Adaptive Simulation for Single-Electron Devices

Nicholas Allec*, Robert Knobel†, Li Shang‡

*ECE Department
Queen's University
Kingston, ON K7L 3N6, Canada
nicholas.allec@ece.queensu.ca

†Physics Department
Queen's University
Kingston, ON K7L 3N6, Canada
knobel@physics.queensu.ca

‡ECE Department
University of Colorado at Boulder
Boulder, CO 80309, U.S.A
li.shang@colorado.edu

Abstract

Single-electron devices have drawn much attention in the last two decades. They have been widely used for device research and also show promise as a potential alternative to complementary metal-oxide-semiconductor circuits due to their ultra low power dissipation. Three techniques have been used for single-electron device modeling in the past, including Monte Carlo simulation, master equation, and SPICE modeling. Among these, Monte Carlo method provides accuracy, but lacks the time efficiency required for large scale simulation. In this work, we introduce an adaptive multi-scale approach to single-electron device simulation using Monte Carlo method as basis, which significantly improves time efficiency while maintaining accuracy. We have shown it is possible to reduce simulation time up to 40 times and maintain an average error of 3.3% compared to non-adaptive Monte Carlo method. Going beyond simplistic approximations, we have modeled important secondary effects including cotunneling and Cooper pair tunneling, which are critical for device research.

I. Introduction

The current through single-electron devices is governed by the Coulomb blockade effect. At low bias voltages and especially at low temperatures, these devices can enter a Coulomb blockade region, where the tunneling of electrons is suppressed, giving them unique properties, for instance, a periodic current-voltage relationship. Metallic single-electron devices, such as a metallic single-electron transistor (SET) (shown in Fig. 1), can become superconducting at sufficiently low temperatures (SSET), and exhibit additional interesting and useful effects, such as Josephson quasi-particle (JQP) peaks.

At the device level, SETs are attractive because of their sensitivity and unique current-voltage characteristics. Applications include the readout of quantum computers [1], measurement of nanometre-scale movement [2], and Coulomb blockade thermometers [3]. On a large scale (i.e., circuit design level), as projected by International Technology Roadmap for Semiconductors, SETs can potentially achieve the lowest projected energy per switching event of any known computation technology (1×10^{-18} J) [4]. Therefore, SETs have a possibility of solving the power consumption challenge of mainstream complementary metal-oxide-semiconductor (CMOS) circuits. The types of single-electron devices that could find use in large scale circuitry include single-electron flip-flops and electron traps for memory [5], [6], and single-electron transistors for logic [7]–[9].

To characterize single-electron device and circuit behaviour, models and simulators are required. Previously, three methods have been used, which include SPICE approach, master equation (ME) approach, and Monte Carlo (MC) approach. Among the three simulation methods, SPICE is the most time efficient. However, existing SPICE models cannot model secondary effects and do not take into account device coupling [10]. Another disadvantage of SPICE is that only devices which have available models can be simulated, meaning that each new type of single-electron device will require a new model. For the ME approach, the probabilities that a circuit will take on a certain charge distribution are solved. The major disadvantage of this method is that the relevant states must be known before simulation, which is not always possible for large circuits since single-electron

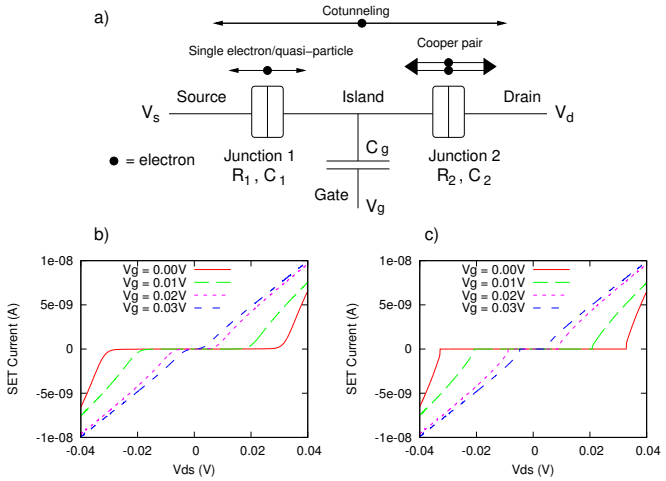

Fig. 1. a) Circuit schematic of a single-electron transistor (SET) showing several different tunneling scenarios that can occur in SETs. b) SEMSIM simulation results at $T = 5$K for various gate voltages of a SET with $R_1 = R_2 = 1\text{M}\Omega$, $C_1 = C_2 = 1$aF, $C_g = 3$aF, and a symmetric bias. The Coulomb blockade region can be seen as the suppression of current near $V_{ds} = 0V$. c) SEMSIM simulation results at $T = 50$mK for various gate voltages of a superconducting SET (SSET) with the same parameters as the non-superconducting SET with $\Delta(0\text{K}) = 0.2$meV and $T_c = 1.2$K. The suppressed current region is enlarged due to the superconducting gap, Δ.

device circuits can potentially occupy an infinite number of states. In MC based simulators, electron tunnel events are simulated to emulate the actual behaviour of electrons. Although this accurately captures the behaviour of electron transport, it requires long simulation times due to the detailed emulation of electron tunneling events.

In this work, our goal is to develop an accurate and efficient single-electron device simulator that can be used for both device research and large scale circuit design. To guarantee modeling accuracy required by device research, we choose MC method as the basis of our simulator and implement detailed secondary effects that include cotunneling and Cooper pair tunneling. The main challenge for developing a simulator targeting large scale circuits using the MC method is time efficiency. We propose an adaptive algorithm, which reduces simulation time significantly by reducing the number of calculations that are carried out after each tunneling event. Using this method, only the dynamic information, which has changed significantly, is updated selectively. This approach allows the investigation of circuit level behaviour in a feasible amount of time, which is not true for conventional MC single-electron device simulators, such as MOSES [8]. The developed adaptive multi-scale single-electron device simulator, called SEMSIM, is ready for public release for free academic and personal use.

II. Background

In this section we discuss background information associated with the single-electron effects modeled by SEMSIM. These effects include single-electron tunneling [11] and cotunneling [12] for non-superconducting devices, and Josephson quasi-particle peaks (JQP) [13], double Josephson quasi-particle peaks (DJQP)

978-3-9810801-3-1/DATE08 © 2008 EDAA

[13], and singularity matching features [14] for superconducting devices.

Single-electron tunneling: Single-electron devices are comprised of conducting (or semiconducting) regions called "islands" separated from leads or other islands by insulating tunnel barriers. These barriers are made sufficiently thin to allow electrons to tunnel through them, which ultimately leads to current flow.

When an island is connected to two leads separated by tunnel junctions, which is the case for SETs, the required energy to add (or remove) an electron from the island is $e^2/2C_\Sigma$, where C_Σ is the capacitance of the island. If this capacitance is sufficiently small, this charging energy can be much greater than the available thermal energy, k_BT. At bias voltages below a certain threshold, this leads to the suppression of electron tunneling. The suppression of current in the region where the bias voltage is below this threshold, is known as Coulomb blockade. Varying the potential on a third (or gate) terminal, which is capacitively coupled to the island, shifts the energy levels of the island, which in turn reduces or raises the threshold voltage. The threshold voltage dependence on gate voltage is a periodic function with period e/C_g.

Cotunneling: While a single-electron device is operating in the Coulomb blockade region, the tunneling of electrons to and from the island is suppressed, but the tunneling of electrons through multiple junctions is possible [12]. In this case, electrons travel through several junctions at once, instead of travelling through a single junction sequentially, and hence can avoid the cost in energy of charging the island. These "cotunneling" events become noticeable when operating in Coulomb blockade regions at low temperatures, providing non-zero current in these regions.

Cotunneling may be divided into elastic and inelastic events. For elastic cotunneling, the same electron travels through several junctions. For inelastic cotunneling, separate electrons tunnel through the junctions in a very short period of time, leaving electron-hole excitations in the intermediate islands. Since elastic cotunneling events are typically negligible (except at very low voltages and temperatures or the number of states on the islands are small) compared to inelastic events [5, 12], they are ignored in this work.

Superconductors: Superconducting materials exhibit certain properties below a material dependent critical temperature T_c. The most important property of these materials is that when they are cooled below T_c, a material dependant temperature, their resistance completely vanishes. This is due to the gap of 2Δ in their density of states. This gap is temperature dependent and decreases with increasing temperature up to T_c, where the gap is no longer in existence [15].

In superconductors, electrons from this gap near the Fermi level are bound in Cooper pairs, which travel without dissipation. The electrons travel in pairs due to their interaction with the material lattice. A simplistic explanation of this phenomenon is that traveling electrons attract the positively charged lattice atoms. The attracted wave of positively charged lattice atoms then attracts an electron in the vacinity. These electrons travel as pairs called Cooper pairs. As one travels, the other one follows due to lattice attractions. Thermal energy can break up these pairs into quasi-particles: single particle electron-like excitations.

In superconducting single-electron devices where all leads and islands are in the superconducting state, electrons can travel alone, as quasi-particles, or in pairs, as Cooper pairs. Quasi-particle tunneling is similar to single electron tunneling in the non-superconducting state except that the reduced current region for quasi-particles is extended due to the superconducting gap, Δ. Cooper pair tunneling, together with quasi-particle tunneling can lead to resonant current peaks below the gap.

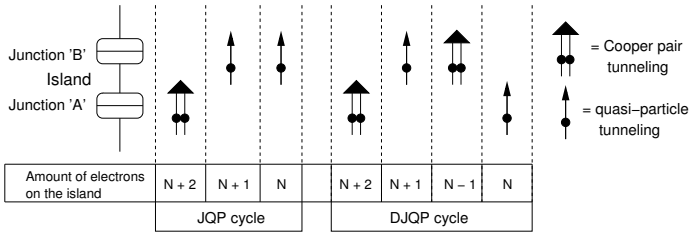

Fig. 2. JQP and DJQP cycle summary

These resonances are known as the Josephson quasi-particle (JQP) and double Josephson quasi-particle (DJQP) resonances.

Josephson quasi-particle peaks: Josephson quasi-particle peaks in SSETs occur at low bias voltages when the change in energy of the system for a Cooper pair tunneling event is approximately zero. After a Cooper pair tunnels to (or from) the island, either another Cooper pair can tunnel to return the island to its initial state, making the net current zero, or a quasi-particle can tunnel from (or to) the island to bring the system closer to its original state. If a quasi-particle tunnels after a Cooper pair does, then another quasi-particle tunnels from (or to) the island to return it to its original state. This process can happen repeatedly causing peaks in the current. The process is summarized in Fig. 2. Either a Cooper pair tunnels through junction 'A' and is followed by two quasi-particle tunneling events through junction 'B' [13], or the reverse.

Double Josephson quasi-particle peaks: Double Josephson quasi-particle processes are similar to JQP processes except that a Cooper pair tunneling event always follows a quasi-particle tunneling event (see Fig. 2). In this process, a Cooper pair tunnels through junction 'A', followed by a quasi-particle tunneling through junction 'B', followed by a Cooper pair tunneling through junction 'B',which is then followed by a quasi-particle tunneling through junction 'A'. The cycle then restarts, leading to resonant current peaks [13, 16]. Both of these effects are significant in superconducting devices because they enable significant current flow in regions that would be in the Coulomb blockade state if only single-particle effects were considered.

Singularity matching: Another interesting feature in low voltage biased SSETs is the appearance of peaks in the current due to singularity matching at finite temperatures ($0 < T < T_c$) [14]. These peaks appear when thermally excited quasi-particles in the singular density of states just above the superconducting gap line up with empty states on the other side of one of the tunnel junctions. At lower temperatures, these sub-gap peaks aren't visible since there are few excited quasi-particles. For a more complete description of singularity matching peaks, see [14, 17].

Other effects that appear in SSETs include parity effects [18], Andreev reflection [19], simultaneous tunneling of Cooper pair and quasiparticle tunneling ($3e$ tunneling) [20], quasiparticle cotunneling [21, 22], and supercurrent [23]. These effects are neglected in this simulator since their impact is only significant for a small range of parameters.

III. Simulator

In this section, we present SEMSIM, the proposed adaptive multi-scale simulator for single-electron devices.

III-A. Modeling

The assumptions of the single-electron tunneling model used in our simulator are based on the orthodox theory [11]. The tunneling rate of an electron through a junction is the probability per unit time that an electron will tunnel across it. The tunneling rate of an electron through a single junction follows [5, 11]:

$$\Gamma(\Delta W) = \frac{I(\Delta W/e)}{e[\exp^{\Delta W/(k_B T)} - 1]} \quad (1)$$

Where ΔW is the change in free energy from after and before the tunneling event; k_B is Boltzmann's constant; T is the absolute temperature; and for the non-superconducting case $I(\Delta W/e)$ is assumed to have the form $(\Delta W/e)/R$, where R is the resistance of the junction being tunneled through. Assuming that nodes i and f are both islands, the change in free energy for an electron to tunnel from i to f is [5]:

$$\Delta W = -e(v_f - v_i) + (C_{ii}^{-1} - 2C_{if}^{-1} + C_{ff}^{-1})e^2/2 \quad (2)$$

Where v_i and v_f are the initial voltages on nodes i and f; and C^{-1} is the n by n inverse capacitance matrix, which contains information on the coupling between nodes, where n is the number of islands.

While the orthodox theory ignores coherent simultaneous tunneling events [11], SEMSIM includes such events up to the second order. The cotunneling rates are calculated using the coexistence principle and associated cotunneling equations and assumptions presented by Fonseca et al. [24].

In the superconducting state, $I(\Delta W/e)$ in Equation 1 is the quasi-particle tunneling I-V function [12] given by [15]:

$$I_{qp} = \frac{G_{nn}}{e} \int_{-\infty}^{\infty} \frac{N_{s1}(E)}{N_1(0)} \frac{N_{s2}(E+eV)}{N_2(0)} [f(E) - f(E+eV)] dE \quad (3)$$

Where G_{nn} is the normal state conductance $(1/R)$, $N_{s1,2}(E,T)/N_{1,2}(0)$ are the BCS reduced densities of states of the superconducting metal on nodes 1 and 2 respectively, and $f(x)$ is the Fermi function. The reduced densities of states are given by:

$$\frac{N_{s1,2}(E,T)}{N_{1,2}(0)} = \frac{|E|}{\sqrt{E^2 - \Delta_{1,2}^2(T)}} \theta[|E| - \Delta_{1,2}(T)] \quad (4)$$

$$\theta[x] = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{otherwise} \end{cases}$$

Where $\Delta(T)$ is the temperature dependent superconducting energy gap.

Cooper pair tunneling is modeled for the regime of high resistance junctions. In this regime, $R_N \gg R_Q$, where R_N is the normal state resistance of the junction and $R_Q = h/4e^2 \simeq 6.5\text{k}\Omega$. Another assumption is that $E_J \ll E_c$, where E_c is the charging energy ($= e^2/2C_\Sigma$ for a SSET) and E_J is the Josephson energy [25].

With the inclusion of the superconducting effects mentioned, SEMSIM can simulate certain current-voltage properties witnessed in SSETs, such as resonant Josephson quasi-particle (JQP) peaks [13], double Josephson quasi-particle (DJQP) peaks [13], and singularity matching features [17].

III-B. Algorithm

In this section, a brief overview of the proposed simulator is given, followed by a detailed explanation of SEMSIM's process flow, which is shown in Fig. 3.

SEMSIM uses a Monte Carlo based method to simulate single-electron device circuits, which includes known secondary effects. The proposed adaptive algorithm reduces the amount of calculations required for each iteration of the Monte Carlo solver, thus making the simulator more efficient at the expense of accuracy.

Circuit information is passed to SEMSIM via an input file containing all the necessary information, which includes component values and simulation time. Once the circuit components have been characterized by the simulator, the Monte Carlo process begins. In this process, one tunnel event is simulated each iteration. In a typical simulation, many tunnel events are simulated to emulate circuit behaviour. At the beginning of each iteration, an adaptive solver is used to calculate the circuit node potentials and single-electron tunnel rate information. If secondary effects are included, or the circuit is superconducting, a non-adaptive solver is used to calculate the tunnel rate information specific to these effects. The tunnel rate information is then used by the event solver to choose a specific tunneling event each iteration. Once the desired simulation time has been reached, or the number of requested tunnel events has been satisfied, the simulation is complete and the results are stored in a file.

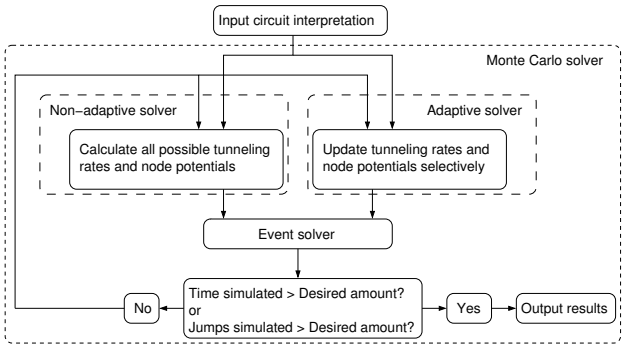

Fig. 3. Flow of the Monte Carlo method used in SEMSIM

Example Input File 1: Single-electron transistor

```
#SET component definitions
junc 1 1 4 1e-6 1e-18
junc 2 2 4 1e-6 1e-18
cap 3 4 3e-18
charge 4 0.0

#Input source information
vdc 1 0.02
vdc 2 -0.02
vdc 3 0.0
symm 1

#Overall node information
num_j 2
num_ext 3
num_nodes 4

#Simulation specific information
temp 5
cotunnel
record 1 2 2
jumps 100000 1
sweep 2 0.02 0.00005
```

Input circuit interpretation: SEMSIM supports a SPICE-like input format to ease device-level and large-scale circuit simulation. Circuits can contain superconducting or non-superconducting elements, but not both. User can specify the simulation parameters, such as the secondary effects being considered, and output methods. As an example, the input file to generate the results seen in Fig. 1 for the non-superconducting SET is presented in Example Input File 1. SEMSIM is also equipped with a parser which supports logic representation of circuit netlist, such as NAND and NOR network, allowing circuit designers to describe large-scale circuits.

Monte Carlo solver: The Monte Carlo process begins after the required circuit parameters have been extracted from the circuit information. The simulator then proceeds into an interative process, where a tunnel event is simulated each iteration, until the desired simulation time is met or the desired amount of tunnel events is satisfied. In typical simulations, many tunnel events are simulated to emulate circuit behaviour.

To emulate tunnel events, the tunneling rates of all possible tunnel events are calculated each iteration. Since the tunnel rate calculation is the most computationally expensive part of the simulation [5], in this work we propose an adaptive solver, which is used to update dynamic node potentials and tunnel

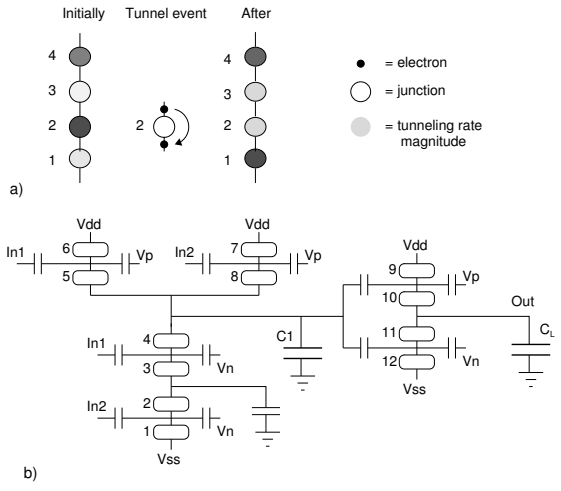

Fig. 4. a) Tunneling example of the circuit below. The tunneling rate magnitudes are shown in grayscale (darker shades refer to higher the tunneling rates) and the junctions are represented by circles. Only junctions that will have their tunnel rates recalculated are shown. b) Circuit schematic for an AND function using voltage state logic implemented with pSETs and nSETs.

rates. The tunnel rates are calculated by either an adaptive solver for single-electron tunnel events, or a non-adaptive solver for secondary or superconducting effects, if they are included. In each iteration, the event solver calculates the time between tunnel events and selects a particular tunnel event randomly, using the tunneling rate probabilities as a probability distribution.

Non-adaptive solver: Similar to conventional MC single-electron device solvers, the non-adaptive solver updates the potential of every single node and recalculates the tunneling rate of every junction at each iteration. This solver is used to calculate secondary or superconducting effects due to the complexity required for them to be solved using an adaptive method while maintaining accuracy. The equations used to calculate effect specific tunneling rates are those given in Section III-A. Since each tunneling rate equation requires the change in free energy for that tunnel event to be known, the change in free energy (Equation 2) must be calculated at least once per tunnel rate calculation (the cotunneling rate equation requires several energy values to be known).

Adaptive solver: In a non-adaptive MC single-electron device approach, the computation of the tunneling rates is the most time consuming part of the simulation [5]. To reduce the amount of tunnel rate calculations per loop, we devised a method to only update the tunnel rates and node potentials which have changed significantly after an electron jump or a significant change in the input voltage.

Algorithm 1 Adaptive analysis

if electron tunnel event occurred **or** AC signal(s) present **then**
  for each junction where a tunnel event occurred, in contact with AC input(s), and neighbour to be tested **do**
    Compute potential change for nodes $n1$ and $n2$ surrounding junction i
    $b(i) = b'(i) + \Delta P_{n1} - \Delta P_{n2}$
    if $(|b(i)| \geq \epsilon|\Delta W_{fw}^{'}|)$ or $(|b(i)| \geq \epsilon|\Delta W_{bw}^{'}|)$ **then**
      Flag junction i for tunneling rate recalculation.
      if junction i has not been visited this iteration **then**
        Go to 2 to test junction i's neighbours
      end if
    end if
  end for
end if
Compute tunneling rates for flagged junctions.

To determine whether or not the tunnel rate will change significantly after a tunneling event, the change in free energy due to a tunnel event (ΔW in Equation 1) is used, since it is the only dynamic parameter of the tunneling rate equation. For ΔW to vary, either the electron distribution of the circuit or the input voltages must change. These occurrences will only have a significant impact on ΔW if the junction nodes are tightly coupled to where the event took place. The adaptive algorithm is described below.

The adaptive algorithm is shown in Algorithm 1. After each tunnel event or change in input potential, the junctions nearest to the tunneling event and/or AC input(s) are each tested. The potential change across junction i, currently being tested, is calculated, where $n1$ and $n2$ are the nodes in contact with the junction, and ΔP_{n1} and ΔP_{n2} are the potential changes on nodes $n1$ and $n2$, respectively. The testing factor $b(i)$ is computed using the potential change across the junction and the accumulated testing factor, $b'(i)$, which has been accumulating since the last time this junction's tunneling rate was calculated. If the testing factor is larger than either the threshold value ϵ times the change in free energy of an electron tunneling forward or backward from the last time this junction's tunneling rate was computed, denoted by $\Delta W_{fw}^{'}$ and $\Delta W_{bw}^{'}$ respectively, this junction's tunneling rate must be recalculated and its neighbouring junctions must be put through the same testing process. Once all the junctions that require their tunneling rates recalculated have been found, the rates are computed and the Monte Carlo process continues.

Since the error of calculating the tunneling rates in this method is cumulative (accumulates every loop), all junction tunneling rates are recalculated periodically to assure the rates remain within a certain error range. Using this adaptive method, the number of tunneling rate calculations can be reduced significantly leading to time efficient MC simulations while maintaining accuracy.

To illustrate how the adaptive method works on a typical circuit, an example is shown in Fig. 4. Initially, the circuit in b) has a certain electronic configuration with a given amount of electrons on each island. Each junction has an associated tunneling rate for the current electronic configuration, which is shown in grayscale in a), where darker shades refer to higher tunneling rates. In the example, a tunnel event through junction 2 is chosen randomly. After the tunnel event, the potentials on the nodes where the electron tunneled to and from will change. The potentials on surrounding nodes will also slightly change due to the coupling between nodes. When the node potentials surrounding a junction change, the tunneling rate will also change (see Equation 2). In the non-adaptive method, all the junctions in the circuit would have their tunneling rates recalculated. In the adaptive method, however, only junctions 1 through 4 would have their tunneling rates recalculated since the change in node potentials surrounding all other junctions would have only increased by an insignificant amount after the tunnel event due to the large capacitance of the metal wire that connects the circuit stages, C1. Although in the example above it was assumed that C1 was sufficiently large to isolate the potential change due to a tunnel event, the adaptive algorithm determines which tunnel rates and node potentials require updating based on the actual circuit coupling values. If the devices and/or circuit stages cannot be considered isolated due to high coupling, they will not be treated as such.

Event solver: In Monte Carlo simulations, tunnel events are treated as independent events with a Poisson probability distribution. The time between tunneling events is found using:

$$\Delta t = -ln(r)/\Gamma_{sum} \tag{5}$$

where r is a random number with a uniform distribution between 0 and 1, and Γ_{sum} is the sum of the tunneling rates.

A slight variation of this method can be found in [5].

To decide which tunnel event will occur during an iteration of the Monte Carlo process, tunneling rates are used as a probability distribution and events are chosen randomly based on this distribution. This means that for each iteration, the tunnel events with the highest probabilities are most likely to occur.

IV. Results

In this section, we evaluate the accuracy and performance of SEMSIM. We first evaluate the accuracy of the simulator at the single device level, and then evaluate the accuracy and performance of the proposed adaptive technique using circuit level benchmarks. It is shown that the adaptive method is capable of decreasing running time by over 40 times compared to the non-adaptive method while mainting accuracy.

To evaluate the accuracy at the single device level, we compare SEMSIM's simulation results with actual experimental results, analytic approximations, and simulation results from other simulators. Specifically, we examine the second order effects presented in Section II for superconducting and non-superconducting devices.

For large scale circuit evaluation, 15 benchmark circuits are used to compare the simulation results of SEMSIM with those from a non-adaptive Monte Carlo approach and an analytical model in SPICE. To investigate the performance of the adaptive technique, simulation times from the three simulation methods are compared. For design metric accuracy evaluation, the propagation delays from SEMSIM and SPICE simulations are compared against the results from the non-adaptive approach, which are assumed to be the most accurate results.

This section is organized as follows. Section IV-A summarizes the experiment results for the single device cases. Section IV-B compares SEMSIM's adaptive approach with alternative methods to verify the accuracy and performance.

IV-A. Single Device Simulation

SEMSIM can model effects for both superconducting and non-superconducting single-electron devices. These effects include single-electron tunneling and second-order cotunneling for non-superconducting devices, and quasi-particle and Cooper pair tunneling for superconducting devices. With the inclusion of these effects, it is possible to simulate unique properties of superconducting devices, such as Josephson quasi-particle peaks [13], double Josephson quasi-particle peaks [13], and singularity matching features [17].

To validate our simulator in the simplest cases, using only single-electron tunneling, we tested the simulator's accuracy against experimental data [26] and simulation results from a SET SPICE model [27], and SIMON [28]. For testing the accuracy of cotunneling, simulations were tested against analytic approximations and SIMON results [29], and excellent agreement was observed.

For validation of superconducting device modeling, simulations were tested against experimental data in [25]. For these experiments the JQP peaks were compared and quantitative agreement was observed. To compare the qualitative features for superconducting devices, we ran simulations using a similar setup as the experiment from [17]. The setup includes testing a SSET at $T = 0.52$K, $R1 = R2 = 210\text{k}\Omega$, $C1 = C2 = 110$aF, $\Delta(0.52\text{K}) = 0.21$meV, $Cg = 14$aF, and a background charge, Q_b/e of 0.65. The current was simulated while the bias and gate voltages were swept. The results found from the experiment in the reference and a contour plot of our simulated results can be seen in Fig.5. The same features can be seen in the simulations that were found in the referenced experiment. The JQP features that correspond to open triangles on the left of Fig. 5 and the singularity matching features that correspond to solid

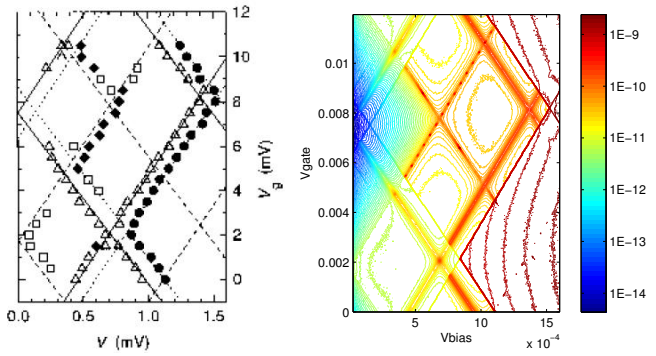

Fig. 5. Left: Experimental results taken from [17]. The lines represent theoretical feature positions. Following the reference, the dotted line is the threshold voltage, the dashed line is the singularity matching voltage, the solid line is the JQP voltage. The symbols represent features found experimentally. Following the reference, the solid circles represent the threshold voltage, the open triangles represent peaks, the solid diamonds represent singularity matching features, the open squares represent where the current stops ramping up and remains constant. Right: Contour plot of our simulation results for the same experiment, showing the current for different bias and gate voltages.

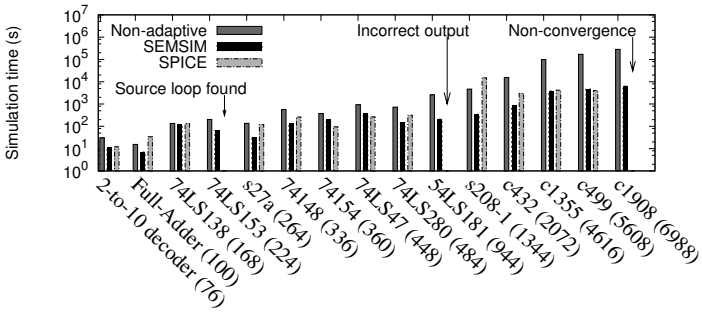

Fig. 6. Performance comparison

diamonds can clearly be seen in our contour plot. The change in current from rising to constant, represented by open squares, can also be seen in the contour plot in the low bias and gate voltage region. These results indicate that SEMSIM can simulate superconducting features found in actual experiments.

IV-B. Large Scale Circuit Simulation

Next, we evaluate the performance and accuracy of SEMSIM for large-scale SET circuit simulation. Experiments were conducted on a Linux workstation with a 2.66GHz Intel processor and 4GB memory. For the experiments, 15 logic benchmarks were tested that include circuits from ISCAS '85 and '89 [30] and other realistic large-scale logic circuits, and range in size from 76 junctions (38 SETs) to 6988 junctions (3494 SETs).

The logic benchmarks were converted into single-electron device circuits using CMOS interpretations of the logic circuits. To mimic the CMOS interpretations of the logic circuits, nSETs and pSETs were implemented. nSETs and pSETs are ordinary SETs with a second gate added that has a constant gate voltage, which shifts the current-voltage characteristic curve in a desired direction, allowing the SET to behave in a similar fashion to nMOS and pMOS transistors. An example circuit using this type of single-electron logic is shown at the bottom of Fig. 4. Note that the feasibility of this implementation is not relevant to its use in testing this simulator: it is used to give a realistic large-scale circuit for simulation, and other implementations can be simulated as well. The analytical SPICE model used in the experiments was an extended version of the model designed by Inokawa et al. [10], which allows for multiple gates that are used here to achieve pSET and nSET designs [27]. For circuit component values, a setup similar to [27] was used.

The relationship between the running time and the amount of

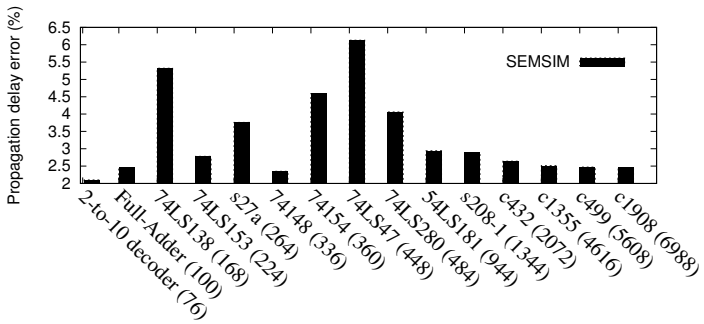

Fig. 7. Accuracy evaluation

junctions is shown in Fig. 6, where the different benchmarks are on the x-axis and the amount of junctions for each benchmark are enclosed in parentheses. Note that the SPICE results for benchmarks 74LS153, 54LS181, and c1908 are not shown since there were either non-convergence issues or incorrect logic outputs using SPICE to model these benchmarks. The running times for five of the larger benchmarks (c1908, c432, c1355, c499, and 54LS181) were extrapolated from shorter running times, and were adjusted for a circuit simulation time of 10μs. The adaptive method proved to be the most time efficient for the largest benchmark where the simulation time was reduced by over 40 times compared to the non-adaptive approach. This is due to the fact that the ratio of the total number of tunnel rate and node potential calculations solved for the adaptive approach over the total number number solved for the non-adaptive approach decreases as the number of junctions increases. The trend in Fig. 6 demonstrates that the adaptive method's efficiency increases with increasing amount of devices. From Fig. 6, it can also be seen that the adaptive simulation time is comparable to that of a SPICE simulation for the same circuit.

Fig. 7 shows the relationship between the amount of junctions and the error present in the propagation delay. Similar to Fig. 6, the different benchmarks are referred to along the x-axis with the number of junctions enclosed in parentheses. To calculate the propagation delay error, SEMSIM and SPICE results were compared to averaged non-adaptive MC simulation results. The averaged non-adaptive MC propagation delays were assumed to be the actual propagation delays. An average error of 9.18% was obtained from the SPICE results (excluding benchmarks 74LS153, 54LS181, and c1908, since there were either non-convergence issues or incorrect logic outputs using SPICE to model these benchmarks). Since SEMSIM is a MC simulator and the propagation delays from each run using different random seeds differ slighlty from each other, the propagation delay errors were calculated for nine different runs of SEMSIM and the average error for those nine runs is shown in Fig. 7. The results show that using an adaptive method, the accuracy is maintained within an acceptable range (average error of 3.30%).

V. Conclusions

In this paper, we have presented an adaptive technique for Monte Carlo simulation of single-electron devices. The proposed adaptive technique was validated against non-adaptive Monte Carlo and SPICE simulations using 15 logic benchmarks, which contain from 38 to 3494 devices. Simulation times, compared to the non-adaptive approach, were reduced by up to 40 times while the average error was 3.3%. Combining the adaptive technique with the secondary and superconducting effects handling allows multi-domain simulation where circuits with differing scales can be simulated using a single tool while maintaining scale specific design metric accuracy. The proposed techniques presented have been implemented as a software tool called SEMSIM, which will be publicly released for free academic and personal use.

Acknowledgment

This work is supported in part by the NSERC CGS scholarship, in part by the NSERC Discovery Grant #29815-04, and in part by the NSERC Discovery Grant #388694-01.

References

[1] Y. Nakamura, Y. A. Pashkin, and J. S. Tsai. Coherent control of macroscopic quantum states in a single-Cooper-pair box. *Nature*, 398:786–788, April 1999.
[2] R. G. Knobel and A. N. Cleland. Nanometer-scale displacement sensing using a single-electron transistor. *Nature*, 424:291–293, July 2003.
[3] J. P. Pekola, K. P. Hirvi, J. P. Kauppinen, and M. A. Paalanen. Thermometry by arrays of tunnel junctions. *Physics Review Ltrs.*, 73:2903–2906, November 1994.
[4] International Technology Roadmap for Semiconductors, 2005. http://public.itrs.net.
[5] C. Wasshuber. *Computational Single-Electronics*. Springer-Verlag/Wien, 2001.
[6] K. K. Yadavalli, A. O. Orlov, G. L. Snider, and A. N. Korotkov. Single electron memory devices: Toward background charge insensitive operation. *J. Vacuum Science Technology B Microelectronics and Nanometer Structures*, 21:2860–2864, 2003.
[7] J R Tucker. Complementary digital logic based on the Coulomb blockade. *J. Applied Physics*, 72(99):4399–4413, 1992.
[8] R. H. Chen, A. N. Korotkov, and K. K. Likharev. Single-electron transistor logic. *Applied Physics Ltrs.*, 68(14):1954–1956, April 1996.
[9] K Uchida, J Koga, R Ohba, and A Toriumi. Programmable single-electron transistor logic for future low-power intelligent LSI: proposal and room-temperature operation. *IEEE Trans. Electron Devices*, 50(7):1623–1630, July 2003.
[10] H. Inokawa and Y. Takahashi. A compact analytical model for asymmetric single-electron tunneling transistors. *IEEE Trans. Electron Devices*, 50(2):455–461, February 2003.
[11] K. K. Likharev. Single-electron devices and their applications. *Proc. IEEE*, 87(4):606–632, April 1999.
[12] T. Dittrich, P. Hanggi, G.-L. Ingold, B. Kramer, G. Schon, and W. Zwerger. *Quantum Transport and Dissipation, Ch. 3: Single-Electron Tunneling (G. Schon), p. 149-212*. Wiley-VCH Verlag, 1998.
[13] M. P. Blencowe, J. Imbers, and A. D. Armour. Dynamics of a nanomechanical resonator coupled to a superconducting single-electron transistor. *New Journal of Physics*, 7:236, November 2005.
[14] Y. Nakamura, A. N. Korotkov, C. D. Chen, and J. S. Tsai. Singularity-matching peaks in a superconducting single-electron transistor. *Phys. Rev. B*, 56:5116–5119, September 1997.
[15] M. Tinkham. *Introduction to Superconductivity*. Krieger Pub Co, 1975.
[16] A. A. Clerk, S. M. Girvin, A. K. Nguyn, and A. D. Stone. Resonant Cooper-pair tunneling: Quantum noise measurement characteristics. *Physics Review Ltrs.*, 89:176804–1–176804–4, October 2002.
[17] A. J. Manninen, Y. A. Pashkin, A. N. Korotkov, and J. P. Pekola. Observation of thermally excited charge transport modes in a superconducting single-electron transistor. *Europhys. Lett.*, 39:305–310, August 1997.
[18] G. Schon and A. Zaikin. Parity effects on electron tunneling through small superconducting islands. *Europhys. Lett.*, 26:650–700.
[19] R. Fitzgerald, S. Pohlen, and M. Tinkham. Observation of Andreev reflection in all-superconducting single-electron transistors. *Phys. Rev. B*, 57:R11073–R11076, May 1998.
[20] P. Hadley, E. Delvigne, E. H. Vissher, S. Lahteenmaki, and J. E. Mooij. $3e$ tunneling processes in a superconducting single-electron tunneling transistor. *Phys. Rev. B*, 58:15317–15320, December 1998.
[21] D. V. Averin, A. N. Korotkov, A. J. Manninen, and J. P. Pekola. Resonant tunneling through a macroscopic charge state in a superconducting single electron transistor. *Physics Review Ltrs.*, 78:4821–4824, June 1997.
[22] J. Siewert. Two-quasiparticle tunneling in all-superconducting single-electron transistors. *Europhys. Lett.*, 46:768–774, June 1999.
[23] T. M. Eiles and J. M. Martinis. Combined Joshephson and charging behaviour of the supercurrent in the superconducting single electron transistor. *Phys. Rev. B*, 50:627–630, July 1994.
[24] L. R. C. Fonseca, A. N. Korotkov, K. K. Likharev, and A. A. Odintsov. A numerical study of the dynamics and statistics of single electron systems. *J. Applied Physics*, 78:3238–3251, September 1995.
[25] Y. Nakamura, C. D. Chen, and J. S. Tsai. Quantitive analysis of Josephson-quasiparticle current in superconducting single-electron transistors. *Phys. Rev. B*, 53:8234–8237, April 1996.
[26] G. Dubejsky. Fabrication and characterization of single electron transistors. Master's thesis, Dept. of Physics, Engineering Physics & Astronomy, Queen's University, Kingston, August 2007.
[27] C. Zhu, Z. Gu, L. Shang, R. P. Dick, and R. G. Knobel. Towards an ultra-low-power architecture using single-electron tunneling transistors. In *Proc. Design Automation Conf.*, June 2007.
[28] C. Wasshuber, H. Kosina, and S. Selberherr. SIMON-a simulator for single-electron tunnel devices and circuits. *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, 16:937–944, September 1997.
[29] Example simulations with SIMON. http://www.lybrary.com/simon/examples.html.
[30] ISCAS Benchmarks. http://www.fm.vslib.cz/~kes/asic/iscas/.