Fast Surface Green's Function Calculations and Point Charge Implementation in ELEQTRONeX

Teo Lara[1] and Saurabh Sawant[2]

[1)]*Massachusetts Institute of Technology, Cambridge, MA*

[2)]*The Center for Computational Sciences and Engineering, Lawrence Berkeley National Laboratory, Berkeley, CA*

The Non-Equilibrium-Green's Function (NEGF) formalism has become widely adopted in quantum transport simulations, which are crucial in developing an understanding of carbon nanotube (CNT) operations. Our work builds on the ELEQTRONeX (Electrostatic Quantum Transport modeling Of Nanomaterials at eXascale) framework, a fast NEGF solver with GPU/MPI parallelization capabilities, in two major ways: we implement fast methods for computationally complex self-energy calculations and add external charge sources to allow for broader modeling applications. A particularly difficult aspect of NEGF simulations are self-energy computations, which generally require large matrix operations. We introduce two methods for self-energy calculations which are much less computationally expensive. Our methods are built on a tight-banding approximation, which allows for the condensation of probe Hamiltonians into their periodic layers. From our condensed Hamiltonian, we present an iterative and analytical approach to self-energy calculations, which we show to apply to arbitrarily complex probe matrices. Using randomly generated matrices, we demonstrate our methods are much more efficient than alternative direct inversions. We implement our methods into ELEQTRONeX, allowing for the handling of more complicated CNT geometries without sacrificing computational efficiency. Motivated by discrepancies with experimental results, we also implement point charges as external boundary conditions, which we demonstrate to have drastic impacts in our simulations. We demonstrate the impacts of point charges on single CNT simulations and more complex topgate setups that are common in experiments. By introducing a point charge density in gate oxide regions, we show simulations approaching experimental results, pointing to gate oxide impurities within topgate experiments.

I. INTRODUCTION

After the development of quantum mechanics in the early twentieth century, engineers have gradually developed devices on increasingly smaller scales. Over the last few decades, nanoscale instruments have become prominent, with one of their most notable applications being in transistors, the building blocks of modern electronics. Since their original fabrication,[1] carbon nanotube (CNT) transistors have seen much improvement in their performance, and we now possess both a theoretical and practical understanding of their operation.[2] Simulations play a critical role in both predicting these devices' behavior and developing more advanced experimental designs.

Due to their size, classical transistor modeling methods are not applicable to nanoscale devices, which are governed in large part, by quantum phenomena. The Non-Equilibrium Green's Function (NEGF) formalism has is a promising approach for these simulations.[3] In the past, it has been used for CNTs,[4] nanowires,[5] and even single molecules.[6] The NEGF formalism is extremely versatile, being able to describe time-dependent transport[7] and arbitrarily complex external conditions.[8] However, the method is computationally complex due to difficult matrix operations and a time-consuming iterative approach to solving the coupled NEGF and electrostatic equations. Traditional CPU implementations sometimes require hours to converge on a single operating point, even when deployed in parallel. The computational time further increases with efforts to model experimentally-relevant device geometries that may require simulations of multiple material channels to study their cross-talk, necessitating faster and more efficient NEGF implementations.

While traditional NEGF programs are parallelized on distributed CPU processors using message-passing-interfaces (MPI), there has been increasing efforts to further accelerate the computationally expensive routines on GPUs. The ELEQTRONeX (Electrostatic Quantum Transport modeling Of Nanomaterials at eXascale) framework, is a self-consistent NEGF implementation built on the AMReX (Adaptive Mesh Refinement at eXascale) library, which uses GPU/MPI parallelization across all components. Unlike previous methods, ELEQTRONeX uses Broyden's second method for self-consistency, greatly accelerating the iterative process. Because of AMReX's GPU-accelerated multigrid capabilities, intricate geometries can be represented as embedded boundary conditions. Although it has mostly been used for CNT modeling, the design of ELEQTRONeX allows for straightforward extensions to model a wide range of nanoscale devices.

An important part of CNT simulations is modeling the influence of electrical contacts onto the quantum device region. These influences, called "self-energies," are currently calculated semi-analytically in ELEQTRONeX, following the work of Guo et al.,[9] this method, while fast, limits the configurations of contacts, and in this paper we discuss the implementation of fast, more general methods for calculating self-energies into the ELEQTRONeX framework. We present two approaches, an implementation of the decimation algorithm of Lopez Sancho et al.,[10] and a computation using a Krylov subspace eigenfunction decomposition based on the presentation of Huand et al..[11] Through these additions, ELEQTRONeX can utilize both analytical and iterative approaches to calculating Σ_j self-energies, allowing for arbitrarily complex contacts of different material properties, defects in CNTs, and
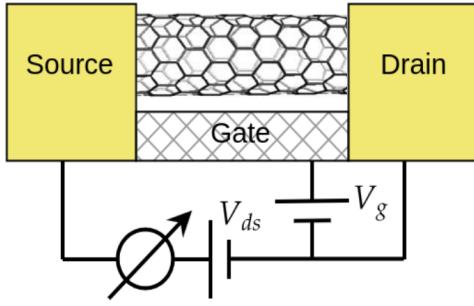
FIG. 1. A simple CNT device, featuring source, drain, gate, and CNT regions.

calculations of other nanoscale phenomena such as phonon propagation.[12]

We also implement point charges as external charge sources modifying our simulation boundary conditions. As boundary conditions are only calculated once during our simulation process, this addition does not impact the computational efficiency of ELEQTRONeX. Our motivation for adding point charges comes from discrepancies between simulated and experimental CNT results. In particular, we model topgate configurations, which are commonly used in CNT experiments,[13–16] and demonstrate the impact of point charge densities within gate oxide regions. Our work supports the hypothesis of impurities within gate oxide regions affecting experimental results.

This paper is organized as follows: §II describes the computational framework of ELEQTRONeX. §II A and §II B outline the existing methodology, and our implemented approaches to self-energy computation are discusses in §II D and §II E. Our addition of point charges is found in §II F. We evaluate the speed and accuracy of our self-energy calculation methods in §III A, followed by validation and comparison studies for our point charge implementation in §III B and §III C. In §IV, we discuss the outcomes of our work and future research directions.

II. COMPUTATIONAL APPROACH

We use ELEQTRONeX to model CNT field-effect transistor (CNTFET) devices, such as the one pictured in figure 1. This device is composed of a CNT conductor between a source and drain voltage lead with voltages V_s and V_d over a gate with voltage V_{gs}. Electrons travel through CNTs due to electrochemical potential differences between source and drain leads, V_{ds}, with their transport being impacted by the gate. Due to the nanoscale of these devices, the transport cannot be described classically, requiring a quantum description modeled with NEGF.

The ELEQTRONeX framework is composed of three major components: an electrostatic module, an NEGF module, and a coupling algorithm. The electrostatic module solves Poisson's Equation, computing electrostatic potential due to induced and external charges. Here, charge sources from de-

vices or boundary conditions are added, allowing for intricate setups of charge sources and drains. The NEGF module computes the inverse, solving for the induced charge within the CNT device region given an electrostatic potential. The charge calculated by NEGF in turn affects the potential, which impacts induced charge, so the system must the solved iteratively until self-consistency is achieved. In ELEQTRONeX, this iterative coupling is done efficiently and in parallel using Broyden's second algorithm. Once consistency is found, the current and conductance are calculated. The process is outlined schematically in figure 2.

In this paper, we focus mostly on the NEGF method, as it is the most relevant to our work. We summarise NEGF in §II A before discussing self-energy calculations.

A. Overview of NEGF Method

In the NEGF method for modeling quantum transport in nanodevices, the system is divided into a device region and two probe regions. These are described by a device Hamiltonian, $\boldsymbol{H}_D$ and left and right probe Hamiltonians, $\boldsymbol{H}_L$ and $\boldsymbol{H}_R$. Device-probe interactions are described with $\boldsymbol{\tau}_j$ operators. It is to the device Hamiltonian $\boldsymbol{H}_D$, that electrostatic potentials $\boldsymbol{U}$ representing charge distributions, are added. As we work on a large-scale, we omit many-body interactions. The Schrödinger equation, written in subspaces for each component is written:

$$\begin{pmatrix} \boldsymbol{H}_L & \boldsymbol{\tau}_L & \boldsymbol{0} \\ \boldsymbol{\tau}_L^\dagger & \boldsymbol{H}_D & \boldsymbol{\tau}_R^\dagger \\ \boldsymbol{0} & \boldsymbol{\tau}_R & \boldsymbol{H}_R \end{pmatrix} \begin{pmatrix} |\psi_L\rangle \\ |\psi_D\rangle \\ |\psi_R\rangle \end{pmatrix} = E \begin{pmatrix} |\psi_L\rangle \\ |\psi_D\rangle \\ |\psi_R\rangle \end{pmatrix} \qquad \text{(II.1)}$$

From equation II.1, the Green's function for the device is expressed as:

$$\boldsymbol{G}_D = (E\boldsymbol{I} - \boldsymbol{H}_D - \boldsymbol{\Sigma}_L - \boldsymbol{\Sigma}_R)^{-1} \qquad \text{(II.2)}$$

$$\boldsymbol{\Sigma}_j := \boldsymbol{\tau}_j^\dagger \boldsymbol{g}_j \boldsymbol{\tau}_j \qquad \text{(II.3)}$$

$$\boldsymbol{g}_j = (E\boldsymbol{I} - \boldsymbol{H}_j)^{-1} \qquad \text{(II.4)}$$

where E denotes the energy of the electron passing through the device. $\boldsymbol{g}_j$ are known as surface green's functions. $\boldsymbol{G}_D$ is used to calculate charge distribution matrix $\boldsymbol{\rho}$:

$$\boldsymbol{\rho} = 2 \times \frac{1}{2\pi} \int_{-\infty}^{\infty} \boldsymbol{G}_D \left(f_L \boldsymbol{\Gamma}_L + f_R \boldsymbol{\Gamma}_R \right) \boldsymbol{G}_D^\dagger dE \qquad \text{(II.5)}$$

$$\boldsymbol{\Gamma}_j := i \left(\boldsymbol{\Gamma}_j - \boldsymbol{\Gamma}_j^\dagger \right) \qquad \text{(II.6)}$$

The multiplication by 2 appears from two angular spin states of an electron. $f_j = f_j(E, \mu_j, T_j)$ is the probe-specific Fermi function, which depends on the electrostatic potential μ_j and the temperature T_j. While we may vary μ_j between probes, we maintain $T_L = T_R$ for simplicity. In practice, to avoid singularities, we perform the transformation: $E \to E + \delta i$, for a small δ, and carry out a contour integral over the imaginary place as discussed in Taylor et al.[17] Once consistency is
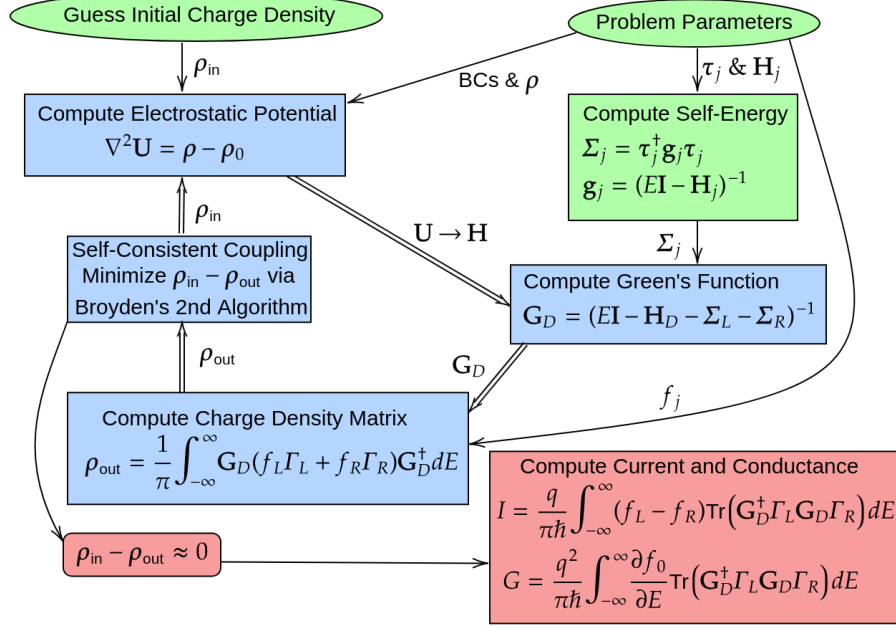
FIG. 2. Workflow for quantum transport simulations

achieved, the current is calculated as:

$$I = \frac{q}{\pi\hbar} \int\limits_{-\infty}^{\infty} (f_L - f_R) \, \mathrm{Tr} \left(G_D^\dagger \Gamma_L G_D \Gamma_R \right) dE \qquad (\text{II.7})$$

Defining f_0 as the Fermi function at equilibrium, the conductance G can also be found:

$$G = \frac{q^2}{\pi\hbar} \int\limits_{-\infty}^{\infty} \frac{\partial f_0}{\partial E} \, \mathrm{Tr} \left(G_D^\dagger \Gamma_L G_D \Gamma_R \right) dE \qquad (\text{II.8})$$

We derive equations II.5-II.8 in §V A.

B. Numerical Description of NEGF Hamiltonians

A typical system used in ELEQTRONeX is illustrated in figure 3. Device and probe Hamiltonians are divided into planes, with each plane consisting of n atoms. The device Hamiltonian consists of N planes, with interactions within atoms of plane j being described by α_j matrices, and interactions between plane j and $j+1$ being described by interaction matrices τ_j. We utilize a nearest-neighbor interaction approximation, such that layers only affect adjacent ones, eliminating the need for other interaction terms. In this scheme, the right probe Hamiltonian can be written in tridiagonal block form:

$$H_R = \begin{pmatrix} \alpha_{N+1} & \tau_{N+1} & 0 & \cdots \\ \tau_{N+1}^\dagger & \alpha_{N+2} & \tau_{N+2} & \cdots \\ 0 & \tau_{N+2}^\dagger & \alpha_{N+3} & \ddots \\ \vdots & \vdots & \ddots & \ddots \end{pmatrix} \qquad (\text{II.9})$$
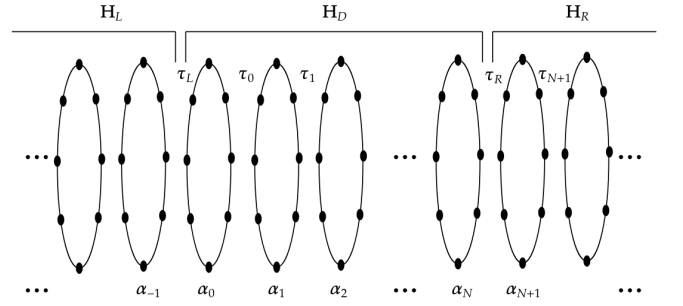

FIG. 3. Schematic for probe-device-probe simulation. The system consists of a device Hamiltonian H_D, and probe Hamiltonians H_L and H_R. The system is divided into atomic planes as illustrated.

Without loss of generality, in this and all following sections, we consider only the right probe; all described approaches can also be applied to the left side. Naturally, surface Green's functions are also written as full matrices:

$$g_R = \begin{pmatrix} g_{0,0} & g_{0,1} & g_{0,2} & g_{0,3} \\ g_{1,0} & g_{1,1} & g_{1,2} & g_{1,3} \\ g_{2,0} & g_{2,1} & g_{2,2} & \cdots \\ g_{3,0} & g_{3,1} & \cdots & \cdots \end{pmatrix} \qquad (\text{II.10})$$

From our nearest-neighbor approximation, interaction terms between probes and devices have only one non-zero block term, appearing in the top right corner for τ_R. So, for calculating self energies, defined in equation II.4, only the top left corner element of g_R is needed, all other terms are superfluous. In the case of the left probe, the bottom right corner block would be needed.

C. Hamiltonian Matrix Condensation

We assume that probes are periodic, such that after P atomic planes layers repeat, allowing us to write:

$$\boldsymbol{H}_{R\,j,k} = \boldsymbol{H}_{R\,nP+j,nP+k}$$
$$(j = 0,1,\ldots,P; j = k, k \pm 1; n = 1,2,\ldots) \tag{II.11}$$

Using this fact, we write the probe Hamiltonian using new block matrices:

$$\boldsymbol{H}_R = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} & \boldsymbol{0} & \cdots \\ \boldsymbol{B}^\dagger & \boldsymbol{C} & \boldsymbol{D} & \cdots \\ \boldsymbol{0} & \boldsymbol{D}^\dagger & \boldsymbol{A} & \ddots \\ \vdots & \vdots & \ddots & \ddots \end{pmatrix} \tag{II.12}$$

With:

$$\boldsymbol{A} := \boldsymbol{H}_{R\,0,0} = \boldsymbol{\alpha}_{N+1} \tag{II.13}$$

$$\boldsymbol{B} := \begin{pmatrix} \boldsymbol{H}_{R\,0,1} & \boldsymbol{0} & \boldsymbol{0} & \cdots \end{pmatrix} = \begin{pmatrix} \boldsymbol{\tau}_{N+1} & \boldsymbol{0} & \boldsymbol{0} & \cdots \end{pmatrix} \tag{II.14}$$

$$\boldsymbol{D} := \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ \vdots \\ \boldsymbol{H}_{R\,P-1,P} \end{pmatrix} = \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ \vdots \\ \boldsymbol{\tau}_{N+P} \end{pmatrix} \tag{II.15}$$

$$\boldsymbol{C} := \begin{pmatrix} \boldsymbol{H}_{R\,1,1} & \boldsymbol{H}_{R\,1,2} & \cdots & \boldsymbol{0} \\ \boldsymbol{H}_{R\,2,1}^\dagger & \boldsymbol{H}_{R\,2,2} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{H}_{R\,P-1,P-1} \end{pmatrix}$$
$$= \begin{pmatrix} \boldsymbol{\alpha}_{N+2} & \boldsymbol{\tau}_{N+2} & \cdots & \boldsymbol{0} \\ \boldsymbol{\tau}_{N+2}^\dagger & \boldsymbol{\alpha}_{N+3} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{\alpha}_{N+P} \end{pmatrix} \tag{II.16, II.17}$$

Calculating the surface Green's functions:

$$\begin{pmatrix} \boldsymbol{EI} - \boldsymbol{A} & -\boldsymbol{B} & \boldsymbol{0} & \cdots \\ -\boldsymbol{B}^\dagger & \boldsymbol{EI} - \boldsymbol{C} & -\boldsymbol{D} & \cdots \\ \boldsymbol{0} & -\boldsymbol{D}^\dagger & \boldsymbol{EI} - \boldsymbol{A} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \begin{pmatrix} \boldsymbol{g}_{0,0} \\ \boldsymbol{g}_{1\sim P-1,0} \\ \boldsymbol{g}_{P0} \\ \boldsymbol{g}_{P+1\sim 2P-1,0} \\ \vdots \end{pmatrix} = \begin{pmatrix} \boldsymbol{I} \\ \boldsymbol{0} \\ \boldsymbol{0} \\ \vdots \end{pmatrix} \tag{II.18}$$

where $\boldsymbol{g}_{1\sim P-1,0}$ represent the Green's surface functions from $\boldsymbol{g}_{10}$ to $\boldsymbol{g}_{P-1,0}$ in a column. We eliminate non-periodic terms using:

$$\begin{pmatrix} \boldsymbol{EI} - \boldsymbol{\Xi}_s & -\boldsymbol{\Pi} & \boldsymbol{0} & \cdots \\ -\boldsymbol{\Pi}^\dagger & \boldsymbol{EI} - \boldsymbol{\Xi} & -\boldsymbol{\Pi} & \cdots \\ \boldsymbol{0} & -\boldsymbol{\Pi}^\dagger & \boldsymbol{EI} - \boldsymbol{\Xi} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \begin{pmatrix} \boldsymbol{g}_{0,0} \\ \boldsymbol{g}_{P,0} \\ \boldsymbol{g}_{2P,0} \\ \vdots \end{pmatrix} = \begin{pmatrix} \boldsymbol{I} \\ \boldsymbol{0} \\ \boldsymbol{0} \\ \vdots \end{pmatrix} \tag{II.19}$$

With:

$$\boldsymbol{\Xi}_s := \boldsymbol{A} + \boldsymbol{B}(\boldsymbol{EI} - \boldsymbol{C})^{-1}\boldsymbol{B}^\dagger \tag{II.20}$$

$$\boldsymbol{\Xi} := \boldsymbol{A} + \boldsymbol{B}(\boldsymbol{EI} - \boldsymbol{C})^{-1}\boldsymbol{B}^\dagger + \boldsymbol{D}(\boldsymbol{EI} - \boldsymbol{C})^{-1}\boldsymbol{D}^\dagger \tag{II.21}$$

$$\boldsymbol{\Pi} := \boldsymbol{B}(\boldsymbol{EI} - \boldsymbol{C})^{-1}\boldsymbol{D} \tag{II.22}$$

From equation II.19, we identify a condensed Hamiltonian, consisting of only repeated periodic regions.

$$\boldsymbol{H}_{R\,\text{cond}} := \begin{pmatrix} \boldsymbol{\Xi}_s & \boldsymbol{\Pi} & \boldsymbol{0} & \cdots \\ \boldsymbol{\Pi}^\dagger & \boldsymbol{\Xi} & \boldsymbol{\Pi} & \cdots \\ \boldsymbol{0} & \boldsymbol{\Pi}^\dagger & \boldsymbol{\Xi} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \tag{II.23}$$

$\boldsymbol{\Xi}$ terms describe the interactions within periodic regions and $\boldsymbol{\Pi}$ terms couple regions to each other. The only computationally burdensome aspect of this condensation is an inversion of $\boldsymbol{EI} - \boldsymbol{C}$. But as this this matrix is both sparse and of size $(P-1)n \times (P-1)n$, this inversion is generally fast. However, we can acceleration this inversion further by recognizing from equations II.20-II.22 that we only need the corner elements of this matrix since $\boldsymbol{B}$ has only one non-zero block. We find these elements recursively as described in §V B. This recursive calculation is more computationally efficient than a matrix inversion.

D. Iterative Approach

As discussed in §II B, due to our tight-banding approximation, the only required element of $\boldsymbol{g}_R$ is $\boldsymbol{g}_{0,0}$, the top left block. Inverting $\boldsymbol{g}_R$ is expensive, and $\boldsymbol{g}_{0,0}$ can be found more efficiently through an iterative technique. Here, we apply the decimation algorithm[10] to our condensed Hamiltonian.

Using equation II.19, we extract two equations:

$$(\boldsymbol{EI} - \boldsymbol{\Xi}_s)\boldsymbol{g}_{0,0} = \boldsymbol{I} + \boldsymbol{\Pi}\boldsymbol{g}_{P,0} \tag{II.24}$$

$$(\boldsymbol{EI} - \boldsymbol{\Xi})\boldsymbol{g}_{jP,0} = \boldsymbol{\Pi}^\dagger\boldsymbol{g}_{(j-1)P,0} + \boldsymbol{\Pi}\boldsymbol{g}_{(j+1)P,0} \tag{II.25}$$

Let us define:

$$\boldsymbol{\mu}_0 := \boldsymbol{EI} - \boldsymbol{\Xi}_s \tag{II.26}$$

$$\boldsymbol{\nu}_0 := \boldsymbol{EI} - \boldsymbol{\Xi} \tag{II.27}$$

$$\boldsymbol{\gamma}_0 := \boldsymbol{\Pi} \tag{II.28}$$

We rewrite equations II.24-II.25:

$$\boldsymbol{\mu}_0\boldsymbol{g}_{0,0} = \boldsymbol{I} + \boldsymbol{\gamma}_0\boldsymbol{g}_{P,0} \tag{II.29}$$

$$\boldsymbol{\nu}_0\boldsymbol{g}_{jP,0} = \boldsymbol{\gamma}_0^\dagger\boldsymbol{g}_{(j-1)P,0} + \boldsymbol{\gamma}_0\boldsymbol{g}_{(j+1)P,0} \tag{II.30}$$

Using equation II.30:

$$\boldsymbol{g}_{(2j+1)P,0} = \boldsymbol{\nu}_0^{-1}\left(\boldsymbol{\gamma}_0^\dagger\boldsymbol{g}_{2jP,0} + \boldsymbol{\gamma}_0\boldsymbol{g}_{(2j+2)P,0}\right) \tag{II.31}$$

With equation II.31 we write:

$$\boldsymbol{\mu}_1\boldsymbol{g}_{0,0} = \boldsymbol{I} + \boldsymbol{\gamma}_1\boldsymbol{g}_{2P,0} \tag{II.32}$$

$$\boldsymbol{\nu}_1\boldsymbol{g}_{2jP,0} = \boldsymbol{\gamma}_1\boldsymbol{g}_{(2j-2)P,0} + \boldsymbol{\zeta}_1\boldsymbol{g}_{(2j+2)P,0} \tag{II.33}$$

Using:

$$\boldsymbol{\mu}_1 := \boldsymbol{\mu}_0 - \boldsymbol{\gamma}_0\boldsymbol{\nu}_0^{-1}\boldsymbol{\gamma}_0^\dagger \tag{II.34}$$

$$\boldsymbol{\nu}_1 := \boldsymbol{\nu}_0 - \boldsymbol{\gamma}_0\boldsymbol{\nu}_0^{-1}\boldsymbol{\gamma}_0^\dagger - \boldsymbol{\gamma}_0^\dagger\boldsymbol{\nu}_0^{-1}\boldsymbol{\gamma}_0 \tag{II.35}$$

$$\boldsymbol{\gamma}_1 := \boldsymbol{\gamma}_0\boldsymbol{\nu}_0^{-1}\boldsymbol{\gamma}_0 \tag{II.36}$$

$$\boldsymbol{\zeta}_1 := \boldsymbol{\gamma}_0^\dagger\boldsymbol{\nu}_0^{-1}\boldsymbol{\gamma}_0^\dagger \tag{II.37}$$

We repeat this process k times:

$$\boldsymbol{\mu}_k \boldsymbol{g}_{0,0} = \boldsymbol{I} + \boldsymbol{\gamma}_k \boldsymbol{g}_{2^k P,0} \tag{II.38}$$

$$\boldsymbol{\nu}_k \boldsymbol{g}_{2^k jP,0} = \boldsymbol{\gamma}_k \boldsymbol{g}_{2^k(j-1)P,0} + \boldsymbol{\zeta}_k \boldsymbol{g}_{2^k(j+1)P,0} \tag{II.39}$$

With:

$$\boldsymbol{\mu}_k := \boldsymbol{\mu}_{k-1} - \boldsymbol{\zeta}_{k-1} \boldsymbol{\nu}_{k-1}^{-1} \boldsymbol{\gamma}_{k-1} \tag{II.40}$$

$$\boldsymbol{\nu}_k := \boldsymbol{\nu}_{k-1} - \boldsymbol{\zeta}_{k-1} \boldsymbol{\nu}_{k-1}^{-1} \boldsymbol{\gamma}_{k-1} - \boldsymbol{\gamma}_{k-1} \boldsymbol{\nu}_{k-1}^{-1} \boldsymbol{\zeta}_{k-1} \tag{II.41}$$

$$\boldsymbol{\gamma}_k := \boldsymbol{\gamma}_{k-1} \boldsymbol{\nu}_{k-1}^{-1} \boldsymbol{\gamma}_{k-1} \tag{II.42}$$

$$\boldsymbol{\zeta}_k := \boldsymbol{\zeta}_{k-1} \boldsymbol{\nu}_{k-1}^{-1} \boldsymbol{\zeta}_{k-1} \tag{II.43}$$

The key idea is that as $k \to \infty$, $\boldsymbol{\gamma}_k \to 0$, since $\boldsymbol{\kappa}_k$ effectively describes interactions between layers that are further and further apart. As distance becomes infinite, and the interaction strength goes to zero, we find:

$$\boldsymbol{g}_{0,0} \approx \boldsymbol{\mu}_k^{-1} \tag{II.44}$$

E. Eigenfunction Decomposition Approach

A second method for Green's surface function calculations is passed on a decomposition in eigenvalue space. The key idea is based on the Bloch wave condition,[18] applicable to our condensed Hamiltonian from the second layer onwards.

$$\begin{aligned} \boldsymbol{H}_{R\,\text{cond}\,j,j} &= \boldsymbol{\Xi}\,(j \neq 0) \\ \boldsymbol{H}_{R\,\text{cond}\,j,k} &= \boldsymbol{\Pi}\,(j = k \pm 1) \end{aligned} \tag{II.45}$$

By omitting the first layer of $\boldsymbol{H}_{R\,\text{cond}}$, we define an entirely periodic semi-inifite chain satisfying the eigenfunction relationship:

$$\boldsymbol{H}_{R\,\text{cond}}(p) = \boldsymbol{H}_{R\,\text{cond}}(p+P) \to |\psi_j(p+P)\rangle = \lambda\,|\psi_j(p)\rangle \tag{II.46}$$

where $\lambda = e^{ijd}$. Using our condensed Hamiltonian (equation II.23), and our Bloch wave condition, equation II.46:

$$-\lambda_j^{-1} \boldsymbol{\Pi}^\dagger |\psi_j(p)\rangle + \lambda_j^{-1}(E\boldsymbol{I} - \boldsymbol{\Xi})|\psi_j(p+P)\rangle \\ - \boldsymbol{\Pi}|\psi_j(p+P)\rangle = 0 \tag{II.47}$$

And writing equations II.46 and II.47 in matrix form:

$$\begin{aligned} &\begin{pmatrix} \boldsymbol{0} & \boldsymbol{I} \\ -\boldsymbol{\Pi}^\dagger & E\boldsymbol{I} - \boldsymbol{\Xi} \end{pmatrix} \begin{pmatrix} |\psi_j(p)\rangle \\ |\psi_j(p+P)\rangle \end{pmatrix} \\ &= \lambda_j \begin{pmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Pi} \end{pmatrix} \begin{pmatrix} |\psi_j(p)\rangle \\ |\psi_j(p+P)\rangle \end{pmatrix} \end{aligned} \tag{II.48}$$

Now, we define a surface Green's function, $\boldsymbol{g}'$, for the second layer onward, which should satisfy equation II.19:

$$(E\boldsymbol{I} - \boldsymbol{\Xi})\boldsymbol{g}'_{P,0} = \boldsymbol{I} + \boldsymbol{\Pi}\boldsymbol{g}'_{2P,0} \tag{II.49}$$

$$(E\boldsymbol{I} - \boldsymbol{\Xi})\boldsymbol{g}'_{jP,0} = \boldsymbol{\Pi}^\dagger \boldsymbol{g}'_{(j-1)P,0} + \boldsymbol{\Pi}\boldsymbol{g}'_{(j+1)P,0} \tag{II.50}$$

Expanding $\boldsymbol{g}'_{P,0}$ in terms of eigenfunctions:

$$\boldsymbol{g}'_{P,0} = \boldsymbol{U}\boldsymbol{C} \tag{II.51}$$

where $\boldsymbol{U}$ contains normalized eigenfunctions, and $\boldsymbol{C}$ holds coefficients. Due to the periodicity of eigenfunctions established in equation II.46, we write:

$$\boldsymbol{g}'_{2P,0} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{C} \tag{II.52}$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix with the elements $\boldsymbol{\Lambda}_{jj} = \lambda_j$. Note that we only use eigenfunctions with $|\lambda| < 1$ to ensure wavefunction normalization. So $\boldsymbol{U}$ is size $n \times M$ for M eigen-funfctions corresponding to $|\lambda| < 1$. We relate Green's surface functions between planes:

$$\boldsymbol{g}'_{2P,0} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{C} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}\boldsymbol{U}\boldsymbol{C} = \boldsymbol{U}\boldsymbol{\Lambda}\tilde{\boldsymbol{U}}\boldsymbol{g}'_{P,0} = \boldsymbol{F}\boldsymbol{g}'_{P,0} \tag{II.53}$$

$$\boldsymbol{F} := \boldsymbol{U}\boldsymbol{\Lambda}\tilde{\boldsymbol{U}} \tag{II.54}$$

$\boldsymbol{U}^{-1}$ does not exist, as in general $\boldsymbol{U}$ will not be a square. We utilize the pseudo inverse, factorizing $\boldsymbol{U} = \boldsymbol{Q}\boldsymbol{R}$ and writing $\boldsymbol{R}\tilde{\boldsymbol{U}} = \boldsymbol{Q}^\dagger$ Applying equation II.53 between any two layers yeilds:

$$\boldsymbol{g}'_{(n+1)P,0} = \boldsymbol{F}\boldsymbol{g}'_{nP,0} \tag{II.55}$$

Using equation II.55 in our Green's functions relationships, equations II.49 and II.50, yields:

$$(E\boldsymbol{I} - \boldsymbol{\Xi} - \boldsymbol{\Pi}\boldsymbol{F})\boldsymbol{g}'_{P,0} = \boldsymbol{I} \tag{II.56}$$

$$(E\boldsymbol{I} - \boldsymbol{\Xi} - \boldsymbol{\Pi}\boldsymbol{F})\boldsymbol{F}\boldsymbol{g}'_{P,0} = \boldsymbol{\Pi}^\dagger \boldsymbol{g}'_{P,0} \tag{II.57}$$

This is solved readily, to find:

$$\boldsymbol{g}'_{P,0} = \boldsymbol{F}\boldsymbol{\Pi}^{\dagger-1} \tag{II.58}$$

And solving for self-energy, which eliminates the need to compute $\boldsymbol{\Pi}^{\dagger-1}$:

$$\boldsymbol{\Sigma}' = \boldsymbol{\Pi}\boldsymbol{g}'_{P,0}\boldsymbol{\Pi}^\dagger = \boldsymbol{\Pi}\boldsymbol{F} \tag{II.59}$$

To find the original surface Green's function $\boldsymbol{g}_{0,0}$, we treat $\boldsymbol{g}'_{P,0}$ as a surface Green's function attaching the first layer with the periodic semi-infinite probe, to compute:

$$\boldsymbol{g}_{0,0} = (E\boldsymbol{I} - \boldsymbol{\Xi}_s - \boldsymbol{\Sigma}')^{-1} \tag{II.60}$$

F. Numerical Addition of External Charge Sources

Point charges are added as boundary conditions, modifying the electrostatic potential through Poisson's equation. We initialize N point charges, modeling each one as a steep Gaussian bump with $\sigma = 2 \times 10^{-10}$ and adding over all contributions as well as any additional external charge sources to create ρ. We use this distribution in our iterative procedure during electrostatic calculations. Since we only compete boundary conditions once, compared to several iterations of our NEGF process, adding arbitrary charge distributions is not expensive and does not increase the complexity of our model. We can efficiently model arbitrarily complex point charge distributions, such as shown in figure 4.

FIG. 4. ELEQTRONeX spelled using 200 point charges. Charge density is plotted, spelling ELEQTRONeX using 200 point charges. The number or distribution of point charges does not slow computations.

TABLE I. Total time complexities for different surface Green's function calculation methods

	Time Complexity
Eigenfunction Decomposition	$(P-1)\mathscr{O}\left(n^3\right)+\mathscr{O}\left(2^3n^3\right)+\mathscr{O}\left(n^3\right)$
Iterative	$(P-1)\mathscr{O}\left(n^3\right)+(M+1)\mathscr{O}\left(n^3\right)$
Inversion of $H_{R\,cond}$	$(P-1)\mathscr{O}\left(n^3\right)+\mathscr{O}\left(P^3n^3\right)$
Inversion of H_R	$\mathscr{O}\left(N^3P^3n^3\right)$

III. RESULTS AND DISCUSSION

A. Comparisons of Computational Costs

Our Hamiltonian condensation algorithm requires the inversion of matrix C, which can be found quickly using the algorithm presented in §V B. The cost of this process is $(P-1)\mathscr{O}\left(n^3\right)$. In principle, the surface Green's function could be calculated from here, directly inverting the condensed matrix. This operation would cost $\mathscr{O}\left(P^3n^3\right)$. The total cost of $(P-1)\mathscr{O}\left(n^3\right)+\mathscr{O}\left(P^3n^3\right)$ is already an improvement over the direction inverse of an uncondensed Hamiltonian, an operation with cost $\mathscr{O}\left(N^3P^3n^3\right)$ from N periodic regions. Due to the assumption of semi-infinite probes, N must be a much larger than P, so reducing the size of the Hamiltonian is already hugely beneficial in terms of time complexity.

Our iterative and eigenvalue methods also accelerate computations, with these having costs of $(M+1)\mathscr{O}\left(n^3\right)$ and $\mathscr{O}\left(2^3n^3\right)+\mathscr{O}\left(n^3\right)$ respectively for M iterations of the decimation algorithm. in ELEQTRONeX CNT simulations, we usually set $n=17$, with n being in no case greater than 30. For these cases, the number of iterations required before $\mathrm{tr}\left(\boldsymbol{\mu}_k-\boldsymbol{\mu}_{k-1}\right)\leq 10^{-12}$ was never more than 10, our iterative algorithm converges extremely quickly. Convergence errors when compared to direct inversions are shown in figure 5. The iterative and eigenvalue techniques are thus roughly comparable in time complexity, both representing tremendous improvements over direct inverse computations to find $\boldsymbol{g}_{0,0}$. All time complexities are displayed in table I.

To validate our methods, we calculate the surface Green's functions in full generality, with $\boldsymbol{\alpha}_j$ and $\boldsymbol{\tau}_j$ matrices containing random complex values of magnitude less than 1. We use $n=17$, and vary the number of layers P that compose the periodic region. For direct Hamiltonian inversion, we invert over $N=100$ periodic regions. We implement our methods in a Python 3.10.12, performing simulations on an Intel Xeon

TABLE II. List of run times (in seconds) for surface Green's function calculations using different methods implemented in Python.

	$P=2$	$P=4$	$P=8$
Eigenfunction Decomposition	0.0117	0.0148	0.0294
Iterative	0.0184	0.0117	0.0159
Inversion of $H_{R\,cond}$	0.5753	0.7039	1.583
Inversion of H_R	3.146	20.64	136.2

processor.

As demonstrated in table II, our eigenfunction and iterative methods are much faster than directly inverting both the complete and condensed probe Hamiltonians. As expected, matrix inversion methods scale poorly as the number of layers increases, but since we held n constant, our eigenfunction and iterative methods do not slow significantly as we add layers. This independence from P and N allows us to model arbitrarily large and complex probes without slowing down simulations.

B. Impacts of Point Charges on Single CNTs

To first examine the impact of single point charges, we test point charges over a single CNT of 10 nm and 100 nm. One point charge of charge q is placed 1 nm above the center of CNT, as shown in figure 6. The setup is otherwise identical to that of figure 1, with a source, drain, and gate voltage affecting the CNT. We test the equilibrium case, setting $V_{ds}=0$, and varying V_g from -2 V to 2 V. We use fixed Dirichlet boundary conditions.

From figure 7, it is clear that point charges have a drastic impact on CNT dynamics. Point charges shift CNT potentials in their vicinity, even in the presence of larger gate voltages. This potential increase also shifts the electron band edges, changing the flow of current within the CNT and modifying conductance.

ON and OFF current factors as well as the CNT subthreshold swing (SS), are of importance in digital electronics, and we calculate these to quantify the effects of point charges on 10 nm and 100 nm CNTs. The ON current, I_{ON} is the current flowing through the CNT when it is conducting, while the OFF current I_{OFF} is the current flowing when the CNT is conducting only a small leakage current. We evaluate I_{ON} by determining the value of V_g at which CNT conductance stabilizes, indicating a current saturation. Calculating I_{OFF} is more difficult, as in CNTs, leakage OFF currents are dominated by complex optical phonon scattering interactions,[19] which are not yet modeled in ELEQTRONeX. However, since these phonons have an energy of approximately 0.2 eV, we approximate I_{OFF} at the gate voltage level at which the conductance band edge is 0.2 eV above the lead Fermi level. SS is defined as:

$$\mathrm{SS}:=\frac{dV_g}{d\left(\log I\right)} \qquad \text{(III.1)}$$

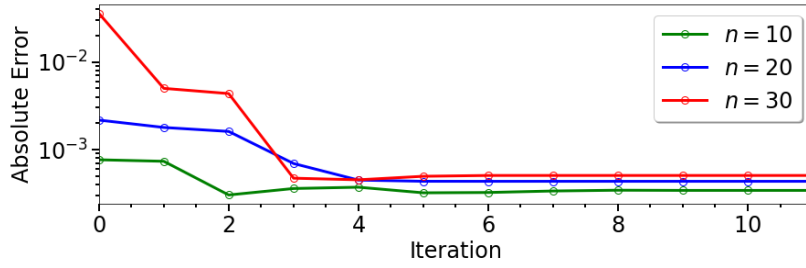Where I is the CNT current at equilibrium. SS describes the

FIG. 5. Error of decimation technique from direct matrix inversions on random matrices for varying n sizes. Error is calculated from the average difference of all matrix elements. We set $N = 100$ and $P = 12$ in this case, noting that error shrinks very rapidly, converging within 6 iterations even for $n = 30$, a larger layer size that we use in simulations.
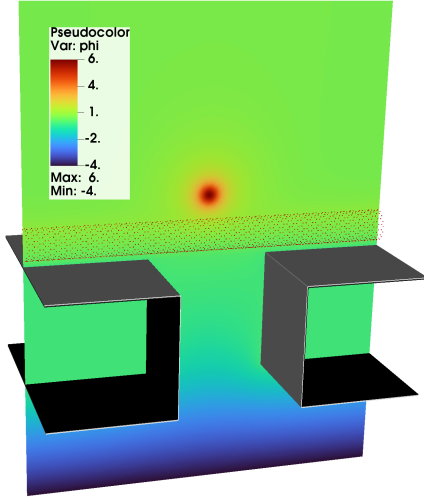

FIG. 6. Point Charge setup for 10 nm cnt. The colormap plots potential, the point charge as a narrow Guassian clearly indicated above the CNT region. The source and drain portions are in grey, placed to the left and right of the figure.

TABLE III. ON and OFF conductance current and subthreshold swings for 10 nm and 100 nm cases with and without point charges. Clearly, point charges have a significant impact on these values.

CNT Geometry	G_{ON} ($\mu S/\mu m$)	G_{OFF} ($\mu S/\mu m$)	SS (mV/dec)
10 nm	64.61	9.869	1912
10 nm Charge	197.7	33.26	2641
100 nm	16.76	1.215e-3	22.85
100 nm Charge	324.3	45.82	499.4

change in gate voltage needed for a decade change in the current, corresponding to the energy required to turn a CNTFET on and off.

From table III, point charges impact these quantities. In both 10 and 100 nm cases, ON and OFF currents are increased in the presence of a point charge, as well as the SS factor. Since point charges effectively serve as potential barriers along CNTs, they increase conductance and SS. While our results have little physical relevance, as the simulated system with a single exposed point charge cannot be recreated exper-

imentally, we have demonstrated the effect on point charges on CNTFETs, showing our code implementation functioning in a physically reasonable manner.

C. Point Charge Densities in Topgate Configuration Experiments

Single CNT setups as described in §III B are uncommon in CNT experiments and only serve for model validation studies. Topgate configurations, shown in figure 8, are much more widespread.[13–16] These setups feature a gate placed above an array of CNTs, separated by a gate oxide region. We model such a configuration, using a 20 nm CNT and varying V_g from -3 to 1 V. We fix V_{ds} =-0.1 V. To model a CNT array, we use periodic boundary conditions at CNT edges.

Motivated by a contrast between experimental topgate results from Lin et al.[13], we speculate that a point charge density, trapped in the gate oxide region due to manufacturing deficiencies caused lower conductance values than those we simulated. We introduce a random distribution of point charges into the gate oxide region, and observe a shift in conductance towards experimental results, shown in figure 9. As in §III B, point charge densities lower CNT conductance values, with a higher density of charges having a greater impact. Of our simulated values, it seems as if a charge density of 0.24 q/nm^3, most closely matches experimental results. We note that while a shift in conductance is evident, the shape of conductance vs V_g curves in logarithmic space do not match experimental results. We hypothesize that this is due to a potential dependence off point charges. Gate oxide charge defects may only activate and act as point charges as certain potential values, instead of always being present as in our simulations. We also observe a bend towards higher conductance values for experimental results near V_g=0.3 that is not captured in our simulations. This is due to phonon transport effects introducing a minimum conductance value which we have not yet introduced in ELEQTRONeX.

While we have shown that point charge densities do allow us to approach experimental topgate results, we cannot yet replicate them without adding more features to our models. But, we now have results to lend credence to our theory of trapped charges in gate oxide regions.
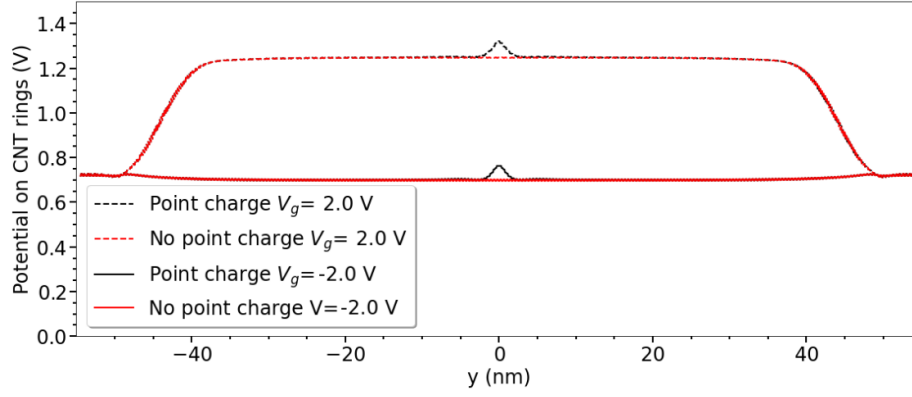
FIG. 7. Point charge impact on potential values for 100 nm CNT. We plot potentials with and without a point charge, for $V_{gs} = -2.0$ V and $V_{gs} = 2.0$ V.
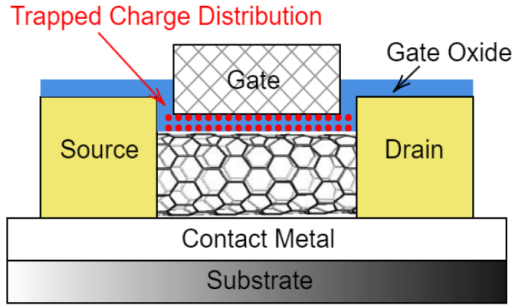

FIG. 8. Topgate CNT configuration, featuring point charges in gate oxide region.

IV. CONCLUSIONS

Herein, we have implemented two methods for self-energy calculations in ELEQTRONeX, demonstrating their computational efficiency for random matrices. Because neither our eigenfunction or decomposition methods depend on the number of layers N used, they can accurately simulate semi-infinite leads. From this, we can expand the modeling capabilities of ELEQTRONeX, with future experimental directions including different lead materials, lead defects, and phonon transport. All of these require complex lead matrix geometries, which cannot currently be implemented by the current simple self-energy analytical implementation in ELEQTRONeX.

We have also implement point charges as embedded boundary conditions, demonstrating they have a significant impacts on CNTs. For single CNTs, conductance and sub-threshold swing values change significantly under the influence of point charges, and for the topgate CNT configuration, point charge densities greatly shift conductance values. This shift points to gate oxide impurities in topgate experiments, however, to capture observed results, we need to allow for point charge potential based activation. By activating point charges only at certain position-dependent potential values, we expect to replicate experimental topgate conductance results. This ac-

tivation is fully implementable in ELEQTRONeX, instead of introducing charges as external boundary conditions, they can be coupled with the self-consistency module, as an additional step before computing NEGF equations. In the future we will also use our self-energy calculation methods to model the impact of phonons, giving more accurate I_{OFF} calculations and capturing another element of topgate experimental results that we cannot presently model.
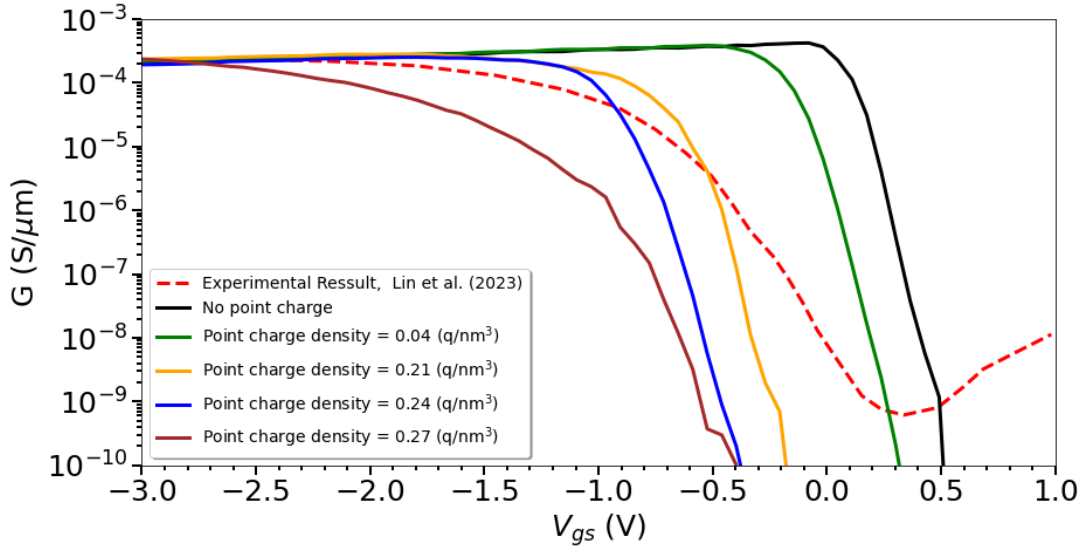
FIG. 9. Conductance values for topgate configurations for V_{gs} values from -3 V to 1 V. A voltage difference of 0.1 between source and drains is used. We introduce point charge densities in the gate oxide region, modifying the conductance. The dotted red line shows experimental results, taken from Lin et al..[13]

V. APPENDIXES

A. Derivation of NEGF Equations

1. Derivation of Charge Distribution Matrix

Beginning with equation II.4, we expand surface Green's function g_j in terms of an eigenbasis:

$$g_j = (E\boldsymbol{I} - \boldsymbol{H}_j)^{-1} = \sum_k \frac{|\boldsymbol{\xi}_{j,k}\rangle \langle \boldsymbol{\xi}_{j,k}|}{E + i\kappa - \varepsilon_{j,k}} \qquad \text{(V.1)}$$

Where $\boldsymbol{\xi}_{j,k}$ are eigenfunctions for probe j. The imaginary κ term in the denominator is an infinitesimal quantity to prevent division by zero. Defining the spectral function:

$$\boldsymbol{a}_j := i\left(\boldsymbol{g}_j - \boldsymbol{g}_j^\dagger\right) \qquad \text{(V.2)}$$

$$= i\sum_k |\boldsymbol{\xi}_{j,k}\rangle \langle \boldsymbol{\xi}_{j,k}| \left(\frac{1}{E + i\kappa - \varepsilon_{j,k}} - \frac{1}{E - i\kappa - \varepsilon_{j,k}}\right) \qquad \text{(V.3)}$$

$$= i\sum_k |\boldsymbol{\xi}_{j,k}\rangle \langle \boldsymbol{\xi}_{j,k}| \left(-\frac{2i\kappa}{\left(E - \varepsilon_{j,k}\right)^2 + \kappa^2}\right) \qquad \text{(V.4)}$$

And, letting κ go to zero:

$$\boldsymbol{a}_j = 2\pi \sum_k \delta\left(E - \varepsilon_{j,k}\right) |\boldsymbol{\xi}_{j,k}\rangle \langle \boldsymbol{\xi}_{j,k}| \qquad \text{(V.5)}$$

Now we consider the case of an incoming wave, propagating through the device into the right contact. A portion of this wave function, $|\psi_{\text{ref}}\rangle$, is completely reflected and the rest, $|\psi_{\text{trans}}\rangle$, is transmitted into the right probe. The Schrödinger equation is written in terms of these two waves:

$$\boldsymbol{H}|\psi_{\text{trans}}\rangle + \boldsymbol{\tau}_R^\dagger |\psi_{\text{ref}}\rangle = E\boldsymbol{I}|\psi_{\text{trans}}\rangle \qquad \text{(V.6)}$$

Solving using Green's functions:

$$|\psi_{\text{trans}}\rangle = \boldsymbol{G}\boldsymbol{\tau}_R^\dagger |\psi_{\text{ref}}\rangle \qquad \text{(V.7)}$$

Since, $|\psi_{\text{ref}}\rangle$ is reflected from the right probe, in our tight-banding approximation it exists entirely within the device, so:

$$|\psi_D\rangle = \boldsymbol{G}_D \boldsymbol{\tau}_R^\dagger |\psi_{\text{ref}}\rangle \qquad \text{(V.8)}$$

Formalizing, and repeating the process for either contact, we write the eigenfunctions of the device in terms of eigenfunctions in either probe:

$$|\boldsymbol{\xi}_{D,j,k}\rangle = \boldsymbol{G}_D \boldsymbol{\tau}_j^\dagger |\boldsymbol{\xi}_{j,k}\rangle \qquad \text{(V.9)}$$

The charge density matrix for the device from incoming waves of the j probe is given as:

$$\tilde{\boldsymbol{\rho}}_j = \sum_k f_j \delta\left(E - \varepsilon_{j,k}\right) |\boldsymbol{\xi}_{D,j,k}\rangle \langle \boldsymbol{\xi}_{D,j,k}| \qquad \text{(V.10)}$$

The sum is carried out over all quantum numbers, the delta function ensuring the sum is performed over eigenenergies. Using equation V.9 and V.5:

$$\tilde{\boldsymbol{\rho}}_j = \sum_k f_j(E, \mu_j) \delta\left(E - \varepsilon_{j,k}\right) \boldsymbol{G}_D \boldsymbol{\tau}_j^\dagger |\boldsymbol{\xi}_{j,k}\rangle \langle \boldsymbol{\xi}_{j,k}| \boldsymbol{\tau}_j \boldsymbol{G}_D^\dagger$$
$$\text{(V.11)}$$

$$= \frac{1}{2\pi} f_j \boldsymbol{G}_D \boldsymbol{\tau}_j^\dagger \boldsymbol{a}_j \boldsymbol{\tau}_j \boldsymbol{G}_D^\dagger \qquad \text{(V.12)}$$

And, introducing the quantity $\boldsymbol{\Gamma}_j := \boldsymbol{\tau}_j^\dagger \boldsymbol{a}_j \boldsymbol{\tau}_j = i\left(\boldsymbol{\Sigma}_j - \boldsymbol{\Sigma}_j^\dagger\right)$:

$$\tilde{\boldsymbol{\rho}}_j = \frac{1}{2\pi} f_j \boldsymbol{G}_D \boldsymbol{\Gamma}_j \boldsymbol{G}_D^\dagger \qquad \text{(V.13)}$$

Summing over both probes and integrating over all energies yields the total charge in the device:

$$\rho = 2 \times \frac{1}{2\pi} \int_{-\infty}^{\infty} G_D \left(f_L \Gamma_L + f_R \Gamma_R \right) G_D^\dagger dE \quad \text{(V.14)}$$

The multiplier of 2 comes from 2 spin states for electrons.

2. Derivation of Device Current and Conductance

Using conservation of probability, and expanding in terms of eigenfunctions, we derive an equation for the probability current within the device:

$$\partial_t \langle \psi_D | \psi_D \rangle = \langle \partial_t \psi_D | \psi_D \rangle + \langle \psi_D | \partial_t \psi_D \rangle \quad \text{(V.15)}$$

$$= \frac{i}{\hbar} \left(\langle \psi_D H | \psi_D \rangle - \langle \psi_D | H \psi_D \rangle \right) \quad \text{(V.16)}$$

$$= \frac{i}{\hbar} \sum_k \left(\left[\langle \xi_{L,k} \tau_L | + \langle \xi_{D,k} H_D | + \langle \xi_{R,k} \tau_R | \right] |\psi_D\rangle \right.$$
$$\left. - \langle \psi_D | \left[\tau_L^\dagger \xi_{L,k} \rangle + |H_D \xi_{D,k}\rangle + |\tau_R^\dagger \xi_{R,k} \rangle \right] \right) \quad \text{(V.17)}$$

$$= \frac{i}{\hbar} \sum_k \left(\left[\langle \xi_{L,k} | \tau_L | \psi_D \rangle - \langle \psi_D | \tau_L^\dagger | \xi_{L,k} \rangle \right] \right.$$
$$\left. + \left[\langle \xi_{R,k} | \tau_R | \psi_D \rangle - \langle \psi_D | \tau_R^\dagger | \xi_{R,k} \rangle \right] \right) \quad \text{(V.18)}$$

Each term can be regarded as the current from source j, such that by multiplying by the charge of an electron, $-q$, the electrical current density in the device in one probe is expressed as:

$$\tilde{I}_j = -\frac{qi}{\hbar} \sum_k \left(\langle \xi_{j,k} | \tau_j | \psi_D \rangle - \langle \psi_D | \tau_j^\dagger | \xi_{j,k} \rangle \right) \quad \text{(V.19)}$$

Now we calculate the current at each end of the device. The formulation is equivalent at both ends of the device, so without loss of generality, we chose to calculate the current at the left side of the device. We do this by calculating the current contribution from the right probe using equation V.19 in terms of the right hand basis wave functions $\xi_{R,k}$. Using equations II.1 and V.9, we show:

$$\xi_{L,k} = g_L \tau_L |\xi_{D,k}\rangle = g_L \tau_L G_D \tau_R^\dagger |\xi_{R,k}\rangle \quad \text{(V.20)}$$

Thus:

$$\tilde{I}_{L \text{ from } R} = -\frac{qi}{\hbar} \sum_k \left(\langle \xi_{R,k} | \tau_L | \psi_D \rangle - \langle \psi_D | \tau_L^\dagger | \xi_{L,k} \rangle \right) \quad \text{(V.21)}$$

$$= -\frac{qi}{\hbar} \sum_k \left(\langle \xi_{R,k} | \tau_R G_D^\dagger \tau_L^\dagger g_L^\dagger \tau_L G_D \tau_R^\dagger |\psi_R\rangle \right.$$
$$\left. - \langle \psi_R | \tau_R G_D^\dagger \tau_L^\dagger g_L \tau_L G_D \tau_R^\dagger |\xi_{R,k}\rangle \right) \quad \text{(V.22)}$$

$$= \frac{qi}{\hbar} \sum_k \langle \xi_{R,k} | \tau_R G_D^\dagger \tau_L^\dagger \left(g_L - g_L^\dagger \right) \tau_L G_D \tau_R^\dagger |\psi_R\rangle \quad \text{(V.23)}$$

$$= \frac{q}{\hbar} \sum_k \langle \xi_{R,k} | \tau_R G_D^\dagger \Gamma_L G_D \tau_R^\dagger |\psi_R\rangle \quad \text{(V.24)}$$

Integrating over energy and multiplying by 2 to account for both spin states yields total current:

$$I_{L \text{ from } R} = \quad \text{(V.25)}$$

$$\frac{2q}{\hbar} \int_{-\infty}^{\infty} f_R \sum_k \delta \left(E - \varepsilon_{R,k} \right) \langle \xi_{R,k} | \tau_R G_D^\dagger \Gamma_L G_D \tau_R^\dagger |\psi_R\rangle dE \quad \text{(V.26)}$$

$$= \frac{2q}{\hbar} \int_{-\infty}^{\infty} f_R \sum_{k,l} \delta \left(E - \varepsilon_{R,k} \right) \langle \xi_{R,k} | \tau_R \phi_{R,l} \rangle$$
$$\langle \phi_{R,l} | G_D^\dagger \Gamma_L G_D \tau_R^\dagger |\psi_R\rangle dE \quad \text{(V.27)}$$

$$= \frac{2q}{\hbar} \int_{-\infty}^{\infty} f_R \sum_l \langle \phi_{R,l} | G_D^\dagger \Gamma_L G_D \tau_R^\dagger$$
$$\left[\sum_k \delta \left(E - \varepsilon_{R,k} \right) |\xi_{R,k}\rangle \langle \xi_{R,k} | \right] \tau_R \phi_{R,l} \rangle dE \quad \text{(V.28)}$$

$$= \frac{2q}{2\pi\hbar} \int_{-\infty}^{\infty} f_R \sum_l \langle \phi_{R,l} | G_D^\dagger \Gamma_L G_D \tau_R^\dagger |a_R| \tau_R \phi_{R,l} \rangle dE \quad \text{(V.29)}$$

$$= \frac{q}{\pi\hbar} \int_{-\infty}^{\infty} f_R \text{Tr} \left(G_D^\dagger \Gamma_L G_D \Gamma_R \right) dE \quad \text{(V.30)}$$

And to get the total current, the current from both ends are subtracted:

$$I = \frac{q}{\pi\hbar} \int_{-\infty}^{\infty} \left(f_L - f_R \right) \text{Tr} \left(G_D^\dagger \Gamma_L G_D \Gamma_R \right) dE \quad \text{(V.31)}$$

The conductance is calculated without bias, at equilibrium at the Fermi level f_0 at which $I = 0$:

$$G = \frac{q^2}{\pi\hbar} \int_{-\infty}^{\infty} \frac{\partial f_0}{\partial E} \text{Tr} \left(G_D^\dagger \Gamma_L G_D \Gamma_R \right) dE \quad \text{(V.32)}$$

B. Algorithm for Hamiltonian Matrix Condensation

Here we present an algorithm for the condensation of the Hamiltonian Matrix as outlined in §II C. This algorithm is based on the work of Huang et al.[11]. Given a block tridiagonal Hamiltonian with n elements per block and a periodicity of P layers as described by equations II.9 and II.11, we proceed as follows:

1. Define:
   $$\tilde{H}_{P,P} := (EI - \alpha_{N+P})^{-1}$$

2. For $k = P - 1, P - 2, \ldots 2$, do:
   $$\tilde{H}_{k,k} = \left(EI - \alpha_{N+k} - \tau_{N+k} \tilde{H}_{k+1,k+1} \tau_{N+k}^\dagger \right)^{-1}$$
   $$\tilde{H}_{k,P} = \tilde{H}_{k,k} \tau_{N+k} \tilde{H}_{k+1,P}$$

3. Define:
$$\tilde{C}_{2,2} := \tilde{H}_{2,2}$$
$$\tilde{C}_{2,P} := \tilde{H}_{2,P}$$

4. For $k = 3, 4, \ldots P$, do:
$$\tilde{C}_{k,k} = \tilde{H}_{k,k} + \tilde{H}_{k,k} \tau_{N+k-1}^{\dagger} \tilde{C}_{k-1,k-1} \tau_{N+k-1} \tilde{H}_{k,k}$$

5. Obtain:
$$\Xi_s = \alpha_{N+1} + \tau_{N+1} \tilde{C}_{2,2} \tau_{N+1}^{\dagger}$$
$$\Xi = \Xi_s + \tau_{N+P} \tilde{C}_{P,P} \tau_{N+P}^{\dagger}$$
$$\Pi = \tau_{N+1} \tilde{C}_{2,P} \tau_{N+P}$$

4. Assemble $\mathbf{\Lambda}$, a diagonal matrix with eigenvalue elements λ_j satisfying $|\lambda_j| < 1$.

5. Find psuedo inverse of U by factorizing $U = QR$, and solve $R\tilde{U} = Q^{\dagger}$.

6. Define:
$$F := U\Lambda\tilde{U}$$
$$\Sigma' = \Pi F$$

7. Obtain:
$$g_{0,0} = (E\mathbf{I} - \Xi_s - \Sigma')^{-1}$$

C. Algorithm for Decimation Technique for Surface Green's Function Calculation

Here we present an algorithm for the decimation method discussed in §II D. Given Ξ_s, Ξ, and Π matrices, we proceed as follows:

1. Define:
$$\mu_0 := E\mathbf{I} - \Xi_s$$
$$\nu_0 := E\mathbf{I} - \Xi$$
$$\gamma_0 := \Pi$$

2. Perform initial iteration:
$$\mu_1 := \mu_0 - \gamma_0 \nu_0^{-1} \gamma_0^{\dagger}$$
$$\nu_1 := \nu_0 - \gamma_0 \nu_0^{-1} \gamma_0^{\dagger} - \gamma_0^{\dagger} \nu_0^{-1} \gamma_0$$
$$\gamma_1 := \gamma_0 \nu_0^{-1} \gamma_0$$
$$\zeta_1 := \gamma_0^{\dagger} \nu_0^{-1} \gamma_0^{\dagger}$$

3. Iterative procedure:
$$\mu_k := \mu_{k-1} - \zeta_{k-1} \nu_{k-1}^{-1} \gamma_{k-1}$$
$$\nu_k := \nu_{k-1} - \zeta_{k-1} \nu_{k-1}^{-1} \gamma_{k-1} - \gamma_{k-1} \nu_{k-1}^{-1} \zeta_{k-1}$$
$$\gamma_k := \gamma_{k-1} \nu_{k-1}^{-1} \gamma_{k-1}$$
$$\zeta_k := \zeta_{k-1} \nu_{k-1}^{-1} \zeta_{k-1}$$

4. At convergence:
$$g_{0,0} = \mu_k^{-1}$$

D. Algorithm for Eigenvalue Technique for Surface Green's Function Calculation

Here we present an algorithm for the eigenvalues method discussed in §II E. Given Ξ_s, Ξ, and Π matrices, we proceed as follows:

1. Define:
$$A := \begin{pmatrix} 0 & I \\ -\Pi^{\dagger} & E\mathbf{I} - \Xi \end{pmatrix}$$
$$B := \begin{pmatrix} I & 0 \\ 0 & \Pi \end{pmatrix}$$

2. Solve the eigensystem:
$$A\vec{\psi} = \lambda B \vec{\psi}$$

3. Assemble U from the first n elements of eigenvectors $\vec{\psi}_j$ with corresponding eigenvalues with $|\lambda_j| < 1$.

[1] S. J. Tans, A. R. M. Verschueren, and C. Dekker, "Room-temperature transistor based on a single carbon nanotube," Nature (1998).

[2] B. M. Kim, T. Brintlinger, E. Cobas, M. S. Fuhrer, H. Zheng, Z. Yu, R. Droopad, J. Ramdani, and K. Eisenbeiser, "High-performance carbon nanotube transistors on srtio3/si substrates," Applied Physics Letters **84** (2004).

[3] S. Datta, *Qunatum transport atom to transistor* (Cambridge University Press, Cambridge, UK, 2005).

[4] F. Léonard and D. A. Stewart, "Properties of short channel ballistic carbon nanotube transistors with ohmic contacts," Nanotechnology **17** (2006).

[5] B. N. Chowdhury and S. Chattopadhyay, "Dual-gate gaas-nanowire fet for room temperature charge-qubit operation: a negf approach," Advanced Quantum Technologies **6** (2023).

[6] G. Cohen and M. Galperin, "Green's function methods for single molecule junctions," The Journal of Chemical Physics **152** (2020).

[7] D. Kienle and F. Léonard, "Terahertz response of carbon nanotube transistors," Physical Review Letters **103** (2009).

[8] D. A. Stewart and F. Léonard, "Photocurrents in nanotube junctions," Physical Review Letters **93** (2004).

[9] J. Guo, S. Datta, and M. Lundstrom, "Toward multiscale modeling of carbon nanotube transistors," International Journal for Multiscale Computational Engineering **2** (2004).

[10] M. P. L. Sancho, J. M. L. Sancho, J. M. L. Sancho, and J. Rubio, "Highly convergent schemes for the calculation of bulk and surface green functions," Journal of Physics F: Metal Physics **15** (1985).

[11] J. Z. Huang, W. C. Chew, Y. Wu, and L. J. Jiang, "Methods for fast evaluation of self-energy matrices in tight-binding modeling of electron transport systems," Journal of Applied Physics **112** (2012).

[12] S. Sadasivam, Y. Che, Z. Huang, L. Chen, S. Kumar, and T. S. Fisher, "The atomistic green's function method for interfacial phonon transport," Annual Review of Heat Transfer **17** (2014).

[13] X. Lin, N. Gao, Z. Zhang, and L.-M. Peng, "Lowering interface state density in carbon nanotube thin film transistors through using stacked y2o3/hfo2 gate dielectric," Applied Physics Letters **113** (2023).

[14] L. Liu, J. Han, L. Xu, J. Zhou, C. Zhao, S. Ding, H. Shi, M. Xiao, L. Ding, Z. Ma, C. Jin, Z. Zhang, and L.-M. Peng, "Aligned, high-density semiconducting carbon nanotube arrays for high-performance electronics," Science **368** (2020).

[15] Z. Zhang, M. Passlack, G. Pitner, S. Natani, S.-K. Su, T.-A. Chao, S. L. Liew, V. D.-H. Hou, C.-F. Hsu, W. E. Shipley, N. Safron, G. Doornbos, T.-E. Lee, I. Radu, A. C. Kummel, P. Bandaru, and H.-S. P. Wong, "Complementary carbon nanotube metal– oxide–semiconductor field-effect transistors with localized solid-state extension doping," Nature Electronics **6** (2023).

[16] Y. Liu, S. Ding, W. Li, Z. Zhang, Z. Pan, Y. Ze, B. Gao, Y. Zhang, C. Jin, L.-M. Peng, and Z. Zhang, "Interface states in gate stack of carbon nanotube array transistors," ACS Nano **18** (2024).

[17] J. Taylor, H. Guo, and J. Wang, "Ab initio modeling of quantum transport properties of molecular electronic devices," Physical Review B **63** (2001).

[18] F. Bloch, "Über die quantenmechanik der elektronen in kristallgittern," Zeitschrift für physik **52** (1929).

[19] S. O. Koswatta, M. S. Lundstrom, M. P. Anantram, and D. E. Nikonov, "Simulation of phonon-assisted band-to-band tunneling in carbon nanotube field-effect transistors," Applied Physics Letters **87** (2005).

[20] M. Luisier, A. Schenk, and W. Fichtner, "Quantum transport in two- and three-dimensional nanoscale transistors: Coupled mode effects in the nonequilibrium green's function formalism," Journal of Applied Physics **100** (2006).

[21] Y.-J. Zeng, Z.-K. Ding, H. Pan, Y.-X. Feng, and K.-Q. Chen, "Nonequilibrium green's function method for phonon heat transport in quantum system," Journal of Physics: Condensed Matter **34** (2022).

[22] Y.-J. Zeng, Z.-K. Ding, H. Pan, Y.-X. Feng, and K.-Q. Chen, "ad-negf: an end-to-end differentiable quantum transport simulator for sensitivity analysis and inverse problems," Physical Review B **108** (2023).

[23] F. Teichert, A. Zienert, J. Schuster, and M. Schreiber, "An improved green's function algorithm applied to quantum transport in carbon nanotubes," Computational Materials Science **169** (2019).

[24] A. Umerski, "Closed-form solutions to surface green's functionss," Physical Review B **55** (1996).

[25] M. P. Anantram, M. S. Lundstrom, and D. E. Nikonov, "Modeling of nanoscale devices," Proceedings of the IEEE **96** (2008).

[26] N. Papior, N. Lorente, T. Frederiksen, A. García, and M. Brandbyge, "Improvements on non-equilibrium and transport green function techniques: The next-generation transiesta," Computer Physics Communications **212** (2017).