M. A. Littlejohn, and T. H. Glisson, "Velocity-field relationship of InAs-InP alloys including the effects of alloy scattering," *Appl. Phys. Lett.*, vol. 28, no. 8, pp. 458–461, 1976.

[47] W. Fawcett, A. D. Boardman, and S. Swain, "Monte Carlo determination of electron transport properties in gallium arsenide," *J. Phys. Chem. Solids*, vol. 31, no. 9, pp. 1963 – 1990, 1970. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0022369770900016

[48] W. Fawcett and E. G. S. Paige, "Negative differential mobility of electrons in germanium: A Monte Carlo calculation of the distribution function, drift velocity and carrier population in the (111) and (100) minima," *J. Phys. C: Solid State Phys.*, vol. 4, no. 13, pp. 1801–1821, Sep. 1971. [Online]. Available: https://doi.org/10.1088/0022-3719/4/13/031

[49] C. Canali, C. Jacoboni, F. Nava, G. Ottaviani, and A. Alberigi-Quaranta, "Electron drift velocity in silicon," *Phys. Rev. B*, vol. 12, pp. 2265–2284, Sep. 1975. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevB.12.2265

[50] C. Jacoboni, R. Minder, and G. Majni, "Effects of band non-parabolicity on electron drift velocity in silicon above room temperature," *J. Phys. Chem. Solids*, vol. 36, pp. 1129–1133, Oct. 1975.

[51] M. V. Fischetti and S. E. Laux, "Monte Carlo simulation of electron transport in Si: The first 20 years," in *Proc. 26th Eur. Solid State Device Research Conf.*, 1996.

[52] H. Shichijo and K. Hess, "Band-structure-dependent transport and impact ionization in GaAs," *Phys. Rev. B*, vol. 23, pp. 4197–4207, Apr. 1981. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevB.23.4197

[53] M. V. Fischetti and S. E. Laux, "Monte Carlo analysis of electron transport in small semiconductor devices including band-structure and space-charge effects," *Phys. Rev. B*, vol. 38, no. 14, pp. 9721–9745, Nov. 1988.

[54] T. Kunikiyo, M. Takenaka, Y. Kamakura, M. Yamaji, H. Mizuno, M. Morifuji, K. Taniguchi, and C. Hamaguchi, "A Monte Carlo simulation of anisotropic electron transport in silicon including full band structure and anisotropic impact-ionization model," *J. Appl. Phys.*, vol. 75, no. 1, pp. 297–312, 1994.

[55] P. D. Yoder, J. M. Higman, J. Bude, and K. Hess, "Monte Carlo simulation of hot electron transport in Si using a unified pseudopotential description of the crystal," *Semicond. Sci. Technol.*, vol. 7, no. 3B, pp. B357–B359, Mar. 1992. [Online]. Available: https://doi.org/10.1088/0268-1242/7/3b/092

[56] J. Bude and R. K. Smith, "Phase-space simplex Monte Carlo for semiconductor transport," *Semicond. Sci. Technol.*, vol. 9, no. 5S, pp. 840–843, May 1994. [Online]. Available: https://doi.org/10.1088/0268-1242/9/5s/119

[57] A. Abramo, L. Baudry, R. Brunetti, R. Castagne, M. Charef, F. Dessenne, P. Dollfus, R. Dutton, W. L. Engl, R. Fauquembergue, C. Fiegna, M. Fischetti, S. Galdin, N. Goldsman, M. Hackel, C. Hamaguchi, K. Hess, K. Hennacy, P. Hesto, and A. Yoshii, "A comparison of numerical solutions of the Boltzmann transport equation for high-energy electron transport silicon," *IEEE Trans. Electron Devices*, pp. 1646 – 1654, Oct. 1994.

[58] D. Chattopadhyay and H. J. Queisser, "Electron scattering by ionized impurities in semiconductors," *Rev. Mod. Phys.*, vol. 53, pp. 745–768, Oct. 1981. [Online]. Available: https://link.aps.org/doi/10.1103/RevModPhys.53.745

[59] H. Kosina and G. Kaiblinger-Grujin, "Ionized-impurity scattering of majority electrons in silicon," *Solid-State Electron.*, vol. 42, no. 3, pp. 331 – 338, 1998. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0038110197001998

[60] N. Takenaka, M. Inoue, and Y. Inuishi, "Influence of inter-carrier scattering on hot electron distribution function in GaAs," *J. Phys. Soc. Jpn.*, vol. 47, no. 3, pp. 861–868, 1979.

[61] M. V. Fischetti, "Effect of the electron-plasmon interaction on the electron mobility in silicon," *Phys. Rev. B.*, 1991.

[62] N. S. Mansour, K. Diff, and K. F. Brennan, "Effect of inclusion of self-consistently determined electron temperature on the electron-plasmon interaction," *J. Appl. Phys.*, vol. 80, no. 10, pp. 5770–5774, 1996.

[63] S. Bosi and C. Jacoboni, "Monte Carlo high-field transport in degenerate GaAs," *J. Phys. C: Solid State Phys.*, vol. 9, no. 2, pp. 315–319, Jan. 1976. [Online]. Available: https://doi.org/10.1088%2F0022-3719%2F9%2F2%2F017

[64] P. Lugli and D. K. Ferry, "Degeneracy in the ensemble Monte Carlo method for high-field transport in semiconductors," *IEEE Trans. Electron Devices*, vol. 32, no. 11, pp. 2431–2437, Nov. 1985.

[65] S. E. Laux and M. V. Fischetti, *The DAMOCLES Monte Carlo Device Simulation Program.* Boston, MA: Springer US, 1991, pp. 87–92. [Online]. Available: https://doi.org/10.1007/978-1-4757-2124-9_16

[66] J.-P. M. Péraud, N. G. Hadjiconstantinou, and C. D. Landon, "Monte Carlo and methods for and solving the and Boltzmann and transport equation," vol. 17, 2014.

[67] Perkins, *Introduction to the High Energy Physics.* Cambridge University Press, 2000.

[68] A. Danner, "An introduction to the empirical pseudopotential method," https://www.ece.nus.edu.sg/stfpage/eleadj/pseudopotential.htm, 2004, accessed: 2017-02-29.

[69] "MATLAB - MathWorks," https://www.mathworks.com/products/matlab.html, accessed: 2019-08-11.

[70] E. Conwell and V. F. Weisskopf, "Theory of impurity scattering in semiconductors," *Phys. Rev.*, vol. 77, no. 3, pp. 388–390, Feb. 1950.

[71] K. Tomizawa, *Numerical Simulation of Submicron Semiconductor Devices.* Artech House, 1993.

[72] P. Palestri, N. Barin, D. Esseni, and C. Fiegna, "Stability of self-consistent monte carlo simulations: effects of the grid size and of the coupling scheme," *IEEE Transactions on Electron Devices*, vol. 53, no. 6, pp. 1433–1442, 2006.

[73] K. Tomizawa, Y. Awano, N. Hashizume, and M. Kawashima, "Monte Carlo simulation of submicron GaAs n+-i(n)-n+ diode," *IEEE Proceedings I Solid State and Electron Devices*, vol. 129, no. 4, p. 131, 1982.

[74] M. A. Jaud, S. Barraud, J. Saint-Martin, A. Bournel, P. Dollfus, and H. Jaouen, "Pearson effective potential vs. multi-subband Monte-Carlo simulation for electron transport in DG nMOSFETs," in *Simulation of Semiconductor Processes and Devices 2007*, T. Grasser and S. Selberherr, Eds. Vienna: Springer Vienna, 2007, pp. 65–68.

[75] M. Jaud, S. Barraud, J. Saint-Martin, A. Bournel, P. Dollfus, and H. Jaouen, "A Pearson effective potential for Monte Carlo simulation of quantum confinement effects in nMOSFETs," *IEEE Trans. Electron Devices*, vol. 55, no. 12, pp. 3450–3458, Dec 2008.

[76] F. M. Bufler and L. Smith, "3D Monte Carlo simulation of FinFET and FDSOI devices with accurate quantum correction," *J. Comput. Electron.*, vol. 12, no. 4, pp. 651–657, Dec. 2013. [Online]. Available: http://dx.doi.org/10.1007/s10825-013-0518-z

[77] R. Venugopal, Z. Ren, S. Datta, M. S. Lundstrom, and D. Jovanovic, "Simulating quantum transport in nanoscale transistors: Real versus mode-space approaches," *Journal of Applied Physics*, vol. 92, no. 7, pp. 3730–3739, 2002.

[78] M. A. Elmessary, D. Nagy, M. Aldegunde, J. Lindberg, W. G. Dettmer, D. Peric, A. J. Garcia-Loureiro, and K. Kalna, "Anisotropic quantum corrections for 3-D finite-element Monte Carlo simulations of nanoscale multigate transistors," *IEEE Trans. Electron Devices*, vol. 63, no. 3, pp. 933–939, Mar. 2016.

[79] T. Sadi, E. Towie, M. Nedjalkov, C. Riddet, C. Alexander, L. Wang, V. Georgiev, A. Brown, C. Millar, and A. Asenov, "One-dimensional multi-subband Monte Carlo simulation of charge transport in Si nanowire transistors," in *2016 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD)*, Sep. 2016, pp. 23–26.

[80] J. Saint-Martin, A. Bournel, F. Monsef, C. Chassat, and P. Dollfus, "Multi sub-band Monte Carlo simulation of an ultra-thin double gate MOSFET with 2D electron gas," vol. 21, no. 4. IOP Publishing, Feb. 2006, pp. L29–L31. [Online]. Available: https://doi.org/110.1088/0268-1242/21/4/l01

[81] C. Sampedro, L. Donetti, F. Gámiz, A. Godoy, F. J. Garcia-Ruiz, V. P. Georgiev, S. M. Amoroso, C. Riddet, E. A. Towie, and A. Asenov, "3D multi-subband ensemble Monte Carlo simulator of FinFETs and nanowire transistors," in *2014 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD)*, Sep. 2014, pp. 21–24.

[82] L. Donetti, C. Sampedro, F. G. Ruiz, A. Godoy, and F. Gamiz, "3D multi-subband ensemble Monte Carlo simulation of ⟨100⟩ and ⟨110⟩ Si nanowire FETs," in *2018 Joint International EUROSOI Workshop and International Conference on Ultimate Integration on Silicon (EUROSOI-ULIS)*, Mar. 2018, pp. 1–4.

[83] "Eigen overview," http://eigen.tuxfamily.org/index.php?title=Main_Page, 2019, accessed: 2019-08-09.

[84] C. Lin, B. Greene, S. Narasimha, J. Cai, A. Bryant, C. Radens, V. Narayanan, B. Linder, H. Ho, A. Aiyar, E. Alptekin, J. An, M. Aquilino, R. Bao, V. Basker, N. Breil, M. Brodsky, W. Chang, L. Clevenger, D. Chidambarrao, C. Christiansen, D. Conklin, C. DeWan, H. Dong, L. Economikos, B. Engel, S. Fang, D. Ferrer, A. Friedman, A. Gabor, F. Guarin, X. Guan, M. Hasanuzzaman, J. Hong, D. Hoyos, B. Jagannathan, S. Jain, S. Jeng, J. Johnson, B. Kannan, Y. Ke, B. Khan, B. Kim, S. Koswatta, A. Kumar, T. Kwon, U. Kwon, L. Lanzerotti, H. Lee, W. Lee, A. Levesque, W. Li, Z. Li, W. Liu, S. Mahajan, K. McStay, H. Nayfeh, W. Nicoll, G. Northrop, A. Ogino, C. Pei, S. Polvino, R. Ramachandran, Z. Ren, R. Robison, I. Saraf, V. Sardesai, S. Saudari, D. Schepis, C. Sheraw, S. Siddiqui, L. Song, K. Stein, C. Tran, H. Utomo, R. Vega, G. Wang, H. Wang, W. Wang, X. Wang, D. Wehelle-Gamage, E. Woodard, Y. Xu, Y. Yang, N. Zhan, K. Zhao, C. Zhu, K. Boyd, E. Engbrecht, K. Henson, E. Kaste, S. Krishnan, E. Maciejewski, H. Shang, N. Zamdmer, R. Divakaruni, J. Rice, S. Stiffler, and P. Agnello, "High performance 14nm SOI FinFET CMOS technology with 0.0174m^2 embedded DRAM and 15 levels of Cu metallization," in *2014 IEEE International Electron Devices Meeting*, Dec. 2014, pp. 3.8.1–3.8.3.

[85] C. Sampedro, F. Gmiz, A. Godoy, R. Valn, A. Garca-Loureiro, N. Rodrguez, I. Tienda-Luna, F. Martinez-Carricondo, and B. Biel, "Multi-subband ensemble Monte Carlo simulation of bulk MOSFETs for the 32nm-node and beyond," *Solid-State Electronics*, vol. 65-66, pp. 88 – 93, 2011, selected Papers from the ESSDERC 2010 Conference. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0038110111002450

[86] L. Donetti, C. Sampedro, F. G. Ruiz, A. Godoy, and F. Gamiz, "Multi-subband ensemble Monte Carlo simulations of scaled GAA MOSFETs," in *2017 Joint International EUROSOI Workshop and International Conference on Ultimate Integration on Silicon (EUROSOI-ULIS)*, Apr. 2017, pp. 220–223.

[87] N. Fasarakis, A. Tsormpatzoglou, D. H. Tassis, I. Pappas, K. Papathanasiou, and C. A. Dimitriadis, "Analytical compact modeling of nanoscale triple-gate FinFETs," in *2012 16th IEEE Mediterranean Electrotechnical Conference*, Mar. 2012, pp. 72–75.

[88] F. Venturi, R. K. Smith, E. C. Sangiorgi, M. R. Pinto, and B. Ricco, "A general purpose device simulator coupling Poisson and Monte Carlo transport with applications to deep submicron MOSFETs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 8, no. 4, pp. 360–369, Apr. 1989.

[89] J.-P. M. Péraud and N. G. Hadjiconstantinou, "Adjoint-based deviational Monte Carlo methods for phonon transport calculations," *Phys. Rev. B*, vol. 91, no. 23, Jun. 2015.

[90] S. Mazumder and A. Majumdar, "Monte Carlo study of phonon transport in solid thin films including dispersion and polarization," *J. Heat Transfer*, vol. 123, no. 4, pp. 749–759, 2001.

[91] Y. Chen, D. Li, J. R. Lukes, and A. Majumdar, "Monte Carlo simulation of silicon nanowire thermal conductivity," *J. Heat Transfer*, vol. 127, no. 10, p. 1129, 2005.

[92] Y. S. Ju and K. E. Goodson, "Phonon scattering in silicon films with thickness of order 100 nm," *Applied Physics Letters*, vol. 74, no. 20, pp. 3005–3007, 1999.

[93] A. J. H. McGaughey and A. Jain, "Nanostructure thermal conductivity prediction by Monte Carlo sampling of phonon free paths," *Appl. Phys. Lett.*, vol. 100, no. 6, p. 061911, Feb. 2012.

[94] M. Verdier, R. Anufriev, A. Ramiere, K. Termentzidis, and D. Lacroix, "Thermal conductivity of phononic membranes with aligned and staggered lattices of holes at room and low temperatures," *Phys. Rev. B*, vol. 95, no. 20, May 2017.

[95] J.-P. M. Péraud and N. G. Hadjiconstantinou, "Efficient simulation of multidimensional phonon transport using energy-based variance-reduced Monte Carlo formulations," *Phys. Rev. B*, vol. 84, no. 20, Nov. 2011.

[96] J. Ma, X. Wang, B. Huang, and X. Luo, "Effects of point defects and dislocations on spectral phonon transport properties of wurtzite GaN," *J. Appl. Phys.*, vol. 114, no. 7, p. 074311, Aug. 2013.

[97] J. M. Ziman, *Electrons and Phonons*. Oxford University Press, 1960.

[98] W. Liu and M. Asheghi, "Phonon-boundary scattering in ultrathin single-crystal silicon layers," *Applied Physics Letters*, vol. 84, no. 19, pp. 3819–3821, 2004.

[99] K. Kukita, I. N. Adisusilo, and Y. Kamakura, "Impact of quasi-ballistic phonon transport on thermal properties in nanoscale devices: A Monte Carlo approach," in *2012 International Electron Devices Meeting*, Dec. 2012, pp. 17.5.1–17.5.4.

[100] A. A. Prokofiev, A. N. Poddubny, and I. N. Yassievich, "Phonon decay in silicon nanocrystals: Fast phonon recycling," *Phys. Rev. B*, vol. 89, p. 125409, Mar. 2014. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevB.89.125409

[101] T. Misawa, S. Oki, and Y. Awano, "Quasi self-consistent Monte Carlo particle simulations of local heating properties in nano-scale gallium nitride FETs," in *2013 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD)*, Sep. 2013, pp. 308–311.

[102] R. Sawabe, N. Ito, and Y. Awano, "Advanced quasi-self-consistent Monte Carlo simulations on high-frequency performance of nanometer-scale GaN HEMTs considering local phonon distribution," in *2017 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD)*, Sep. 2017, pp. 285–288.

[103] Q. Hao, H. Zhao, Y. Xiao, and M. B. Kronenfeld, "Electrothermal studies of GaN-based high electron mobility transistors with improved thermal designs," *Int. J. Heat Mass Transfer*, vol. 116, pp. 496–506, Jan. 2018.

[104] D. Nagy, M. A. Elmessary, M. Aldegunde, R. Valin, A. Martinez, J. Lindberg, W. G. Dettmer, D. Peric, A. J. Garcia-Loureiro, and K. Kalna, "3-D finite element Monte Carlo simulations of scaled Si SOI FinFET with different cross sections," *IEEE Trans. Nanotechnology*, vol. 14, no. 1, pp. 93–100, Jan. 2015.

[105] D. Vasileska, "Modeling self-heating in nanoscale devices," in *2015 IEEE 15th International Conference on Nanotechnology (IEEE-NANO)*, Jul. 2015, pp. 200–203.

[106] M. Shrivastava, M. Agrawal, S. Mahajan, H. Gossner, T. Schulz, D. K. Sharma, and V. R. Rao, "Physical insight toward heat transport and an improved electrothermal modeling framework for FinFET architectures," *IEEE Trans. Electron Devices*, vol. 59, no. 5, pp. 1353–1363, May 2012.

[107] C. W. Chang, S. E. Liu, B. L. Lin, C. C. Chiu, Y. H. Lee, and K. Wu, "Thermal behavior of self-heating effect in FinFET devices acting on back-end interconnects," in *2015 IEEE International Reliability Physics Symposium*, Apr. 2015.

[108] Y. Cengel, *Heat and Mass Transfer: Fundamentals and Applications*. McGraw Hill Education, 2011.

[109] Texas Instruments, "Calculating useful lifetimes of embedded processors," http://www.ti.com/lit/an/sprabx4a/sprabx4a.pdf, 2017, accessed: 2019-08-11.