

Scaling of the energy gap in pattern-hydrogenated graphene: Supporting Information

Roberto Grassi,^{1,2,*} Tony Low,^{1,†} and Mark Lundstrom¹

¹*School of Electrical & Computer Engineering,
Purdue University, West Lafayette, Indiana 47906, USA*

²*ARCES and DEIS, University of Bologna,
Viale Risorgimento 2, 40136 Bologna, Italy*

(Dated: October 12, 2011)

TRIANGULAR VERSUS HONEYCOMB SUPERLATTICE

While patterned hydrogenation has been experimentally demonstrated only for graphene on iridium, other substrate materials that can accommodate a monolayer graphene on their surface, e.g. ruthenium [1], could in principle be used. The resulting superlattice can be of the triangular or honeycomb type. Let m and n be the size of the supercell in units of the graphene and substrate lattice constants, respectively. Here it is demonstrated that a honeycomb superlattice is obtained whenever $m/n = (3M + 1)/(3N)$ with $M, N \in \mathbb{Z}^+$.

Figs. 1a and 1b show the unit cell for the honeycomb graphene lattice and the typical triangular substrate surface layer, respectively. According to the definition of m and n given above, we assume that an $m \times m$ graphene supercell is commensurate with an $n \times n$ substrate supercell. Therefore, denoting the lattice vectors of graphene, substrate, and superlattice by \mathbf{a}_j , \mathbf{b}_j , and \mathbf{c}_j ($j = 1, 2$), respectively, we have

$$\mathbf{c}_j = m\mathbf{a}_j = n\mathbf{b}_j. \quad (1)$$

Each pair (m, n) and its multiples map to a unique composite system, e.g. $(4, 3)$ is the same as $(8, 6)$. Hence, we consider only the case where m and n are prime to each other,

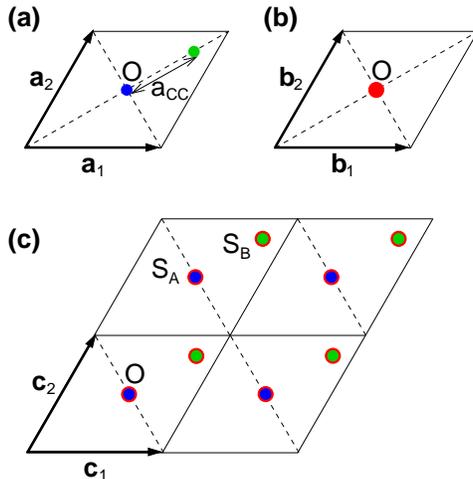


FIG. 1. (a) Graphene unit cell: the two carbon atoms are indicated with different colors. (b) Substrate unit cell. (c) Honeycomb superlattice generated by the superposition of the graphene and substrate lattices. S_A (S_B) is the point inside the supercell where a carbon atom of the A (B) graphene sublattice sits on top of a substrate atom.

so that \mathbf{c}_j are the primitive lattice vectors of the superlattice. We further assume that, at some point O inside the supercell, a carbon atom sits directly on top of a substrate atom: such arrangement was found to be an energetically stable configuration [2]. We indicate the superlattice points that are equivalent to O as S_A .

As shown by Fig. 1c, in order to generate a honeycomb superlattice, there must be another point S_B inside the supercell where a carbon atom belonging to the opposite sublattice sits on top of a substrate atom. Also, for the superlattice to be regular, it can be derived that the distance between S_A and S_B must be equal to $|\mathbf{c}_1 + \mathbf{c}_2|/3$. Inspecting the graphene lattice tells us that S_A and S_B have to be separated by a distance of $(3M + 1)a_{CC}$, where $M \in \mathbb{Z}^+$ and a_{CC} is the carbon-carbon distance. Hence, we can write

$$\frac{1}{3} |\mathbf{c}_1 + \mathbf{c}_2| = \frac{m}{3} |\mathbf{a}_1 + \mathbf{a}_2| = (3M + 1)a_{CC} \quad \Rightarrow \quad m = 3M + 1, \quad M \in \mathbb{Z}^+. \quad (2)$$

In a similar fashion for the substrate, the S_A and S_B have to be separated by a distance of Na_{SS} , where $N \in \mathbb{Z}^+$ and a_{SS} is the interatomic distance of the substrate. We can then write

$$\frac{1}{3} |\mathbf{c}_1 + \mathbf{c}_2| = \frac{n}{3} |\mathbf{b}_1 + \mathbf{b}_2| = Na_{SS} \quad \Rightarrow \quad n = 3N, \quad N \in \mathbb{Z}^+. \quad (3)$$

In conclusion, the superlattice with the similar honeycomb structure as graphene, shown in Fig. 1c, can be generated by satisfying Eqs. 2-3. Otherwise, the superlattice would produce a triangular structure instead, with only repeated units of S_A . Examples of honeycomb superlattices are the cases $m/n = 10/9$ (graphene on iridium, also indicated as SL10 in the manuscript) and $m/n = 13/12$ (indicated as SL13 in the manuscript).

MODEL FOR PATTERNED HYDROGENATION

The model for generating the hydrogen clusters is described below.

Given a sample of pristine graphene on a certain substrate, we divide the structure in supercells, the supercell subdivision being chosen so that each supercell contains one S_A and one S_B point in symmetric positions (see Fig. 1a of the main text for an illustration: the centers of the two circles correspond to an S_A or S_B point). Each carbon atom can be denoted by the pair of indexes (l, i) , where l is the index of the half-supercell to which it belongs and i is the atom index inside the whole supercell. We introduce a binary random

variable $Z_{l,i} \in \{0, 1\}$ for each carbon atom: the atom is hydrogenated when $Z_{l,i} = 1$. $Z_{l,i}$ in turn is written as the product of other two binary random variables X_l and Y_i , whose probability distributions are given below.

Y_i , controls the cluster formation inside each supercell. We propose the following formula for the probability $P(Y_i = 1)$:

$$P(Y_i = 1) \equiv f(w_i), \quad f(w_i = 0) = 0, \quad \frac{df}{dw_i} \geq 0, \quad (4)$$

w_i being a quantity defined for each carbon atom as

$$w_i = \frac{d_i \left| d_i - \frac{1}{3} \sum_{\langle j \rangle} d_j - c \right|}{a_{CC}^2}, \quad (5)$$

where d_i is the xy -plane distance between the carbon atom of index i and its nearest-neighbor substrate atom (let \mathbf{r} be the position vector in the xy -plane parallel to the surface),

$$d_i = \min_k |\mathbf{r}_{C_i} - \mathbf{r}_{S_k}|, \quad (6)$$

the summation over j is restricted to the three carbon atoms that are nearest neighbor to the carbon atom of index i , and c is simply a constant,

$$c = \frac{a_{SS}}{\sqrt{3}} - \sqrt{\frac{a_{SS}^2}{3} + a_{CC}^2 - \frac{a_{SS}a_{CC}}{\sqrt{3}}}. \quad (7)$$

Eqs. 4–5 can be justified by the following considerations. Experimentally, the hydrogen clusters tend to form around the regions where one graphene sublattice is located on top of substrate atoms, while the other sublattice is far from substrate atoms and can bind to hydrogen atoms on the opposite face [3]. This translates in two conditions for the generic carbon atom to be hydrogenated. First, it should be located in between substrate lattice sites. The probability for adsorption would then increase as its distance d_i from the nearest-neighbor substrate atom increases. This effect is captured by the prefactor in (5). However, if the considered carbon atom is located at the maximum distance from substrate atoms, equal to $a_{SS}/\sqrt{3}$, the probability for adsorption should distinguish between the case in which the three nearest-neighbor carbon atoms are located close to substrate atoms (high probability, Fig. 2a) and the case in which also the three nearest neighbors are between substrate atoms (low probability, Fig. 2b). We can note that in the first case $d_i \gg \frac{1}{3} \sum_{\langle j \rangle} d_j$, while in the second case $d_i \sim \frac{1}{3} \sum_{\langle j \rangle} d_j$. This explains the second factor in (5), where the constant c serves only to set the probability to 0 for the worst case (Fig. 2b).

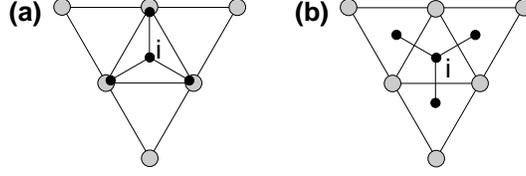


FIG. 2. Best (a) and worst (b) cases for the probability of hydrogenation of a carbon atom of index i located at a distance $d_i = a_{\text{SS}}/\sqrt{3}$ from the nearest-neighbor substrate atoms. Carbon (substrate) atoms are represented with black (gray) balls.

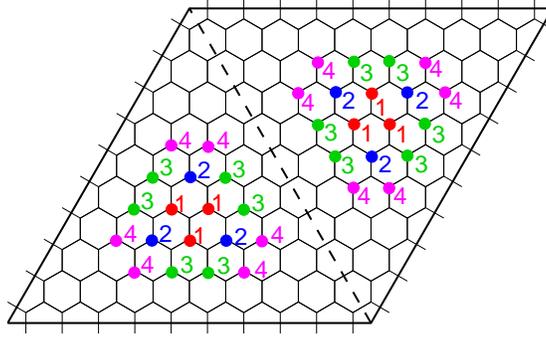


FIG. 3. Supercell of graphene on iridium: location of the carbon atoms with the four largest values of w_i , i.e. $w_i = w^{(1)}$, $w_i = w^{(2)}$, etc.

In (4) we have omitted the actual functional dependence of $P(Y_i = 1)$ on w_i . Since this relationship depends on the physical hydrogenation process and it is unknown, we choose here a simple cut-off model. For a given superlattice unit cell, all the possible values of w_i are computed and labeled in decreasing order as $w^{(1)}, w^{(2)}, \dots$ (the location of the corresponding carbon atoms is shown in Fig. 3 for the case of iridium substrate). Then, the probability $P(Y_i = 1)$ is assigned as

$$P(Y_i = 1) \equiv f(w_i) = \begin{cases} 1 & \text{if } w_i = w^{(j)} \text{ with } j < N_w, \\ 0.5 & \text{if } w_i = w^{(j)} \text{ with } j = N_w, \\ 0 & \text{if } w_i = w^{(j)} \text{ with } j > N_w. \end{cases} \quad (8)$$

With this method, a cluster of hydrogen atoms is formed around the sites where the quantity w_i tends to grow (i.e. around S_A and S_B). The input parameter N_w controls the size of this cluster. The disorder is only located at the cluster edges.

X_I , instead, is used to generate the superlattice disorder, which consists in some hydrogen

clusters being randomly missing from the superlattice. The probability $P(X_l = 1)$ is set equal to the input parameter n_c , with $0 \leq n_c \leq 1$, which therefore assumes the meaning of the ratio between the average number of hydrogenated half-supercells (or equivalently average number of clusters) and the total number of half-supercells.

The hydrogenation model described above produces clusters inside which only one graphene sublattice is hydrogenated. This leads to the formation of midgap states in the electronic structure, associated with dangling bonds. However, these states are believed to be an artefact of the TB model, due to the fact that bond relaxation is neglected. To avoid this, after hydrogen atoms are generated according to the method described above, a final step is introduced: additional hydrogen atoms are placed on top of the carbon atoms that have at least two nearest neighbors being hydrogenated.

CALCULATION OF MOMENTUM-ENERGY RESOLVED DENSITY OF STATES

We consider a sample composed of N_1 and N_2 graphene unit cells along the directions of \mathbf{a}_1 and \mathbf{a}_2 , respectively, with periodic boundary conditions on both directions. N_1 and N_2 are chosen to be multiples of m , the size of the supercell in units of the graphene lattice constant, so that the sample contains exactly $\frac{N_1}{m} \times \frac{N_2}{m}$ supercells. The sample is then hydrogenated as described in the previous section. An example, obtained with $N_1 = N_2 = 20$, $m/n = 10/9$ (graphene on iridium), $n_c = 0.75$, and $N_w = 4$, is shown in Fig. 4a. The actual samples that are simulated are much larger: $N_1 = N_2 = 120$ is used for the SL10 case, while $N_1 = N_2 = 117$ for the SL13 case. Such values have been checked to be large enough to ensure the convergence of $A(\mathbf{k}, \mathbf{k}; E)$ (see next section).

The generic orbital of the TB representation can be indicated as $|\mathbf{l}, q\rangle$, where \mathbf{l} is the graphene lattice vector, i.e. $\mathbf{l} = \sum_i n_i \mathbf{a}_i$ with $n_i = 1, \dots, N_i$, and q is the orbital index within the cell ($q = 1, 2$ for the A and B carbon orbitals, respectively). This is the real space representation. However, one could also work in the k -space representation, which is obtained by restricting $q = 1, 2$ (projected space for carbon atoms only) and by using as basis the set $\{|\mathbf{k}, q\rangle\}$ defined by $\langle \mathbf{l}, q_1 | \mathbf{k}, q_2 \rangle = \delta_{q_1, q_2} e^{i\mathbf{k} \cdot \mathbf{l}} / \sqrt{N_1 N_2}$. The \mathbf{k} vectors are discrete because a finite volume is considered: if \mathbf{d}_i are the primitive vectors of the reciprocal lattice, i.e. $\mathbf{a}_i \cdot \mathbf{d}_j = 2\pi\delta_{i,j}$, we have $\mathbf{k} = (m_1/N_1)\mathbf{d}_1 + (m_2/N_2)\mathbf{d}_2$, where $m_1, m_2 \in \mathbb{Z}$ (it can be proved that the K point is included in the grid if both N_1 and N_2 are multiple of 3).

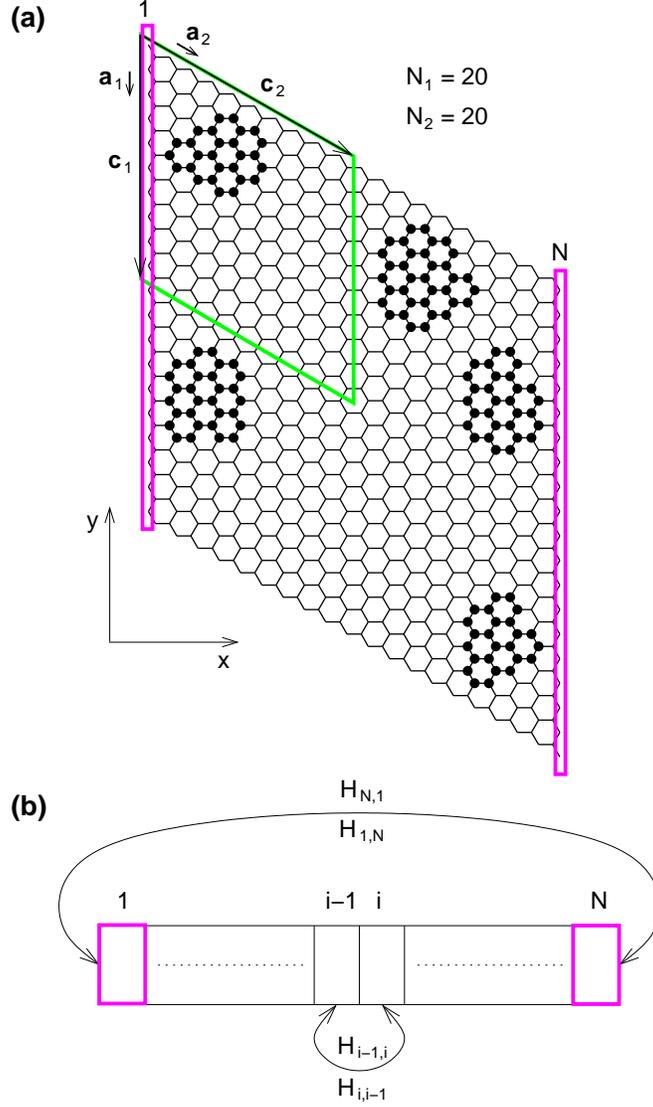


FIG. 4. (a) Example of sample used for calculating the momentum-energy resolved density of states. The structure is made of $N_1 \times N_2$ graphene unit cells, with N_1 and N_2 being chosen so that the sample is a multiple of the superlattice unit cell (region enclosed by the green line). Periodic boundary conditions are applied on both the \mathbf{a}_1 and \mathbf{a}_2 directions. (b) The same structure can be viewed as a linear chain of $N = 2N_2$ slabs (each slab corresponding to a row of atoms) with a periodic closure at the two ends.

Also, since the set of \mathbf{l} vectors is discrete, it follows that only a number $N_1 N_2$ of \mathbf{k} vectors, spanning a Brillouin zone, give rise to independent basis vectors.

Let H be the electron Hamiltonian. We recall that the spectral function at the energy

E is defined as $A = i(G^r - G^a)$, where $G^r = [(E + i\eta)I - H]^{-1}$ is the retarded Green's function and $G^a = G^{r\dagger}$ is the advanced one (η is an infinitesimal positive quantity). While the diagonal elements of A in real space have the meaning of a density of states in energy and physical space, its diagonal elements in k -space give the density of states in energy and momentum, which in turn corresponds to the physical quantity measured by ARPES. The diagonal elements of A in k -space can be expanded as

$$\begin{aligned} A(\mathbf{k}, q; \mathbf{k}, q; E) &= \sum_{\mathbf{l}_1, q_1} \sum_{\mathbf{l}_2, q_2} \langle \mathbf{k}, q | \mathbf{l}_1, q_1 \rangle A(\mathbf{l}_1, q_1; \mathbf{l}_2, q_2; E) \langle \mathbf{l}_2, q_2 | \mathbf{k}, q \rangle \\ &= \frac{1}{N_1 N_2} \sum_{\mathbf{l}_1} \sum_{\mathbf{l}} e^{-i\mathbf{k} \cdot \mathbf{l}} A(\mathbf{l}_1, q; \mathbf{l}_1 - \mathbf{l}, q; E), \end{aligned} \quad (9)$$

where we recognize a discrete Fourier transform with respect to the relative variable $\mathbf{l} = \mathbf{l}_1 - \mathbf{l}_2$. We define

$$A(\mathbf{k}, \mathbf{k}; E) = \frac{A_s}{(2\pi)^2} \sum_q A(\mathbf{k}, q; \mathbf{k}, q; E), \quad (10)$$

where $A_s = a_{\text{CC}}^2 3\sqrt{3}/2$ is the area of the graphene unit cell, so that $A(\mathbf{k}, \mathbf{k}; E)/(2\pi)$ gives the number of states per unit energy, per unit \mathbf{k} , and per graphene unit cell (apart from spin degeneracy). The calculation of $A(\mathbf{k}, \mathbf{k}; E)$ is thus performed by first computing the spectral function in real space and then Fourier transforming according to (9-10).

A method to compute A in real space is by diagonalization of H . Indeed, if $\{|\psi_\alpha\rangle\}$ are the orthonormal eigenstates of H with corresponding eigenvalues $\{\epsilon_\alpha\}$, we have

$$\begin{aligned} A(\mathbf{l}_1, q_1; \mathbf{l}_2, q_2; E) &= \sum_\alpha \frac{2\eta}{(E - \epsilon_\alpha)^2 + \eta^2} \psi_\alpha(\mathbf{l}_1, q_1) \psi_\alpha^*(\mathbf{l}_2, q_2) \\ &\xrightarrow{\eta \rightarrow 0} 2\pi \sum_\alpha \delta(E - \epsilon_\alpha) \psi_\alpha(\mathbf{l}_1, q_1) \psi_\alpha^*(\mathbf{l}_2, q_2), \end{aligned} \quad (11)$$

with $\psi_\alpha(\mathbf{l}, q) = \langle \mathbf{l}, q | \psi_\alpha \rangle$ the generic eigenfunction in real space. The numerical computation of (11) can be efficiently done by setting a finite value of η and by finding for each energy E the eigenvalues (and corresponding eigenvectors) that are closest to it, using well known methods for large and sparse eigenvalue problems [4]. Nevertheless, we propose here an alternative method based on Green's functions. Although in the case considered here our method does not improve the computational time with respect to the diagonalization technique, because almost all the off-diagonal of the spectral function in real space need to be calculated, the method could be interesting in other situations, such as for the calculation

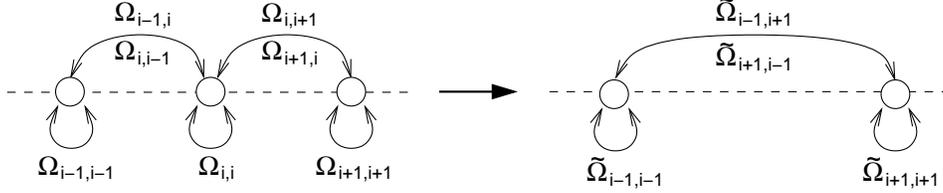


FIG. 5. Pictorial representation of the renormalization method in [5]: the circles represent nodes (or slabs) and the arrows the different types of coupling between them. The method consists in eliminating node i , while leaving unchanged the solution for all the remaining nodes by a proper renormalization of the matrix blocks of Ω .

of the local density of states in real space, where only few elements of the spectral function are needed. The method could also be useful when the storage of the eigenvectors is a major problem.

As illustrated in Fig. 4a, we divide the sample in $N = 2N_2$ slabs along the \mathbf{a}_2 direction so that each slab corresponds to a row of atoms (the choice between \mathbf{a}_1 and \mathbf{a}_2 is arbitrary). The structure is therefore of the type in Fig. 4b: a linear chain of slabs with a periodic boundary conditions at the two ends. Using block matrix notation in real space, $\Omega = (E + i\eta)I - H$ has the form

$$\Omega = \begin{pmatrix} \Omega_{1,1} & \Omega_{1,2} & & & \Omega_{1,N} \\ \Omega_{2,1} & \Omega_{2,2} & \Omega_{2,3} & & \\ & \Omega_{3,2} & \ddots & \ddots & \\ & & \ddots & \ddots & \Omega_{N-1,N} \\ \Omega_{N,1} & & & \Omega_{N,N-1} & \Omega_{N,N} \end{pmatrix}, \quad (12)$$

where each block represents the coupling between a pair of slabs. We notice that only the elements of the spectral function that connect orbitals belonging to the same graphene sublattice (i.e. same q) are needed in (9-10). From Fig. 4a, it can be seen that there is a correspondence between $q = 1, 2$ and the odd and even slabs, respectively. Therefore, only the matrix block of A (and thus of G^r) connecting slabs with the same parity need to be calculated. In order to avoid the calculation of the unnecessary matrix blocks, the renormalization method [5] is employed: an equivalent Ω^{odd} (Ω^{even}) matrix for the odd (even) slabs alone is computed by decimating the even (odd) ones. We recall that the decimation of a single node i from a linear chain, as depicted in Fig. 5, is achieved by renormalizing the

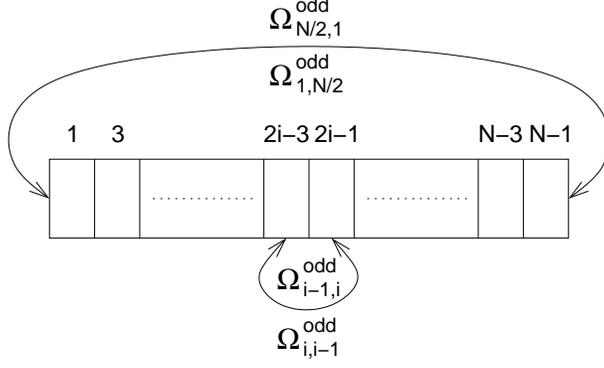


FIG. 6. Same structure as in Fig. 4b, where the even slabs have been decimated.

matrix blocks of Ω for the adjacent nodes according to the formulas [5]:

$$\begin{aligned}
 \tilde{\Omega}_{i-1,i-1} &= \Omega_{i-1,i-1} - \Omega_{i-1,i}(\Omega_{i,i})^{-1}\Omega_{i,i-1}, \\
 \tilde{\Omega}_{i-1,i+1} &= -\Omega_{i-1,i}(\Omega_{i,i})^{-1}\Omega_{i,i+1}, \\
 \tilde{\Omega}_{i+1,i-1} &= -\Omega_{i+1,i}(\Omega_{i,i})^{-1}\Omega_{i,i-1}, \\
 \tilde{\Omega}_{i+1,i+1} &= \Omega_{i+1,i+1} - \Omega_{i+1,i}(\Omega_{i,i})^{-1}\Omega_{i,i+1}.
 \end{aligned} \tag{13}$$

By repeated use of (13), the following algorithm can be derived to compute the renormalized matrix Ω^{odd} for the odd slabs alone (Fig. 6):

1. for $i = 1, \dots, N/2$ initialize

$$\Omega_{i,i}^{\text{odd}} = \Omega_{2i-1,2i-1}; \tag{14}$$

2. for $i = 1, \dots, N/2$

- set $j = 2i$, $k = \text{mod}(i, \frac{N}{2}) + 1$, $l = \text{mod}(j, N) + 1$
- compute

$$\begin{aligned}
 \Omega_{i,i}^{\text{odd}} &= \Omega_{i,i}^{\text{odd}} - \Omega_{j-1,j}(\Omega_{j,j})^{-1}\Omega_{j,j-1}, \\
 \Omega_{k,k}^{\text{odd}} &= \Omega_{k,k}^{\text{odd}} - \Omega_{l,j}(\Omega_{j,j})^{-1}\Omega_{j,l}, \\
 \Omega_{i,k}^{\text{odd}} &= -\Omega_{j-1,j}(\Omega_{j,j})^{-1}\Omega_{j,l}, \\
 \Omega_{k,i}^{\text{odd}} &= -\Omega_{l,j}(\Omega_{j,j})^{-1}\Omega_{j,j-1}.
 \end{aligned} \tag{15}$$

A similar algorithm can be derived for Ω^{even} . Both Ω^{odd} and Ω^{even} have the same shape as Ω (Eq. 12), but with $N \rightarrow N/2$. In the following, we will therefore refer to Ω for brevity, implicitly assuming that what we say must be applied separately to Ω^{odd} and Ω^{even} .

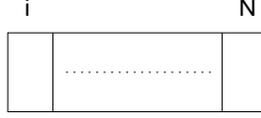


FIG. 7. Structure corresponding to $g^{r\mathcal{R},(i)}$.

The retarded Green's function at the energy E is obtained by inverting Ω . Here we present an algorithm for recursively calculating the blocks of G^r , extending the one in [6] for the case of $\Omega_{1,N}, \Omega_{N,1} \neq 0$. The algorithm is based on Dyson's equations. We recall that if the Hamiltonian is expressed as $H = H_0 + H_1$ (H_0 is called the unperturbed Hamiltonian and H_1 the perturbation one), the Dyson equations are given by $G^r = G_0^r + G_0^r H_1 G^r = G_0^r + G^r H_1 G_0^r$, where G_0^r is the retarded Green's function corresponding to H_0 . While the formulas that are presented here directly exploit time reversal symmetry, i.e. the fact that $G^r = (G^r)^T$, their extension to the general case is straightforward.

The first part of the algorithm consists in the calculation of certain blocks of $g^{r\mathcal{R},(i)}$ for $i = 1, \dots, N$, where $g^{r\mathcal{R},(i)}$ is the retarded Green's function corresponding to the structure composed of only the nodes from i to N without the periodic closure (Fig. 7). Indeed, by applying Dyson's equations to the structure in Fig. 7 with the coupling blocks $-\Omega_{i,i+1}$ and $-\Omega_{i+1,i}$ treated as the perturbation Hamiltonian, it is possible to relate $g^{r\mathcal{R},(i)}$ to $g^{r\mathcal{R},(i+1)}$ and derive the following equations:

1. initialize

$$g_{N,N}^{r\mathcal{R},(N)} = (\Omega_{N,N})^{-1}; \quad (16)$$

2. for $i = N - 1, \dots, 1$ compute

$$g_{i,i}^{r\mathcal{R},(i)} = (\Omega_{i,i} - \Omega_{i,i+1} g_{i+1,i+1}^{r\mathcal{R},(i+1)} \Omega_{i+1,i})^{-1}, \quad (17)$$

$$g_{i,N}^{r\mathcal{R},(i)} = -g_{i,i}^{r\mathcal{R},(i)} \Omega_{i,i+1} g_{i+1,N}^{r\mathcal{R},(i+1)}, \quad (18)$$

$$g_{N,i}^{r\mathcal{R},(i)} = (g_{i,N}^{r\mathcal{R},(i)})^T, \quad (19)$$

$$g_{N,N}^{r\mathcal{R},(i)} = g_{N,N}^{r\mathcal{R},(i+1)} - g_{N,i+1}^{r\mathcal{R},(i+1)} \Omega_{i+1,i} g_{i,N}^{r\mathcal{R},(i)}. \quad (20)$$

The blocks $g_{N,N}^{r\mathcal{R},(i)}$ for $i > 1$ can be discarded. The second part of the algorithm is obtained by applying again Dyson's equations, but to the original structure in Fig. 4b, with the perturbation Hamiltonian given by the coupling blocks $-\Omega_{i-1,i}$, $-\Omega_{i,i-1}$, $-\Omega_{N,1}$, and $-\Omega_{1,N}$. The formulas are as follows

3. initialize

$$G_{1,1}^r = \left[I + g_{1,N}^{r\mathcal{R},(1)} \Omega_{N,1} - g_{1,1}^{r\mathcal{R},(1)} \left(I + \Omega_{1,N} g_{N,1}^{r\mathcal{R},(1)} \right)^{-1} \Omega_{1,N} g_{N,N}^{r\mathcal{R},(1)} \Omega_{N,1} \right]^{-1} \times \\ \times \left[g_{1,1}^{r\mathcal{R},(1)} - g_{1,1}^{r\mathcal{R},(1)} \left(I + \Omega_{1,N} g_{N,1}^{r\mathcal{R},(1)} \right)^{-1} \Omega_{1,N} g_{N,1}^{r\mathcal{R},(1)} \right]; \quad (21)$$

4. for $i = 1, \dots, N$

– if $i > 1$ compute

$$G_{i,i}^r = g_{i,i}^{r\mathcal{R},(i)} - g_{i,i}^{r\mathcal{R},(i)} \Omega_{i,i-1} G_{i-1,i}^r - g_{i,N}^{r\mathcal{R},(i)} \Omega_{N,1} G_{1,i}^r, \quad (22)$$

– for $j = i + 1, \dots, N$ compute

$$G_{i,j}^r = -G_{i,j-1}^r \Omega_{j-1,j} g_{j,j}^{r\mathcal{R},(j)} - G_{i,1}^r \Omega_{1,N} g_{N,j}^{r\mathcal{R},(j)}, \quad (23)$$

$$G_{j,i}^r = (G_{i,j}^r)^T. \quad (24)$$

As soon as each block of G^r is computed, its contribution to (9) can be evaluated and the block is ready for being discarded unless is used later by the algorithm. It can be checked that the first row of G^r has to be fully saved, while $G_{i,i}^r$ is needed to calculate $G_{i,j+1}^r$ and $G_{i,i+1}^r$ to calculate $G_{i+1,i+1}^r$. In conclusion, compared to the direct inversion of the matrix in (12), the proposed algorithm allows to reduce of about a factor of 4 the number of blocks that are calculated, due to both the decimation of nodes with different parity and the exploitation of time reversal symmetry; moreover, the blocks are recursively calculated one after the other, thus saving memory. We note that our algorithm could also be used for partial inversion of (12), as in the case of the calculation of the density of states in energy, where only the diagonal elements of the spectral function in real space are required: in this case, for $i > 1$, (23) can be limited to $j = i + 1$ and (24) is not required. Regarding the stability of the overall algorithm, it should be pointed out that a value of $\eta = 10^{-3}$ eV was necessary in the simulations to avoid numerical artefacts; however, the corresponding broadening introduced in the $A(\mathbf{k}, \mathbf{k}; E)$ plot is way much smaller than the one due to disorder.

CONVERGENCE STUDY W.R.T. SAMPLE SIZE (AND ENSEMBLE SIZE)

In order to account for disorder across different cells of the superlattice, each sample has to be large enough so that it contains a sufficient number of clusters and the effect of the

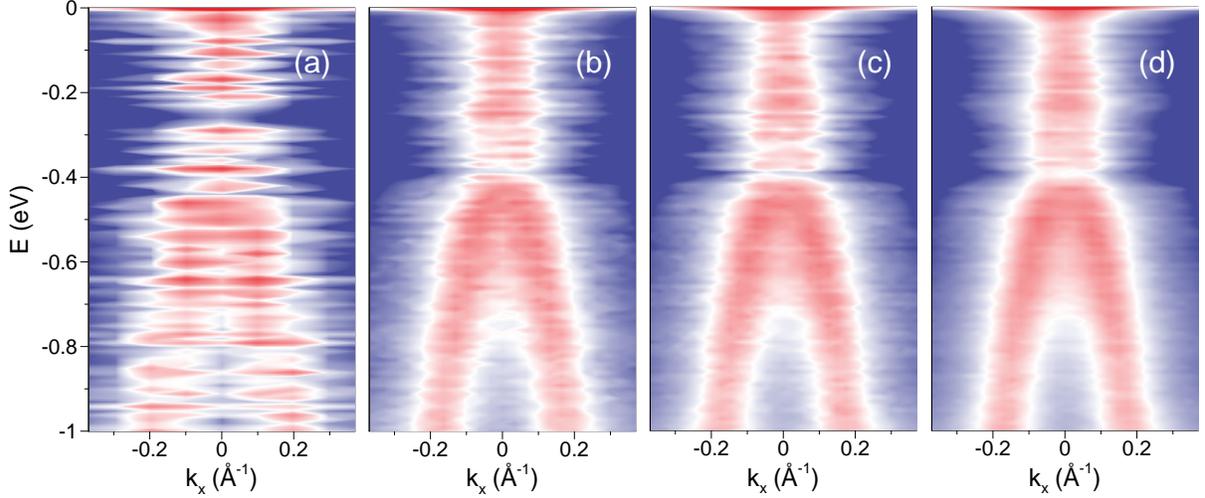


FIG. 8. Averaged $A(\mathbf{k}, \mathbf{k}; E)$ for ensembles with different sample size or ensemble size, all generated with the same supercell size (SL10), filling factor ($n_c = 0.75$), and cluster size ($N_w = 4$): (a) $N_1 \times N_2 = 30 \times 30$, 20 samples; (b) 90×90 , 20 samples; (c) 120×120 , 20 samples; (d) 120×120 , 50 samples. The color scale and path in k -space are the same as in the main text.

periodic boundary condition is washed out. In addition, the size of the sample determines the discretization step in k -space (the total number of \mathbf{k} points along the red line in the inset of Fig. 1c of the main text is equal to N_2): for a high-quality plot of $A(\mathbf{k}, \mathbf{k}; E)$, each sample has to be large enough so that the grid in k -space is sufficiently fine.

In Figs. 8, we report a comparison of $A(\mathbf{k}, \mathbf{k}; E)$ plots obtained by varying the sample size or the ensemble size. For the case with $N_1 \times N_2 = 30 \times 30$ and 20 samples (Fig. 8a), the number of \mathbf{k} points is small and the dispersion looks quite vague (also note that this is a zoomed view around the K point, so that only about 1/4 out of the N_2 points along the path are shown). Interestingly, the number of clusters seems to be already large enough to destroy the superlattice band structure: no repeated bands are visible, while the states are clearly arising from the graphene Dirac cone. The plot greatly improves when the size of the samples is increased to 90×90 (Fig. 8b). However, further increasing the sample size (Fig. 8c) or the number of samples (Fig. 8d), gives only a slight improvement, which means that the result has already well converged.

PROCEDURE FOR BAND-GAP EXTRACTION

The band gap is extracted from each (ensemble-averaged) $A(\mathbf{k}, \mathbf{k}; E)$ plot using a fitting technique. We recall that the path in k -space is the one shown in the inset of Fig. 1c in the main text, so that $\mathbf{k} = (k_x, K_y)$, where K_y is the k_y -coordinate of the K point. Since k_y is fixed, we use the simplified notation $A(k_x, k_x; E)$. The fitting procedure is composed of the following steps.

1. Manually choose a range of energies $[E_1, E_2]$ where to apply the fitting.
2. Find for each energy $E \in [E_1, E_2]$ the k_x coordinate where the intensity is maximum, separately for positive and negative k_x :

$$k_x^+(E) \text{ such that } A(k_x^+, k_x^+; E) = \max_{k_x \geq 0} A(k_x, k_x; E), \quad (25)$$

$$k_x^-(E) \text{ such that } A(k_x^-, k_x^-; E) = \max_{k_x \leq 0} A(k_x, k_x; E). \quad (26)$$

3. Compute for each $E \in [E_1, E_2]$ the values $w^+(E)$ and $w^-(E)$ as follows

$$w^+(E) = \frac{A(k_x^+(E), k_x^+(E); E)}{\max_{E' \in [E_1, E_2]} A(k_x^+(E), k_x^+(E); E')}, \quad (27)$$

$$w^-(E) = \frac{A(k_x^-(E), k_x^-(E); E)}{\max_{E' \in [E_1, E_2]} A(k_x^-(E), k_x^-(E); E')}. \quad (28)$$

4. Apply a least-square fitting to the set of points $\{(E, k_x^+(E))\} \cup \{(E, k_x^-(E))\}$ with $E \in [E_1, E_2]$, by using $w^+(E)$ and $w^-(E)$ as weights and one of the following dispersion relations as fitting curve:

$$E = \pm \left(\hbar v |k_x| + \frac{E_g}{2} \right), \quad (29)$$

$$E = \pm \left(\frac{\hbar^2 k_x^2}{2m} + \frac{E_g}{2} \right), \quad (30)$$

$$E = \pm \sqrt{\frac{\hbar^2 E_g k_x^2}{2m} + \left(\frac{E_g}{2} \right)^2}. \quad (31)$$

The result of the fitting for negative E is shown in Fig. 9, superimposed to the original $A(\mathbf{k}, \mathbf{k}; E)$ plot.

For each $A(\mathbf{k}, \mathbf{k}; E)$ plot, a measure of the broadening is also extracted. We consider a specific k_x value, k_x^B , and compute the quantity $2B$ as the difference between the two energies at which the function $A(k_x^B, k_x^B; E)$ decreases to half of its maximum value.

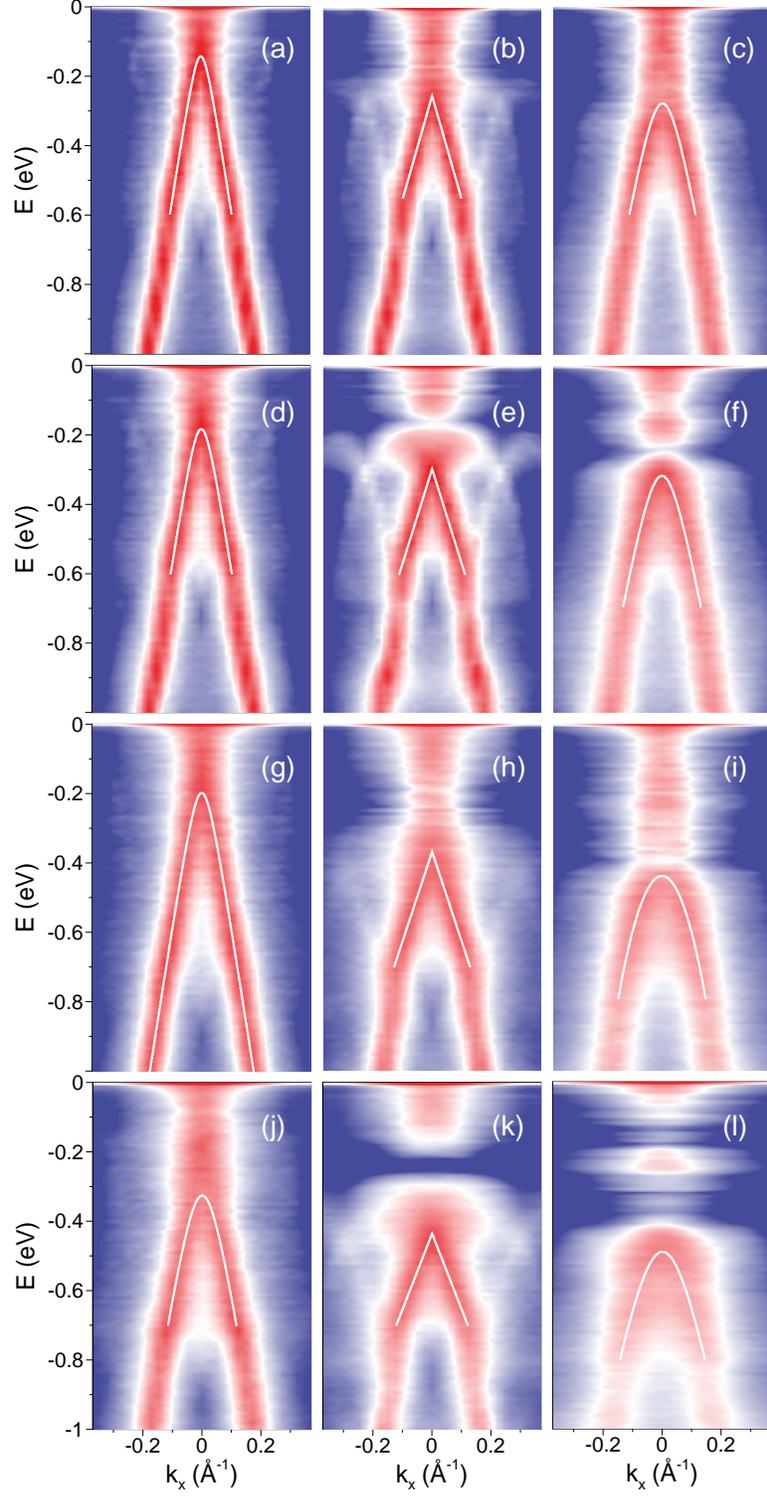


FIG. 9. Plot of averaged $A(\mathbf{k}, \mathbf{k}; E)$ and fitting curve, for all the set of samples studied in this work. (i) and (l) are the same plots as in Fig. 1c of the manuscript, (j) and (k) the same as in Fig. 2a of the manuscript.

set	SL	n_c	N_w	E_1 (eV)	E_2 (eV)	fit.	$E_g/2$ (eV)	v/v_F	m/m_0	k_x^B (\AA^{-1})	$2B$ (eV)
(a)	13	0.75	2	0.05	0.6	LP	0.285		0.035	0.101	0.14
(b)	13	0.75	3	0.2	0.55	L	0.507	0.537		0.101	0.15
(c)	13	0.75	4	0.25	0.6	LP	0.555		0.094	0.101	0.28
(d)	13	1	2	0.1	0.6	LP	0.365		0.046	0.101	0.13
(e)	13	1	3	0.3	0.6	L	0.594	0.491		0.101	0.18
(f)	13	1	4	0.25	0.7	LP	0.633		0.110	0	0.12
(g)	10	0.75	2	0.1	1	LP	0.396		0.048	0.098	0.18
(h)	10	0.75	3	0.4	0.7	L	0.736	0.465		0.098	0.18
(i)	10	0.75	4	0.4	0.8	P	0.876		0.236	0	0.23
(j)	10	1	2	0.3	0.7	LP	0.650		0.087	0.098	0.22
(k)	10	1	3	0.35	0.7	L	0.868	0.393		0.098	0.19
(l)	10	1	4	0.35	0.8	LP	0.979		0.192	0	0.23

TABLE I.

Parameters of the $A(\mathbf{k}, \mathbf{k}; E)$ fitting and broadening extraction.

The input and output parameters of the band-gap and broadening extraction are collected in Table I for each set of sample: L, P, and LP refer to the fitting curves (29), (30) and (31), respectively, $v_F = (3/2)a_{CC}|\gamma|/\hbar$ is the graphene Fermi velocity, and m_0 is the electron rest mass.

The use of different fitting curves deserves an explanation. We notice that relation (31) is the most physical one since it describes the 1D quantization of the graphene dispersion relation. However, for $N_w = 3$, i.e. cases (b), (e), (h), and (k) of Fig. 9 and Table I, the averaging effect of disorder at the cluster edges seems to be not strong enough to reach the typical dispersion relation (in fact, the repeated bands of the superlattice are still slightly visible in the $A(\mathbf{k}, \mathbf{k}; E)$ plot), and the functional dependence in (29) gives a better fitting. Also, when the parabolicity is large in the energy range of interest, such as in cases (i) and (l) of Fig. 9 and Table I, the use of (30) instead of (31) does not make a significant difference

in the gap value.

TRANSPORT CALCULATION

We consider a structure such as the one represented in Fig. 10a. The structure is divided in slabs along the longitudinal direction so that each slab corresponds to an atomic row. The slabs inside the device region are numbered from 1 to N .

The zero temperature conductance G is given by the transmission function $T(E)$ at the

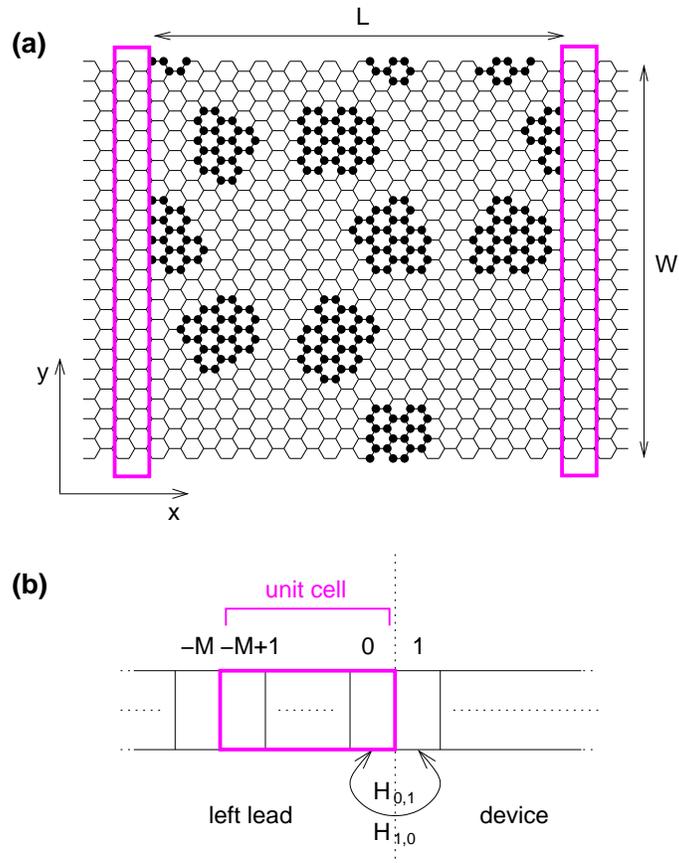


FIG. 10. (a) Example of device used in transport simulations: hydrogenated channel of size $W \times L$ between two leads of pristine graphene. The device is aligned so that its longitudinal direction corresponds to an armchair direction. (b) Slab representation of the same structure. The unit cell inside the lead regions can be viewed as being made of $M = 4$ slabs, where each slab corresponds to a row of carbon atoms.

Fermi energy E_F ,

$$G = \frac{2e^2}{h} T(E = E_F), \quad (32)$$

where in turn the transmission function is computed through Green's functions as [7]

$$T(E) = \text{Tr} [\Gamma^L G^r \Gamma^R G^a]. \quad (33)$$

In this equation, $\Gamma^{L/R} = i(\Sigma^{r,L/R} - \Sigma^{a,L/R})$ is the broadening function due to the left/right lead, where $\Sigma^{r,L/R}$ is the self-energy representing the renormalization of the Hamiltonian of the device region alone due to the presence of the semi-infinite left/right lead, and $\Sigma^{a,L/R} = (\Sigma^{r,L/R})^\dagger$. Since the only non-null block of $\Sigma^{r,L}$ is $\Sigma_{1,1}^{r,L}$ and the only non-null one of $\Sigma^{r,R}$ is $\Sigma_{N,N}^{r,R}$, (33) can be rewritten as

$$T(E) = \text{Tr} [\Gamma_{1,1}^L G_{1,N}^r \Gamma_{N,N}^R G_{N,1}^a]. \quad (34)$$

The calculation of the G^r blocks can be efficiently performed using well known methods, such as the already mentioned recursive algorithm [6], or a combination of the recursive and the renormalization algorithms [8], and therefore it is not treated here. Instead, we focus on the calculation of the lead self-energies.

We notice that in our case the unit cell of each lead is made of $M = 4$ slabs (Fig. 10b). Here, we propose an algorithm, based on the renormalization method [5], to reduce the size of the unit cell to only one slab, such that the usual Sancho-Rubio algorithm [9] can then be applied on matrices having a reduced size, thus saving computational time. We notice that, in the specific case considered in this work, analytical expressions for the self-energies could have been used [10]. However, the numerical technique presented here is more general: for example, it can also be applied in the presence of a magnetic field.

We consider only the case of a left lead, the generalization to the right case being straightforward. The self-energy due to the left lead is defined as

$$\Sigma_{1,1}^{r,L} = \Omega_{1,0} g_{0,0}^r \Omega_{0,1}, \quad (35)$$

where $\Omega = (E + i\eta)I - H$ and g^r is the retarded Green's function for the case in which the coupling between the device and the leads is set to zero [7]. Suppose that the unit cell of

* E-mail: rgrassi@arces.unibo.it

† Current address: IBM T.J. Watson Research Center, Yorktown Heights, New York 10598, USA

- [1] A. L. V. de Parga, F. Calleja, B. Borca, J. M. C. G. Passeggi, J. J. Hinarejos, F. Guinea, and R. Miranda, *Phys. Rev. Lett.* **100**, 056807 (2008).
- [2] A. T. N'Diaye, S. Bleikamp, P. J. Feibelman, and T. Michely, *Phys. Rev. Lett.* **97**, 215501 (2006).
- [3] R. Balog, B. Jørgensen, L. Nilsson, M. Andersen, E. Rienks, M. Bianchi, M. Fanetti, E. Lægsgaard, A. Baraldi, S. Lizzit, Z. Sljivancanin, F. Besenbacher, B. Hammer, T. G. Pedersen, P. Hofmann, and L. Hornekær, *Nature Materials* **9**, 315 (2010).
- [4] <http://www.caam.rice.edu/software/ARPACK/>.
- [5] G. Grosso, S. Moroni, and G. P. Parravicini, *Phys. Rev. B* **40**, 12328 (1989).
- [6] R. Lake, G. Klimeck, R. C. Bowen, and D. Jovanovic, *J. Appl. Phys.* **81**, 7845 (1997).
- [7] S. Datta, *Electronic Transport in Mesoscopic Systems* (Cambridge University Press, 1997).
- [8] T. Low and J. Appenzeller, *Phys. Rev. B* **80**, 155406 (2009).
- [9] M. P. L. Sancho, J. M. L. Sancho, and J. Rubio, *J. Phys. F: Met. Phys.* **15**, 851 (1985).
- [10] P. Zhao and J. Guo, *J. Appl. Phys.* **105**, 034503 (2009).