

# Ab Initio Transport Calculations for Functionalized Graphene Flakes on a Supercomputer

Michael Walz, Alexei Bagrets, Ferdinand Evers, and Ivan Kondov

**Abstract** We present ab initio transport studies of large graphene flakes focusing on the local current density  $\mathbf{j}(\mathbf{r})$  as it arises from a dc-transport measurement. Such ab initio transport calculations for sufficiently large flakes can be successfully tackled only using well scaling ab initio packages capable for transport calculations in thin film geometries. We employ the FHI-AIMS/AITRANSS packages to study the effect of disorder on the local current density in graphene flakes, in particular, the effect of chemical functionalization on mesoscopic fluctuations of the current density. Simulating graphene flakes with several thousands of atoms, we clearly see the qualitative effects of quantum interference and mesoscopic fluctuations in such systems. We also discuss the parallelization and optimization techniques, which we implemented into the transport module AITRANSS to allow efficient ab initio transport calculation on Cray XE6 and XC40 supercomputers.

## 1 Introduction

Graphene is an atomically thin layer of carbon atoms that self-organize in a honeycomb lattice. Since several years, such layers are routinely manufactured in laboratories worldwide. Electron transport in graphene and other graphene properties are investigated intensively and form one of the hottest fields in condensed matter and materials sciences. The driving force results from prospective applications in information and nanotechnology, e.g., as field effect transistors or

---

M. Walz

Institute of Nanotechnology and Institut für Theorie der Kondensierten Materie, Karlsruhe Institute of Technology, Karlsruhe, Germany  
e-mail: [michael.walz@kit.edu](mailto:michael.walz@kit.edu)

A. Bagrets • F. Evers

Institute of Nanotechnology, Karlsruhe Institute of Technology, Karlsruhe, Germany

I. Kondov (✉)

Steinbuch Centre for Computing, Karlsruhe Institute of Technology, Karlsruhe, Germany  
e-mail: [ivan.kondov@kit.edu](mailto:ivan.kondov@kit.edu)

spin quantum bits. Also, chemical functionalization triggers proposals for additional technologies like hydrogen storage or sensor applications.

In most applications, the electron transport properties of graphene-based films are crucial. Despite intensive research, the transport properties are far from being fully understood, especially concerning the influence of disorder, as it is induced, e.g., by functionalization with adsorbates. In computational work, disorder effects have been treated so far mostly on the level of tight-binding calculations that incorporate aspects of  $\pi$ -electron physics; genuine *ab initio* treatments of transport in functionalized molecular films were not yet available. Partially, this is because electronic structure calculations based on the density functional theory (DFT) for large, sufficiently representable flakes are computationally highly demanding. They can be successfully treated only by the most advanced *ab initio* packages. An additional difficulty is that such packages are often not capable for transport calculations in film geometries.

Recently, the FHI-AIMS code [1] has been complemented with a versatile transport module AITRANSS [2–5], which implements the non-equilibrium Green's function formalism (NEGF) and uses DFT results as input. It enables computing *ab initio* transport properties of single molecules but also of more extended structures such as molecular flakes, films or nanotubes.

In this work we study the effect of disorder (e.g. hydrogen adsorbates) and chemical functionalization on the conductance of graphene flakes. In contrast to the presently available tight-binding treatments, our DFT approach includes the effects of charging and screening of impurities, and lattice distortion, i.e., strain. As a consequence, we can study the cross-talk between different impurities which is not possible with present tight-binding implementations.

The sizes of computationally tractable graphene flakes reach 2500 carbon atoms. This is sufficient to see the qualitative effects of quantum interference and mesoscopic fluctuations in such systems. Reaching such flake sizes was only possible by achieving an excellent DFT performance on the HLRS systems when using the FHI-AIMS package. In addition, we improved the parallel scaling of the transport module AITRANSS.

The structure of the paper is as follows: in Sect. 2.1, we describe our transport method as implemented in the code, and we comment on the employed parallelization techniques (Sect. 2.2). In Sect. 3.1, the key findings are summarized. For reference, we provide the numerical parameters used in a typical calculation in Sect. 3.2. Next, the achieved excellent performance on the HLRS computing systems is shown, first for the DFT simulation (Sect. 3.3), second for the transport part (Sect. 3.4). In Sect. 3.5, we present two optimizations, necessary for achieving the excellent performance.

## 2 Methods

Our transport calculations are performed in a two-step procedure. First, we perform a DFT calculation using the all-electron DFT code FHI-AIMS including relaxation effects. Second, we perform a transport calculation using AITRANSS with the Kohn–Sham orbitals from the previous DFT step as input. We summarize the transport calculation in the following and point out where parallelization is necessary. The implementation description of FHI-AIMS and AITRANSS is outlined in [1] and [5], respectively.

### 2.1 Transport Calculations

We extract the Kohn–Sham (KS) Green’s function  $\mathbf{G}_0^{\text{KS}}(E) = (E\mathbf{1} - \mathbf{H}^{\text{KS}} + i0)^{-1}$  for a finite disordered graphene flake from the DFT calculations. To model the infinite extension of the system in current flow direction, we compute the self-energies  $\Sigma_{L/R}$  using absorbing boundary conditions [5, 6]. The self-energies  $\Sigma_{L/R}$  represent the influences of the leads. The resulting Green’s function

$$\mathbf{G}(E)^{-1} = \mathbf{G}_0^{\text{KS}}(E)^{-1} - \Sigma_L(E) - \Sigma_R(E) \quad (1)$$

describes the propagation of KS particles in the device in the presence of leads and is used to calculate the transmission  $\mathcal{T}(E) = \text{Tr}\{\Gamma_L \mathbf{G} \Gamma_R \mathbf{G}^\dagger\}$ . Here,  $\Gamma_{L/R}$  denote the anti-Hermitian parts of the self-energies, i.e.,  $\Gamma_{L/R} = i(\Sigma_{L/R} - \Sigma_{L/R}^\dagger)$ . They account for the level broadenings in the finite graphene flake due to the coupling to the leads.

Using the retarded Green’s function, we calculate the non-equilibrium Keldysh Green’s function  $\mathbf{G}^< = i\mathbf{G}[f_L \Gamma_L + f_R \Gamma_R] \mathbf{G}^\dagger$ . Assuming that the scattering states originating from the left/right lead are occupied/unoccupied, the Keldysh Green’s function  $\mathbf{G}^<(E)$  simplifies to

$$\mathbf{G}^<(E) = i\mathbf{G}(E)\Gamma_L(E)\mathbf{G}^\dagger(E). \quad (2)$$

Here, we assumed zero temperature and an energy inside the voltage window, so that  $f_L = 1$  and  $f_R = 0$  for the occupation in the leads.

We use orthonormal basis functions  $\tilde{\varphi}_i(\mathbf{r})$  [constructed via Löwdin-orthogonalization from the DFT basis functions  $\varphi_i(\mathbf{r})$ ] to transform the Keldysh Green's function into real space:

$$\mathbf{G}^<(\mathbf{r}, \mathbf{r}', E) = \sum_{ij} \tilde{\varphi}_i(\mathbf{r}) \mathbf{G}_{ij}^<(E) \tilde{\varphi}_j^*(\mathbf{r}'). \quad (3)$$

The current density (per spin and energy) is then expressed as

$$\mathbf{j}(\mathbf{r}, E) = \frac{1}{2\pi} \frac{\hbar}{2m} \lim_{\mathbf{r}' \rightarrow \mathbf{r}} (\nabla_{\mathbf{r}'} - \nabla_{\mathbf{r}}) G^<(\mathbf{r}, \mathbf{r}', E). \quad (4)$$

## 2.2 Implementation: Parallelization Efforts

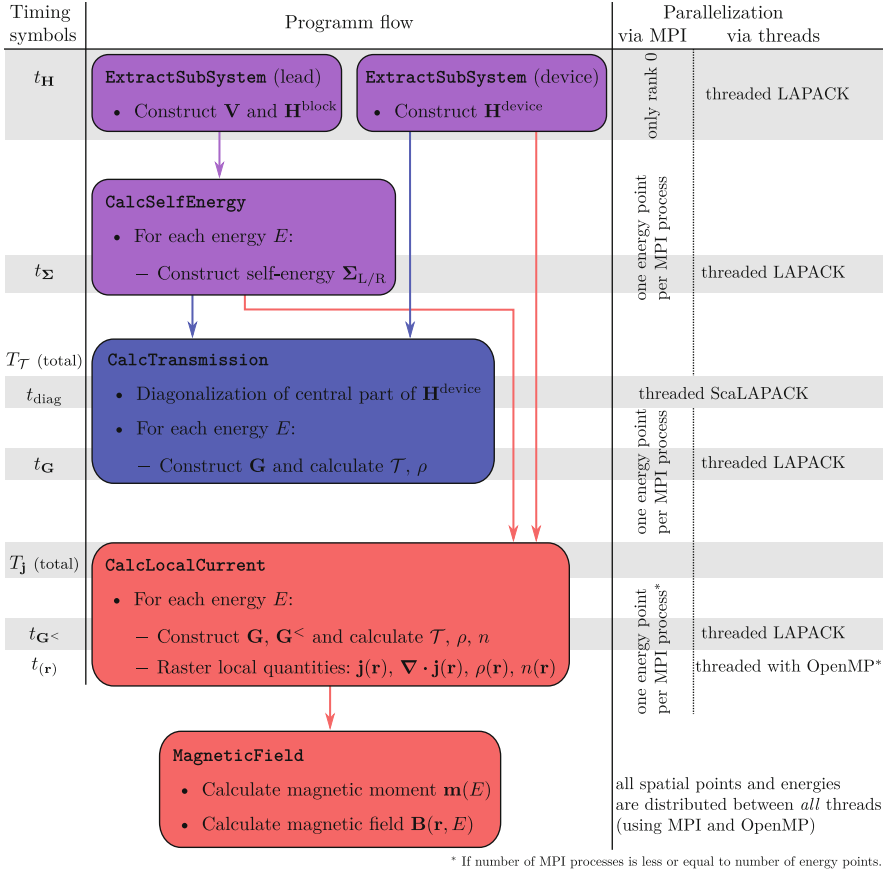
Because the computational demand of the DFT and transport steps scale very similar with flake size (see Sect. 3.5), none of the steps represents a single bottleneck for the whole simulation. Thus, we also optimized and parallelized the transport module AITRANSS to enable study of large flakes.

The AITRANSS code uses two types of parallelization:

**shared-memory parallelization**, in which several threads within one process have access to the same data, i.e., the same energy point  $E$ . We use threaded LAPACK as implemented in the Intel MKL for matrix operations and OpenMP for parallelization of loops over real space, etc.

**distributed-memory parallelization**, in which each process works with separate data sets for possibly different energy points  $E$ . The Message Passing Interface (MPI) is used for work balancing, e.g., for distributing different energy points to different processes, cutting and distributing the real-space grid.

A workflow diagram of the parallelized modules is shown in Fig. 1.



**Fig. 1** Workflow diagram of the transport module AITRANSS. The two most common module sequences are depicted in *blue* (transmission calculation) and *red* (current density and magnetic field calculation). Both sequences include the reconstruction of the Kohn–Sham Hamiltonian and the calculation of the self-energies (*purple*). *Left*: timing symbols used in the following for performance analysis, cf. Figs. 4 and 5. *Right*: overview of the parallelization techniques used in each module

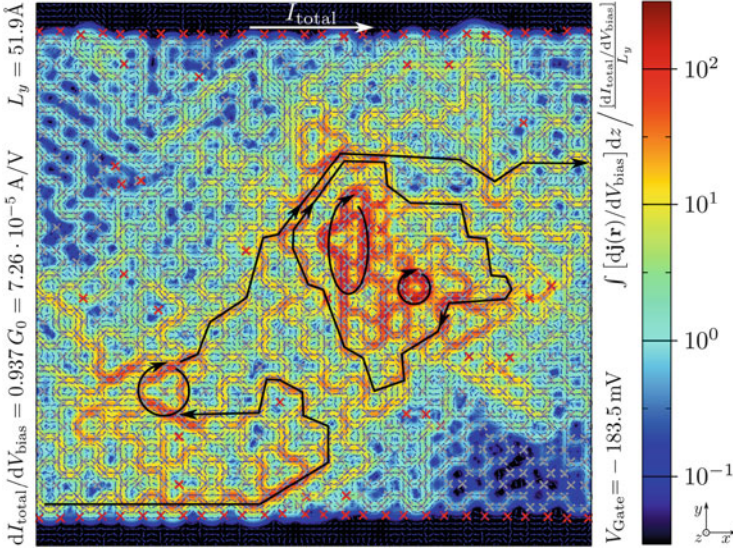
### 3 Results

The results of this work are twofold. First, we summarize the physics results for the investigated graphene flakes in Sect. 3.1. Second, we present our development of computational techniques: (a) we demonstrate that our applied codes scale sufficiently when using a few thousand cores and (b) we discuss the improvements in our transport module that were necessary to achieve good scalability.

### 3.1 Overview of the Key Findings

As a preparation for this work, we calculated the local current density response  $\mathbf{j}(\mathbf{r}, E)$  of pristine armchair graphene nanoribbons (AGNRs) with varying width [7]. We observe pronounced current patterns, which we call “streamlines”, with three-fold periodicity in the ribbon width. They arise as a consequence of quantum confinement in the direction transverse to the current flow. Neighboring streamlines are separated by stripes of almost vanishing flow. This effect can be explained in a tight-binding toy model. The response of the current density to adatoms is very sensitive to the placement: adatoms placed within the current filaments lead to strong backscattering; while in other regions, adatoms have almost no impact.

Then, we switched to larger graphene flakes calculating the local current density  $\mathbf{j}(\mathbf{r}, E)$  in the presence of hydrogen adsorbates [8], an example with 5 % hydrogen is shown in Fig. 2. We discovered pronounced local current patterns, ring currents (current vortices), that go along with orbital magnetism. Importantly, the magnitude of the ring currents can exceed the average transport current by orders of magnitude. The associated magnetic fields exhibit drastic fluctuations with large field gradients reaching up to  $1 \text{ T nm}^{-1} \text{ V}^{-1}$ . These observations are relevant for spin relaxation in systems with very weak spin–orbit interaction, e.g., organic semiconductors. In



**Fig. 2** Local current density response (integrated over the out-of-plane direction) in an AGNR41 ( $24 \times 41$ ) with 5 % hydrogen adsorbates. The current density exhibits very strong mesoscopic fluctuations which exceed the average current by over two orders of magnitude in the logarithmic color scale. The current density is plotted relative to average current density. Plot shows current amplitude (color), current direction (arrows), carbon atoms (grey crosses) and hydrogen atoms (red crosses). Some arbitrary current paths (black lines) are drawn into the plot for illustration

such systems, spin relaxation induced by bias-driven orbital magnetism competes with the hyperfine interaction. Both appear to be of similar strength. As a result of our calculation, we propose an NMR-type experiment combined with a dc-transport setup to observe the spatial fluctuations of the induced magnetic fields. We studied several impurity concentrations and different graphene flake sizes. The described physics seems to be independent and, therefore, should also be present in larger mesoscopic samples which are more common in experiments.

We also studied the statistical distribution of the current density in the graphene flakes [9]. The distribution function of the current density follows a log-normal distribution in a wide range. Its typical value is larger than the average current. Therefore, there are always significant contributions to the current density which do not contribute to the conductance, i.e., they form current rings. This work is still on-going, but so far, these features seem to be remarkably stable, in a wide range of impurity concentrations (5–30 %) and system sizes (up to 2500 carbon atoms), and even survive an averaging over several scattering states, e.g., when a finite bias voltage is applied to the system.

### 3.2 Typical Numerical Parameters

In Table 1, we list the numerical parameters of a typical calculation performed for hydrogenated graphene flakes. First, the graphene flake is structurally relaxed using FHI-AIMS until the remaining forces decrease below  $10^{-2}$  eV/Å. This is, by far, the most expensive part of the calculation. Then a final DFT run for the relaxed structure

**Table 1** An overview of a typical calculation performed on Cray XC40 (Hornet) for a graphene flake (with 1312 carbon atoms) whose central  $24 \times 41$  carbon atoms have been functionalized with hydrogen (compare with Fig. 2)

	FHI-AIMS		AITRANS	
	Relaxation	DFT	Transmission <sup>a</sup>	Current density <sup>b</sup>
Numbers of processes $N_{\text{MPI}}$	2688	96	60	3
Cores per process $p$	1	1	4	8
Total number of cores $P$	2688	96	240	24
Number of nodes $n$	112	4	10	1
Wall time $T_{\text{wall}}$	36.4 h <sup>c</sup>	1.85 h	0.80 h	7.37 h
Core hours $T_{\text{cores}}$	97 800 h	178 h	192 h	177 h
Memory usage per process $\mathcal{M}$	0.55 GB	0.55 GB	14.8 GB	21.5 GB
Memory usage per node $\mathcal{M}_{\text{node}}$	13.2 GB	13.2 GB	88.7 GB	64.4 GB

Note that some information is redundant, i.e.,  $P = p N_{\text{MPI}} = 24 n$ ,  $T_{\text{cores}} = T_{\text{wall}} P$ ,  $\mathcal{M}_{\text{node}} = \mathcal{M} N_{\text{MPI}} / n$

<sup>a</sup>Transmission  $\mathcal{T}(E)$  has been calculation at 2423 energies values

<sup>b</sup>The current density  $\mathbf{j}(\mathbf{r}, E)$  has been calculation at 6 energies values

<sup>c</sup>The relaxation calculation was broken down into several jobs, each with a wall time below 24 h

is performed and the output written to disk. This is used by AITRANSS to perform a wide scan over the transmission function (the self-energies  $\Sigma$  are pre-calculated since they only depend on the system size, not on the impurity configuration). Eventually, a few interesting energy points are taken from the transmission function and the current density is calculated at those energy points.

### 3.3 Achieved DFT Performance on Cray XE6 (Hermit) and Cray XC40 (Hornet)

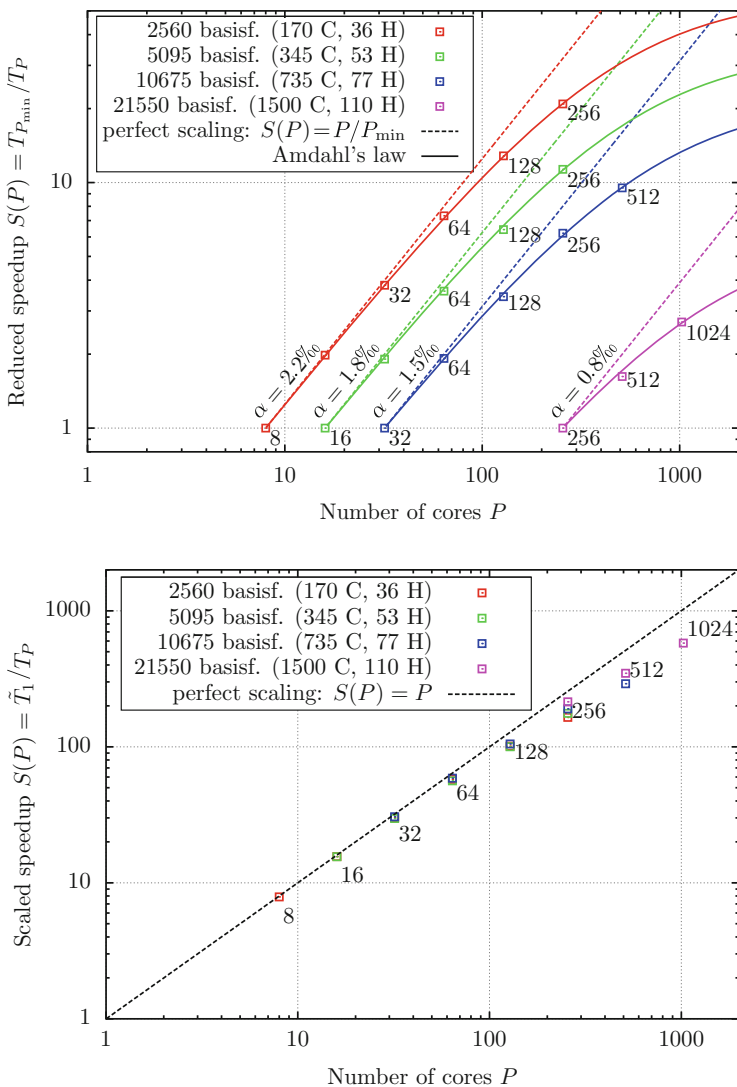
Within a test project on Hermit for which a budget of 40,000 core hours has been granted, we carried out code porting, tuning and performance measurements for DFT calculations of graphene to ensure that the FHI-AIMS code scales as necessary for the completion of the project goals and that the envisaged computing system Hermit was appropriate for the planned productive simulations.

The scalability of the FHI-AIMS code was demonstrated running the DFT module for pure planar graphene flakes, in which dangling bonds at the boundaries were saturated with hydrogen atoms with 170 ( $10 \times 17$ ), 345 ( $15 \times 23$ ), 735 ( $21 \times 35$ ) and 1500 ( $30 \times 50$ ) carbon atoms, each represented by a `tier 1` basis set, i.e. 14 basis functions per C atom and 5 per H atom. The total number of basis functions  $N$  is 2560, 5095, 10,675 and 21,550, respectively.

Figure 3 (upper plot) presents the scaling of the code where the data from each set (color) are plotted as reduced speedup  $T_n/T_P$ .  $T_P$  is the wall time of calculation on  $P$  processor cores and  $T_n$  is the wall time on the minimum number of cores  $n$  necessary for the job to finish within wall time limit (24 h). Thus,  $n$  is 8, 16, 32 or 256, depending on the graphene flake size, i.e., each color in Fig. 3 is for a different  $n$ . The normalization with  $n$  is also necessary to allow easy comparison of the speedups for the four graphene flake sizes. Then, the data for each set have been fitted to Amdahl's law (the solid lines) showing that the serial fraction of the work  $\alpha$  is always very small, around 1 % ( $= 0.001$ ) and furthermore decreases with increasing the graphene flake size.

Figure 3 (lower plot) shows the scaling for graphene using the same data as in Fig. 3 (upper plot) now represented by the speedup  $S(P) = \tilde{T}_1/T_P$  achieved on  $P$  cores, where  $\tilde{T}_1$  is a hypothetical time for which the same calculation would take when a single core were used. Note that for a fixed number of cores used for computations, e.g., set of data points for  $P = 256$ , the speedup is improved when the size of the problem increases approaching the ideal speedup (Gustafson's law).

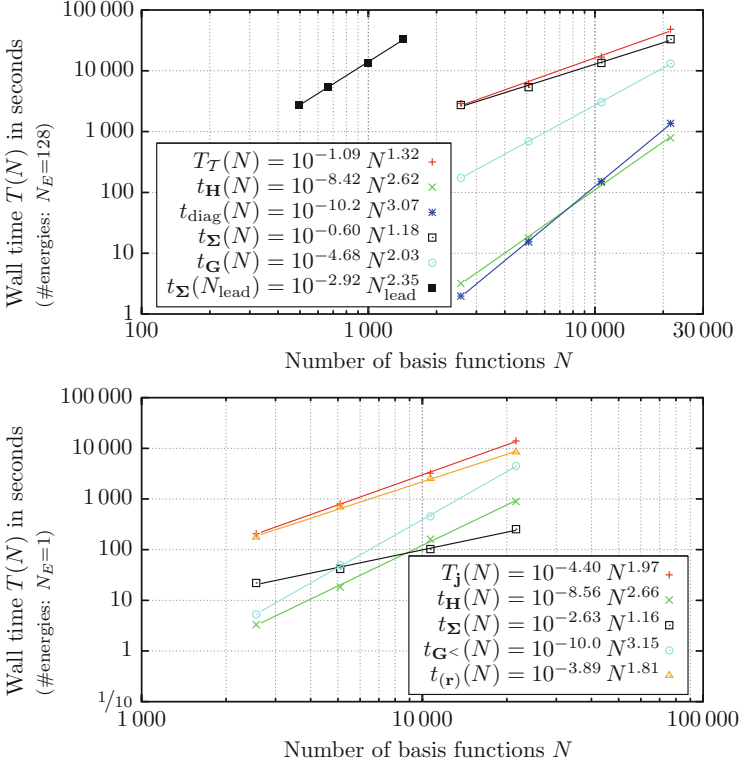




**Fig. 3** Strong scaling of FHI-AIMS for pure graphene flakes on the CRAY XE6 (Hermit) cluster at HLRS (*upper plot*). Scaling of FHI-AIMS for pure graphene flakes of different size on the CRAY XE6 (Hermit) cluster at HLRS (*weak scaling, lower plot*)

### 3.4 Parallelization of the Transport Module AITRANSS and Achieved Performance

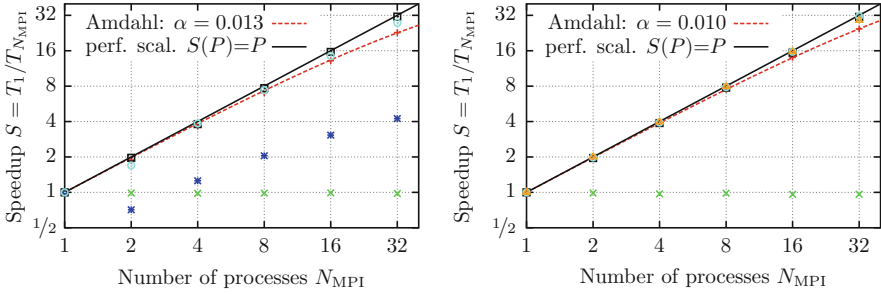
In Fig. 4, we present detailed measurements of the performance of our transport code for realistic system sizes. In the tests, we distinguish between calculation of the transmission and of local observables. Transmission calculations ( $\mathcal{T}(E)$ )



**Fig. 4**  $\mathcal{O}(N^n)$ -scaling: Performance measurements with varying system size for a transmission (upper plot) and local observable (lower plot) calculation for a fixed number of CPU cores ( $P=32$ ). Symbols: number of basis functions  $N$ , number of energy points  $N_E$ , number of CPU cores  $P$

also include the density of states  $\rho(E)$ . Local observables are the local current density  $\mathbf{j}(\mathbf{r}, E)$  but also its divergence  $\nabla \cdot \mathbf{j}(\mathbf{r}, E)$ , the non-equilibrium density  $n(\mathbf{r}, E)$  and the local density of states  $\rho(\mathbf{r}, E)$ .

In the upper panel, the wall time for calculating  $N_E = 128$  transmission and density of states of hydrogen-saturated AGNRs (the same as in Sect. 3.3) is plotted depending on the number of basis functions  $N$ . The total time  $T_{\mathcal{S}}(N)$  is divided into four groups ( $t_H$ ,  $t_{\text{diag}}$ ,  $t_{\Sigma}$ ,  $t_G$ , cf. Fig. 1). Because the calculation (via 200 iterations in the decimation technique [5]) of the self-energy representing the leads depends directly on the number of basis functions  $N_{\text{lead}}$  of the leads (and only indirectly on the basis functions  $N$  of the device region), it is plotted separately. The main effort for a transmission calculation is reconstructing the self-energy of the leads; therefore it makes sense to save them on the hard disk if several different impurity configurations are processed which all use the same leads. Then, the main effort is spent for reconstructing the Green's function ( $t_G$ ). The diagonalization ( $t_{\text{diag}}$ , see Sect. 3.5.3) does not significantly contribute to the overall effort for the considered system sizes since it is performed only once and not for every energy point.



**Fig. 5** MPI-parallelization: speedup for a fixed system size (735 ( $21 \times 35$ ) carbon atoms) and for a fixed number of CPU cores per MPI process ( $p=8$ ). *Left*: speedup for transmission calculations. *Right*: speedup for local observable calculation. Symbols are the same as in Fig. 4. The  $p$  is the number of CPU cores per MPI process, and  $N_{\text{MPI}}$  is the number of MPI processes ( $P = p N_{\text{MPI}}$ )

In the lower panel of Fig. 4, the same quantities are plotted for a single ( $N_E = 1$ ) current density calculation. The number of grid points used is proportional to the system size (one grid point every  $0.2 \text{ \AA}$ , 31 in  $z$ -direction). We first note, that the main effort is dominated by the discretization of the local quantities ( $t_{(\mathbf{r})}$ ). We were able to optimize local observable calculation to scale below  $N^2$  employing the locality of the basis functions.<sup>1</sup>

In Fig. 5, we discuss the parallelization efficiency of the transmission calculation and current density calculation for a fixed system size ( $21 \times 35 = 735$  carbon atoms). The speedup  $S$  for many MPI processes compared to a single process is shown and compared to Amdahl's law:  $T_{N_{\text{MPI}}} = T_1 [\alpha + (1 - \alpha)/N_{\text{MPI}}]$ ,  $\alpha \approx 1\%$ . We see a good scalability for the total wall time, the self-energy construction and the Green's function construction ( $T_{\mathcal{G}}$ ,  $t_{\Sigma}$ ,  $t_{\mathbf{G}}$ ). The reconstruction of the KS Hamiltonian does not speedup since only the first MPI process is involved (cf. Fig. 1). For the diagonalization, we even observe that using two processes (using ScaLAPACK) is slower than using only one process (using LAPACK). Therefore, our code now only uses ScaLAPACK starting with 4 MPI processes.

### 3.5 Optimization of the Transport Module AITRANSS

#### 3.5.1 Overview: Code Optimization

In the following sections, the two most important optimizations used throughout our code development are presented. Their performance impact is summarized in Table 2. Please note, that both optimizations have larger impact for larger systems.

<sup>1</sup>Naively the evaluation of the current or its divergence scales as  $N^3$  with  $N$  since the number of spatial grid points  $\mathbf{r}$  scales linearly, for constant grid spacing, and the summation of  $i, j$  [see Eq. (3)] gives additional  $N^2$ . Please, refer to Sect. 3.5.2 for optimization details.

**Table 2** Increase in wall time when specific optimizations *are removed* from the code

Optimization	Changed quantity	398 atoms (345 carbon atoms)	812 atoms (735 carbon atoms)
SpaceBlocks: $\mathbf{j}$	Local quantities: $t_{(\mathbf{r})}$	$\times 18.3$ ; 557s $\rightarrow$ 10177s	$\times 41.1$ ; 1845s $\rightarrow$ 75811s
MatrixInverse: $\mathcal{T}$	Constructing $\mathbf{G}$ : $t_{\mathbf{G}}$	$\times 7.0$ ; 203s $\rightarrow$ 1414s	$\times 10.6$ ; 1342s $\rightarrow$ 14264s

To get a feeling about the dependence on the system size, two different sizes are shown. The computational parameters and the systems are identical to the ones used in Fig. 4. Symbols are used to distinguish transmission ( $\mathcal{T}$ ) and current density ( $\mathbf{j}$ ) calculations. The self-energy was always read from hard disk. To measure the wall time without optimization “SpaceBlocks” for 810 atoms, we doubled the grid spacing (taking only every eighth grid point) and used linear extrapolation to approximate  $t_{(\mathbf{r})}$  for the full grid

The optimization “SpaceBlocks” is vital because without it calculations for systems with more than 1000 atoms become unfeasible. Also the optimization “MatrixInverse” is quite handy because it allows quick transmission scans before turning to more expensive current density calculations.

### 3.5.2 SpaceBlocks: Dividing Space into Blocks

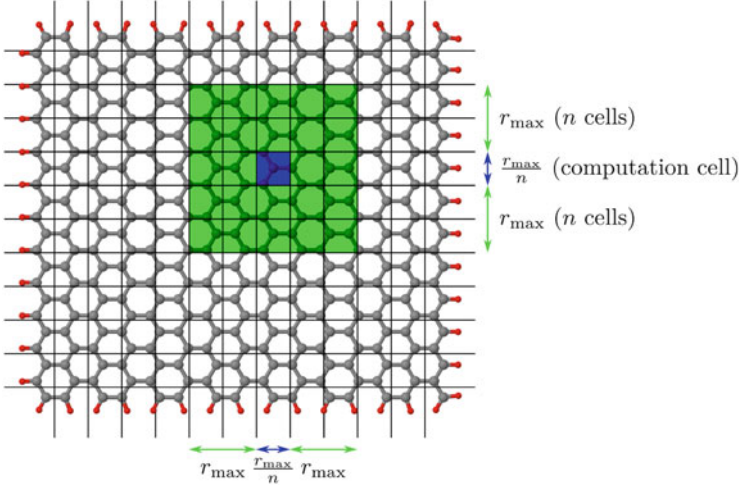
Here, we discuss how to evaluate the formulas for space-depending local quantities such as the current density

$$\mathbf{j}(\mathbf{r}, E) \propto \sum_{ij} \varphi_i(\mathbf{r}) \mathbf{G}_{ij}^{\text{as}}(\nabla \varphi_j(\mathbf{r})), \quad \mathbf{G}_{ij}^{\text{as}} = \frac{1}{2}(\check{\mathbf{G}}_{ij}^< - \check{\mathbf{G}}_{ji}^<). \quad (5)$$

In principle, the double sum runs over *all* basis functions of the underlying DFT simulation. FHI-AIMS uses numerically tabulated atom-centered orbitals (NAOs), i.e. localized basis functions. When restricting the spatial point  $\mathbf{r}$  to a small region, most basis functions are vanishing inside this region. (These basis functions are, of course, nonzero elsewhere.) This locality in the basis set can be exploited in the following way.

First, we define  $r_{\text{max}}$  as the maximal radial extent of all basis functions (i.e. all basis functions are zero at points which are further away from the central atom than  $r_{\text{max}}$ ). Second, the 3D space is divided into little cubes with edge length  $r_{\text{max}}/n$  with  $n$  being an integer (see Fig. 6 for an example). When performing the sum of Eq. (5) for any spatial point  $\mathbf{r}$  inside the blue shaded area, the only basis functions taken into account are centered around atoms in the green (and blue) shaded area. All other basis functions do not contribute to this area.

Hence, we divide the space into cubes of length  $r_{\text{max}}/n$  and distribute them to separate MPI processes. The integer  $n$  is chosen such that every MPI process works on at least five blocks to alleviate load imbalance due to different block sizes. Then, for each inner (blue) block, we restrict the Green’s function  $\mathbf{G}^<$  to the basis functions localized at atoms in the extended (green) block.



**Fig. 6** Dividing space into 156 ( $13 \times 12 \times 1$ ) non-overlapping blocks, exemplary for a graphene flake with 398 atoms. (2D model with  $n = 2$ ,  $r_{\max} = 5.05 \text{ \AA}$ )

### 3.5.3 MatrixInverse: Calculating the Green's Function Inverse

As the self-energy can be read from hard disk, the most expensive part in a transmission calculation is the matrix inversion in calculating the retarded Green's function  $\mathbf{G}$ , cf. Eq. (1). According to Fig. 4 (upper panel),  $\mathbf{G}$  can be constructed in  $\mathcal{O}(N^2)$ . Without this optimization the matrix inversion in calculating the Green's function would scale as  $\mathcal{O}(N^3)$  and would therefore dominate for large systems.

**Partitioning of the Green's function:** The Green's function inverse, see Eq. (1), can be calculated by transforming the Hamiltonian so that it is diagonal in the regions where the self-energies  $\Sigma$  are zero. We partition the indices in the Green's function inverse such that the self-energy contribution of the leads only appears in subblock  $\mathbf{D}$ , i.e.,

$$\begin{aligned} \mathbf{G}^{-1} &= E\mathbb{1} - \mathbf{H} - \Sigma_L(E) - \Sigma_R(E) \\ &= \begin{pmatrix} E\mathbb{1}_{AA} - \mathbf{H}_{AA} & -\mathbf{H}_{AD} \\ -\mathbf{H}_{DA} & E\mathbb{1}_{DD} - \mathbf{H}_{DD} - \Sigma_L(E) - \Sigma_R(E) \end{pmatrix}^{-1} =: \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix}^{-1}, \end{aligned} \quad (6)$$

with the subscripts AA, AD, DA, DD denoting the restriction to the respective matrix subspace.

As advantage of this partitioning, the only non-trivial energy dependence appears in subblock  $\mathbf{D} = E\mathbb{1}_{DD} - \mathbf{H}_{DD} - \Sigma_L(E) - \Sigma_R(E)$ . The block  $\mathbf{A}$  can be diagonalized for all energies in a single eigenvalue problem: the eigenvalues are given by the

diagonal matrix  $\tilde{\mathbf{A}} = E\mathbb{1} - \tilde{\mathbf{H}}_{AA}$  where  $\tilde{\mathbf{H}}_{AA}$  denotes the diagonal matrix with the eigenvalues of  $\mathbf{H}_{AA}$ . The transformation matrix  $\mathbf{V}$  ( $\tilde{\mathbf{H}}_{AA} = \mathbf{V}^{-1}\mathbf{H}_{AA}\mathbf{V}$ ) is constructed by filling its columns with the (right) eigenvectors of  $\mathbf{H}_{AA}$ . The off-diagonal blocks stay energy independent, i.e.,  $\tilde{\mathbf{B}} = -\mathbf{V}^{-1}\mathbf{H}_{AD}$ .

**General matrix:** For the matrix inversion, we first tend to a general matrix which we divide into four blocks

$$\begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix}, \quad (7)$$

so that the submatrices  $\mathbf{A}$  and  $\mathbf{D}$  are square matrices. The inverse is given by

$$\begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix}^{-1} = \begin{pmatrix} \mathbf{A}^{-1}(\mathbb{1} + \mathbf{B}\mathbf{E}^{-1}\mathbf{C}\mathbf{A}^{-1}) & -\mathbf{A}^{-1}\mathbf{B}\mathbf{E}^{-1} \\ -\mathbf{E}^{-1}\mathbf{C}\mathbf{A}^{-1} & \mathbf{E}^{-1} \end{pmatrix} \text{ with } \mathbf{E} := \mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B} \quad (8)$$

as is easily checked by direct matrix multiplication.

Transforming  $\mathbf{A}$  into diagonal form  $\tilde{\mathbf{A}}$ , i.e.,  $\mathbf{A} = \mathbf{V}\tilde{\mathbf{A}}\mathbf{V}^{-1}$ , makes the calculation of the inverse  $\tilde{\mathbf{A}}^{-1}$  trivial and we get:

$$\begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix}^{-1} = \begin{pmatrix} \mathbf{V}\tilde{\mathbf{A}}^{-1}(\mathbb{1} + \tilde{\mathbf{B}}\mathbf{E}^{-1}\tilde{\mathbf{C}}\tilde{\mathbf{A}}^{-1})\mathbf{V}^{-1} & -\mathbf{V}\tilde{\mathbf{A}}^{-1}\tilde{\mathbf{B}}\mathbf{E}^{-1} \\ -\mathbf{E}^{-1}\tilde{\mathbf{C}}\tilde{\mathbf{A}}^{-1}\mathbf{V}^{-1} & \mathbf{E}^{-1} \end{pmatrix} \text{ with } \mathbf{E} := \mathbf{D} - \tilde{\mathbf{C}}\tilde{\mathbf{A}}^{-1}\tilde{\mathbf{B}} \quad (9)$$

using the abbreviations  $\tilde{\mathbf{C}} := \mathbf{C}\mathbf{V}$  and  $\tilde{\mathbf{B}} := \mathbf{V}^{-1}\mathbf{B}$ .

**Exploiting symmetries of  $\mathbf{G}$ :** In general, the Hamiltonian  $\mathbf{H}$  is Hermitian and the self-energies  $\boldsymbol{\Sigma}$  are non-Hermitian. In most cases, we can restrict ourselves to a real symmetric Hamiltonian and complex symmetric self-energies. In that case, the Green's function  $G$  is also (complex) symmetric,  $\tilde{\mathbf{B}}$  and  $\tilde{\mathbf{C}}$  are related by transposition, the eigenvalue problem simplifies to a real symmetric one<sup>2</sup> which makes the transformation matrix  $\mathbf{V}$  orthogonal, i.e.,  $\mathbf{V}^{-1} = \mathbf{V}^T$ .

**Basis change for non-local quantities:** If we are only interested in non-local quantities such as the transmission or the density of states, we can go one step further. Such quantities do not depend on the spatial basis and we can transform the Green's function so that the Hamiltonian is diagonal in the subblock  $\mathbf{A}$ :

$$\mathbf{G} \rightarrow \mathbf{S}^{-1}\mathbf{G}\mathbf{S}, \quad \mathbf{S} = \begin{pmatrix} \mathbf{V} & 0 \\ 0 & \mathbb{1}_{DD} \end{pmatrix}. \quad (10)$$

<sup>2</sup>For real symmetric eigenvalue problems, efficient implementations such as ScaLAPACK [10] or ELPA [11] exist that are parallelized over many computing nodes.

In practice, we indirectly perform this transformation by omitting the respective factors of  $\mathbf{V}$  in Eq. (9). All in all, the inverse is given by:

$$\mathbf{G} = \begin{pmatrix} \tilde{\mathbf{A}}^{-1}(\mathbb{1} + \tilde{\mathbf{B}}\mathbf{E}^{-1}\tilde{\mathbf{B}}^T\tilde{\mathbf{A}}^{-1}) & -\tilde{\mathbf{A}}^{-1}\tilde{\mathbf{B}}\mathbf{E}^{-1} \\ \left[-\tilde{\mathbf{A}}^{-1}\tilde{\mathbf{B}}\mathbf{E}^{-1}\right]^T & \mathbf{E}^{-1} \end{pmatrix} \text{ with } \mathbf{E} := \mathbf{D} - \tilde{\mathbf{B}}^T\tilde{\mathbf{A}}^{-1}\tilde{\mathbf{B}} \quad (11)$$

using the abbreviation  $\tilde{\mathbf{B}} := \mathbf{V}^T\mathbf{B}$ .

**Optimization traits:** In Eq. (11), no matrix operations for matrices of size of  $\mathbf{H}_{AA}$  appear (except for the initial eigenvalue problem): the inverse  $\tilde{\mathbf{A}}^{-1}$  is trivial since  $\tilde{\mathbf{A}}$  is diagonal. Therefore, this optimization is extremely useful for large systems where the contact regions to the leads are only a small part of the overall system, i.e.,  $N_A \gg N_D$  with  $N_{A/D}$  denoting the size of the square matrices  $\mathbf{A}$ ,  $\mathbf{D}$ , respectively.

For a short complexity analysis, we assume that multiplication and eigenvalue problem of  $N \times N$ -matrices have computational complexity  $\mathcal{O}(N^3)$ . Then, without above optimization, the direct matrix inversion used to calculate the Green's function has complexity  $\mathcal{O}((N_A + N_D)^3) \xrightarrow{N_A \gg N_D} \mathcal{O}(N_A^3)$ .

In the above optimization, the complexity of the preparation process containing the eigenvalue problem and the calculation of  $\tilde{\mathbf{B}}$  is  $\mathcal{O}(N_A^3 + N_A^2 N_D) \xrightarrow{N_A \gg N_D} \mathcal{O}(N_A^3)$ . All the following inversions using Eq. (11) only are of complexity  $\mathcal{O}(N_D^3 + N_A N_D^2 + N_A) \xrightarrow{N_A \gg N_D} \mathcal{O}(N_A N_D^2)$ . The summands stand for inversion of  $\mathbf{E}$ , products of  $N_A \times N_D$ -matrices with  $N_D \times N_D$ -matrices like  $\tilde{\mathbf{B}}\mathbf{E}^{-1}$  and inversion of  $\tilde{\mathbf{A}}$ , respectively.

Strictly speaking, the optimization still scales cubically in  $N_A$  due to the initial eigenvalue problem. Nevertheless, for energy sweeps over the density of states or the transmission, the complexity of each inversion step dominates and this effort could be reduced to complexity  $\mathcal{O}(N_A N_D^2)$  for large systems,<sup>3</sup> cf. Fig. 4 (lower panel).

As stated above, the optimization only applies for non-local quantities. For local quantities such as current densities, the transformation matrix  $\mathbf{V}$  cannot be omitted from Eq. (9) and we are back to cubic complexity.

## 4 Conclusions

In this work, we calculated the local current density in large hydrogenated graphene flakes. The current flow shows complicated patterns, a fact that is ignored in most studies focusing only on the total conductance. These patterns show large local fluctuations. The idea behind the nontrivial local current patterns is very general:

<sup>3</sup>For the AGNRs used for Fig. 4, the central part scales linearly,  $N_A \in \mathcal{O}(N)$ , but the contact regions scale with the square root,  $N_D \in \mathcal{O}(\sqrt{N})$  because they only grow transverse to the current direction but not in current direction. This gives the observed overall complexity  $\mathcal{O}(N^2)$ .

scattering states of mesoscopic samples have an inner, nontrivial structure. The latter is seen in the electric current density but it is easily generalized for other observables, e.g. for heat. A mesoscopic device (like a hydrogenated graphene ribbon) which is connected to reservoirs with different temperatures will show fluctuations in the local temperatures as a result of the nontrivial structure of the scattering states.

Along the way, we parallelized and optimized our transport module AITRANSS to benefit from a supercomputer, thus to enable ab initio transport calculations for large graphene flakes. We showed that our techniques feature good scalability and discussed necessary optimizations. Future work will benefit from the fact that ab initio current density calculations for disordered systems are now available for large 2D film materials and for medium sized 3D materials. Essentially, the approach is only limited by the available computing power.

**Acknowledgements** We gratefully acknowledge HLRS Stuttgart for granting us computing time on the Hermit and Hornet systems.

## References

1. Blum, V., Gehrke, R., Hanke, F., Havu, P., Havu, V., Ren, X., Reuter, K., Scheffler, M.: *Comput. Phys. Commun.* **180**(11), 2175 (2009). doi:10.1016/j.cpc.2009.06.022. <http://www.sciencedirect.com/science/article/pii/S0010465509002033>
2. Bagrets, A.: *J. Chem. Theory Comput.* **9**(6), 2801 (2013). doi:10.1021/ct4000263. <http://pubs.acs.org/doi/abs/10.1021/ct4000263>
3. Wilhelm, J., Walz, M., Stendel, M., Bagrets, A., Evers, F.: *Phys. Chem. Chem. Phys.* **15**, 6684 (2013). doi:10.1039/C3CP44286A. <http://dx.doi.org/10.1039/C3CP44286A>
4. Arnold, A., Weigend, F., Evers, F.: *J. Chem. Phys.* **126**(17), 174101 (2007). doi:<http://dx.doi.org/10.1063/1.2716664>. <http://scitation.aip.org/content/aip/journal/jcp/126/17/10.1063/1.2716664>
5. Walz, M., Bagrets, A., Evers, F.: Local current density calculations for molecular films from ab initio. *J. Chem. Theory Comput.* (accepted, 2015).
6. Evers, F., Arnold, A.: In: Röhlig, C., Vojta, M. (eds.) *CFN Lectures on Functional Nanostructures - Volume 2: Nanoelectronics*. Springer, Berlin/Heidelberg (2011)
7. Wilhelm, J., Walz, M., Evers, F.: *Phys. Rev. B* **89**, 195406 (2014). doi:10.1103/PhysRevB.89.195406. <http://link.aps.org/doi/10.1103/PhysRevB.89.195406>
8. Walz, M., Wilhelm, J., Evers, F.: *Phys. Rev. Lett.* **113**, 136602 (2014). doi:10.1103/PhysRevLett.113.136602. <http://link.aps.org/doi/10.1103/PhysRevLett.113.136602>
9. Walz, M., Evers, F.: (in preparation)
10. Blackford, L.S., Choi, J., Cleary, A., D'Azevedo, E., Demmel, J., Dhillon, I., Hammarling, S., Henry, G., Petitet, A., Stanley, K., Walker, D., Whaley, R.C.: *ScaLAPACK User's Guide*. Society for Industrial and Applied Mathematics, Philadelphia (1997)
11. Marek, A., Blum, V., Johanni, R., Havu, V., Lang, B., Auckenthaler, T., Heinecke, A., Bungartz, H.J., Lederer, H.: *J. Phys. Condens. Matter* **26**(21), 213201 (2014). <http://stacks.iop.org/0953-8984/26/i=21/a=213201>