Divide-and-conquer quantum mechanical material simulations with exascale supercomputers

Lin-Wang Wang

ABSTRACT
Recent developments in large-scale materials science simulations, especially under the divide-and-conquer method, are reviewed. The pros and cons of the divide-and-conquer method are discussed. It is argued that the divide-and-conquer method, such as the linear-scaling 3D fragment method, is an ideal approach to take advantage of the heterogeneous architectures of modern-day supercomputers despite their relatively large prefactors among linear-scaling methods. Some developments in graphics processing unit (GPU) electronic structure calculations are also reviewed. The accelerators like GPU could be an essential part for the future exascale supercomputing.

Keywords: $O(N)$ method, electronic structure calculations, exascale supercomputing, density functional theory, LS3DF

THE DRAMATIC INCREASE IN SUPERCOMPUTER SPEED

Ever since computers have been used for scientific computing, we have witnessed an exponential growth in their speeds, as illustrated in Fig. 1 [1]. The fastest supercomputers of today are more than a million times faster than the supercomputers used 25 years ago when this author started his career in this field. Today, a typical simulation run on the Titan machine at the Oak Ridge Leadership Computing Facility (OLCF) will consist of 50 000 computer cores with a theoretical speed more than one petaflops. One hour of such a calculation by Titan is more than 100 years of computation to a mainframe computer of 25 years ago. In other words, if a job submitted to a mainframe computer 25 years ago has just finished today, one could resubmit the same job now and expect it to be finished within 15 minutes; however, this is assuming that the original code can be run as efficiently on the 50 000 computer cores as on a single computer core. In reality, this is far from the case. Running an application on a massive parallel computer presents a unique challenge that requires significantly modifying the computer code and sometimes changing the algorithms starting with fundamental physics approximations.

Due to the consideration of power consumption and heat dissipation, the clock speed of a modern computer chip has been fixed at around a few GHz, as shown in Fig. 2 [2]. As a result, the floating point operation on a single core is not getting any faster. Instead, the reduction of transistor size and the corresponding increase in the number of transistors on a single integrated circuit die have translated into an increase of the number of CPU cores on a chip [2]. This multi-core computation has dominated high-performance computing (HPC) in the last 20 years. About five years ago, the HPC community has geared up for the exascale computing ($10^{18}$ FLOPS, floating point operation per second). The goal for the US Department of Energy (DOE) is to have an exascale computer around 2020 [3]. China is now a major player in the race with the United States, Japan, and Europe to develop an exascale computer. The Tianhe-1A and Tianhe-2 supercomputers developed by the National University of Defense Technology (NUDT) in China have been crowned the No. 1 computer on the Top 500 list in November 2010, June 2013, and November 2013 [4]. The total electric power consumption of the NUDT supercomputers is immense—for example, Tianhe-2 consumes as much as 17 MW [4].
The US DOE has set a goal of 20 MW for its exascale supercomputer [3]. The pressure to reduce electric power consumption has forced the designers of such machines to use lean core accelerators. In the case of Tianhe-2, it is the Intel Many Integrated Core architecture [4]. For OLCF’s Titan, currently the No. 2 computer on the Top 500 list, it is the graphics processing unit (GPU) accelerator [5]. Such accelerators often feature the single instruction multiple data computational paradigm [6], with all the cores executing the same instructions (programs) simultaneously using different data sets. As a result, the core itself can be very lean, consuming minimum amounts of energy.

The introduction of accelerators has further complicated supercomputer architectures. Instead of a homogeneous multi-core or multi-node computer, as it has been for the last 20 years, we now face a truly hierarchical architecture with node, CPU core, and accelerators. Fig. 3 shows the schematics of one node of the Cray XK7 machine [7], which makes up the Titan machine. Each node consists of several AMD opteron chips (each chip with 16 CPU cores), and one or two Nvidia GPU chips (each GPU chip has one or a few thousand Compute Unified Device Architecture or CUDA cores and a few TFLOPS of computing power). A supercomputer like Titan has 18 688 such nodes connected by the Cray Gemini interconnect in a 3D torus topology. The future exascale computers could have about a million such nodes [5]. It is a tremendous challenge to run scientific simulation codes efficiently on such supercomputers. This requires us to re-think about the fundamental algorithms, at both the

Figure 1. The increase of computer speed over the years. © Advanced Micro Devices Inc. AMD, the AMD logo, AMD and combinations thereof are trademarks of, and used with permission from, Advanced Micro Devices, Inc.

Figure 2. The continued increase of the number of transistors on a die and the saturation of the clock speed. Reprinted with permission from [2]. Copyright 2010 Herb Sutter.
computer science level and the physics approximation level. One needs to think about the simulation as a whole, e.g. what kind of physics approximation could fit best with the exascale computer architectures. Sometimes this also requires the collaboration between domain scientists (e.g. material scientists) and computer scientists. For the past 20 years, computer languages for parallelization have been dominated by Message Passing Interface (MPI) \[8, 9\]. Now new language has been added to program the accelerators, e.g. the CUDA \[10\] for the GPU processors.

**LARGE-SCALE MATERIAL SIMULATIONS USING DENSITY FUNCTIONAL THEORY**

Numerical simulation has long emerged as the third cornerstone of modern scientific investigations, along with experiment and analytical theory. In materials science, the analytical theory also heavily depends on numerical simulations. The materials science simulations can be divided into three categories: (1) the continuous field simulations (e.g. Navier–Stokes equation for fluid dynamics, Maxwell’s equations for electrostatics and photonics, and the Ginzburg–Landau theory for material phase transitions or magnetic properties), (2) the classical atomic force field methods, and (3) the quantum mechanical (QM) methods. In this paper, we will focus on the QM method, which is the most accurate but also the most time-consuming approach. In a survey a few years ago, it was found that 70% of computer time for materials science simulation allocated within the National Energy Research Scientific Computing Center (NERSC) was devoted to QM simulations \[11\]. In 2013, the Nobel Prize in Chemistry was awarded to pioneers who combined the classical atomic force field with QM simulations in a quantum mechanical/molecular mechanical (QM/MM) approach \[12\]. In 1998, the Nobel Prize in Chemistry was awarded to Walter Kohn for the development of density functional theory (DFT) \[13, 14\] and to John A. Pople for the development of quantum chemistry computational methods.

The development of DFT has significantly accelerated \textit{ab initio} materials science simulations. Nowadays, the majority of QM simulations are carried out using DFT. This is due to its relative simplicity and surprising good accuracy. The DFT simulations with recently developed exchange correlation functionals can be more accurate than the quantum chemistry Møller–Plesset perturbation (MP2) method, but with much lower computational costs \[15\]. Besides DFT, there are other QM approaches, most noticeably the quantum Monte Carlo method \[16\] and all the quantum chemistry methods \[17\]. In this paper, we will focus on the DFT method.

In DFT, the many-body wave function $\Psi(r_1, r_2, ..., r_N)$ has been replaced with single particle Kohn–Sham orbitals $\tilde{\Psi}_i(r)$ for $i = 1, N$. Here $N$ is the number of electrons in the system. The Kohn–Sham orbital is solved via the Kohn–Sham equation:

$$\left\{ -\frac{1}{2} \nabla^2 + \int \frac{\rho(r')}{|r - r'|} d^3r' + \sum_R \frac{Z_R}{|r - R|} + \mu_{xc}[\rho(r)] \right\} \psi_i(r) = \varepsilon_i \psi_i(r). \quad (1)$$

Here $R$ is the nuclear position, $Z_R$ is the nuclear charge, and $\rho(r) = \sum_{i=1,N} |\psi_i(r)|^2$ is the electron charge density of the system. $\mu_{xc}[\rho]$ is the exchange correlation potential, which equals to a functional derivative $\partial E_{xc}[\rho]/\partial \rho$ of the exchange correlation functional $E_{xc}[\rho]$. The $E_{xc}[\rho]$ is the essential functional of DFT. Over the years, many $E_{xc}[\rho]$, both within DFT or beyond DFT
REVIEW

ε in hybrid functional, a part of the exchange energy correlation functional:

\[ \sum \text{explicit exchange integral} \]

described as a Jacob’s ladder by J. Perdew, as illustrated in Fig. 4 [18]. At the lowest level, there is the local density approximation (LDA) [19], where

\[ E_{\text{xc}} = \int \rho(r) \varepsilon_{\text{xc}}(\rho(r)) \, d^3r, \]

and the function \( \varepsilon_{\text{xc}}(\rho) \) is obtained from a quantum Monte Carlo simulation of homogeneous electron gas. At the next level, in the generalized gradient approximation (GGA) [20], the gradient of \( \rho \) is used in \( \varepsilon_{\text{xc}}(\rho, \nabla \rho) \). At the meta-GGA level [21], which strictly speaking is beyond the initial DFT formalism, the kinetic energy density

\[ \tau(r) = \sum_i |\nabla \psi_i(r)|^2 \]

is also used in the exchange correlation functional:

\[ \varepsilon_{\text{xc}}(\rho, \nabla \rho, \nabla^2 \rho, \tau). \]

In the hybrid functional, a part of the exchange energy in \( \varepsilon_{\text{xc}}(\rho, \nabla \rho, \nabla^2 \rho) \) has been replaced with the explicit exchange integral

\[ \sum_{i,j} \int \psi_i(r) \psi_j(r) \nu(r-r') \psi_i(r') \psi_j(r') \, d^3r \, d^3r', \]

where \( \nu(r-r') \) kernel is either the original Coulomb interaction, such as in B3LYP [22] or a screened Coulomb interaction, such as in HSE [23]. One can also start with the meta-GGA functional, then mix the exchange energy part with the explicit exchange integrals, such as in the M06 functional [24].

Finally, in the random phase approximation (RPA) [25–28], the correlation energy in \( \varepsilon_{\text{xc}} \) is replaced by the correlation energy calculated from the dielectric response function of the system. This can introduce the truly nonlocal correlation effects as in van der Waals’ interaction; however, there is a computational trade-off. While in hybrid or lower functionals in the Jacob’s ladder, only the occupied single particle orbitals \( \Psi_i \) need to be calculated, in RPA, all the unoccupied orbitals also need to be calculated to yield the dielectric response function. The RPA under GW approximation [29,30] is a many-body perturbation theory, which starts with a single Slater determinant. On the other hand, the dynamic mean field theory (DMFT) [31] explicitly calculates the on-site (one atom) correlation effects using a local configuration interaction approach. It can be used to calculate strongly correlated systems, e.g. f-state problems with multi-configurations (beyond single Slater determinant approximations). The benchmark for some of these functionals can be found in [15,24,32].

The availabilities of petascale and future exascale computers, together with the DFT method, have opened up vast opportunities in material simulations, especially to match the material complexities in real systems. While the traditional \textit{ab initio} simulations were focused on bulk crystals or small molecules, the future exascale computation will allow us to simulate mesoscale systems and phenomena where many of the real material properties reside on. There are three major challenges in simulating such mesoscale phenomena: (1) the accuracy, (2) the long-temporal scale, and (3) the large-size scale. The situation for accuracy has been summarized in the Jacob’s ladder of Fig. 4 and in the discussion of functional in the last section of this paper. In practice, current-day simulations are mostly stopped at the hybrid level. Besides the discussion provided above, there are many empirical approaches in actual calculations. For example, the semi-empirical LDA+U or GGA+U [33–35] have been used to deal with the on-site strongly correlated systems (instead of the more rigorous DMFT approach). The hybrid methods (e.g. HSE [21]) with adjustable parameters (the mixing parameter) have been used to correct the single electron eigen energy and band gap error (instead of the GW method).

Empirical classical pair potentials [36] as well as approximated nonlocal functionals based on the charge density \( \rho(r) \) [37,38] have been used instead of the expensive RPA calculations to describe the van der Waals’ interaction. However, with the availability of exascale computers, it is natural to expect that we will progress toward higher level methods in the Jacob’s ladder, especially perhaps in combination with the divide-and-conquer (DaC) method to combat the high power law size scalabilities in these high-level methods.

The temporal scale is another extremely challenging problem. While many physical processes happen in the timescale of nanosecond, microsecond, or even seconds, the \textit{ab initio} molecular dynamics (MD) simulations usually can only be carried out for hundreds of picosecond with 1–2 fs per MD step. 
Each MD step takes about 1 minute of wall-clock time to calculate. Due to the causality requirement, it is rather difficult to speed up the temporal scale using parallel computation. One possible approach is to develop accelerated MD (AMD) methods [39,40], where one abandons the calculations of exact microscopic molecular trajectories but tries to maintain the correct statistical properties (e.g. the rate of the rare events, such as the potential valley hopping). Since the system needs to run for a long time before a rare event happens, parallel replication and acceleration become possible to reduce the simulation wall-clock time [41]. The faster computational speed from the future exascale computer, e.g. provided by the accelerators, can perhaps reduce the 1 minute wall-clock time for each MD step to about a few seconds. This can bring an algorithmic paradigm change, for example, to enable the use of \textit{ab initio} calculations for AMD algorithm, which requires thousands of steps before a rare event happens. Currently, the AMD methods are mostly used with classical force fields. Another commonly used approach is the kinetic Monte Carlo (KMC) method [42], with the energy barrier or individual transition rate pre-calculated. The challenge for KMC is to identify the correct transition paths and events. Perhaps the resource provided by exascale computers will enable one to search all the possible pathways and nearby hopping sites in a systematic and automatic on-the-flight fashion.

The last challenge for mesoscale simulation is the system size, which is the problem we will focus on in this paper. It is now possible to calculate systems with hundreds or even 1000 to 2000 atoms using direct DFT methods based on Equation (1) using plane wave or atomic orbital basis sets on petascale computers. However, to study many nanoscale and mesoscale problems, such as large quantum dots, dislocation, incommensurate lattice interfaces, complex grain boundaries, and amorphous materials, one easily runs into systems containing tens of thousands of atoms, or even millions of atoms. In such cases, direct application of Equation (1) is inappropriate. In Equation (1), the number of single particle orbital \( N \) is proportional to the size of the problem. The number of basis function (e.g. plane wave) is also proportional to the problem size. Since orthonormalization enforcement among orbital pairs is an essential step in the calculation, the computational cost scales as \( O(N^3) \). Thus, a 1000-fold computer power increase from petascale to exascale can only increase the system size by a factor of 10, assuming the same computational efficiency. This is not the best way to use exascale computers for material simulations. When the system size and computer power are both increased, there are better computational approaches to be adopted, such as the \( O(N) \) method [43,44]. The \( O(N) \) method takes the advantage of the locality (nearsightedness) of the QM effects [45] to create algorithms with their computational complexities (cost) scale linearly to the system size. While there are many \( O(N) \) methods, we will focus on DaC methods, which fit well with the hierarchical supercomputer architecture discussed above, and hence allows efficient parallel scalability up to the exascale.

**DaC METHODS**

One of the major challenges for efficient use of HPC is to reduce the communication between processors. With about a million processors for future exascale supercomputers, the system will stagnate if every processor is communicating with all the other processors. To reduce global communication, the best way is to assign a group of processors (e.g. a few computer nodes with all their CPU and GPU processors) to work on an isolated system, and only combine the results at the end of these independent calculations to yield the result of the whole system. In this way, the major communications will be confined within each processor group. To use this strategy, one has to divide the whole system into small problems, which is the DaC approach. In a DaC method, the whole system is spatially divided into small fragments, and each fragment will be calculated using a few computer nodes for their QM properties. The QM results (the energies and charge densities) of the fragments will then be assembled into the results of the whole system. Since the electrostatic Coulomb interaction is long range, it is thus necessary to solve Poisson’s equation globally for the full system when the self-consistent field (SCF) iterations are carried out. However, since the most time-consuming part of the calculation is the QM calculations of the fragments, the computational complexity scales linearly to the size of the system. This is because the number of fragments will increase linearly with the size of the whole system while the size of each fragment can remain the same. If more computer nodes are used with increased system size (hence the increased number of fragments), the total computational time can remain the same. Thus, a perfect weak scaling for such calculations can be achieved.

The key to the DaC approach is to divide the full system into fragments and to assemble the fragments to yield the full system properties. The electron charge density \( \rho(r) \) is an essential property under DFT since it is used to construct all the other properties within the DFT formalism. There
are different strategies to divide and assemble the systems. In the embedding method [46,47], the total energy of the whole system is rewritten using fragment wave functions $\Psi_{F,i}$ as

$$E = \sum_F \left\{ -\frac{1}{2} \sum_i \int \psi_{F,i} \nabla^2 \psi_{F,i} \, d^3r - T[\rho_F] \right\}$$

$$+ T[\rho] + U[\rho].$$

(2)

Here $F$ is the fragment index, $\rho_F = \sum_i \psi_{F,i}^2$ is the fragment charge density, and $\rho = \sum_F \rho_F$ is the total charge density of the system. $U[\rho]$ is the total potential energy due to $\rho$, including Hartree energy $1/2 \int \rho(r) \rho(r')/|r-r'| \, d^3r \, d^3r'$, electron-ionic energy $-\int \rho(r) \sum R \rho(R)/|r-R| \, d^3r$, here $Z_R$ is the nuclear charge of atom $R$, DFT exchange correlation energy $\int \rho(r) \epsilon_{xc} \, d^3r$, and ion-ion Ewald energy $1/2 \sum_{R,R'} Z_R Z_{R'}/|R-R'|$. Here for simplicity, we have not used pseudopotential expression [48] for the ionic potential. In reality, especially when plane waves are used as the basis set, the ionic potential $Z_R/|r-R|$ will be replaced by the pseudopotential, and a nonlocal potential operator will exist in the total energy expression [48]. $T[\rho_F]$ and $T[\rho]$ are approximated kinetic energy as a functional of charge density $\rho$. One possible choice is to use the Thomas–Fermi approximation, where $T[\rho] = \frac{1}{10} (3\pi^2)^2/3 \int \rho(r) \frac{\hbar^2}{m} \, d^3r$. Taking the minimum of $E$ in Equation (2) with respect to fragment wave function $\Psi_{F,i}$: $\partial E/\partial \psi_{F,i} = \epsilon_{F,i} \psi_{F,i}$, one obtains the corresponding Schrödinger equation for fragment wave function. The term $\partial(T[\rho] - T[\rho_F])/\partial \rho_F$ provides a confinement potential for the fragment wave function.

While elegant in its formalism and efficient in its computation, the embedding methods can suffer from inaccuracy due to the approximate nature of the kinetic energy terms $T[\rho_F], T[\rho]$, in treating the boundary regions. In a way, the boundary region is not covered by accurate QM treatment. In other words, the fragments are not overlapping, and the total system charge density is simply the sum of the fragment charge densities. Embedding methods can also take a nonhomogeneous form, e.g. treat a central fragment with quantum chemistry calculations and all the other areas with orbital-free calculations (based purely on $\rho(r)$) [46]. In a way, the QM/MM method is also an embedded method while the central part is treated quantum mechanically and all the other parts are treated classically [12].

Another set of methods use overlapping fragments. One such scheme was originally proposed by Weitao Yang in 1991 [49]. In that approach, only the central part of a fragment is used to construct the global charge density: $\rho(r) = \sum_F W_F(r) \rho_F(r)$, where $W_F(r) \leq 1$ is a partition function, which satisfies $\sum_F W_F(r) = 1$. Typically, for a fragment, only the central part of its charge density is used, and hence the fragments are overlapping with each other. This ensures that every point in space will be covered by the central part of one fragment. Under this approach, the total energy of the system can be expressed as

$$E = \sum_F \sum_i \rho(\epsilon_{F,i}, E_F) \int W_F(r) \nabla^2 \psi_{F,i} \, d^3r$$

$$+ U[\rho].$$

(3)

Here $\rho(\epsilon_{F,i}, E_F)$ is the occupation number determined by the fragment eigen energy $\epsilon_{F,i}$ and a global Fermi energy $E_F$, using e.g. the Fermi–Dirac distribution function. $E_F$ is adjusted to yield the correct total electron charge of the whole system. The fragment charge density is then calculated as $\rho_F = \sum_i \rho(\epsilon_{F,i}, E_F) |\psi_{F,i}|^2$.

One problem is that if Equation (3) is used to represent the total energy and $\partial E/\partial \psi_{F,i} = \epsilon_{F,i} \psi_{F,i}$ is used to yield the Schrödinger equation for the fragment wave function $\Psi_{F,i}$, an unphysical equation involving terms such as $\nabla W_F \cdot \nabla \psi_{F,i}$ will be resulted. Thus, in practice, the variational principle has to be abandoned. Instead, a conventional Schrödinger equation for the fragment wave function $\Psi_{F,i}$ is used:

$$\left\{ -\frac{1}{2} \nabla^2 + V(r) \right\} \psi_{F,i}(r) = \epsilon_{F,i} \psi_{F,i}(r).$$

(4)

Here $V(r)$ is the full potential from the global potential energy $U[\rho]$ defined as $V(r) = \partial U[\rho]/\partial \rho$. The fragment wave function $\Psi_{F,i}$ and Equation (4) are defined within the fragment domain $\Omega_i$ (e.g. on a real-space grid, or using a corresponding atomic orbital basis set), which prevents the fragment wave function from becoming delocalized.

Over the years, P. Vashishta, A. Nakano, and their collaborators have implemented this DaC method on massive parallel computers, using either real-space grids or plane waves to describe the fragment wave functions [50–52]. In an impressive test case, they have used this approach to calculate an 11-million-atom system on the IBM BlueGene/L supercomputer [51]. They have also used this method to carry out MD where the atoms can move in and out of the spatially defined fragments [52]. Fig. 5 shows one such simulation and the fragments used [46]. Recently, they have also combined this
approach with other electronic structure calculations methods, in particular the nonadiabatic electron dynamics, to simulate the electron and exciton dynamics in the system [52]. In another work [53], T. Duy and T. Ozaki developed a DaC method using atomic orbitals as the electron-wave-function basis set, and the Krylov subspace method to obtain the on-site density and other properties for each atom using a recursion method under a Green’s function formalism. By limiting the recursion process within a local domain surrounding each atom [53], the calculation for the on-site density for that atom becomes independent of the overall size of the full system. As a result, the total computational cost scales linearly to the size of the system. Fig. 6 shows one of their systems and the distribution of processors to calculate different groups of atoms [53]. Finally, in [54], Touma et al. have introduced the time-dependent Hartree–Fock algorithm in the DaC approach. All these methods scale linearly to the size of the total system and also to the number of computer processors owing to the minimum communication requirement of the DaC method. However, in most of these methods, a global Fermi energy $E_F$ is used. This effectively treats each fragment or domain as a metallic system, and can sometimes introduce errors when balancing the potentials between different domains is required.

Recently, the author’s group has developed a different approach to divide the global system into fragments and to patch the fragments into the global system [55–57]. Instead of using partition functions $W_F(r)$, this linear scaling 3D fragment (LS3DF) method uses positive and negative fragments, and different fragment sizes. In its simplest scheme, this method divides the space into a $m_1 \times m_2 \times m_3$ grid; and at each grid corner, it generates eight fragments with their sizes specified as (using the grid space as size 1): $2 \times 2 \times 2$, $2 \times 2 \times 1$, $1 \times 2 \times 2$, $2 \times 2 \times 1$, $1 \times 2 \times 1$, $1 \times 1 \times 2$, and $1 \times 1 \times 1$. Each fragment $F$ will have the sign $\alpha(F)$. It is $+1$ for $2 \times 2 \times 2$, $2 \times 1 \times 1$, $1 \times 2 \times 1$, and $1 \times 1 \times 2$; and $-1$ for $2 \times 2 \times 1$, $2 \times 1 \times 2$, $1 \times 2 \times 2$, and $1 \times 1 \times 1$. Then the total charge density is assembled from the fragment charge density as $\rho(r) = \sum_F \alpha(F) \rho_F(r)$. Here the fragment index $F$ includes the eight fragments specified at each grid point of the $m_1 \times m_2 \times m_3$ grid. These fragments are illustrated schematically in Fig. 7 [58]. It can be shown that the boundary effects will be cancelled out between different fragments (due to the $+1$ and $-1$ fragments), and there will remain one copy of the charge density at every position $r$ from the center of one fragment. The total energy of the LS3DF can be expressed as

$$E = \sum_F \left\{ -\frac{\alpha(F)}{2} \sum_i \int \frac{|\nabla^2 \psi_{F,i}|^2}{\psi_{F,i}} d^3r \right\} + U[\rho]. \quad (5)$$

Figure 5. A DaC method calculation of liquid Rd for a MD. The lines show the fragment domain and buffer region used. Reprinted with permission from [51]. Copyright 2008 American Physics Society.

Figure 6. The different domains in a DaC scheme for a carbon ribbon. Different colors represent different fragment domains. Reprinted with permission from [53]. Copyright 2014 Elsevier Inc.

Figure 7. The fragments in the LS3DF calculation. For each fragment (color), a buffer region also needs to be added. There are such fragments at each corner of the dashed line grid. Reprinted with permission from [58].
If the minimum (maximum for negative \( \alpha(F) \) fragments) of \( E \) with respect to \( \Psi_{F,i} \) is obtained via \( \frac{\partial E}{\partial \psi_{F,i}} = \varepsilon_{F,i} \psi_{F,i} \), one obtains the conventional Schrödinger equation as Equation (4). In this way, the variational principle is restored. It allows us to use the Hellmann–Feynman theorem to calculate the atomic forces for atomic relaxation and MD. One can also introduce artificial surface passivation in Equation (4) for the fragments [56], so each fragment can have an energy band gap. As a result, there is no need to use a global Fermi energy. In practice, it is found that about 100 to 200 atoms are needed for the largest fragment in order to have an accuracy of a few meV/atom compared with the direct calculations. For example, \( 2 \times 2 \times 2 \) fragment sizes of 8, 64, and 216 Si atoms give total energy errors of about 200, 6 and 1 meV/atom, respectively, compared to direct calculations for a bulk Si system, while their charge density errors are 1.1, 0.14, and 0.08% [56]. Theoretically, the errors decay exponentially with the fragment size. Fig. 8 shows a ZnO nanorod and some of its fragments in a typical LS3DF calculation [58]. Poisson’s equation is solved based on the global grid and global charge density. The SCF iteration converges with the same rate as for the direct DFT calculation. Fig. 9 shows a LS3DF SCF convergence for a random alloy ZnTe:O system [58]. The LS3DF method is similar to the fragment molecular orbital (FMO) method used in quantum chemistry [59], which is also a linear-scaling method. They both use positive and negative fragments. While the FMO deals mostly with organic chain systems (e.g., it uses monomers and dimers as the fragments), the LS3DF calculates 3D crystal structures and related nanosystems.

As we discussed before, all the DaC methods have supreme scalabilities both for the physical system size and the number of computer cores. Fig. 10 shows the schematics for how to divide the number of processors into groups, with each group calculating a group of fragments. The only communication between the processor groups is for the assembly of the fragment charge density and for passing down the global potential into each fragment. These are not data-intensive communications compared with the wave functions. As a result, a perfect scaling can be demonstrated as shown in Fig. 11.

The LS3DF has been used to study a variety of problems that all involve thousands of atoms, from random alloys [60] to nanostructures [61]. Fig. 12 shows one such recent calculation where a MoSe\(_2\) monolayer is placed on top of a MoS\(_2\) monolayer [62]. Due to the small lattice mismatch, this bilayer system shows a moiré pattern in its atomic alignment. However, it is unknown whether such a moiré pattern in the atomic alignment will have a significant effect on its electronic structure. The LS3DF calculation shows that the valence band maximum (VBM) state will be localized following the atomic
Figure 11. The weak scaling of the LS3DF run for alloy ZnTe:O systems. The test system size scales proportionally to the number of CPU cores. Jaguar was a Cray XT5 machine at OLCF, Intrepid was an IBM BlueGene/P machine at the Argonne Leadership Computing Facility, and Franklin was a Cray XT4 machine at the NERSC. Reprinted with permission from [58].

Figure 12. A MoSe$_2$ and MoS$_2$ double layer showing a moiré pattern in its atomic arrangement from the top view. The isosurface plot shows the localization of the VBM state. The charge density and potential of the system are calculated by the LS3DF method, followed by the FSM for the band edge states. Reprinted with permission from [62]. Copyright 2013, American Chemical Society.

moiré pattern [62]. This will have significance for its transport properties especially at low temperature, and can potentially be used to manipulate its electronic structure for device applications.

Before closing this section, we would like to point out that the DaC approach is not limited to total energy calculations. It can also be used to calculate electronic structures. Fig. 13 presents one such example [63]. To calculate the charge transfer between a CdSe/CdS core/shell nanorod and an attached ferrocene molecule, one needs to construct the Hamiltonian of such a system, which contains more than 4000 atoms. One could use LS3DF to solve such a problem. However, in [63], a simpler approach is used. In this approach, the whole system is divided into two parts: one is the core/shell nanorod and the other is a small system consisting of a small piece of the nanocrystal plus the attached molecule. The atomic positions of the molecule and its attachment to the surface can be obtained via ab initio atomic relaxation of the small system. The charge density of the core/shell nanorod is calculated with the charge patching method [64]. The charge density of the full system is then patched from the charge density of the core/shell nanorod and the charge density of the small system. The calculation of the full system, especially the coupling strength between the core/shell nanorod VBM and the ferrocene molecule VBM, shows that the charge transfer between these two states is a direct coherence transfer without the help of any intermediate states [63]. The calculated charge transfer rate agrees well with the experimental observations [63].

NEW PROGRAMMING MODEL: GPU

The DaC scheme can be used to solve the large-size scale problem, changing the $O(N^3)$ computational complexity scaling to $O(N)$ scaling. It can be used to solve the communication problem in exascale computation. The DaC scheme can also be used efficiently with the new accelerator architectures which will be deployed in exascale computers. One predominant accelerator is the GPU shown in Fig. 14. In the LS3DF method, the plane wave pseudopotential (PWP) is used to solve the fragment wave function [65]. Recently, it has been shown that the PWP calculations can be efficiently accelerated using GPU [66–70]. The maximum acceleration of the GPU calculation compared with the CPU calculation is about a factor of 20 [70].

The key for GPU calculations is to provide enough data and computational tasks to the GPU cores, and to avoid heavy communication between GPU nodes as well as between GPU memory and CPU memory. In a CPU parallelization of a PWP code, it is common to use G-space parallelization over many CPU processors, which is to distribute the plane wave basis functions of a wave function basis set among different CPU cores [71]. In other words, a fast Fourier transform (FFT) is parallelized. One of the original reasons for G-space parallelization is to avoid storing a single wave function on a single CPU core to reduce the memory requirement. However, as computer memory space has increased over the years, this is rarely a problem. In addition, for problems in LS3DF, the fragment size is never very large (e.g. it is always less than 200 atoms), and thus it is entirely possible to place one or several wave functions within one CPU core or one GPU card as well as all the real-space potentials and non-local potential projectors [65]. As a result, the parallel FFT is no longer necessary. The parallel FFT requires all-to-all communication, and it fragmentizes
Figure 13. A CdSe/CdS nanorod attached to a ferrocene molecule. The charge densities of (A) and (E) are used to construct the charge density of the attached system shown in (C). The dashed line in (C) shows how the charge densities of the two subsystems are patched together. Reprinted with permission from [63]. Copyright 2014, American Chemical Society.

Figure 14. A schematic showing the two different parallel partitions of data (electron wave functions) in a GPU DFT calculation. The left partitions the data among the plane wave G basis set, while the right partitions the data among the wave function index. P_i denotes the processors, G_i denotes the G basis set, and \Psi_i denotes the wave function index. One MPI all-to-all communication transposes the data from one type of partitioning to the other type.

Table 1. The total computation time and overall GPU speedup compared with CPU on a number of CPU/GPU computing units. The number of cores indicates the number of CPU or GPU units in a given run. PETot_CPU and PETot_GPU stand for the CPU and GPU computational times, respectively, for one molecular dynamic step of a 512 atom GaAs system (with 2048 electrons). The data is taken from [73].

| Number of cores | PETot_CPU (s) | PETot_GPU (s) | Speedup (with PETot_CPU) |
|-----------------|---------------|---------------|--------------------------|
| 16              | 921           | 44.3          | ×20.9                    |
| 32              | 496           | 25.22         | ×19.84                   |
| 64              | 284           | 15.42         | ×18.9                    |
| 128             | 178           | 9.4           | ×18.9                    |
| 256             | 125           | 7.71          | ×16.1                    |

CHALLENGES AND OPPORTUNITIES FOR FUTURE EXASCALE SIMULATIONS USING D4C SCHEMES

There are countless critical materials science problems that can benefit significantly from future exascale ab initio simulations. Here we take battery research as one example. Battery development will play a major role in our struggle against global warming, e.g. by converting fossil-fuel-burning vehicles to electric vehicles. Battery development depends critically on the discovery of new cathode and anode materials, e.g. for lithium intercalation. It is realized that high-capacity materials might inevitably accompany large mechanical distortion or volume expansion during the charging and discharging process. For example, silicon can be used to absorb lithium as battery anode to the maximum amount of 22 lithium atoms per 5 silicon atoms [74]. This is accompanied by a 400% volume expansion [74]. After the silicon is fully charged with lithium, the system becomes amorphous [75]. Fig. 15 shows the charging
Figure 15. A silicon nanowire and the charging of Li atoms into the wire. Left panel: initial Si wire with no Li; right: completely Li-charged wire in an amorphous state. Reprinted with permission from [75]. Copyright 2008 Nature Inc.

process of a silicon nanowire and its corresponding morphology change [75]. This charging and discharging process accompanies a dramatic change in the nanowire’s shape, which presents a significant challenge for the battery design to maintain its stability [75]. This phase change process is too dramatic to be described by any perturbative analytical treatments, such as the cluster-expansion method for alloys study [76]. It will be tremendously helpful if one can simulate the entire charging and discharging process directly by \textit{ab initio} simulations. Simulating such a process requires the calculation of systems containing millions of atoms for seconds at a time. Besides the ionic movement, one also needs to simulate the electronic current and the effects of applied voltages. It is quite possible that the size problem can be overcome using the DaC approach. However, to apply the DaC method to such a problem, one needs to develop a robust framework; for example, the fragment can change with time, and the atoms can exchange between fragments during the MD. In a way, the framework developed by Vashishta’s group already has such features [50–52]. As for the long time needs to be simulated, one possibility is to use the AMD method based on \textit{ab initio} calculations as discussed before. Currently, the AMD method is used based on classical force fields. One of the reasons is the large number of MD steps needed in such an algorithm. Although the total number of simulation steps is much less than those of traditional MD simulations, the AMD still needs hundreds of thousands of steps. Today’s \textit{ab initio} MD running on a CPU usually takes about one minute per MD step. Thus, 100 000 steps correspond to two months of simulation time, which is impractical for most simulations. As discussed above, the GPU calculation can reduce the wall-clock time to 10 seconds per MD step. If future developments (from both hardware and software) can further reduce this to one second per MD step, then 100 000 steps correspond to one day, which becomes practical for actual research.

As a result, one can tackle directly the Si/Li charging and discharging problem for a nanosize silicon electrode.

Besides software development, other methodology developments in fundamental physics approximations are also necessary. For example, in the above battery charging or discharging processes, the electron movement (electric current) and applied voltage bias also play critical roles. This is particularly important for materials where the electron conductivity is very small (e.g. LiFePO$_4$ as battery cathode [77]). This requires the development of corresponding algorithms to incorporate the adequate physics in the simulation. There could be different situations for electron transport that require different approximations. If the electron wave function is in a delocalized itinerant state, perhaps a small percentage of the global electronic states near the Fermi energy need to be calculated using algorithms such as the folded spectrum method (FSM) [78]. These delocalized and transporting electronic states can be added to the DaC treatment for the rest of the electronic system to provide the total charge density of the system. Other approaches to dealing with open shell systems with a DaC approach have also been tested [79]. These techniques are not limited to dynamics problems such as in the battery charging and discharging process; they can also be used to study problems where a few electronic states are intrinsically delocalized, and hence cannot be treated directly with the DaC approach (e.g. a shallow impurity state [80]). In another situation, the transporting electrons are highly localized, e.g. in a polaronic form, which is the case for many oxides (e.g. in LiFePO$_4$) [81]. One should be able to use the DaC approach to treat such localized state transport, since a localized polaron can be described within one fragment. This has been tested in [52], where localized electron and exciton state transports are calculated with DaC and eventually connected with KMC simulations. The voltage bias is another challenge. This is an important issue in studying the solid electrolyte interphase (SEI) problem in batteries [82] where the chemical potential and electron Fermi energy change significantly across this interface. Not much is known about the atomic mechanism for this SEI; thus, a direct \textit{ab initio} simulation will be extremely useful. The voltage bias means the whole system is not in equilibrium in its electronic structure. A local Fermi energy can be introduced. For example, different fragments
can have a different Fermi energy. However, the profile of the Fermi energy needs to be coupled with the electric current. In a way, the Fermi energy profile determines the local electric current $J(r)$, while the $\nabla \bullet J = \partial \rho / \partial t$ serves as a condition to determine the Fermi energy profile.

Finally, regarding the accuracy of DFT calculations, the availability of exascale computers will allow us to test new methods. In solid state simulations, one such approach is the RPA, high in the Jacob’s ladder shown in Fig. 4. RPA can be used to calculate the total energy as well as the single particle eigen states [28]. In its calculation, it includes all the single particle excite states; thus, it can incorporate the dynamic screening in its formalism, which is an important part of the correlation effect. Such high-level QM calculations usually have high-power scaling to the system size. For example, the MP2 scales as $O(N^4)$, while the quantum chemistry gold standard CCSD(T) method scales as $O(N^7)$ [83]. However, with the DaC scheme, one can apply the high-level calculations to the fragments. Since the fragment size is fixed while the full system size increases, this can solve the size scaling problem. Because most of the correlation effects are short range [84], one expects to get accurate results using this approach.

While the possibility for exascale material simulations is boundless, it is a tremendous challenge to develop efficient software for such simulations. With the heterogeneous and hierarchical architecture of exascale supercomputers, perhaps the best ways to run applications on these advanced systems will also be heterogeneous and hierarchical. Nowadays, an \textit{ab initio} simulation will deploy one standard package, e.g. a plane wave DFT code or an atomic orbital code, which uses a homogeneous algorithm and approximation. However, for mesoscale systems, the problems themselves might have multiple scales and multiple physics, e.g. involving atomics and electronic transport, and having different regions with different materials that require different treatments and approximations. As a result, the best strategy is to have a flexible framework to link different codes and methods together to simulate them simultaneously for different regions and physics, but with interactions between these regions to allow a self-consistent calculation over the whole system. In a way, the current DaC methods are the first step in that direction which links the fragment calculations at different spatial locations to reach an overall SCF solution. However, current DaC methods, like the LS3DF, deploy fixed division schemes which are not flexible enough. Further developments in their flexibilities and stabilities will be needed.

To design such a flexible framework or an overall package with high efficiency is not a trivial task. Perhaps a co-design strategy is necessary [85] in which the design of a software framework is in coherence with the design of exascale computer architecture. The best computer science practices need to be combined with the best physics approximations. Perhaps the raw computer power to simulate many of the challenging materials science problems already exists or will be provided soon by exascale supercomputers. The challenge, however, is to provide the corresponding software to efficiently use such computers. Future investment in this area is of paramount importance.

CONCLUSIONS

This is an exciting time for large-scale material simulations. The power of simulation increases exponential with time. The new computer power allows us to adopt new computational paradigms—moving up a rung in Jacob’s ladder—that will provide more accurate and more reliable results. Meanwhile, the DaC can provide an approach to calculate mesoscale systems where many of the real material properties reside. With faster simulations, especially those provided by the accelerators, algorithms that usually require tens of thousands of steps and can only be used with empirical force fields nowadays can be used with \textit{ab initio} calculations in the future. It is tantalizing to imagine that someday we will be able to predict a material’s properties as accurately and as reliably as today’s civil engineer. Nowadays, one can design a bridge, a building, a car, or even a large Boeing 787 airplane on a computer. With the further development of \textit{ab initio} material simulations and exascale computational power, we will one day be able to design any material on a computer for various applications. That will uplift us to a new level in human civilization. To reach that goal, we will not only need to construct exascale computers, but we will also need to invest in the development of algorithms and software that can be used efficiently on such massively parallel computers.

FUNDING

The author is grateful for the support by the Director, Office of Science, Office of Basic Energy Science, Materials Science and Engineering Division, of the US Department of Energy under Contract No. DE-AC02–05CH11231 through the Theory of Materials program at Lawrence Berkeley National Laboratory.
REFERENCES

1. Advanced Micro Devices Inc. High-Performance Computing Milestones. Sunnyvale, CA: Advanced Micro Devices Inc., 2010.
2. Sutter, H. The free lunch is over: a fundamental turn toward concurrency in software. Dr. Dobbs’ J. 2005; 30, March 2005; or see [http://www.gotw.ca/publications/concurrency-ddj.htm](http://www.gotw.ca/publications/concurrency-ddj.htm) (20 August 2014, date last accessed).
3. Summary Report of the Advanced Scientific Computing Advisory Committee (ASCAC) subcommittee. The Opportunities and Challenges of Exascale Computing. Washington DC: Office of Science, Department of Energy, 2010.
4. Top 500 Supercomputer Sites. [http://www.top500.org](http://www.top500.org) (20 August 2014, date last accessed).
5. Introduction to the Titan machine. [https://www.olcf.ornl.gov/titan](https://www.olcf.ornl.gov/titan) (20 August 2014, date last accessed).
6. Single Instruction stream, Multiple Data streams (SIMD) architectures. [http://arstechnica.com/features/2000/03/simd10.1093/rst/nwu060.html](http://arstechnica.com/features/2000/03/simd10.1093/rst/nwu060.html) (20 August 2014, date last accessed).
7. Cray Inc. Cray XK7 Brochure. [http://www.cray.com/Products/Computing/XK7.aspx](http://www.cray.com/Products/Computing/XK7.aspx) (20 August 2014, date last accessed).
8. Sur, S, Koop, MJ and Panda, DK. High-performance and scalable MPI over InfiniBand with reduced memory usage: an in-depth performance analysis, In: SC’06 Proceedings of the 2006 ACM/IEEE Conference on Supercomputing. New York: ACM. 2006, Article No. 105.
9. MPI Documents [http://www.mpi-forum.org/docs/](http://www.mpi-forum.org/docs/) (20 August 2014, date last accessed).
10. Nvidia CUDA [http://www.nvidia.com/object/cuda_home_new.htm](http://www.nvidia.com/object/cuda_home_new.htm) (20 August 2014, date last accessed).
11. Wang, LW. A survey of codes and algorithms used in NERSC material science allocations. LBNL Report 61051
12. Senn, HM and Thiel, W. QM/MM methods for biomolecular systems. Angew. Chem. Int. Ed. Engl. 2009; 48: 1198–229.
13. Hohenberg, P and Kohn, W. Inhomogeneous electron gas. Phys. Rev. 1964; 136: B864–71.
14. Kohn, W and Sham, L.J. Self-consistent equations including exchange and correlation effects. Phys. Rev. 1965; 140: A1133–8.
15. Becke, AD. Perspective: fifty years of density-functional theory in chemical physics. J. Chem. Phys. 2014; 140: 18A301.
16. Foulkes, WMC, Mitas, L and Needs, RJ et al. Quantum Monte Carlo simulations of solid. Rev. Mod. Phys. 2001; 73: 33–83.
17. Gavogolu, K and Simoes, A. Neither Physics nor Chemistry: A History of Quantum Chemistry. Cambridge: MIT Press, 2011.
18. Funche, F and Perdew, JP. The performance of semi-local and hybrid density functionals in 3d transition-metal chemistry. J. Chem. Phys. 2006; 124: 044103.
19. Perdew, JP and Zunger, A. Self-interaction correction to density-functional approximations for many-electron systems. Phys. Rev. B 1981; 23: 5048–9.
20. Perdew, JP and Burke, K. Generalized gradient approximation made simple. Phys. Rev. Lett. 1996; 77: 3865–8.
21. Tao, JM, Perdew, JP and Staroverov, VN et al. Climbing the density functional ladder: nonempirical meta-generalized gradient approximation designed for molecules and solids. Phys. Rev. Lett. 2003; 91: 146401.
22. Kim, K and Jordan, KD. Comparison of density functional and MP2 calculations on the water monomer and dimer. J. Phys. Chem. 1994; 98: 10089–94.
23. Heyd, J, Scuseria, GE and Ernzerhof, M. Hybrid functionals based on a screened Coulomb potential. J. Chem. Phys. 2003; 118: 8207.
24. Zhao, Y and Troullier, DG. The M06 suite of density functionals for main group thermochemistry, thermochemical kinetics, noncovalent interactions, excited states, and transition elements: two new functionals and systematic testing of four M06-class functionals and 12 other functionals. Theor. Chem. Acc. 2008; 120: 215–41.
25. Hari, J and Kresse, G. Accurate bulk properties for approximate many-body techniques. Phys. Rev. Lett. 2009; 103: 056401.
26. Furche, F. Developing the random phase approximation into a practical post-Kohn–Sham correlation model. J. Chem. Phys. 2008; 129: 114105.
27. Fuchs, M and Gonze, X. Accurate density functionals: approaches using the adiabatic-connection fluctuation-dissipation theorem. Phys. Rev. B 2002; 65: 235109.
28. Wang, LW. Relationship between the random-phase approximation ground-state total energy and GW quasiparticle energy. Phys. Rev. B 2010; 82: 115111.
29. Hedin, L and Lundqvist, S. Effects of electron-electron and electron-phonon interactions on the one-electron states of solids. Solid State Phys. 1969; 23: 1–81.
30. Heybertsen, MS and Louie, SG. Electron correlation in semiconductors and insulators: band gaps and quasiparticle energies. Phys. Rev. B 1986; 34: 5390–413.
31. Georges, A, Kotliar, G and Krauth, W et al. Dynamical mean-field theory of strongly correlated fermion systems and the limit of infinite dimensions. Rev. Mod. Phys. 1998; 68: 13–125.
32. Yan, J and Norskov, JK. Calculated formation and reaction energies of 3d transition metal oxides using a hierarchy of exchange-correlation functionals. Phys. Rev. B 2013; 88: 245204.
33. Anisimov, VI, Aryasetiawan, F and Liechtenstein, AI. First-principles calculations of the electronic structure and spectra of strongly correlated systems: the LDA+U method. J. Phys. Condens. Matter 1997; 9: 767–808.
34. Dudarev, SL, Liechtenstein, AI and Castell, MR et al. Surface states on NiO (100) and the origin of the contrast reversal in atomically resolved scanning tunneling microscope images. Phys. Rev. B 1997; 56: 4900–8.
35. Mosey, NJ and Carter, EA. Ab initio evaluation of Coulomb and exchange parameters for DFT+U calculations. Phys. Rev. B 2007; 76: 155123.
36. Grimme, S. Semiempirical GGA-type density functional constructed with a long range dispersion correction. J. Comput. Chem. 2006; 27: 1787–99.
37. Dion, M, Rydberg, H and Schroder, E et al. Van der Waals density functional for general geometries. Phys. Rev. Lett. 2004; 92: 246401.
38. Lee, K, Murray, ED and Kong, L et al. Higher-accuracy van der Waals density functional. Phys. Rev. B 2010; 82: 081101(R).
39. Voter, AF. A method for accelerating the molecular dynamics simulation of infrequent events. J. Chem. Phys. 1997; 106: 4665–77.
40. Voter, AF, Montalenti, F and Germann, TC. Extending the time scale in atomistic simulation of materials. Annu. Rev. Mater. Res. 2002; 32: 321–46.
41. Voter, AF. Parallel replica method for dynamics of infrequent events. Phys. Rev. B 1998; 57: R13985–8.
42. Chattarjee, A and Vlachos, DG. An overview of spatial microscopic and accelerated kinetic Monte Carlo methods. J. Comput.-Aided Mater. Des. 2007; 14: 253–308.
43. Goedecker, S. Linear scaling electronic structure methods. Rev. Mod. Phys. 1999; 71: 1085–123.
44. Bowler, DR and Miyazaki, T. O(N) methods in electronic structure calculations. Rep. Prog. Phys. 2012; 75: 036503.
45. Prodan, E and Kohn, W. Nearsightedness of electronic matter. Proc. Natl. Acad. Sci. 2005; 102: 11635–8.
46. Huang, C, Pavone, M and Carter, EA. Quantum mechanical embedding theory based on a unique embedding potential. *J. Chem. Phys.* 2011; **134**: 154110.

47. Wesolowski, TA and Warshel, A. Frozen density functional approach for ab initio calculations of solvated molecules. *J. Phys. Chem.* 1993; **97**: 8050–3.

48. Schwertfeger, P. The pseudopotential approximation in electronic structure theory. *Chem. Phys. Chem.* 2011; **12**: 3143–55.

49. Yang, WT. Direct calculation of electron density in density functional theory. *Phys. Rev. Lett.* 1991; **66**: 1438–41.

50. Shimoo, F, Kalia, RK and Nakano, A et al. Embedded divide-and-conquer algorithm on hierarchical real-space grids: parallel molecular dynamics simulation based on linear-scaling density functional theory. *Comput. Phys. Commun.* 2005; **167**: 151–64.

51. Shimoo, F, Kalia, RK and Nakano, A et al. Divide-and-conquer density functional theory on hierarchical real-space grids: parallel implementation and applications. *Phys. Rev. B* 2008; **77**: 085103.

52. Shimoo, F, Hattori, S and Kalia, RK et al. A divide-conquer-recombine algorithmic paradigm for large spatiotemporal quantum molecular dynamics simulations. *J. Chem. Phys.* 2014; **140**: 18AS29.

53. Duy, TVT and Ozaki, T. A three-dimensional domain decomposition method for large scale DFT electronic structure calculations. *Comput. Phys. Commun.* 2014; **185**: 777–89.

54. Touma, T, Kobayashi, M and Nakai, H. Time-dependent Hartree–Fock frequency-dependent polarizability calculation applied to divide-and-conquer electronic structure method. *Chem. Phys. Lett.* 2010; **485**: 247–52.

55. Wang, LW, Zhao, Z and Meza, J. Linear-scaling three-dimensional fragment method for large scale electronic structure calculations. *Phys. Rev. B* 2008; **77**: 165113.

56. Zhao, Z, Meza, J and Wang, LW. A divide-and-conquer linear scaling three dimensional fragment method for large scale electronic structure calculations. *J. Phys. Condens. Matter* 2008; **20**: 294203.

57. Wang, LW, Lee, B and Shan, H et al. Linear scaling 3D fragment method for large scale electronic structure calculations. In: *SC’08 Proceedings of the 2008 ACM/IEEE Conference on Supercomputing*. Piscataway: IEEE Press, 2008, Article No. 65.

58. Wang, LW and Hules, J. Electronic structure calculations: divide and conquer for tractable computations. *SciDAC Rev.* 2009; **14**: 22–31.

59. Kitaura, K, Ikeo, E and Asada, T et al. Fragment molecular orbital method: an approximate computational method for large molecules. *Chem. Phys. Lett.* 1999; **313**: 701–6.

60. Lee, B and Wang, LW. Electronic structure of ZnTe:O and its usability for intermediate band solar cell. *Appl. Phys. Lett.* 2010; **96**: 071903.

61. Dag, S, Wang, SZ and Wang, LW. Large surface dipole moment in ZnO nanorods. *Nano Lett.* 2011; **11**: 2348–52.

62. Kang, J, Li, J and Li, SS et al. Electronic structural Moire pattern effects on MoS2/MoS2 2D heterostructures. *Nano Lett.* 2013; **13**: 5485–90.

63. Tarafdar, K, Surendranath, Y and Olshansky, JH et al. Hole transfer dynamics from a CdSe/CdS quantum rod to a tethered ferrocene derivative. *J. Am. Chem. Soc.* 2014; **136**: 5121–31.

64. Li, J and Wang, LW. Band-structure-corrected local density approximation study of semiconductor quantum dot dows and wires. *Phys. Rev. B* 2005; **72**: 125325.

65. Payne, MC, Teter, MP and Allan, OC et al. Iterative minimization techniques for ab initio total energy calculations: molecular dynamics and conjugate gradients. *Rev. Mod. Phys.* 1992; **64**: 1045–96.

66. Maintz, S, Eck, B and Drongowski, R. Speeding up plane-wave electronic-structure calculations using graphics-processing units. *Comput. Phys. Commun.* 2011; **182**: 1421–7.

67. Spiga, F. Accelerating the plane-wave self-consistent field (PWscf) package using NVIDIA GP-GPU. *Presented at 7th Advanced School of Parallel Computing*, Casalecchio di Reno, Italy, 10–14 October, 2011

68. Tomono, H, Aoki, M and Itaka, T et al. GPU based acceleration of first principles calculation. *J. Phys. Conf. Ser* 2010; **215**: 012121.

69. Wang, L, Wu, Y and Jia, W et al. Large scale plane wave pseudopotential density functional theory calculations on GPU clusters. In: *SC’11 Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis*. New York: ACM, 2011. Article No. 71.

70. Jia, W, Fu, J and Cao, Z et al. Fast plane wave density functional theory molecular dynamics calculations on multi-GPU machines. *J. Comput. Phys.* 2013; **251**: 102–15.

71. Canning, A, Wang, LW and Williamson, A et al. Parallel empirical pseudopotential electronic structure calculations for million atom Systems. *J. Comput. Phys.* 2000; **160**: 29–41.

72. Canning, A, Shalf, J and Wang, LW. A comparison of different communication structures for scalable three dimensional FFTs in first principle codes. In: Chapman, B, Desperez, F and Joubert, GR et al. (eds). *Parallel Computing: From Multicores and GPUs to Petascale*. Amsterdam: IOS Press, 2010, 107–16.

73. Jia, W, Cao, Z and Wang, L et al. The analysis of a plane wave pseudopotential density functional theory code on a GPU machine. *Comput. Phys. Commun.* 2013; **184**: 9–18.

74. Fister, TT, Esbenshade, J and Chen, X et al. Lithium intercalation behavior in multilayer silicon electrodes. *Adv. Energy Mater.* 2014; **4**, doi: 10.1002/aenm.201301494.

75. Chan, CK, Peng, H and Liu, G et al. High-performance lithium battery anodes using silicon nanowires. *Nat. Nanotechnol.* 2008; **3**: 31–5.

76. Laks, DB, Ferreira, LG and Froyen, S et al. Efficient cluster expansion for sub-stituutional systems. *Phys. Rev. B* 1992; **46**: 12587–605.

77. Padhi, AK, Nanjundaswamy, KS and Goodenough, JB. Phospho-olivines as positive-electrode materials for rechargeable lithium batteries. *J. Electrochem. Soc.* 1997; **144**: 1188–94.

78. Wang, LW and Zunger, A. Solving Schrodinger’s equation around a desired energy: application to silicon quantum dots. *J. Chem. Phys.* 1994; **100**: 2394–7.

79. Kobayashi, M and Nakai, H. How does it become possible to treat delocalized and/or open shell systems in fragmentation-based linear-scaling electronic structure calculations? The case of the divide-and-conquer method. *Phys. Chem. Chem. Phys.* 2012; **14**: 7629–39.

80. Zhang, G, Canning, A and Gronbech-Jensen, N et al. Shallow impurity level calculations in semiconductors using *ab initio* method. *Phys. Rev. Lett.* 2013; **110**: 166404.

81. Maxisch, T, Zhou, F and Ceder, G. *Ab initio* study of the migration of small polaron in olivine LiFePO4 and their association with lithium ions and vacancies. *Phys. Rev. B* 2006; **73**: 104301.

82. Balbuena, PB and Wang, Y. Lithium-Ion Batteries: Solid-Electrolyte Interphase. London: Imperial College Press, 2004.

83. Jensen, F. *Introduction to Computational Chemistry*. West Sussex: John Wiley & Sons, Ltd, 2007.

84. Yang, J, Chan, GK and Manby, FR et al. The orbital-specific-virtual local coupled cluster singles and doubles method. *J. Chem. Phys.* 2012; **136**: 144105.

85. Barrett, RF, Borkar, S and Dosanjh, SS. On the role of co-design in high performance computing. In: D’Holland, EH, Dongarra, JJ and Foster, I et al. (eds) *Transition of HPC Towards Exascale Computing*. Amsterdam: IOS Press, 2013, 141–55.