1-25-2010

Numerical Strategies Towards Peta-Scale Simulations of Nanoelectronics Devices

Mathieu Luisier
Network for Computational Nanotechnology, Purdue University

Gerhard Klimeck
Network for Computational Nanotechnology, gekco@purdue.edu

Parallel Computing 36 (2010) 117–128

Contents lists available at ScienceDirect

Parallel Computing

journal homepage: www.elsevier.com/locate/parco

Numerical strategies towards peta-scale simulations of nanoelectronics devices

Mathieu Luisier *, Gerhard Klimeck

Network for Computational Nanotechnology, Purdue University, West Lafayette, IN 47907, USA

ARTICLE INFO

Article history:
Received 28 July 2009
Received in revised form 18 January 2010
Accepted 20 January 2010
Available online 25 January 2010

Keywords:
Nanoelectronics
Atomistic device simulations
High performance computing
Peta-scale

ABSTRACT

We address two challenges with the development of next-generation nanotransistors, (i) the capability of modeling realistically extended structures on an atomistic basis and (ii) predictive simulations that are faster and cheaper than experiments. We have developed a multi-dimensional, quantum transport solver, OMEN, towards these goals. To approach the peta-scale, the calculation of the open boundary conditions connecting the simulation domain to its environment is interleaved with the computation of the device wave functions and the work load of each task is predicted prior to any calculation, resulting in a dynamic core allocation. OMEN uses up to 147,456 cores on Jaguar with four levels of MPI parallelization and reaches a sustained performance of 504 TFlop/s, running at 37% of the machine peak performance. We investigate 3D nanowire transistors with diameters up to 10 nm, reproduce experimental data of high electron mobility 2D transistors, and expect increased capabilities by using over 300,000 cores in the future.

1. Introduction

Nanoscale field-effect transistors (FETs) are expected to outperform conventional planar Si MOSFETs, reduce the power consumption of integrated circuits, and operate at very high frequency. Several device structures are considered for future applications as low power logic gates [1] like single- or double-gate ultra-thin bodies [2], gate-all-around nanowires [3,4], or graphene nanoribbons [5]. Device concepts such as III–V high electron mobility transistors (HEMTs) [6] or band-to-band tunneling FETs (TFETs) are also investigated [7,8]. The fabrication process of these devices is currently not mature enough to reach cheap mass production or even research production so that the development of a physics-based device simulator is of high interest for industry and academic to guide the experimental work and optimize the device performances.

We have developed a quad-level parallel computer aided design tool, OMEN, dedicated to the simulation of these next generation nanotransistors that might be available in 3–5 years [9–11]. OMEN is a 1D, 2D, and 3D atomistic quantum transport solver based on the self-consistent solution of Schrödinger and Poisson equations with open boundary conditions (OBCs) and on the nearest-neighbor tight-binding (NN TB) model [12]. Different declinations of the NN TB approach including single *s* orbital, $sp^3, sp^3s^*, sp^3d^5s^*$, with and without spin–orbit coupling are available for Si, Ge, GaAs, InAs, C and many other materials [13–16]. Furthermore, crystals with an hexagonal, zincblende, or nanotube structure and any transport direction can be treated.

At the nanometer scale the widely-accepted continuous effective mass approximation (EMA) fails [17] and is replaced by a full-band and atomistic description of the simulation domain to obtain accurate and reliable results. The wave function

* Corresponding author.
E-mail address: mluisier@purdue.edu (M. Luisier).

0167-8191/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.parco.2010.01.003

(WF) formalism used in OMEN [9] requires for each energy, momentum, and bias point that (1) two full eigenvalue problems are solved to model the OBCs, (2) a block-tri-diagonal Hamiltonian matrix "A" containing the OBCs and a vector "b" characterizing the injection mechanism are assembled, and (3) the matrix "A" is factorized and the sparse linear system of equations (LSE) "$Ax = b$" is solved. The size of "A" is comprised between 1e5 and 1e6, its bandwidth is in the order of 1e3 or more for 3D structures. Nowadays, each eigenvalue problem and LSE taken individually is easily manageable, but when hundred of thousands of them have to be handled the computational burden becomes a critical issue. This is the case in nanoelectronics device simulations where 10–100 bias points are usually considered, 1–50 momentum points, and 500–10,000 energy points, resulting in a total of 5000–50 millions combinations.

The calculation of the bias, momentum, and energy points forms a quasi-embarrassing three-level parallelization that allows OMEN to simultaneously solve thousands of quantum transport problems with an almost perfect scaling of the simulation time [11]. The fourth level of parallelism, labeled "spatial domain decomposition", arises from the computation of the OBCs and the solution of the LSE "$Ax = b$", it requires more inter-processor communication, and does not scale efficiently beyond 2 cores. In effect the OBCs eigenvalue problems cannot be parallelized, but each open contact of the device, typically two, the source and the drain, can be distributed to a different CPU [11]. Consequently, the cross section of the 3D structures is limited to about 5 nm × 5 nm while research labs and semiconductor companies are mainly interested in structures larger than 10 nm × 10 nm.

To make OMEN a useful tool for the industry and the scientific community the size of the devices that can be investigated must be increased and the simulation time minimized. After a short description of the current status of OMEN in Section 2 we show in Section 3 how the calculation of the OBCs and of the LSE can be interleaved to scale beyond 2 cores, consume less memory, profit from distributed and shared memory parallelization, and we apply it to the simulation of nanowire tunneling FETs with a diameter up to 10 nm. In Section 4 the computational performances of OMEN are analyzed up to 65,536 cores and the load balance across different group of processors is optimized to reach a parallel efficiency of 90% and a sustained performance of 173 TFlop/s on Kraken, a CRAY XT5 with AMD processors (2.3 GHz) [18]. Finally, the approaches of Sections 3 and 4 are combined in Section 5 to simulate a realistic high electron mobility transistor (HEMT). Good agreement with experimental data is demonstrated [20] with a potential sustained performance of 504 TFlop/s on 147,456 cores on Jaguar, the CRAY XT5 from NCCS, Oak Ridge (2.3 GHz AMD cores) [19].

2. Review of device simulator

2.1. Physical models

The numerical algorithms of OMEN are briefly reviewed in this section to point out their deficiencies before resolving them. A special emphasis is put on the solution of the Schrödinger equation with open boundary conditions, which builds the core of the simulator, limits the maximum size of the simulation domain, and is responsible for most of the computational burden.

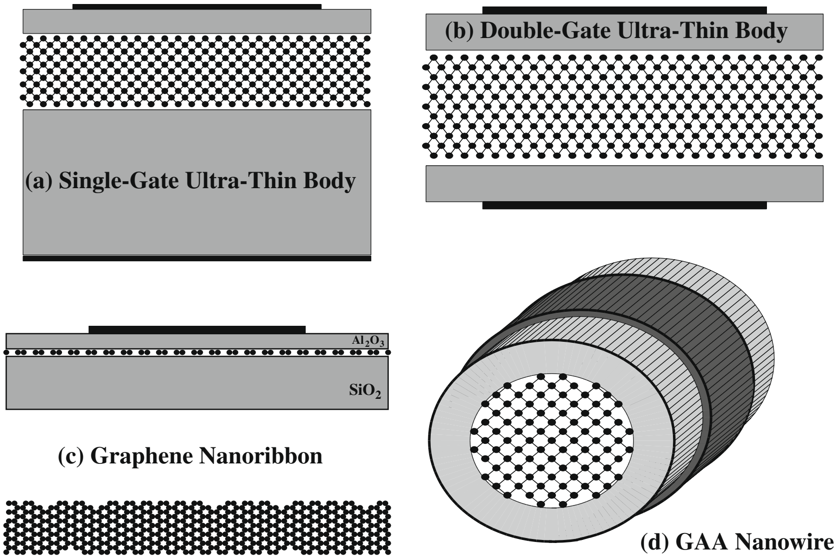

(a) Single-Gate Ultra-Thin Body
(b) Double-Gate Ultra-Thin Body
Al_2O_3
SiO_2
(c) Graphene Nanoribbon
(d) GAA Nanowire

Fig. 1. Typical nanoelectronics device structures. (a) Single-gate ultra-thin-body (SG UTB), (b) double-gate ultra-thin-body (DG UTB), (c) single-gate graphene nanoribbon (SG GNR), and (d) gate-all-around nanowire transistors (GAA NW). The dots represent the atoms that compose the active region of the devices.

OMEN is a multi-dimensional, ballistic, nanoelectronics device simulator written in C++ and dedicated to post-CMOS transistors like single- or double-gate ultra-thin-body (UTB), gate-all-around nanowire (GAA NW), or graphene nanoribbon (GNR) field-effect transistors as sketched in Fig. 1. It solves 1D, 2D, and 3D Schrödinger equations with open boundary conditions (OBCs) in a tight-binding basis. The energy-dependent wave function $|\Psi_E\rangle$ in $H|\Psi_E\rangle = E|\Psi_E\rangle$ is given by the linear combination of atomic orbitals

$$\Psi(\mathbf{r},E) = \langle \mathbf{r}|\Psi_E\rangle = \sum_{\sigma,\mathbf{k},i} c_i^{\sigma}(E,\mathbf{k}) \cdot \phi_{\sigma}(\mathbf{r}-\mathbf{R_i}) \cdot \exp(i\mathbf{k}\cdot\mathbf{R_i}), \tag{1}$$

where the $\phi_{\sigma}(\mathbf{r}-\mathbf{R_i})$ are orbital functions centered around $\mathbf{R_i}$ [12], σ refers to the type of the orbitals in the tight-binding model (s, p, s^*, or d), $\mathbf{R_i}$ to the position of the atoms in the simulation domain, x is the transport direction, and $\mathbf{k}$ models the directions with periodic boundary conditions ($\mathbf{k} = (k_y, k_z)$ in 1D, $\mathbf{k} = k_z, y$ is confined in 2D, and $\mathbf{k} = 0, y$ and z are confined in 3D). In the nearest-neighbor basis of Eq. (1) the Schrödinger equation becomes [9]

$$(E - H_{ii}(\mathbf{k})) \cdot C_i(E,\mathbf{k}) - H_{ii+1}(\mathbf{k}) \cdot C_{i+1}(E,\mathbf{k}) - H_{ii-1}(\mathbf{k}) \cdot C_{i-1}(E,\mathbf{k}) = 0. \tag{2}$$

The vectors $C_i(E,\mathbf{k})$ contain all the unknown expansion coefficients $c_i^{\sigma}(E,\mathbf{k})$ of the device wave function in one atomic layer (all the atoms with the same x-coordinate) situated at $x = x_i$. The sparse blocks H_{ii} and $H_{ii\pm1}$ connect one layer with itself and with its adjacent neighbors, respectively and are of size $N_a t_b$ where N_a is the number of atoms per layer and t_b the number of atomic orbitals considered in the tight-binding model. A detailed description of the Hamiltonian matrix elements can be found in Ref. [12].

The OBCs are calculated using a multi-band scattering boundary ansatz [21] combined with a shift-and-invert procedure which finally consists in finding all the eigenvalues and eigenvectors of the following equation at different energies E and momentum $\mathbf{k}$

$$M(E,\mathbf{k}) \cdot \varphi_n(E,k_{xn},\mathbf{k}) = \frac{1}{\exp(ik_{xn}\Delta) - 1} \cdot \varphi_n(E,k_{xn},\mathbf{k}). \tag{3}$$

The contact unit cell has a width Δ along the x-axis, the general matrix $M(E,\mathbf{k})$ is non-symmetric, non-hermitian, and defined as in Refs. [9,10]. Solving Eq. (3) is about 2 orders of magnitude faster than iterative [22] or general eigenvalue problem [23,24] approaches. It must be repeated for each contact with open boundaries.

The eigenvectors $\varphi_n(E,k_{xn},\mathbf{k})$ are separated into two classes, the forward propagating states (from contact to device), cast into the matrix φ_+ and the backward propagating states (from device back to contact) in matrix φ_-. The OBCs are coupled to Eq. (2) through a boundary self-energy matrix $\Sigma_{ii}(E,\mathbf{k})$ and an injection vector $S_i(E,\mathbf{k})$ calculated for each contact i using

$$\begin{aligned}
&\Sigma_{ii}(E,\mathbf{k}) = H_{ij}(\mathbf{k}) \cdot \varphi_-(E,\mathbf{k}) \cdot g_{jj}^R(E,\mathbf{k}) \cdot \varphi_-^{\dagger}(E,\mathbf{k}) \cdot H_{ji}(\mathbf{k}),\\
&S_i(E,\mathbf{k}) = H_{ij}(\mathbf{k}) \cdot \varphi_+(E,\mathbf{k}) - \Sigma_{ii}(E,\mathbf{k}) \cdot \varphi_+(E,\mathbf{k}) \cdot \exp(ik_{x+}\Delta),\\
&g_{jj}^R(E,\mathbf{k}) = \left(\varphi_-^{\dagger}(E,\mathbf{k}) \cdot H_{ji}(\mathbf{k}) \cdot \varphi_-(E,\mathbf{k}) \cdot \exp(-ik_x\Delta)\right)^{-1}.
\end{aligned} \tag{4}$$

The ith layer is situated within the device while the jth layer is the first one in the semi-infinite contact attached to the device. Layers i and j are adjacent in the nearest-neighbor approximation. Note that the matrix $\varphi_+(E,\mathbf{k})$ only contains states whose imaginary part of the corresponding wave vector k_{x+} vanishes.

Finally, a block tri-diagonal sparse linear system of equations $A \cdot C = S$ is formed from Eqs. (2) and (4). For a device structure containing N_B atomic layers (the index E and $\mathbf{k}$ are dropped for clarity) one has

$$\begin{pmatrix}
A_{11} & A_{12} & 0 & \cdots & 0\\
A_{12}^{\dagger} & A_{22} & A_{23} & 0 & \cdots\\
0 & \ddots & \ddots & \ddots & 0\\
\vdots & \cdots & \ddots & \ddots & A_{N_B-1N_B}\\
0 & \cdots & 0 & A_{N_B-1N_B}^{\dagger} & A_{N_BN_B}
\end{pmatrix} \cdot
\begin{pmatrix} C_1\\ C_2\\ \vdots\\ C_{N_B-1}\\ C_{N_B} \end{pmatrix} =
\begin{pmatrix} S_1\\ 0\\ \vdots\\ 0\\ S_{N_B} \end{pmatrix}, \tag{5}$$

where $A_{nm} = E\delta_{nm} - H_{nm}$, except A_{11} and $A_{N_BN_B}$ that are defined as $A_{11} = E - H_{11} - \Sigma_{11}$ and $A_{N_BN_B} = E - H_{N_BN_B} - \Sigma_{N_BN_B}$, respectively. Two open contacts are assumed, the source at $x = x_1$ and the drain at $x = x_{NB}$. The size of the matrix A is $N_A t_b$ where N_A is the total number of atoms in the transistor structure and t_b the number of orbitals on each atom. Eq. (5) is solved in parallel either with distributed memory softwares like MUMPS [25], SuperLU$_{dist}$ [26], and a block cyclic reduction (BCR) approach that we implemented [27] or with shared memory codes like Pardiso [28] and an OpenMP version of our BCR [29]. For 3D transistor structures BCR outperforms the other solvers by a factor of 2 on a single CPU, when spin–orbit coupling is not included, and it scales better on multiple CPUs [27], even though it requires more floating point operations than, for example, the straight-forward LU factorization from MUMPS, SuperLU, or Pardiso.

The wave function coefficients $C_i(E,\mathbf{k})$ are calculated at each injection energy E and momentum $\mathbf{k}$ to give carrier and current densities [9,10]. These quantities are self-consistently coupled to the finite-element [30] solution of Poisson equation [31]. This operation is repeated for different bias points to obtain the complete output characteristics of a device.

2.2. Parallelization scheme

The distribution of the tasks in OMEN is crucial to reduce the simulation times. Four natural levels of parallelism have been identified, the distribution of (i) the bias points, (ii) the momentum points, (iii) the energy points, and (iv) a spatial decomposition of the simulation domain based on the parallel solution of Eqs. (3)–(5). All these parallelization levels have been implemented via MPI [33] and have been tested on various TeraGrid platforms [34]. To avoid any complication it has been decided that the number of cores attributed to each level is selected by the user at the beginning of the simulation. Starting from a total of P_0 cores, N_{Vg} bias points are simultaneously treated on $P_{v_g} = P_0/N_{V_g}$ cores, where P_{v_g} is an input parameter. The number of momentum points per bias point N_k as well as the number of cores per momentum point P_k are fixed numbers determined by the user to reach a desired physical accuracy and to match $P_{v_g} = N_k \times P_k$. The number of energy points per momentum N_E depends on the contact characteristics, not on the user, and might be different for each momentum. At the lowest parallel level the number of cores assigned to the solution of Eqs. (3)–(5), P_E, is chosen by the user. The subsequent sections show how we improved the scalability and the algorithms of OMEN to finally obtain 504 TFlop/s of sustained performance on 147,456 cores.

The Poisson equation is also solved in parallel with an iterative solver, Aztec [32]. Typically, 32–256 cores are used for that purpose so that the time to solve the Poisson equation is negligible as compared to the time to calculate the transport properties of a device. Hence, the solution of the Poisson equation does not affect the scaling performances of OMEN.

3. Large 2D and 3D simulation domains

3.1. Parallel block cyclic reduction

The computational and the memory burden related to the calculation of the open boundary conditions and of the wave function coefficients becomes a fundamental problem in the simulation of 2D and 3D device structures with large cross sections like circular nanowires with diameters up to 10nm and should therefore be minimized. As pointed out before, Eqs. (3) and (4) cannot be parallelized on more CPUs than the number of contacts with OBCs. However, the factorization and solution of Eq. (5) scales well up to 16 cores and even more depending on the matrix size and bandwidth [29]. Before introducing a novel approach based on a parallel block cyclic reduction (BCR) of the matrix "A" in Eq. (5) and on an "interleaved" calculation of the OBCs and of the wave function coefficients $C_i(E, \mathbf{k})$, we shortly summarize the principle of the BCR algorithm and underline its advantages over other linear solvers.

The BCR algorithm relies on a Gaussian elimination of alternate atomic layers in Eq. (5) (the diagonal blocks A_{ii}) until only the first and the last blocks of the matrix A remain connected and form the following reduced system of equations [27]

$$\begin{pmatrix} E - \widehat{H}_{11} - \Sigma_{11} & -\widehat{H}_{1N_B} \\ -\widehat{H}_{N_B,1} & E - \widehat{H}_{N_BN_B} - \Sigma_{N_BN_B} \end{pmatrix} \cdot \begin{pmatrix} C_1 \\ C_{N_B} \end{pmatrix} = \begin{pmatrix} S_1 \\ S_{N_B} \end{pmatrix}. \tag{6}$$

The blocks $\widehat{H}_{11}, \widehat{H}_{N_BN_B}, \widehat{H}_{1N_B}$, and $\widehat{H}_{N_B,1}$ carry a hat to indicate that they are renormalized by the elimination of all the other blocks. It is important to note that the boundary self-energies Σ and the injection vectors S are not modified by the reduction of the matrix A.

In 3D structures, without spin–orbit coupling, the blocks A_{ij} are real so that the transformation from Eqs. (5) and (6) is performed in "double" arithmetic. Other solvers like MUMPS, SuperLU$_{dist}$, or Pardiso require that all the elements of A are processed as "double complex" because of the complex self-energies Σ. Furthermore, the BCR algorithm only requires to store the upper and lower parts S_1 and S_{N_B} of the right-hand-side reducing the memory consumption when multiple states are injected into the device. Typically, at a given energy E and momentum $\mathbf{k}$, 1–250 states enter the device from the source and the drain contacts.

Nevertheless, a 3D circular structure with a diameter $d = 8$ nm, a total length L = 60 nm, composed of $N_A = 108,207$ atoms, and described in the sp^3s^* tight-binding model generates matrices A of size $N_At_b = 541,035$. More than 20 GB of memory are easily required to handle Eqs. (3)–(5) for a single momentum/energy combination using Blas [38], Lapack [39], and BCR [27]. On a machine like the CRAY XT5 Kraken [18] this means that at least 2 nodes with 16 GB of memory and 8 cores are needed for each energy point. To run efficiently and not waste computational resources OMEN should therefore be able to treat not only Eq. (5) on 9–16 cores, but also the calculation of the open boundary conditions in Eqs. (3) and (4).

3.2. Interleaved decomposition approach

The decoupling scheme of the BCR algorithm offers a solution to the scaling issue of the spatial domain decomposition. The idea consists in starting to calculate the open boundary conditions and the wave function coefficients at the same time so that no CPU remains idle at any moment.

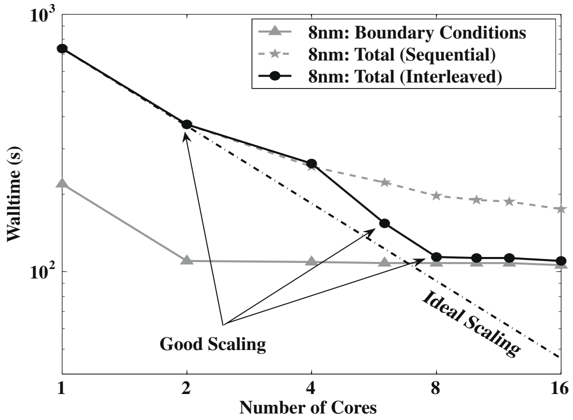
Standard packages like MUMPs, SuperLU$_{dist}$, or Pardiso require that Eqs. (3)–(5) are sequentially solved since the matrix A, which contains the OBCs in its first and last diagonal blocks, is an input parameter. While a speed-up factor of 7–8 can be obtained to solve Eq. (5) on 16 cores as compared to 1 core for a nanowire with $d = 8$ nm and L=60 nm, a speed-up of 2 only is reached for Eqs. (3) and (4). Hence, using $P_E = 16$ cores per energy point to calculate the OBCs and the wave function coefficients results in a total speed-up of 4.2 as shown in the fifth column of the Table in Fig. 2.

The advantage of the parallel block cyclic reduction is that it does not need the boundary self-energies Σ_{ii} and injection vectors S_i before its very last step when the first and the last atomic layers are decoupled in Eq. (6). We can therefore interleave the solution of Eqs. (3) and (4) with the reduction of Eqs. (5) and (6). This method is labeled "interleaved" (OBCs and LSE "$Ax = b$" tasks in parallel) in Fig. 2 and compared to the "sequential" approach (first OBCs and then LSE). In each experiment the matrix "A" is factorized using BCR. A speed-up factor of about 6.5 is measured on 8 cores with respect to 1 core for the "interleaved" approach to compute the OBCs and solve "$Ax = b$". This is about 1.7×faster than the "sequential" approach. However, in going from 8 to 16 cores no further improvement is observed since the total time is dominated by the time to calculate the OBCs.

Shared memory parallelized Blas and Lapack libraries can be used to overcome the scaling limitation of the "interleaved" approach beyond 8 cores, assigning 2 threads per MPI task. On the CRAY XT5 Kraken [18], the shared memory version of the gotoblas library reduces the total time of the "interleaved" approach from 114 s on 8 cores to 80 s on 16 cores with 2 threads per MPI task for the nanowire with $d = 8$ nm (9.2×faster than on a single CPU). As compared to the sequential approach with pure MPI parallelization, we demonstrate an overall performance increase of 2.2×for the $d = 8$ nm nanowire (speed-up goes from 4.2 to 9.2). Without the "interleaved" approach, no device structure with a diameters larger than 5 nm could be self-consistently simulated.

3.3. Application to tunneling transistors

To illustrate the benefit of the "interleaved" approach, we investigate the properties of tunneling field-effect transistors (TFETs) [35]. TFETs are characterized by a very steep subthreshold slope (SS) which determines the amount of gate voltage required to increase their drain current by one order of magnitude. A low SS means that a small voltage is needed to switch a transistor from its OFF- to its ON-state, significantly reducing its power consumption. Silicon MOSFETs are limited to $SS = 60$ mV/decade at room temperature, but TFETs could theoretically exhibit values below 20 mV/decade. Before the semiconductor industry decides to invest resources into this novel technology it would like to know under which conditions TFETs properly work and whether they can reach mass production. The transfer characteristics $I_d - V_{gs}$ of InAs gate-all-



	Sequential				Interleaved	
P_0	BC (s)	Solve (s)	Total (s)	Sp. Up	Total (s)	Sp. Up
1	219	517	736	1×	736	1×
2	110	263	373	1.97×	373	1.97×
4	109	147	256	2.88×	263	2.80×
6	108	114	222	3.31×	154	4.78×
8	**108**	**89**	**197**	**3.73×**	**114**	**6.46×**
10	108	82	190	3.87×	113	6.51×
12	108	79	187	3.94×	113	6.51×
16	106	69	175	4.21×	110	6.69×

Fig. 2. Strong scaling results on 1–16 cores on Kraken [18] for the computation of one energy point of a circular nanowire (diameter $d = 8$ nm, length L=60 nm, matrix size $N_A t_b = 541,035$, matrix bandwidth $b = 2805$, number of atoms $N_A = 108,207$). The solid line with triangles refers to the time to compute the open boundary conditions only, Eqs. (3) and (4), the dashed curve to the time to compute the OBCs and then the wave function coefficients in a sequential way using the BCR algorithm. The solid curve with circles is the same as the dashed curve, but when the OBCs and the sparse LSE in Eq. (4) are treated in an interleaved way. The dashed-dotted line indicates the slope of ideal scaling. All the numerical results are reported in the table as well as the speed-up factor with respect to the time on a single processor.

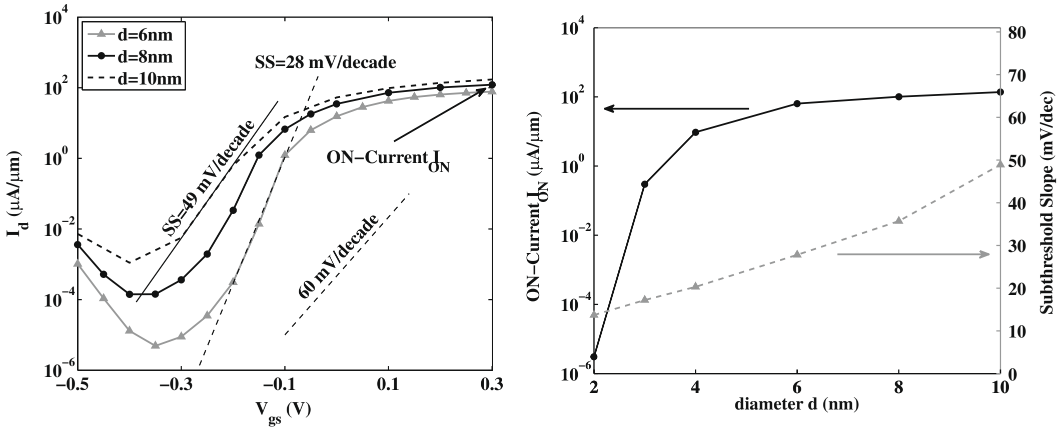


Fig. 3. (Left) Transfer characteristics $I_d - V_{gs}$ (at $V_{ds} = 0.2\,\text{V}$) of tunneling transistors based on gate-all-around nanowires with diameter $d = 6\,\text{nm}$ (solid line with triangles), $d = 8\,\text{nm}$ (solid line with circles), and $d = 10\,\text{nm}$ (dashed line) and length L=60nm. Two important metrics of such transistors are their subthreshold slope *SS* given in mV/decade and their ON-current I_{ON} in μA/μm. Both quantities are reported in the left subplot for diameters ranging from 2nm to 10nm.

	4,096 seq.	4,096 interleaved	8,192 interleaved	16,384 interleaved
Time (s)	3136	1971	1037	566
TFlop/s	4.1	6.6	12.6	23.1

Fig. 4. Walltime and TFlop/s performances on 4096–16,384 cores on Kraken for the calculation of one self-consistent iteration of an InAs GAA NW tunneling transistor (diameter $d = 8\,\text{nm}$, length $L = 60\,\text{nm}$) at $V_{gs} = -0.5$ V and $V_{ds} = 0.2\,\text{V}$. Two levels of parallelism are used, energy (5876 points) and spatial domain decomposition (8 MPI tasks and shared memory BLAS and LAPACK libraries).

around nanowire TFETs with three different diameters is given in Fig. 3 as well as the evolution of the device ON-current and *SS* as function of the nanowire diameter. OMEN predicts that *SS* can be maintained below 60mV/dec even for devices with $d = 10\,\text{nm}$, but the ON-current (130 μA/μm at $d = 10\,\text{nm}$) is too low to match the ITRS specifications [1] and requires further structure optimization [35].

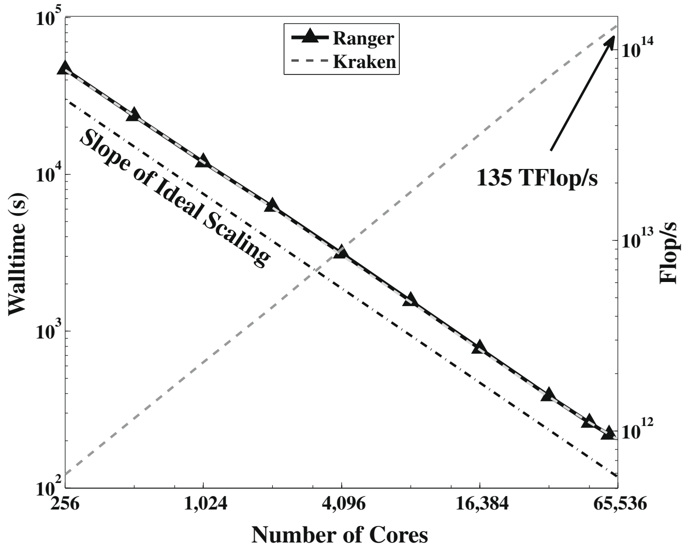
From a numerical point of view, a self-consistent Poisson iteration (time to solve Eqs. (3)–(5) for all energy points, to calculate carrier and current densities, and to solve Poisson equation in parallel) for a 3D InAs nanowire with $d = 8\,\text{nm}$ and $L = 60\,\text{nm}$ takes 3136 s on 4096 cores with the "sequential" approach and 1971 s with the "interleaved" approach. At the same time the sustained performance increases from 4.1 to 6.6 TFlop/s, corresponding to an improvement factor of 1.6. Each energy point, out of $N_E = 5876$, is computed using 2 nodes, 8 MPI tasks, 2 threads per MPI task, and the BCR algorithm. Further results are reported in Fig. 4. For example on 16,384 cores, one self-consistent iteration lasts 566 s and reaches 23.1 TFlop/s (improvement of 3.5 as compared to 4096 CPUs). If 20 bias points were simulated in parallel OMEN could work on more than 300,000 cores with very good performances.

4. Very high performance computing

4.1. Scaling up to 65,536 cores

With the four-level parallelization scheme described in Section 2.2, OMEN can scale almost perfectly up to 59,904 cores on Ranger, the Sun Constellation Star from TACC (2.3 GHz AMD processors) [36] and up to 65,536 cores on Kraken from NICS [18], where it reaches a sustained performance of 135 TFlop/s as shown in Fig. 5. The four MPI levels of parallelization of OMEN are used to simulate a two-dimensional (2D) Si double-gate (DG) ultra-thin-body (UTB) field-effect transistor (FET) with a body thickness $t_{body} = 5\,\text{nm}$ and a gate length $L_g = 22\,\text{nm}$ [37]. The device simulation includes 16 bias points, 16 momentum, from 800 to 1400 energy points per momentum, depending on the value of **k**, and spatial domain decomposition on 2 cores (block cyclic reduction method). OMEN exhibits an almost identical scaling behavior on Ranger and Kraken, both machines having the same AMD processors, and it scales almost perfectly up to the full machine capabilities (91% of parallel efficiency on Ranger, 89% on Kraken in going from 256 to 59,904 and 65,536 cores, respectively).

The scaling behavior of OMEN up to 65,536 cores shown in Fig. 5 can be further extended towards peta-scale computing if (i) a larger number of CPUs is used, (ii) the numerical algorithms are improved, (iii) and the parallelization scheme is optimized. All these issues are addressed in the following sections.



P_0	a (s)	Sp. Up	b (s)	Sp. Up
256	46730	1×	46277	1×
512	23540	1.99×	23521	1.97×
1024	11950	3.91×	11935	3.88×
2048	6218	7.51×	6138	7.54×
4096	3134	14.9×	3047	15.2×
8192	1557	30.0×	1530	30.2×
16384	775	60.4×	761	60.8×
32768	385	121×	381	121×
59904	219	213×	-	-
65536	-	-	203	227×

Fig. 5. Scaling performances of OMEN up to 65,536 cores for a 2D double-gate ultra-thin-body transistor structure using four MPI levels of parallelism (16 bias points, 16 momentum points, 800–1400 energy points, spatial decomposition on 2 cores with BCR). The time on 256 cores is the reference to calculate the speed-up factor. In experiment "a" (solid line with triangles) the simulations were run on Ranger. Experiment "b" (dashed line, lies on top of line with triangles) is the same as "a", but on Kraken. The Flop/s count for the experiment "b" is also given.

4.2. Work load balance

This section is dedicated to the improvement of OMEN's parallel performances on Kraken using a pure MPI approach. The main task consists in optimally balancing the work load among the different processors. The lowest parallelization level, spatial domain decomposition, has been treated in Section 3. The distribution of the energy points has been described in Ref. [11] and works properly. It remains the distribution of the bias points, which can be embarrassingly parallelized and is straight-forward, and the parallelization of the momentum points. We focus on the latter issue since the fixed number of cores per momentum group, P_k, becomes a limiting factor if the number of energy points N_E per momentum strongly differs from one $\mathbf{k}$ to the other. If some momentum groups have much more energy points to handle than others and the same number of cores working on them, they will take more time to finish their calculations. This can be resolved by dynamically assigning a variable number of cores to each momentum group according to their respective work load. The problem is that at the beginning of a device simulation $N_E(\mathbf{k})$ is not known and evolves at each Poisson iteration and for each new bias point. Hence, OMEN should be able to modify P_k at any time during the simulation.

To understand how this can be done the process of determining $N_E(\mathbf{k})$ is briefly described and summarized in the subplots (a) and (b) of Fig. 6. The bandstructure of the source $E_S(k_x, \mathbf{k})$ and drain $E_D(k_x, \mathbf{k})$ semi-infinite contacts is calculated as function of k_x, the wave vector along the transport direction. First, the subband minima of $E_S(k_x, \mathbf{k})$ and $E_D(k_x, \mathbf{k})$ are identified. Around these energy points the density-of-states of the device exhibits sharp peaks that must be accurately captured. This means that the energy vector needs a finer discretization and more points around the subband minima of $E_S(k_x, \mathbf{k})$ and $E_D(k_x, \mathbf{k})$. Obviously, the position of the subband minima as well as the number of subbands depend on $\mathbf{k}$, and are therefore different for each momentum group. This makes the energy vector E $\mathbf{k}$-dependent.

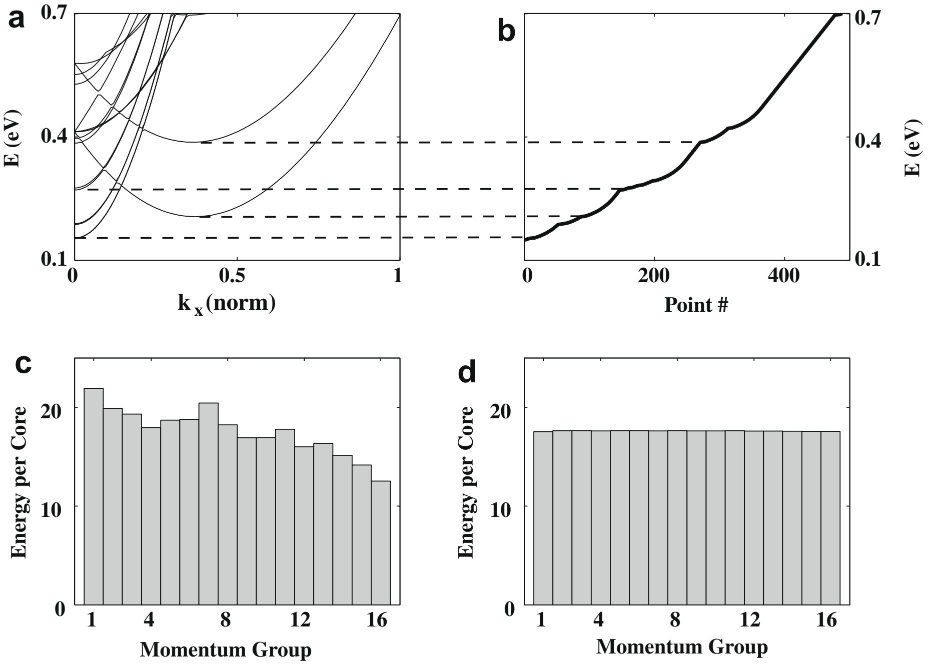


Fig. 6. Generation and parallelization of the momentum-dependent energy vector. (a) Conduction bandstructure $E(k_x, k_z = 0)$ of the drain contact for a 2D DG UTB transistor with a body thickness $t_{body} = 5\,\text{nm}$. (b) First 500 points of the corresponding energy vector. The energy is better resolved around subband minima. (c) Number of energy points per core for each of the momentum point (homogeneous distribution, 16 momentum points, 128 cores per momentum, 2 cores per OBCs and LSE, total of 2048 cores). (d) Same as (c), but with improved work load balance. The number of cores per momentum depends on the size of the energy vector. Here, 160 cores are attributed to k_{z1} (1403 energy points), but only 92 to k_{z16} (802 energy points).

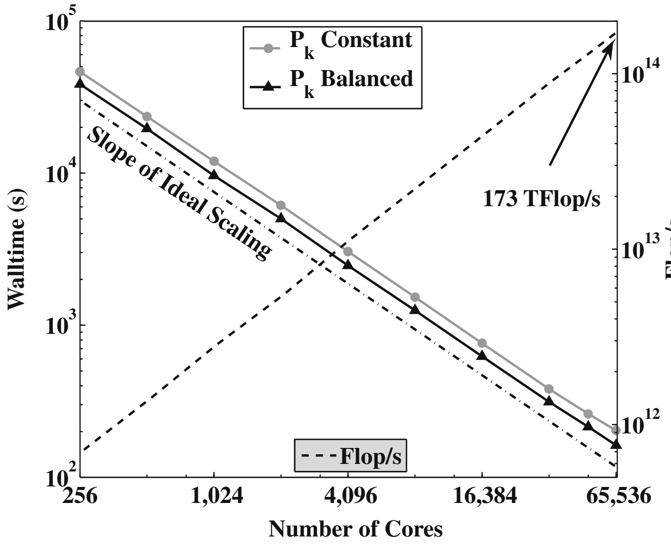
The improvement of the work load balance is realized by first defining at the beginning of the simulation one group "k_group" and one communicator "k_comm" for each momentum $\mathbf{k}$ containing the same number of cores P_k. This group and communicator are used to compute $E_S(k_x, \mathbf{k}), E_D(k_x, \mathbf{k})$, and to determine $N_E(\mathbf{k})$ according to the methodology given above. These operations require exactly the same amount of work for each $\mathbf{k}$. Based on these results the total number of energy points $N_{E,tot}$, the average number of energy points per core $N_{E,av}$, and the new number of cores per momentum $\widehat{P}_k(\mathbf{k})$ are calculated

$$\begin{aligned} N_{E,tot} &= \sum_{\mathbf{k}} N_E(\mathbf{k}), \quad N_{E,av} = N_{E,tot}/P_{V_g}, \\ \widehat{P}_k(\mathbf{k}) &= \text{round}(N_E(\mathbf{k})/N_{E,av}), \end{aligned} \tag{7}$$

where P_{V_g} is the number of cores per bias point. A new group "eq_k_group" and a new communicator "eq_k_comm" containing $\widehat{P}_k(\mathbf{k})$ cores are created for each momentum and used to solve Eqs. (3)–(5) for all the energy points. An example of the optimized distribution of the CPUs among the momentum points is presented in Fig. 6(c) and (d) for the same 2D nanodevice and the same simulation conditions as in Fig. 5. Without the improved work load balance each processor solves 22 energy points in one of the momentum group ($P_k = 128, N_E = 1400$ and $P_E = 2$) while each processor treats only 13 energy points in another group ($P_k = 128, N_E = 800$ and $P_E = 2$). In the improved version of OMEN each processor in any momentum group deals with 18 energy points, $\widehat{P}_k(\mathbf{k})$ going from 92 cores in the group where $N_E = 800$ to 160 where $N_E = 1400$. This results in a more than 20% decrease of the total simulation time. For each new Schrödinger–Poisson self-consistent iteration the value of $\widehat{P}_k(\mathbf{k})$, "eq_k_group" and "eq_k_comm" are re-adapted so that the work balance remains optimal.

Using the same 2D Si ultra-thin-body transistor as in the previous sections with 16 bias points, 16 momentum, from 800 to 1400 energy points per momentum, spatial domain decomposition on 2 cores (BCR algorithm), and the optimal work load balance given above, OMEN is tested up to 65,536 cores on Kraken. The scaling results and the sustained performances are shown is Fig. 7. The line with triangles refers to the new version of the code with an optimal work load balance, the line with circles to the version without. As a reference the transistor is first simulated on 128 cores, all being assigned to the same momentum point ($P_{V_g} = P_k = 128$). In this special configuration there is no problem of work load balance because one single $\mathbf{k}$-point is considered at the time. Then, from 256 to 65,536 cores, all the momentum points are simultaneously computed and the CPUs are dynamically allocated.

We see that the more balanced OMEN reaches a parallel efficiency of 88% in going from 128 to 65,536 cores, the simulation time is reduced by more than 20% as compared to the previous version of the code, and the sustained performance increases to 173 TFlops on 65,536 cores, a factor 1.28×better than in Fig. 5.



P_0	Walltime (s)	Speed Up	TFlop/s	|| Efficiency
128	73405	1×	0.37	100%
256	38372	1.91×	0.72	96%
512	19315	3.75×	1.41	94%
1024	9609	7.64×	2.87	95%
2048	5002	14.7×	5.5	92%
4096	2467	29.8×	11.4	93%
8192	1249	58.8×	22.5	92%
16384	624	117×	44.9	92%
32768	313	234×	89.8	92%
49152	215	341×	131	89%
65536	162	452×	173	88%

Fig. 7. Scaling performances of OMEN up to 65,536 cores for the same 2D DG UTB transistor structure as in Fig. 5, but with the improved work load balance described in Fig. 6. All simulations are performed on Kraken. The time on 128 cores is the reference for the speed-up factor. The solid line with circles corresponds to the experiment "b" in Fig. 5. The solid line with triangles shows the effect of the better distribution of the cores among the momentum points. The Flop/s performance is depicted by the dashed line. The table contains the measured walltimes, the speed-up factors with respect to 128 cores, the TFlop/s counts, and the parallel efficiency.

5. Application to an InAs high electron mobility transistor

The scaling results in the previous sections are based on the same 2D field-effect transistor structure that has not been fabricated yet and remains therefore a fictitious example. By combining the "interleaved" approach described in Section 3 to efficiently treat large simulation domains and the improved work load balance presented in Section 4 to distribute the CPUs among the momentum groups, we are now able to simulate real devices, reproduce experimental data, and propose device optimizations.

Recently III–V high electron mobility transistors (HEMTs) have started to attract a lot of attention for their possible application as low power circuit components [6]. A realistic InAs HEMT with a 40 nm gate length and a multi-quantum-well InGaAs-InAs-InGaAs active region is sketched in Fig. 8. InAs HEMTs usually extend over several micro meters, but their active region can be safely limited to a 2D rectangular domain along the x- and y-axis of size 12 nm × 140 nm and composed of $N_A = 38,556$ atoms. Periodic boundary conditions are applied along the z-axis. In this representation the source and drain extensions are modeled through series resistances attached to the active region. Poisson equation is solved on a larger domain of size 50 nm × 140 nm. The transfer and output characteristics simulated with OMEN are compared to experimental data and reported in Fig. 8. A good quantitative agreement is achieved over a wide range of bias points confirming the quality of the physical models [20].

The size and the complexity of the transistor structure depicted in Fig. 8 make it a particularly interesting case to test the "interleaved" approach and the improved work load balance. The time and sustained performances to simulate 16 bias point of this structure are given in Fig. 9 using the four-levels of parallelism of OMEN. The calculation of each bias point is limited

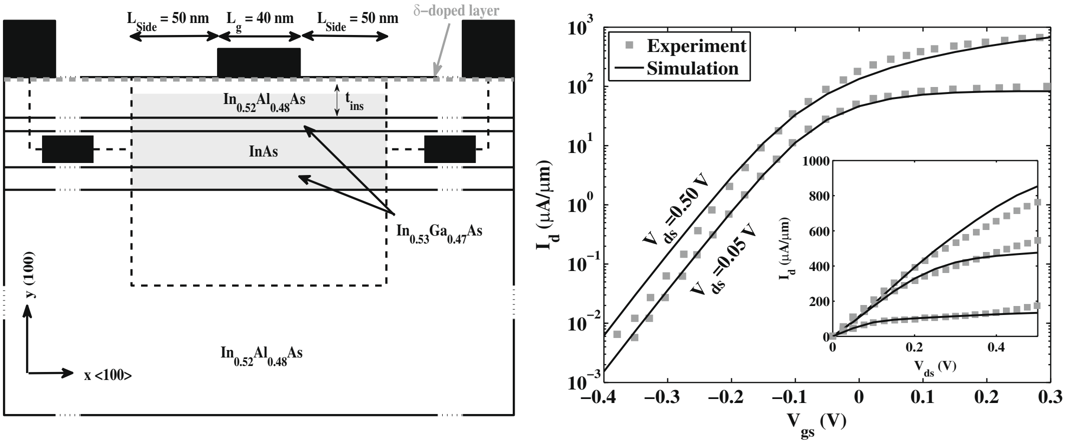


Fig. 8. (Left) Schematic view of a III-V HEMT composed of 5 nm InAs quantum-well embedded between two $In_{0.53}Ga_{0.47}As$ layers (2 nm on top and 3 nm on the bottom), deposed on a thick $In_{0.52}Al_{0.48}As$ layer, and separated from the gate contact by another 4 nm $In_{0.52}Al_{0.48}As$ insulator layer. The 12 nm × 140 nm shaded region delimits the active part of the transistor containing $N_A = 38,556$ atoms resulting into a matrix A of size $N_A t_b = 385,560$ in Eq. (5). (Right) Simulated (solid lines) and experimentally measured (squares) transfer and output characteristics of the HEMT.

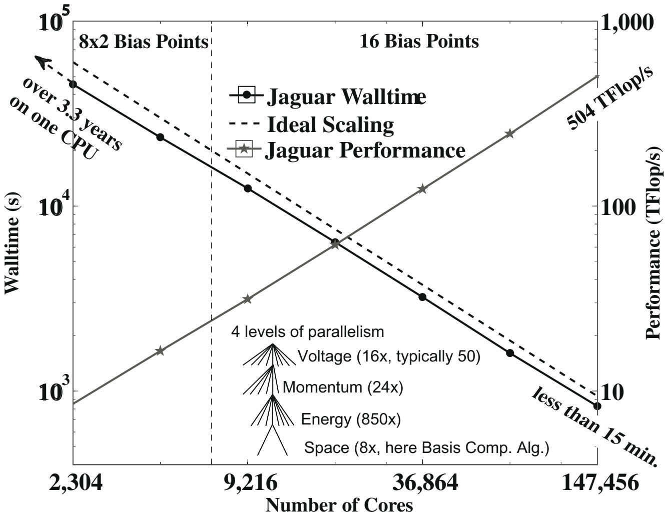


Fig. 9. OMEN scaling performances on Jaguar [19] from 2,304 to 147,456 cores to simulate 16 bias points of the HEMT transistor depicted in Fig. 8. A total of 24 momentum points are used, the number of energy points $N_E(\mathbf{k})$ varies from 124 to 866, and spatial domain decomposition is performed with the BCR algorithm and the "interleaved" approach on 8 cores. The line with circles refer to the simulation time, the line with stars to the sustained performance.

to three self-consistent Poisson iterations, it includes 24 momentum points, the number of energy points $N_E(\mathbf{k})$ is different for each momentum and is comprised between 124 and 866, requiring an adaptive distribution of the work load. The block cyclic reduction algorithm and the interleaved approach introduced in Section 3.2 are used to spatially decompose the simulation domain on 8 cores.

The simulation time is reduced to less than 15 min on 147,456 cores on Jaguar, the CRAY XT5 at NCCS, while it would have lasted more than 3.3 years on a single CPU. OMEN reaches a scaling performance of 504 TFlop/s on 147,456 cores, corresponding to about 37% of the peak performance (1.357 PFlop/s on 147,456 cores). On a computer with a peak performance of 2.7 PFlop/s, OMEN would break the peta-scale barrier. For example, this could be achieved if more bias points were computed and sufficient CPUs were available.

6. Conclusion

We have presented recent progresses in the development of the nanoelectronics device simulator OMEN. An "interleaved" approach to simultaneously calculate the open boundary conditions and the device wave function and an optimized work load balance of the momentum points have been proposed to accelerate the simulation of nanotransistors and increase the simulation domains. An almost ideal scaling of the simulation time up to 59,904 and 65,536 cores on two TeraGrid machines, Ranger at TACC and Kraken at NICS has been demonstrated with a sustained performance of 173 TFlop/s on 65,536 cores as well as a sustained performance of 504 TFlop/s on 147,456 cores on Jaguar at NCCS in the simulation of a realistic HEMT device. OMEN has the potential to reach 1 PFlop/s on a machine with a peak performance of 2.7 PFlop/s or more.

The next steps consists in improving the physical models by including additional effects like electron-phonon scattering, to extend the material database beyond conventional semiconductors, to still improve the parallel linear solvers and the computation of the open boundary conditions, to reproduce further experimental data, and to help experimentalists design better transistors. However, the upper device size that can be simulated is limited by the efficiency of the numerical algorithms: the current parallelization of the spatial domain decomposition does not allow to break the device structure along its cross section, only along its length. Hence, when the device cross section increases beyond 80 nm^2, the required memory per core drastically increases and the computational cost becomes too important. The lower device size that can be treated depends on the accuracy of the device description. When the device dimension does not exceed 2–3 nm, the position of the atoms can no more be assumed ideal and molecular dynamics simulations have to be performed first to obtain the relaxed atom positions.

A simplified version of OMEN running on 1–256 cores is currently available on nanohub.org [40] and is accessible to everybody. We plan to deploy the full capabilities of OMEN on the nanohub.org and submit the jobs to the largest available computational resources.

Acknowledgement

This work was partially supported by NSF grant EEC-0228390 that funds the Network for Computational Nanotechnology, by NSF PetaApps grant number 0749140, and by NSF through TeraGrid resources provided by TACC and NICS. The authors would like to thank Dr. T. Minyard at TACC, Dr. B. Loftis at NICS, and Dr. D. Kothe at NCCS for providing them with full machine runs on Ranger, Kraken, and Jaguar, respectively.

References

[1] http://www.itrs.net/reports.html.

[2] B. Doris et al, Extreme scaling with ultra-thin Si channel MOSFETs, IEDM Tech. Dig. (2002) 267–270.

[3] Y. Cui, L.J. Lauhon, M.S. Gudiksen, J. Wang, C.M. Lieber, Diameter-controlled synthesis of single-crystal silicon nanowires, Appl. Phys. Lett. 78 (2001) 2214.

[4] S.D. Suk et al, Investigation of nanowire size dependency on TSNWFET, IEDM Tech. Dig. (2007) 891–894..

[5] X. Wang, Y. Ouyang, X. Li, H. Wang, J. Guo, H. Dai, Room-temperature all-semiconducting sub-10-nm graphene nanoribbon field-effect transistors, Phys. Rev. Lett. 100 (2008) 206803.

[6] D.H. Kim, J.A. del Alamo, 30-nm InAs Pseudomorphic HEMTs on an InP substrate with a current-gain cutoff frequency of 628 GHz, IEEE Elec. Dev. Lett. 29 (2008) 830–833.

[7] J. Appenzeller, Y.-M. Lin, J. Knoch, Ph. Avouris, Band-to-band tunneling in carbon nanotube field-effect transistors, Phys. Rev. Lett. 93 (2004) 196805.

[8] W.Y. Choi, B.-G. Park, J.D. Lee, T.-J. King Liu, Tunneling field-effect transistors (TFETs) with subthreshold swing (SS) less than 60 mV/dec, IEEE Elec. Dev. Lett. 28 (2007) 743–745.

[9] M. Luisier, G. Klimeck, A. Schenk, W. Fichtner, Atomistic simulation of nanowires in the $sp^3d^5s^*$ tight-binding formalism: from boundary conditions to strain calculations, Phys. Rev. B 74 (2006) 205323.

[10] M. Luisier, A. Schenk, Atomistic simulation of nanowire transistors, J. Comput. Theor. Nanosci. 5 (2008) 1031–1045.

[11] M. Luisier, G. Klimeck, A multi-level parallel simulation approach to electron transport in nano-scale transistors, in: Proceedings of the 2008 ACM/IEEE Conference on Supercomputing, article 12, (2008).

[12] J.C. Slater, G.F. Koster, Simplified LCAO method for the periodic potential problem, Phys. Rev. 94 (1954) 1498–1524.

[13] T.B. Boykin, G. Klimeck, F. Oyafuso, Valence band effective-mass expressions in the $\text{sp}^3\text{d}^5\text{s}^*$ empirical tight-binding model applied to a Si and Ge parametrization, Phys. Rev. B 69 (2004) 115201.

[14] T.B. Boykin, G. Klimeck, R. Chris Bowen, F. Oyafuso, Diagonal parameter shifts due to nearest-neighbor displacements in empirical tight-binding theory, Phys. Rev. B 66 (2002) 125207.

[15] G. Klimeck, R. Chris Bowen, T.B. Boykin, T.A. Cwik, sp^3s^* Tight-binding parameters for transport simulations in compound semiconductors, Superlattices Microstruct. 27 (2000) 519–524.

[16] J.M. Jancu, R. Scholtz, F. Beltram, F. Bassani, Empirical $spds^*$ tight-binding calculation for cubic semiconductors: general method and material parameters, Phys. Rev. B 57 (1998) 6493–6507.

[17] J. Wang, A. Rahman, A. Ghosh, G. Klimeck, M.S. Lundstrom, On the validity of the parabolic effective-mass approximation for the current-voltage calculation of silicon nanowire transistors, IEEE Trans. Elec. Dev. 52 (2005) 1589–1595.

[18] http://www.nics.tennessee.edu/computing-resources/kraken.

[19] http://www.nccs.gov/computing-resources/jaguar/.

[20] M. Luisier, G. Klimeck, Full-band and atomistic simulation of realistic 40 nm InAs HEMT, IEDM Tech. Dig. (2008) 887–890.

[21] D.Z.-Y. Ting, E.T. Yu, T.C. McGill, Multiband treatment of quantum transport in interband tunnel devices, Phys. Rev. B 45 (1992) 3583–3592.

[22] M.P. Lopez Sancho, J.M. Lopez Sancho, J. Rubio, Highly convergent schemes for the calculation of bulk and surface Green functions, J. Phys. F: Met. Phys. 15 (1985) 851–858.

[23] C. Rivas, R. Lake, Non-equilibrium Green function implementation of boundary conditions for full band simulations of substrate-nanowire structures, Phys. Stat. Sol. B 239 (2003) 94–102.

[24] M. Städele, B.R. Tuttle, K. Hess, Tunneling through ultrathin SiO_2 gate oxides from microscopic models, J. Appl. Phys. 89 (2001) 348–363.

[25] P.R. Amestoy, I.S. Duff, J.-Y. L'Excellent, Multifrontal parallel distributed symmetric and unsymmetric solvers, Comput. Methods Appl. Mech. Eng. 184 (2000) 501.

[26] X.S. Li, J.W. Demmel, SuperLU_DIST: a scalable distributed memory sparse direct solver for unsymmetric linear systems, ACM Trans. Math. Softw. 29 (2003) 110.

[27] T.B. Boykin, M. Luisier, G. Klimeck, Multi-band transmission calculations for nanowires using an optimized renormalization method, Phys. Rev. B 77 (2008) 165318.

[28] O. Schenk, K. Gärtner, Solving unsymmetric sparse systems of linear equations with PARDISO, J. Future Gener. Comput. Syst. 20 (2004) 475.

[29] M. Luisier, A. Schenk, W. Fichtner, T.B. Boykin, G. Klimeck, A parallel sparse linear solver for nearest-neighbor tight-binding problems, in: Proceedings of the 14th International Euro-Par Conference on Parallel Processing, (2008), pp. 790–800.
[30] P.M. Gresho, R.L. Sani, Incompressible Flow and the Finite Element Method: Isothermal Laminar Flow, John Wiley and Sons, New York, 2000.
[31] R.E. Bank, D.J. Rose, W. Fichtner, Numerical methods for semiconductor device simulation, IEEE Trans. Electron Dev. 30 (1983) 1031.
[32] R.S. Tuminaro, M. Heroux, S.A. Hutchinson, J.N. Shadid, Official Aztec User's Guide: Version 2.1, (1999).
[33] W. Gropp, E. Lusk, N. Doss, A. Skjellum, A high-performance, portable implementation of the MPI message passing interface standard, Parallel Comput. 22 (1996) 789.
[34] C. Catlett et al, TeraGrid: analysis of organization, system architecture, and middleware enabling new types of applications, in: Lucio Grandinetti (Ed.), HPC and Grids in Action, Advances in Parallel Computing, IOS Press, Amsterdam, 2007.
[35] M. Luisier, G. Klimeck, Atomistic, full-band design study of InAs band-to-band tunneling field-effect transistors, IEEE Elec. Dev. Lett. 30 (2009) 602.
[36] http://www.tacc.utexas.edu/resources/hpcsystems/#constellation.
[37] M. Luisier, Gerhard Klimeck, Full-band and atomistic simulation of n- and p-doped double-gate MOSFETs for the 22 nm technology node, in: Proceedings of the International Conference on the Simulation of Semiconductor Processes and Devices (SISPAD) 2008, vol. 17, (2008).
[38] J. Dongarra, Basic linear algebra subprograms technical forum standard, Int. J. High Perform. Appl. Supercomput. 16 (2002) 1–111.
[39] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, D. Sorensen, LAPACK User's Guide, third ed., SIAM, Philadelphia, 1999.
[40] http://www.nanohub.org/resources/omenwire.