11-16-2008

Multiband transmission calculations for nanowires using an optimized renormalization method

Timothy B. Boykin
University of Alabama in Huntsville

Mathieu Luisier
Integrated Systems Laboratory, Zurich

Gerhard Klimeck
Purdue University - Main Campus, gekco@purdue.edu

Multiband transmission calculations for nanowires using an optimized renormalization method

Timothy B. Boykin
Department of Electrical and Computer Engineering, The University of Alabama in Huntsville, Huntsville, Alabama 35899, USA

Mathieu Luisier
Integrated Systems Laboratory, ETH Zurich, Gloriastrasse 35, 8092 Zurich, Switzerland

Gerhard Klimeck
Network for Computational Nanotechnology, School of Electrical and Computer Engineering, Purdue University, West Lafayette, Indiana 47907, USA and Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Road, MS 169-315, Pasadena, California 91109, USA

(Received 21 December 2007; revised manuscript received 15 February 2008; published 10 April 2008)

The discovery of an interesting nanostructure behavior or the design of useful nanodevices requires state-of-the-art physical models. Realistic, multiband nanowire calculations especially tend to be computationally intensive and slow. Here, we develop optimizations to the renormalization method of Grosso *et al.* [Phys. Rev. B **40**, 12328 (1989)] specifically for nanowires with [100]- or [111]-oriented axes. For no-spin-orbit models, our optimizations give far superior performance to other available methods, while for spin-orbit models on a single processor, our results are at least as good as the best alternative. More importantly, the parallel scalability of our optimizations is superior to that of other available methods, making optimized renormalization very attractive for multiple-processor computers. We demonstrate the method with calculations for Si nanowires.

DOI: 10.1103/PhysRevB.77.165318 PACS number(s): 73.63.Nm, 73.21.Hb

I. INTRODUCTION

Calculations of the electronic structure or transmission characteristics of realistic nanostructures using multiband tight-binding models present challenges in terms of both numerical stability and computational burden. Because the presence of evanescent states throughout the energy spectrum leads to the spectacular failure of the simple transfer matrix method in these calculations, several numerically stable methods[1–6] have been developed for layered structures (quantum wells, resonant tunneling diodes, nanowires, etc.). All of these methods have proven successful in overcoming the numerical instability problems of the transfer matrix method; yet, in terms of computational burden, they can differ greatly depending on the specific application.

The question of numerical efficiency is not merely a computational issue. Transmission or electronic structure calculations generally serve as the starting point for including other effects such as inelastic scattering or electromagnetic fields, and such calculations can be very computationally intensive. It is therefore important that the electronic structure or transmission calculation consumes as little computation time as possible. As we shall show here, the renormalization method[6] can be modified and optimized so as to realize significant efficiency in nanowire calculations.

A brief review of the properties of the Hamiltonian matrix for a layered nanostructure is helpful for understanding how differences in the numerically stable methods will affect computational efficiency. Because tight-binding models have interactions only over a finite range of neighboring atoms (e.g., up to nearest or second nearest), for a layered structure, it is generally possible to construct units or layers of one or more atomic planes normal to the structure axis, which interact only with themselves and their immediately adjacent layers. (For a nearest-neighbor model such as that we employ here, the units can be single-atomic planes.) When open-system boundary conditions are enforced for transmission calculations, the Hamiltonian matrix takes a block tridiagonal form. In general, the first and last blocks incorporating the boundary conditions are of larger dimension than the others. However, for the nearest-neighbor model considered here, only adjacent atomic planes are coupled so that the first and last blocks remain the same size as the others. For transmission calculations, there is a unit incident flux, so that the linear system becomes a complex, general $\underline{\mathbf{A}}\mathbf{x}=\mathbf{b}$ problem.

The resulting sparse $\underline{\mathbf{A}}\mathbf{x}=\mathbf{b}$ problem can be solved using one of several efficient direct algorithms.[7–10] While such solvers do take advantage of some of the properties of the matrix $\underline{\mathbf{A}}$ to achieve efficiencies, they can nevertheless miss some important structural details which can lead to great speed improvements. For example, in nanowire calculations without the spin-orbit interaction, usually, only the terminal (emitter and collector) blocks of the matrix are complex. An algorithm which deals with the terminal blocks last could employ real arithmetic for most of the calculation, realizing significant savings. Complex-complex multiplies cost four times as much as do real-real multiplies, while complex-real multiplies cost twice as much. Even in calculations which include the spin-orbit interaction, the geometry of many nanowires permits similar efficiencies. When properly modified and optimized, the renormalization method[6] becomes attractive for nanowire calculations since it allows considerable flexibility in the ordering of the solution process.

The renormalization method was originally introduced for multiband superlattice electronic structure calculations.[6] Subsequently, it was used for multiband bound-state calculations of [001]-oriented quantum wells,[11] and single-band, two-dimensional nanowire calculations.[12] Because superlattices and quantum wells are effectively infinite in planar extent, Bloch sums in the appropriate wave vector form the

Hamiltonian basis states, so that the size of the block matrices of the Hamiltonian is set by the number of orbitals per atom, typically 10–20. For single-band nanowire calculations, the block size is set by the number of atoms transverse to the nanowire axis; in Ref. 12, this number is typically 50.

Our calculations involve rather larger Hamiltonians since we study realistic, three-dimensional nanowires modeled with the $sp^3d^5s^*$ basis[13] (with and without spin-orbit interaction). Such nanowires typically have 30–80 atoms in a plane, so that when spin orbit is included, the Hamiltonian blocks are of dimension 600–1600, an order of magnitude or more larger than the previous calculations. For such large problems, the renormalization method[6] must be optimized and modified to achieve good computational performance. Here, we show that for [100]- and [111]-oriented nanowires, the Hamiltonian matrix has a readily exploited mathematical structure. We present this modified and optimized version, showing that significant performance improvements are possible for many nanowires. We also show that this optimized version has excellent parallel scalability, and we use it to study the transmission characteristics and densities of states of [100]- and [111]-oriented nanowires. Section II presents the optimized method, Sec. III presents our results, and Sec. IV the conclusions.

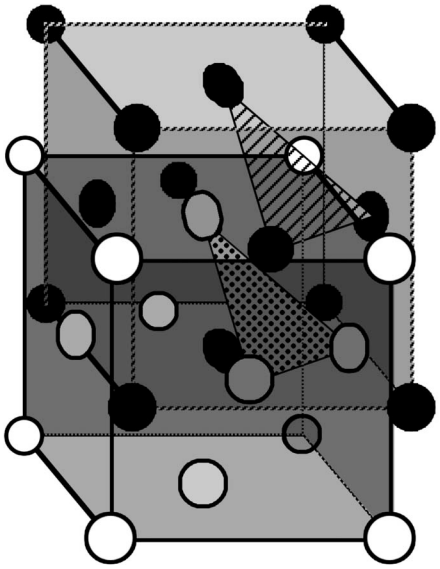

FIG. 1. Atomic positions for diamond. The two fcc sublattices ("anion" and "cation") are shown as open and filled circles. The planes orthogonal to [100] lie parallel to the cube faces, while triangles show planes orthogonal to [111]. Note that in both cases, atoms of a given plane belong to only one of the fcc sublattices. Neither of the planes has both anions and cations.

II. METHOD

A. Nanowire geometry and Hamiltonian structure

To illustrate the importance of the choice of solution method for [100]- and [111]-oriented nanowire calculations, it is necessary to examine in detail how the nanowire geometry and tight-binding model affect the mathematical properties of the Hamiltonian matrix. Consider a nanowire grown on a [100] substrate of a diamond or zinc blende material. The primitive cell for such a perfect nanowire consists of four (finite) atomic planes, as shown in Fig. 1; we will refer to this unit as a layer. The Schrödinger equation for a transmission problem through a nanowire of L layers is written[1,2,5,14] as

$$\begin{bmatrix} \underline{\underline{\mathbf{H}}}_{1,1}+\underline{\underline{\boldsymbol{\Sigma}}}_1-\underline{\underline{\mathbf{1}}}E & \underline{\underline{\mathbf{H}}}_{1,2} & \underline{\underline{\mathbf{0}}} & \cdots & \underline{\underline{\mathbf{0}}} \\ \underline{\underline{\mathbf{H}}}_{1,2}^{\dagger} & \underline{\underline{\mathbf{H}}}_{2,2}-\underline{\underline{\mathbf{1}}}E & \underline{\underline{\mathbf{H}}}_{2,3} & \underline{\underline{\mathbf{0}}} & \underline{\underline{\mathbf{0}}} \\ \underline{\underline{\mathbf{0}}} & \ddots & \ddots & \ddots & \underline{\underline{\mathbf{0}}} \\ \vdots & \ddots & \ddots & \ddots & \underline{\underline{\mathbf{H}}}_{L-1,L} \\ \underline{\underline{\mathbf{0}}} & \cdots & \underline{\underline{\mathbf{0}}} & \underline{\underline{\mathbf{H}}}_{L-1,L}^{\dagger} & \underline{\underline{\mathbf{H}}}_{L,L}+\underline{\underline{\boldsymbol{\Sigma}}}_L-\underline{\underline{\mathbf{1}}}E \end{bmatrix} \begin{bmatrix} \vec{\mathbf{C}}_1 \\ \vec{\mathbf{C}}_2 \\ \vdots \\ \vdots \\ \vec{\mathbf{C}}_L \end{bmatrix} = \begin{bmatrix} \vec{\mathbf{V}}_I \\ \vec{\mathbf{0}} \\ \vdots \\ \vdots \\ \vec{\mathbf{0}} \end{bmatrix}, \tag{1}$$

where the self-energies $\underline{\underline{\boldsymbol{\Sigma}}}_1, \underline{\underline{\boldsymbol{\Sigma}}}_L$, respectively, couple the emitter and collector layers to the semi-infinite contact regions (zero-bias continuations of the nanowire structure), the vectors $\vec{\mathbf{C}}_j$ are the orbital coefficients for the jth layer, and the vector $\vec{\mathbf{V}}_I$ represents the injection from the left contact. (Any injection from the right contact would appear in the last block of the right-hand vector.) Double-underlined matrices and arrow-superscripted vectors cover layers. (The Hamiltonian for a [111]-oriented nanowire adopts the same tridiagonal form, except that the primitive cell now has six atomic planes.)

We remark on some general properties of Eq. (1) before proceeding. First, Eq. (1) could be written in other forms. For example, to recover the form of Eq. (1) from Eq. (17) of Ref. 5, one uses standard block elimination to decouple slabs 0 and (N_S+1) from the rest of that matrix by eliminating blocks L_{10} and $R_{N_SN_S+1}$ (in the notation of Ref. 5). The result is a matrix of three block diagonals, where the central (and largest) block and inhomogeneous vector correspond to Eq. (1) and are decoupled from the first and last blocks. Second, the tridiagonal structure of the system in Eq. (1) is clearly advantageous but is not its only useful mathematical property. For nearest-neighbor models, the structure of the submatrices can be exploited to simplify the solution process.

To understand this structure, it is important to keep in mind a few points about orthogonal tight-binding models for diamond or zinc blende semiconductors. First, in nearest-neighbor orthogonal tight binding, the only interatomic coupling is anion-cation. There are no anion-anion or cation-cation matrix elements. (For elemental semiconductors, we employ this terminology strictly to designate the two fcc sublattices of diamond.) Second, if the spin-orbit interaction is neglected, then the orbitals are conventionally chosen as real and all of the submatrices in Eq. (1) except the terminal diagonal blocks, (1, 1) (L,L), are real. Third, when the spin-orbit interaction is included using Chadi's model[15] (a same-atom, p-only matrix element), the only complex submatrices are the diagonal blocks. There are no interatomic matrix elements between orbitals of different spin quantum numbers. Our optimizations take advantage of these properties and therefore assume the use of a nearest-neighbor, orthogonal tight-binding model, where Chadi's prescription[15] is used for the spin-orbit interaction (when included).

The geometry of [100]- and [111]-oriented nanowires together with a nearest-neighbor orthogonal tight-binding model leads to perhaps the most important mathematical properties of the system in Eq. (1). Figure 1 shows the two fcc sublattices of the diamond or zinc blende structure. The cube faces are of course normal to the [100] directions, while planes normal to the [111] directions are shown as striped and shaded triangles. Note that for both wire geometries, only a single atom type (anion or cation) lies in each plane, so that only atoms in adjacent planes are coupled together. There are no same-plane, different atom matrix elements, so that the blocks in Eq. (1) adopt a much simpler form.

We illustrate this simplification for the [100] case; the extension to the [111] case is obvious. Denoting the atomic planes $n=1,2,\ldots,N$, $N=4L$, the diagonal blocks take the form

$$\underline{\underline{\mathbf{H}}}_{l,l} = \begin{bmatrix} \underline{\mathbf{H}}'_{4l-3,4l-3} & \underline{\mathbf{H}}_{4l-3,4l-2} & \underline{\mathbf{0}} & \underline{\mathbf{0}} \\ \underline{\mathbf{H}}^{\dagger}_{4l-3,4l-2} & \underline{\mathbf{H}}_{4l-2,4l-2} & \underline{\mathbf{H}}_{4l-2,4l-1} & \underline{\mathbf{0}} \\ \underline{\mathbf{0}} & \underline{\mathbf{H}}^{\dagger}_{4l-2,4l-1} & \underline{\mathbf{H}}_{4l-1,4l-1} & \underline{\mathbf{H}}_{4l-1,4l} \\ \underline{\mathbf{0}} & \underline{\mathbf{0}} & \underline{\mathbf{H}}^{\dagger}_{4l-1,4l} & \underline{\mathbf{H}}'_{4l,4l} \end{bmatrix}, \quad l=1,2,\ldots,L, \tag{2}$$

where single-underlined matrices denote single-atomic plane blocks. The matrices $\underline{\mathbf{H}}'_{p,p}$ are different only for the terminal blocks since they incorporate the open-system boundary conditions. These conditions affect only the first and last planes of the nanowire,

$$\underline{\mathbf{H}}'_{p,p} = \underline{\mathbf{H}}_{p,p}, \quad p \neq 1,4L, \quad \underline{\mathbf{H}}'_{1,1} = \underline{\mathbf{H}}_{1,1} + \underline{\mathbf{\Sigma}}_1,$$
$$\underline{\mathbf{H}}'_{4L,4L} = \underline{\mathbf{H}}_{4L,4L} + \underline{\mathbf{\Sigma}}_{4L}, \tag{3}$$

$$\vec{\mathbf{V}}_l = [\mathbf{V}_1, \mathbf{0}, \mathbf{0}, \mathbf{0}]^T, \tag{4}$$

where superscript T denotes the transpose. Because the self-energies $\underline{\mathbf{\Sigma}}_p$ incorporate the open-system boundary conditions, the first and last diagonal blocks are non-Hermitian. Observe that if the spin-orbit interaction is neglected, the diagonal blocks of Eq. (2), $\underline{\mathbf{H}}_{p,p}$, are themselves real and diagonal, consisting of the orbital same-atom parameters. The off-diagonal blocks in Eq. (1) are even simpler,

$$\underline{\underline{\mathbf{H}}}_{l,l+1} = \begin{bmatrix} \underline{\mathbf{0}} & \underline{\mathbf{0}} & \underline{\mathbf{0}} & \underline{\mathbf{0}} \\ \underline{\mathbf{0}} & \underline{\mathbf{0}} & \underline{\mathbf{0}} & \underline{\mathbf{0}} \\ \underline{\mathbf{0}} & \underline{\mathbf{0}} & \underline{\mathbf{0}} & \underline{\mathbf{0}} \\ \underline{\mathbf{H}}_{4l,4l+1} & \underline{\mathbf{0}} & \underline{\mathbf{0}} & \underline{\mathbf{0}} \end{bmatrix}, \quad l=1,2,\ldots,(L-1), \tag{5}$$

since only the last plane of layer l and the first plane of layer $(l+1)$ are coupled. The planar block matrices in Eqs. (2) and (5) are of dimension $N_a N_{orb}$, where N_a is the number of atoms in a plane and N_{orb} is the number of orbitals per atom.

Because even the planar block matrices can be quite large in a realistic model, it is critical to take advantage of the efficiencies afforded by treating the system in Eq. (1) as a planar coupling system (as opposed to a layer coupling system). Other efficiencies may also be exploited. Since each atom couples to only four nearest neighbors, the Hamiltonian blocks coupling adjacent atomic planes will themselves be quite sparse. In the [100]-oriented nanowire discussed here, each atom couples to two atoms in the plane ahead, and two in the plane behind; only orbitals of the same spin quantum number are coupled. Thus, each block row of the matrix coupling atomic planes p and $(p+1)$, $\underline{\mathbf{H}}_{p,p+1}$, only has two nonzero block columns. In summary, then, an efficient method for solving the system [Eq. (1)] should take advantage of the sparse nature of the Hamiltonian and individual planar blocks, as well as the fact that the interplane coupling matrices are real.

B. Optimized renormalization method

The renormalization method[6] is attractive for solving [100] and [111] nanowire transmission problems because it allows great flexibility in the order of execution of the $\underline{\mathbf{A}}\mathbf{x}=\mathbf{b}$ calculation. As noted above, when spin-orbit coupling is neglected, all of system (1) is real except for the terminal blocks, and even when spin-orbit is included, the vast majority of the system is real. In addition, the planar blocks are quite sparse. As we shall see below, the renormalization method[6] allows us to take advantage of these properties to an extent unmatched by other methods.

The basic process in the renormalization method[6] is decoupling an atomic plane from its neighbors. Because the details of this process are precisely what allow us to introduce optimizations, we briefly sketch the method below. Since we need the wave function in the original basis, this process is most conveniently presented as a transformation. (In the original formulation of the method,[6] only the eigenvalues of the superlattice Hamiltonian were of interest.) An interior plane p (one not belonging to either of the terminal blocks) is decoupled by observing that

$$H = M^{-1}_{L,p} \tilde{H} M^{-1}_{R,p}, \tag{6}$$

$$H = \begin{bmatrix} \ddots & \ddots & & & \\ \ddots & \underline{\mathbf{H}}_{p-1,p-1} - \underline{\mathbf{1}}E & \underline{\mathbf{H}}_{p-1,p} & \underline{\mathbf{0}} & \\ \underline{\mathbf{0}} & \underline{\mathbf{H}}_{p-1,p}^{\dagger} & \underline{\mathbf{H}}_{p,p} - \underline{\mathbf{1}}E & \underline{\mathbf{H}}_{p,p+1} & \\ & & \underline{\mathbf{H}}_{p,p+1}^{\dagger} & \underline{\mathbf{H}}_{p+1,p+1} - \underline{\mathbf{1}}E & \underline{\mathbf{H}}_{p+1,p+2} \\ & & \ddots & \ddots & \ddots \end{bmatrix}, \tag{7}$$

$$\tilde{H} = \begin{bmatrix} \ddots & \ddots & & & \\ \ddots & \hat{\underline{\mathbf{H}}}_{p-1,p-1} - \underline{\mathbf{1}}E & \underline{\mathbf{0}} & \hat{\underline{\mathbf{H}}}_{p-1,p+1} & \\ \underline{\mathbf{0}} & \underline{\mathbf{0}} & \underline{\mathbf{H}}_{p,p} - \underline{\mathbf{1}}E & \underline{\mathbf{0}} & \\ & \hat{\underline{\mathbf{H}}}_{p-1,p+1}^{\dagger} & \underline{\mathbf{0}} & \hat{\underline{\mathbf{H}}}_{p+1,p+1} - \underline{\mathbf{1}}E & \underline{\mathbf{H}}_{p+1,p+2} \\ & & & \ddots & \ddots \end{bmatrix}, \tag{8}$$

$$M_{L,p}^{-1} = \begin{bmatrix} \ddots & \underline{\mathbf{0}} & \ddots & \cdots & \cdots \\ \ddots & \underline{\mathbf{1}} & \underline{\mathbf{X}}_p^{\dagger} & \underline{\mathbf{0}} & \ddots \\ \ddots & \underline{\mathbf{0}} & \underline{\mathbf{1}} & \underline{\mathbf{0}} & \ddots \\ \vdots & \underline{\mathbf{0}} & \underline{\mathbf{Y}}_p^{\dagger} & \underline{\mathbf{1}} & \ddots \\ \vdots & \ddots & \ddots & \underline{\mathbf{0}} & \ddots \end{bmatrix}, \quad M_{R,p}^{-1} = (M_{L,p}^{-1})^{\dagger}, \tag{9}$$

$$\underline{\mathbf{X}}_p = (\underline{\mathbf{H}}_{p,p} - \underline{\mathbf{1}}E)^{-1}\underline{\mathbf{H}}_{p-1,p}^{\dagger}, \quad \underline{\mathbf{Y}}_p = (\underline{\mathbf{H}}_{p,p} - \underline{\mathbf{1}}E)^{-1}\underline{\mathbf{H}}_{p,p+1}, \tag{10}$$

$$\hat{\underline{\mathbf{H}}}_{p-1,p+1} = -\underline{\mathbf{H}}_{p-1,p}\underline{\mathbf{Y}}_p, \quad \hat{\underline{\mathbf{H}}}_{p-1,p-1} = \underline{\mathbf{H}}_{p-1,p-1} - \underline{\mathbf{H}}_{p-1,p}\underline{\mathbf{X}}_p,$$
$$\hat{\underline{\mathbf{H}}}_{p+1,p+1} = \underline{\mathbf{H}}_{p+1,p+1} - \underline{\mathbf{H}}_{p,p+1}^{\dagger}\underline{\mathbf{Y}}_p, \tag{11}$$

with the other blocks unchanged. Because each plane is decoupled once, indexing the transformation matrices $M_{L,p}^{-1}, M_{R,p}^{-1}$ by just the decoupled plane index is sufficient. Also, the matrix $M_{L,p}$ is trivially obtained from Eq. (9),

$$M_{L,p} = \begin{bmatrix} \ddots & \underline{\mathbf{0}} & \ddots & \cdots & \cdots \\ \ddots & \underline{\mathbf{1}} & -\underline{\mathbf{X}}_p^{\dagger} & \underline{\mathbf{0}} & \ddots \\ \ddots & \underline{\mathbf{0}} & \underline{\mathbf{1}} & \underline{\mathbf{0}} & \ddots \\ \vdots & \underline{\mathbf{0}} & -\underline{\mathbf{Y}}_p^{\dagger} & \underline{\mathbf{1}} & \ddots \\ \vdots & \ddots & \ddots & \underline{\mathbf{0}} & \ddots \end{bmatrix}, \quad M_{R,p} = M_{L,p}^{\dagger}. \tag{12}$$

The matrices $\underline{\mathbf{X}}_p, \underline{\mathbf{Y}}_p$ and the inverses $\hat{\underline{\mathbf{H}}}_{p,p}^{-1}$ used to generate them are saved for use in wave function reconstruction. (A circumflex is used since a plane may have been renormalized prior to decoupling.)

One finds the wave function by repeatedly decoupling planes and accumulating transformation matrices until the decoupled Hamiltonian $\tilde{H}$ is block diagonal. Only the last step, decoupling terminal planes 1 and 4 L, is different because the terminal blocks are not Hermitian due to the open-system boundary conditions,

$$M_{L,4L}^{-1} = \begin{bmatrix} \underline{\mathbf{1}} & \underline{\mathbf{0}} & \cdots & \underline{\mathbf{0}} & \underline{\mathbf{X}}_{4L}' \\ \underline{\mathbf{0}} & \ddots & \ddots & \ddots & \underline{\mathbf{0}} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \underline{\mathbf{0}} \\ \underline{\mathbf{0}} & \cdots & \cdots & \underline{\mathbf{0}} & \underline{\mathbf{1}} \end{bmatrix},$$

$$M_{R,4L}^{-1} = \begin{bmatrix} \underline{\mathbf{1}} & \underline{\mathbf{0}} & \cdots & \cdots & \underline{\mathbf{0}} \\ \underline{\mathbf{0}} & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \underline{\mathbf{0}} & \ddots & \ddots & \ddots & \underline{\mathbf{0}} \\ \underline{\mathbf{X}}_{4L} & \underline{\mathbf{0}} & \cdots & \underline{\mathbf{0}} & \underline{\mathbf{1}} \end{bmatrix}, \tag{13}$$

$$\underline{\mathbf{X}}_{4L} = [\hat{\underline{\mathbf{H}}}_{4L,4L} + \underline{\mathbf{\Sigma}}_{4L} - \underline{\mathbf{1}}E]^{-1}\hat{\underline{\mathbf{H}}}_{1,4L}^{\dagger},$$
$$\underline{\mathbf{X}}_{4L}' = \hat{\underline{\mathbf{H}}}_{1,4L}[\hat{\underline{\mathbf{H}}}_{4L,4L} + \underline{\mathbf{\Sigma}}_{4L} - \underline{\mathbf{1}}E]^{-1}, \tag{14}$$

$$\tilde{\underline{\mathbf{H}}}_{1,1} = \hat{\underline{\mathbf{H}}}_{1,1} - \hat{\underline{\mathbf{H}}}_{1,4L}\underline{\mathbf{X}}_{4L}. \tag{15}$$

Note that for an incident state as in Eq. (1), multiplication of the inhomogeneous vector by the matrices $M_{L,p}$ has no effect. Thus, to reconstruct the wave function, one multiplies the inhomogeneous vector by the inverse of the now block-diagonal $\tilde{H}$ followed by repeated multiplication by the transformation matrices $M_{R,p}$, in reverse decoupling order. The transformation matrices are not multiplied together; instead, this operation proceeds as the far more efficient matrix-vector multiplication. Because each plane is decoupled only once, a simple recursion relation yields the subvectors of wave function coefficients for each atomic plane. From Eq. (4), only plane 1 has an inhomogeneous part, so that only the inverse $[\tilde{\underline{\mathbf{H}}}_{1,1} + \underline{\mathbf{\Sigma}}_1 - \underline{\mathbf{1}}E]^{-1}$ need be computed. Furthermore, each of the matrices $M_{R,p}$ alters only the subvector for plane p, which is not affected by subsequent matrix multiplications. Because this subvector was initially zero, the result is the wave function subvector for plane p, $\mathbf{C}_p$. The recursion is thus

$$\mathbf{C}_1 = [\underline{\tilde{\mathbf{H}}}_{1,1} + \underline{\tilde{\mathbf{\Sigma}}}_1 - \underline{\mathbf{1}}E]^{-1}\mathbf{V}_1, \quad \mathbf{C}_{4L} = -\underline{\mathbf{X}}_{4L}\mathbf{C}_1, \tag{16}$$

$$\mathbf{C}_p = -\underline{\mathbf{X}}_p\mathbf{C}_{p-m} - \underline{\mathbf{Y}}_p\mathbf{C}_{p+n}, \tag{17}$$

where the recursion proceeds in reverse decoupling order and, prior to decoupling, the plane p was connected to planes $(p-m)$ and $(p+n)$, $p \neq 1$, $4L$, $m,n \geq 1$.

There is, of course, great flexibility in the ordering of the decoupling and it is here that we can optimize the method. It is not at all necessary to "roll up" the nanowire from one end to the other. In fact, that procedure is very inefficient. To understand why, note from Eqs. (4)–(9) that so long as only interior planes are being decoupled, the new same-plane matrices (e.g., $\hat{\underline{\mathbf{H}}}_{p+1,p+1}$) remain Hermitian, and the new interplane coupling matrices remain Hermitian conjugates of one another, e.g., $\hat{\underline{\mathbf{H}}}_{p+1,p-1} = \hat{\underline{\mathbf{H}}}^{\dagger}_{p-1,p+1}$, thus saving matrix multiplications. In the no-spin-orbit case, intra- and interplane matrices for all interior planes are purely real, so all but the terminal layers can be decoupled using real arithmetic, where multiplications cost only one-fourth that of complex arithmetic. In the spin-orbit case, as shown in the Appendix, when the orbitals in a plane are ordered by spin quantum number, then atom, and finally spatial orbital, all interior-plane Hamiltonian blocks (both intra- and interplane) take a special form which persists under inversion and matrix addition and multiplication,

$$\underline{\mathbf{H}} = \begin{bmatrix} \underline{\mathbf{a}} & \underline{\mathbf{b}} \\ -\underline{\mathbf{b}}^* & \underline{\mathbf{a}}^* \end{bmatrix}, \tag{18}$$

so that renormalized blocks such as $\hat{\underline{\mathbf{H}}}_{p+1,p+1}$ and $\hat{\underline{\mathbf{H}}}_{p-1,p+1}$ retain this form as well. As a result, only one-half of each matrix need be computed using matrix multiplication [the remainder follows using $O(N^2)$ operations]. In contrast, "rolling up" the nanowire from one end to the other would introduce non-Hermitian diagonal blocks and complex arithmetic (without any special structure) from the very first step, destroying all of the special structure of the interior matrices.

Even greater savings can be achieved by repeated decoupling of alternate interior planes because this procedure preserves the sparse nature of the Hamiltonian blocks for as long as possible. We emphasize that these savings *do not* depend on the alternate planes being bulklike, as is the case in Ref. 6, which proposes decoupling of all layers of a given species. To see this, consider the initial decoupling step. If only planes of a single atomic type (anion or cation) are decoupled first, then the diagonal block inversions are either trivial (no-spin-orbit case) or nearly so (spin-orbit case) since there are no same-plane interatomic parameters. One multiplies these inverses by the interplane coupling matrices, which are likewise initially sparse, as discussed above. Decoupling every other remaining plane in the second step, and repeating this procedure until only the terminal layers remain coupled, maintains as much of the sparse nature of both the intra- and interplane coupling matrices for as long as possible. This procedure results in substantial computational savings because one-half of the atomic planes are decoupled with simple sparse-matrix operations, and a further one quarter are decoupled with a mixture of full- and sparse-matrix operations; only the final quarter of the nanowire requires full-matrix operations. The savings are often dramatic. In practice, decoupling the first three-fourths of the planes takes only about one-third of the total decoupling time.

Thus, the savings do not depend on all planes being bulklike, only on the fact that there are no same-plane interatomic couplings. For example, in a random-alloy AlGaAs nanowire, cation planes $p,(p+2),(p+4),\ldots$ all will be different, due to different atomic compositions and orderings. Likewise, in a biased nanowire, where the bias is treated in the conventional manner, as a same-atom, same-orbital interaction,[16–18] all cation planes are different (as are all anion planes) due to the varying potential. However, because there are no same-plane interatomic couplings in either case, the computational savings are the same. This optimized process demonstrates yet another inefficiency of rolling up the wire from one end to the other since that procedure produces full matrices from the beginning.

Finally, observe that the renormalization method[6] lends itself naturally to parallelization. Obviously, one can simultaneously decouple alternating planes independently. We have found the following parallelization scheme to be particularly efficient. For an N_{proc} processor computer, we divide the Hamiltonian into N_{proc} sets of atomic planes, assigning one set to each processor. Each processor only has the part of the full Hamiltonian for its own set of planes. We apply the optimized renormalization algorithm discussed above to each set until only the first planes of the sets remain connected. Thus, prior to the final renormalization, the first plane of set n now only couples to the first plane of set $(n-1)$ and the first plane of set $(n+1)$. This method reduces communication between the processors operating on the various sets of planes.

III. RESULTS

Figure 2 shows transmission and density of states results for a Si nanowire with axis along [100], including the spin-orbit interaction, as calculated with the optimized renormalization method presented above and MUMPS;[7] the wire dimensions are $L_x \times L_y \times L_z = 30 \times 2.1 \times 2.1$ nm^3. The $sp^3d^5s^*$ nearest-neighbor tight-binding model[13] is used for the calculations and the Si parameters are taken from Ref. 19. The curves lie atop one another because both methods give essentially the same results, with relative errors of 8.0×10^{-8} for the transmission and 8.4×10^{-7} for the density of states. The wire is unbiased and perfect, so sharp transmission steps occur as new channels open, exactly as expected. The density of states results correlate nicely with the transmission results, showing peaks as new channels open. This test establishes the reliability of the optimized renormalization method for nanowires modeled with realistic, multiband tight-binding approaches.

In Tables I and II, we compare the optimized renormalization method presented here to other methods for transmission calculation. All simulations were run on a 64 bit Sun Fire X4600 with 4×2.8 GHz Dual Core Opteron processors. Table I presents single-processor times (in seconds) to compute one energy point for the device wave function (or in the

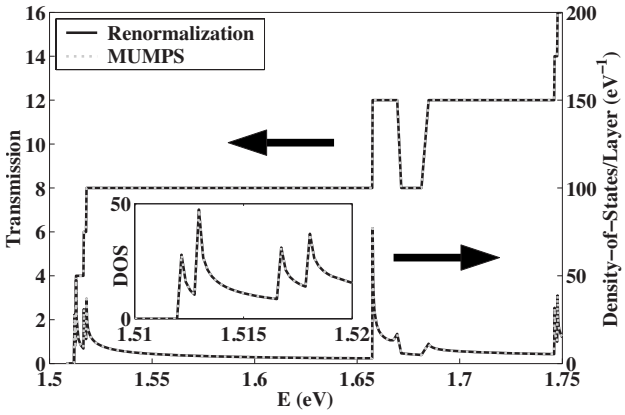

FIG. 2. Transmission and density of states (DOS) per layer for the $L_x \times L_y \times L_z = 30 \times 2.1 \times 2.1$ nm^3 Si nanowire with axis along [100] including spin-orbit coupling, as discussed in the text as calculated with the optimized renormalization method presented here and MUMPS (Ref. 7). The two curves are essentially the same, with maximum relative errors of 8.0×10^{-8} for the transmission curve and 8.4×10^{-7} for the density of states. Inset: Detailed DOS near the turn on of the nanowire.

case of the recursive Green function method, the Green function) after the open-system boundary conditions have been computed as well as processing time relative to that of optimized renormalization; results are listed as absolute/relative time. The wire has dimensions $L_x \times L_y \times L_z = 30 \times 2.1 \times 2.1$ nm^3, with transport axis along either [100] or [111], and we employ the $sp^3d^5s^*$ nearest-neighbor tight-binding model[13] with Si parameters taken from Ref. 19. For no-spin-orbit calculations, optimized renormalization is typically two to three times faster than other methods. For the [100] nanowire with spin orbit, MUMPS[7] is slightly faster, but for the [111] nanowire with spin orbit, optimized renormalization is slightly faster. Comparing the [111] to the [100] cases reveals that optimized renormalization takes only slightly longer (139 s vs 130 s), whereas MUMPS[7] takes significantly longer (172 s vs 117 s). Thus, for a single processor, optimized renormalization is superior for no-spin-orbit cases and on average at least as good as the best alternative when spin orbit is included.

It is in terms of parallel scalability where optimized renormalization clearly takes the lead. In Table II, we present execution times (in seconds) and parallel scalings for a circular nanowire with axis along [111], length $L=50$ nm, and diameter $d=4$ nm. The same tight-binding model is used but without spin-orbit coupling. Figure 3 shows the transmission and density of states results for this nanowire. As with the nanowire in Fig. 2, the two methods give essentially the same results. The optimized renormalization method shows an impressive parallel behavior, as expected from the discussion in Sec. II B. above. Discounting the anomalous performance of SuperLU$_{\text{dist}}$,[9] discussed below, the optimized renormalization method has the best parallel scaling behavior, in addition to being the fastest method of all. The two- and four-processor scalings come close to theoretical maxima, and even the eight-processor scaling is very impressive (5.85×).

We believe the superlinear performance of the SuperLU$_{\text{dist}}$ (Ref. 9) algorithm reported here (two processors: 2.9×; four processors: 5.4×) to be anomalous. We have found that on some single processors, it gives unusually poor performance and in these cases we have seen a superlinear speed improvement between one and two processors. Conversely, where the single-processor performance is reasonable, we have seen only the expected sublinear improvement. This processor dependence is likely due to the fact that SuperLU$_{\text{dist}}$ (Ref. 9) was designed specifically for parallel computers. As further evidence that the superlinear figures are the result of comparing

TABLE I. Execution times (measured in seconds) and execution times relative to the optimized renormalization for a silicon nanowire $L_x \times L_y \times L_z = 30 \times 2.1 \times 2.1$ nm^3 with axis along the [100] or [111] directions with (SO) or without (no SO) spin-orbit coupling. The test computation is one energy point once the open-system boundary conditions have been computed, for the device wave function (Refs. 7–10) or the Green function (Ref. 3) via the recursive Green Function (RGF) method. First row, N, is the size of the sparse matrix $[E-H-\Sigma]$. Entries are listed as absolute time (s)/relative time (dimensionless, to optimized renormalization). All simulations were run on a 64 bit Sun Fire X4600 with 4×2.8 GHz Dual Core Opteron processors.

	[100] no SO	[100] SO	[111] no SO	[111] SO
N	70400	140800	67200	134400
Optimized renormalization	11.2/1.0	130/1.0	11.3/1.0	139/1.0
Umfpak 5.0.1[a]	24.6/2.20	176/1.35	28.4/2.51	185/1.33
PARADISO[b]	27.7/2.47	147/1.13	63.5/5.62	447/3.22
SuperLU$_{\text{dist}}$[c]	40.6/3.62	203/1.56	58.7/5.19	420/3.02
MUMPS 4.6.3[d]	21.4/1.91	117/0.90	26/2.30	172/1.24
RGF[e]	95/8.48	708/5.45	104/9.20	754/5.42

[a]Reference 8.
[b]Reference 10.
[c]Reference 9.
[d]Reference 7.
[e]Reference 3.

TABLE II. Execution times (in seconds) and parallel scalings for a circular silicon nanowire (length $L=50$ nm; diameter $d=4$ nm) with axis along [111] without spin-orbit coupling. The test computation is one energy point once the open-system boundary conditions have been computed, for the device wave function (Refs. 7–10) or the Green function (Ref. 3) via the recursive Green function (RGF) method. The size of the sparse matrix $[E-H-\Sigma]$ is $N=319\,060$. Entries are listed as time (s)/speed (relative to one processor). All simulations were run on a 64 bit Sun Fire X4600 with 4×2.8 GHz Dual Core Opteron processors.

CPUs	1	2	4	8
Optimized renormalization	316/1×	162/1.95×	87.4/3.61×	54/5.85×
Umfpak 5.0.1[a]	1070	NA	NA	NA
PARADISO[b]	1660/1×	852/1.95×	475/3.5×	330/5.03×
$\text{SuperLU}_{\text{dist}}$[c]	2950/1×	1009/2.9×	547/5.4×	477/6.18×
MUMPS 4.6.3[d]	667/1×	362/1.84×	211/3.16×	160/4.16×
RGF[e]	3560	NA	NA	NA

[a]Reference 8.
[b]Reference 10.
[c]Reference 9.
[d]Reference 7.
[e]Reference 3.

reasonable multiple-processor performance to unusually poor single-processor results, we calculate from Table II the speed improvements between two and four processors (1.84×) and four and eight processors (1.15×). These results show the expected sublinear scaling. In comparison, for optimized renormalization, the two- to four-processor improvement is 1.85×, while that for four to eight processors is 1.62×. We therefore conclude that optimized renormalization has superior scaling, in addition to being the fastest method for no-spin-orbit calculations.

IV. CONCLUSIONS

We have optimized the renormalization method[6] for use in [100]- and [111]-axis nanowire transmission calculations. Our optimizations exploit mathematical properties of the Hamiltonian matrix arising from the relationships between the device geometry and the tight-binding model. One key result is that the open-system boundary conditions should be handled only as the last step since doing so allows us to exploit the mathematical structure of the Hamiltonian to the greatest extent possible. For no-spin-orbit models, our optimizations allow the use of real arithmetic for the vast majority of the transmission or wave function calculation, while for spin-orbit models, we reduce the computational effort by almost one-half via exploitation of the structure in the blocks of the Hamiltonian matrix. Furthermore, we have shown that the optimized renormalization method has an excellent parallel scaling behavior. Thus, optimized renormalization is well suited for modeling nanowires with realistic tight-binding models; its superior performance will aid in device design and simulation and clarification of nanodevice behavior.

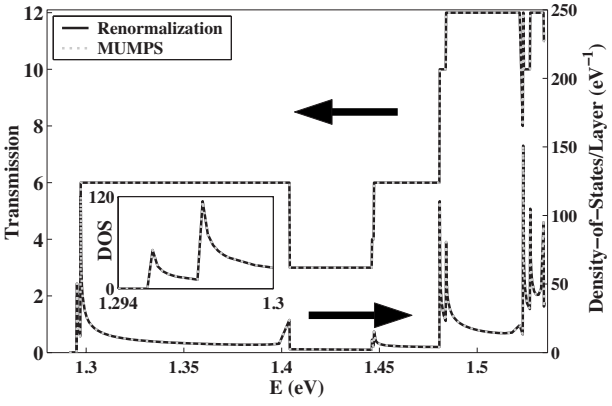

FIG. 3. Transmission and density of states per layer for the circular Si nanowire (length $L=50$ nm; diameter $d=4$ nm) with axis along [111] without spin-orbit coupling, as calculated with the optimized renormalization method presented here and MUMPS (Ref. 7). As with Fig. 2, the two curves are essentially the same. Inset: Detailed DOS near the turn on of the nanowire.

ACKNOWLEDGMENTS

We thank E. Ingram for expert help with the figures. This work was supported by the Semiconductor Research Corporation and the Army Research Office. M.L. acknowledges financial support from the Swiss National Science Foundation (NEQUATTRO SNF Project No. 200020-117613/1). The work described in this publication was carried out in part at the Jet Propulsion Laboratory, California Institute of Technology under a contract with the National Aeronautics and Space Administration, and Jet Propulsion Laboratory. nanohub.org computational resources provided by the Network for Computational Nanotechnology, funded by the National Science Foundation, were used.

APPENDIX

Here, we show that the special form taken by the Hamiltonian blocks in the spin-orbit case persists up to the last renormalization. To establish this assertion, we must show

that this form persists under (i) matrix inversion and (ii) multiplication of two matrices of this form.

When the spin-orbit interaction is treated as a p-orbital, same-atom only parameter, the Hamiltonian block for a single atom, l, takes the form

$$\begin{bmatrix} \underset{\sim}{\mathbf{H}}_l^{\uparrow\uparrow} & \underset{\sim}{\mathbf{H}}_l^{\uparrow\downarrow} \\ -\underset{\sim}{\mathbf{H}}_l^{\uparrow\downarrow *} & \underset{\sim}{\mathbf{H}}_l^{\uparrow\uparrow *} \end{bmatrix} = \begin{bmatrix} \underset{\sim}{\mathbf{h}}_d + \underset{\sim}{\mathbf{h}}_{so}^{\uparrow\uparrow} & \underset{\sim}{\mathbf{h}}_{so}^{\uparrow\downarrow} \\ -\underset{\sim}{\mathbf{h}}_{so}^{\uparrow\downarrow *} & \underset{\sim}{\mathbf{h}}_d + \underset{\sim}{\mathbf{h}}_{so}^{\uparrow\uparrow *} \end{bmatrix}, \quad \text{(A1)}$$

where the matrix $\underset{\sim}{\mathbf{h}}_d$ is real and diagonal in the orbitals. Because none of the additional orbitals used in our model has spin-orbit interactions, it suffices to specify the matrices in Eq. (A1) for the sp^3 basis; the extension to the $sp^3d^5s^*$ case is obvious. For the sp^3 basis, with orbital ordering $\{|s\rangle, |p_x\rangle, |p_y\rangle, |p_z\rangle\}$, the matrices appearing in Eq. (A1) are

$$\underset{\sim}{\mathbf{h}}_d = \begin{bmatrix} E_s & 0 & 0 & 0 \\ 0 & E_p & 0 & 0 \\ 0 & 0 & E_p & 0 \\ 0 & 0 & 0 & E_p \end{bmatrix}, \quad E_s, E_p \in \mathrm{Re}, \quad \text{(A2)}$$

$$\underset{\sim}{\mathbf{h}}_{so}^{\uparrow\uparrow} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -i\lambda & 0 \\ 0 & i\lambda & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \underset{\sim}{\mathbf{h}}_{so}^{\uparrow\downarrow} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda \\ 0 & 0 & 0 & -i\lambda \\ 0 & -\lambda & i\lambda & 0 \end{bmatrix},$$
$$\lambda \in \mathrm{Re}. \quad \text{(A3)}$$

From Eq. (A3), it is clear that $-(\underset{\sim}{\mathbf{h}}_{so}^{\uparrow\downarrow})^* = (\underset{\sim}{\mathbf{h}}_{so}^{\uparrow\downarrow})^\dagger$, as demanded by Hermiticity; the form adopted in Eq. (A1) is more useful for our purposes.

Next, consider the same-plane Hamiltonian block for an interior plane p of n atoms, where the basis is ordered $\{\{|\omega\uparrow;1\rangle\}, \ldots, \{|\omega\uparrow;n\rangle\}, \{|\omega\downarrow;1\rangle\}, \ldots, \{|\omega\downarrow;n\rangle\}\}$, and $\{|\omega\sigma;l\rangle\}$ denotes the full set of spatial orbitals ω of spin σ on atom l. (Since this is an interior plane, it does not incorporate the open-system boundary conditions. These appear only in the terminal planes.) From Eqs. (A1)–(A3), it is clear that prior to any renormalization, this block takes the form

$$\underset{\sim}{\mathbf{H}}_{(p,p)} = \begin{bmatrix} \underset{\sim}{\mathbf{H}}_{p,p}^{\uparrow\uparrow} & \underset{\sim}{\mathbf{H}}_{p,p}^{\uparrow\downarrow} \\ -\underset{\sim}{\mathbf{H}}_{p,p}^{\uparrow\downarrow *} & \underset{\sim}{\mathbf{H}}_{p,p}^{\uparrow\uparrow *} \end{bmatrix}, \quad \text{(A4)}$$

$$\underset{\sim}{\mathbf{H}}_{p,p}^{\uparrow\uparrow} = \mathrm{diag}[\underset{\sim}{\mathbf{H}}_1^{\uparrow\uparrow}, \ldots, \underset{\sim}{\mathbf{H}}_n^{\uparrow\uparrow}], \quad \underset{\sim}{\mathbf{H}}_{p,p}^{\uparrow\downarrow} = \mathrm{diag}[\underset{\sim}{\mathbf{H}}_1^{\uparrow\downarrow}, \ldots, \underset{\sim}{\mathbf{H}}_n^{\uparrow\downarrow}], \quad \text{(A5)}$$

where "diag" denotes a block-diagonal matrix, and the blocks in Eq. (A5) are defined in Eq. (A1). Note that the form of Eq. (A4) is the same as that of Eq. (A1).

To show that this form persists under inversion, we consider an invertible matrix $\underset{\sim}{\mathbf{h}}$ and its inverse,

$$\underset{\sim}{\mathbf{h}} = \begin{bmatrix} \underset{\sim}{\mathbf{a}} & \underset{\sim}{\mathbf{b}} \\ -\underset{\sim}{\mathbf{b}}^* & \underset{\sim}{\mathbf{a}}^* \end{bmatrix}, \quad \underset{\sim}{\mathbf{h}}^{-1} = \begin{bmatrix} \underset{\sim}{\boldsymbol{\alpha}} & \underset{\sim}{\boldsymbol{\beta}} \\ \underset{\sim}{\boldsymbol{\gamma}} & \underset{\sim}{\boldsymbol{\delta}} \end{bmatrix}, \quad \text{(A6)}$$

where all blocks are of the same size. Demanding that $\underset{\sim}{\mathbf{h}} \cdot \underset{\sim}{\mathbf{h}}^{-1} = \underset{\sim}{\mathbf{1}}$ results in four equations to be solved for the blocks of $\underset{\sim}{\mathbf{h}}^{-1}$. The result is

$$\underset{\sim}{\boldsymbol{\alpha}} = [\underset{\sim}{\mathbf{a}} + \underset{\sim}{\mathbf{b}}(\underset{\sim}{\mathbf{a}}^*)^{-1}\underset{\sim}{\mathbf{b}}^*]^{-1}, \quad \underset{\sim}{\boldsymbol{\beta}} = -\underset{\sim}{\mathbf{a}}^{-1}\underset{\sim}{\mathbf{b}}\underset{\sim}{\boldsymbol{\alpha}}^*, \quad \underset{\sim}{\boldsymbol{\delta}} = \underset{\sim}{\boldsymbol{\alpha}}^*,$$
$$\underset{\sim}{\boldsymbol{\gamma}} = -\underset{\sim}{\boldsymbol{\beta}}^*. \quad \text{(A7)}$$

Hence, the form of Eq. (A6) is preserved under inversion. The proof of persistence under multiplication is even simpler (under matrix addition, it is obvious),

$$\begin{bmatrix} \underset{\sim}{\mathbf{a}} & \underset{\sim}{\mathbf{b}} \\ -\underset{\sim}{\mathbf{b}}^* & \underset{\sim}{\mathbf{a}}^* \end{bmatrix} \begin{bmatrix} \underset{\sim}{\mathbf{c}} & \underset{\sim}{\mathbf{d}} \\ -\underset{\sim}{\mathbf{d}}^* & \underset{\sim}{\mathbf{c}}^* \end{bmatrix} = \begin{bmatrix} \underset{\sim}{\mathbf{a}}\underset{\sim}{\mathbf{c}} - \underset{\sim}{\mathbf{b}}\underset{\sim}{\mathbf{d}}^* & \underset{\sim}{\mathbf{a}}\underset{\sim}{\mathbf{d}} + \underset{\sim}{\mathbf{b}}\underset{\sim}{\mathbf{c}}^* \\ -\underset{\sim}{\mathbf{a}}^*\underset{\sim}{\mathbf{d}}^* - \underset{\sim}{\mathbf{b}}^*\underset{\sim}{\mathbf{c}} & \underset{\sim}{\mathbf{a}}^*\underset{\sim}{\mathbf{c}}^* - \underset{\sim}{\mathbf{b}}^*\underset{\sim}{\mathbf{d}} \end{bmatrix}. \quad \text{(A8)}$$

The persistence of this form throughout the renormalization process until the last step follows from observing that the matrices to be computed, such as,

$$\underset{\sim}{\mathbf{X}}_4 = \underset{\sim}{\mathbf{h}}_{4,4}^{-1}\underset{\sim}{\mathbf{H}}_{3,4}^\dagger, \quad \hat{\underset{\sim}{\mathbf{h}}}_{3,3} = \underset{\sim}{\mathbf{h}}_{3,3} - \underset{\sim}{\mathbf{H}}_{3,4}\underset{\sim}{\mathbf{X}}_4, \quad \hat{\underset{\sim}{\mathbf{H}}}_{3,5} = -\underset{\sim}{\mathbf{H}}_{3,4}\underset{\sim}{\mathbf{h}}_{4,4}^{-1}\underset{\sim}{\mathbf{H}}_{4,5}, \quad \text{(A9)}$$

where prior to the first renormalization the adjacent-plane coupling matrices are spin independent and block diagonal, $\underset{\sim}{\mathbf{H}}_{3,4} = \mathrm{diag}[\underset{\sim}{\mathbf{H}}_{3,4}^{\uparrow\uparrow}, \underset{\sim}{\mathbf{H}}_{3,4}^{\uparrow\uparrow}]$ and real. Note that this form is a special case of the form Eq. (A6). Because the matrices computed in Eq. (A9) all retain the form Eq. (A6), subsequent renormalizations will likewise retain this form until the last step, where the open-system boundary conditions enter the process. Therefore, only the upper half of each matrix need be computed, reducing the computational burden by almost one-half.

[1] T. B. Boykin, J. P. A. van der Wagt, and J. S. Harris, Jr., Phys. Rev. B **43**, 4777 (1991).
[2] D. Z.-Y. Ting, E. T. Yu, and T. C. McGill, Phys. Rev. B **45**, 3583 (1992).
[3] R. Lake, G. Klimeck, R. C. Bowen, and D. Jovanovic, J. Appl. Phys. **81**, 7845 (1997).
[4] M. P. Lopez Sancho, J. M. Lopez Sancho, and J. Rubio, J. Phys. F: Met. Phys. **15**, 851 (1985).
[5] M. Luisier, A. Schenk, W. Fichtner, and G. Klimeck, Phys. Rev. B **74**, 205323 (2006).
[6] G. Grosso, S. Moroni, and G. P. Parravicini, Phys. Rev. B **40**, 12328 (1989).
[7] P. R. Amestoy, I. S. Duff, and J.-Y. L'Excellent, Comput. Methods Appl. Mech. Eng. **184**, 501 (2000).
[8] T. A. Davis, ACM Trans. Math. Softw. **30**, 165 (2004).
[9] X. S. Li and J. W. Demmel, ACM Trans. Math. Softw. **29**, 110 (2003).
[10] O. Schenk and K. Gartner, FGCS, Future Gener. Comput. Syst. **20**, 475 (2004).
[11] G. Grosso, G. Pastori Parravicini, and C. Piermerocchi, Phys. Rev. B **61**, 15585 (2000).
[12] A. Cresti, R. Farchioni, G. Grosso, and G. P. Parravicini, Phys.

Rev. B **68**, 075306 (2003).
[13] J.-M. Jancu, R. Scholz, F. Beltram, and F. Bassani, Phys. Rev. B **57**, 6493 (1998).
[14] T. B. Boykin, Phys. Rev. B **54**, 7670 (1996).
[15] D. J. Chadi, Phys. Rev. B **16**, 790 (1977).
[16] M. Graf and P. Vogl, Phys. Rev. B **51**, 4940 (1995).
[17] T. B. Boykin, R. C. Bowen, and G. Klimeck, Phys. Rev. B **63**, 245314 (2001).
[18] T. B. Boykin and P. Vogl, Phys. Rev. B **65**, 035202 (2001).
[19] T. B. Boykin, G. Klimeck, and F. Oyafuso, Phys. Rev. B **69**, 115201 (2004).