Journal of Physics: Condensed Matter

<i>Ab initio </i> electronic structure calculations using a real-space Chebyshev-filtered subspace Iteration method

To cite this article before publication: Qiang Xu *et al* 2019 *J. Phys.: Condens. Matter* in press https://doi.org/10.1088/1361-648X/ab2a63

Ab initio Electronic Structure Calculations Using a Real-space Chebyshev-filtered Subspace Iteration Method

Qiang Xu[†,1], Sheng Wang[†,1], Lantian Xue[†], Xuecheng Shao[†], Pengyue Gao[†], Jian Lv[†], Yanchao Wang[*,†], and Yanming Ma[*,†,#]

[†]*State Key Lab of Superhard Materials & Innovation Center of Computational Physics Methods and Software, College of Physics, Jilin University, Changchun 130012, China*

[#]*International Center of Future Science, Jilin University, Changchun 130012, China*

Corresponding Authors:

[*]*E-mail: wyc@calypso.cn*

[*]*E-mail: mym@calypso.cn*

[1]These authors contributed equally.

Ab initio electronic structure calculations within Kohn–Sham density functional theory requires a solution of the Kohn–Sham equation. However, the traditional self-consistent field (SCF) approach of solving the equation using iterative diagonalization exhibits an inherent cubic scaling behavior and becomes prohibitive for large systems. The Chebyshev-filtered subspace iteration (CheFSI) method holds considerable promise for large-system calculations by substantially accelerating the SCF procedure. Here, we employed a combination of the real space finite-difference formulation and CheFSI to solve the Kohn–Sham equation and implemented this approach in *Ab initio* Real-space Electronic Structure (ARES) software in a multi-processor, parallel environment. An improved scheme was proposed to generate the initial subspace of Chebyshev filtering in ARES efficiently, making it suitable for large-scale simulations. The accuracy, stability, and efficiency of the ARES software were illustrated by simulations of large-scale crystalline systems containing thousands of atoms.

1. Introduction

Kohn–Sham density functional theory (KS-DFT) [1, 2] is the most widely used quantum mechanical method for obtaining the electronic structures of condensed matter and plays a crucial role in understanding the physical and chemical properties of complex materials at the microscopic level. However, the solution of the Kohn–Sham equations requires solving a nonlinear eigenvalue problem. The traditional self-consistent field (SCF) approach using iterative diagonalization is the most popular method for solving the KS-DFT equation and has been implemented in several packages, including VASP [3,4], ABINIT [5], CASTEP [6], and Quantum Espresso [7], which have been used to simulate systems comprising a few hundred atoms. In practice, a large-scale simulation of complicated structures (e.g., heterointerfaces, dislocations) modeled by a big unit cell containing thousands of atoms are required sometimes. The traditional SCF method typically exhibits an inherent cubic scaling behavior, making large-scale simulations prohibitive [8]. Furthermore, the KS-DFT equation must be solved a large number of times for practical applications of density functional theory (DFT) simulations for dynamic processes, such as structure relaxation and molecular dynamics simulations. Therefore, there is an urgent need to develop an efficient Kohn−Sham equation solution strategy to reduce the computational cost in the framework of the DFT method for large-scale simulations.

Research has focused on developing efficient DFT methods whose computational costs scale linearly as a function of the number of atoms. In these methods, truncated localized basis sets, such as atomic and pseudo-atomic orbitals, wavelets, or B-spline functions, coupled with the divide-and-conquer scheme [9–13] have been introduced to extend the applicability and increase the accessible length scales of DFT to large-scale systems [14]. There are numerous software packages (e.g., ONETEP [15, 16], CONQUEST [17,18], QUICKSTEP [19], and BigDFT [20, 21]) in the framework of linear scaling methods that have increased the maximum system size considerably to many thousands of atoms [14, 22]. Currently, the reduced-scaling methods combined with the availability of high-performance computing resources make million-atom DFT computations affordable, while maintaining the same accuracy as the traditional cubic scaling approaches [14, 22]. However, these methodologies using the truncated localized basis sets depend on the possibility of localizing Kohn–Sham orbitals or short-range density matrices; thus, their applications to metallic systems have been limited [14, 22].

The pole expansion and selected inversion (PEXSI) [23, 24] technique provides an alternative strategy for efficiently solving the Kohn–Sham problem without using a diagonalization procedure. The computational cost of the PEXSI technique scales at most as $O(N^2)$ for general 3D bulk systems. The technique has been implemented in several DFT-based packages, including SIESTA-PEXSI [25] and ABACUS [26], and has been applied to simulations of systems comprising thousands of atoms. However, it is usually more difficult to achieve a good load balance and memory distribution in inverse algorithms [27].

Recently, the Chebyshev-filtered subspace iteration (CheFSI) [28, 29] method has been proposed to solve the KS-DFT equation. In this approach, the explicit eigenvectors of the intermediate linearized Kohn–Sham eigenvalue problems are replaced by approximate basis vectors of a progressively refined subspace, leading to the substantial reduction of the diagonalization cost and allowing large-scale simulations with currently available computing resources. The method has been implemented in several DFT-based packages, including PARSEC [29], RESCU [27], and SPARC [30,31], and has been applied to simulations of large-scale systems containing thousands of atoms on a modest computer cluster.

In this work, a combination of the real-space finite-difference formulation [4, 5] and CheFSI method introduced by Zhou et al. [28] was used to solve the KS-DFT equation for periodic systems and was implemented in our DFT-based package, *Ab initio* Real-space Electronic Structure software (ARES), with sequential and parallel architectures. The reliability of ARES has been benchmarked by numerical simulations for a wide variety of condensed matter, encompassing metals, semiconductors, and insulators. The simulated results demonstrate that ARES can substantially reduce the computational cost of DFT simulations in a periodic system without sacrificing accuracy.

The remainder of this manuscript is organized as follows. Section 2 briefly introduces KS-DFT and details of the implementation of the ARES package. In Section 3, we present the results of testing on crystalline systems to demonstrate the computational stability, accuracy, and efficiency of ARES. Finally, conclusions are presented in Section 4.

2. Theory and Implementation

2.1 Kohn–Sham Theory

We briefly introduce the equations of KS-DFT for electronic structure calculations. Note that atomic units $\left(e=\hbar=m_e=1\right)$ are used throughout this paper. The central task in KS-DFT calculation is solving the Kohn–Sham equation [2],

$$\hat{H}^{KS}\psi_i(\vec{r})=\varepsilon_i\psi_i(\vec{r}), \tag{1}$$

where ψ_i and ε_i are the i-th Kohn–Sham eigenfunctions and eigenvalues, respectively. The Kohn–Sham Hamiltonian, H^{KS}, is given by

$$\hat{H}^{KS}=-\frac{1}{2}\nabla^2+\hat{V}_H[\rho]+\hat{V}_{XC}[\rho]+\hat{V}_{ion}, \tag{2}$$

where $\hat{V}_H, \hat{V}_{XC}$, and $\hat{V}_{ion}$ are the Hartree potential, exchange-correlation potential, and ionic potential, respectively. The Kohn–Sham Hamiltonian depends on the electronic density, $\rho(r)$, which can be determined from the occupied Kohn–Sham orbitals (spin degeneracy is assumed here) as

$$\rho(\vec{r})=2\sum_{i=1}^{Occ.}\left|\psi_i(\vec{r})\right|^2. \tag{3}$$

Considering crystalline systems under Bloch periodic boundary conditions, the Kohn–Sham eigenfunctions in Eq. (1) then become

$$\psi_{n,\vec{k}}\left(\vec{r}\right)=\frac{1}{\sqrt{N_k}}e^{i\vec{k}\cdot\vec{r}}u_{n,\vec{k}}\left(\vec{r}\right), \tag{4}$$

where N_k is the number of k-points in the Brillouin zone, which is typically equivalent to the number of unit cells in the Born von Karmen (BvK) supercell under periodic boundary conditions. Here, the normalization of wavefunctions is over the BvK cell.

$u_{n,\vec{k}}(\vec{r})$ is a periodic function on the lattice and the normalization is over the unit cell, $\frac{1}{N_k}\int_V \left|u_{n,\vec{k}}(\vec{r})\right|^2 d^3r = \int_\Omega \left|u_{n,\vec{k}}(\vec{r})\right|^2 d^3r = 1$. $V = N_k\Omega$ is the BvK cell volume, where Ω is the volume of the unit cell. The electron density can be computed as

$$\rho(\vec{r}) = \frac{1}{N_k}\sum_{\vec{k}}\sum_{n=1}^{N_s} f_{n,\vec{k}}\left|u_{n,\vec{k}}(\vec{r})\right|^2, \tag{5}$$

where $f_{n,\vec{k}}$ is the Fermi–Dirac occupation function. Eq. (1) can be written as

$$\hat{H}_{\vec{k}}[\rho]u_{n,\vec{k}} = \varepsilon_{n,\vec{k}}u_{n,\vec{k}} \quad n=1,\ 2,\ \cdots,\ N_s;\ \vec{k}=\vec{k}_1,\ \vec{k}_2,\ \cdots,\ \vec{k}_{N_k}, \tag{6}$$

where $\varepsilon_{n,\vec{k}}$ are eigenvalues of the *k*-point-dependent Kohn–Sham Hamiltonian

$$\hat{H}_{\vec{k}}[\rho] := -\frac{1}{2}\left(\nabla^2 + 2i\vec{k}\cdot\nabla - \left\|\vec{k}\right\|^2\right) + \hat{V}_H[\rho] + \hat{V}_{XC}[\rho] + \hat{V}_{loc} + \hat{V}_{nl}^{\vec{k}}. \tag{7}$$

It is necessary to determine the ground electron density from the self-consistent solution of the Kohn–Sham equations in Eqs. (5–7). First, Eq. (7) uses the initially estimated charge density to yield a set of *k*-point-dependent Hamiltonians. Second, the solution of Eq. (6) with the current Hamiltonians (Eq. (7)) is used to obtain the eigenvalues and corresponding set of eigenfunctions (wave functions). Then, the new charge density can be evaluated using Eq. (5). This process is repeated until the variation of the electron density is smaller than the given tolerance. The most expensive step in the process is obtaining the eigenpairs (eigenvalues and eigenstates) by solving Eq. (6).

2.2 Real-Space Representation of Kohn–Sham Equation

Real-space representations can effectively simplify the application of localization constraints [34]. In the representations, the wave functions and potentials are directly evaluated on real-space grid points. Accordingly, the Hamiltonian in Eq. (7) is discretized on real-space uniform grids as an $N_b \times N_b$ matrix, where each eigenfunction is a $N_b \times 1$ size vector with N_b denoting the total number of real-space grids. Although the Hamiltonian matrix within a real-space representation of the Kohn–Sham equation is larger than that in other basis-dependent approaches, the Hamiltonian matrix is discretized to be a spare matrix whose nonzero elements are confined within a diagonal band. The extent of the nonzero elements in the off-diagonal positions only depends on the order of the finite difference expansion of the kinetic energy operator.

The Laplacian and gradient operator for the kinetic term in the Hamiltonians in Eq. (7) can be represented by finite difference expansion, discretized into a sparse matrix [35]. The general form of the Laplacian operator on a non-orthorhombic grid is given by [33]

$$\nabla^2 = \sum_{i=1}^{6} f_i \frac{\partial^2}{\partial x_i^2}. \tag{8}$$

The Laplacian form is represented by a combination of derivatives along the following six directions, $\{\hat{x}_i\}$: three along the original lattice vectors and three additional derivatives in the nearest-neighbor directions. For an orthorhombic grid, the Laplacian form is reduced to three vectors along the original lattice vectors. The coefficients, f_i, in Eq. (8) refer to Ref. [33]. The gradient operator on a non-orthorhombic grid is given by

$$\nabla = \hat{e}_x \sum_{i=1}^{3} B_{1i}^T \frac{\partial}{\partial x_i} + \hat{e}_y \sum_{i=1}^{3} B_{2i}^T \frac{\partial}{\partial x_i} + \hat{e}_z \sum_{i=1}^{3} B_{3i}^T \frac{\partial}{\partial x_i}, \tag{9}$$

where matrix B is the inverse of normalized lattice matrix $\tilde{A} = [\hat{a}_1,\ \hat{a}_2,\ \hat{a}_3]$, $\hat{a}_i = \frac{a_i}{\|a_i\|}$.

The *m*-th ($m = 1,\ 2$) derivative along the $\hat{x}_i$ direction can be approximated by the expansion of the high-order finite difference as

$$\frac{\partial^m}{\partial x_i^m} u(x_i) \approx \sum_{n=-N_{ord}}^{N_{ord}} \frac{C_n^m}{h_i^m} u(x_i + n h_i \hat{x}_i), \tag{10}$$

where N_{ord} and h_i are the order of the finite-difference expansion and spacing of the grid along the $\hat{x}_i$ direction, respectively. The coefficients of C_n^m are available in Refs. [36] and [37].

The potential term in Eq. (7) consists of the Hartree potential, $\hat{V}_H[\rho]$, the exchange-correlation potential, $\hat{V}_{XC}[\rho]$, the local ionic pseudopotential, $\hat{V}_{loc}$, and the non-local ionic pseudopotential, $\hat{V}_{nl}^{\vec{k}}$. The first three potential terms only contribute to the leading diagonal of the Hamiltonian matrix in real-space representation.

The Hartree potential is given by solving the Poisson equation as[38]

$$\nabla^2 V_H[\rho](\vec{r}) = -4\pi[\rho(\vec{r}) - \rho_0]\cdot \tag{11}$$

Where ρ_0 is average electron density of the system. Both real-space and fast Fourier transform (FFT)-based calculations of the Hartree potential are adopted in ARES. However, the computational cost of the FFT-based method is lower than that of the real-space method for periodic systems [39]. Therefore, the FFT-based method was also employed in this work.

The exchange-correlation potential is estimated by

$$V_{XC}[\rho](\vec{r}) = \frac{\delta E_{XC}[\rho]}{\delta\rho(\vec{r})}, \tag{12}$$

where $E_{XC}[\rho]$ is the exchange-correlation energy functional. Currently, two typical exchange-correlation functionals of local density approximation (LDA) [40] and Perdew–Burke–Ernzerhof generalized gradient approximation (GGA) [41] have been implemented. The other exchange and correlation functionals, such as LDA, GGA, and meta-GGA functionals available from LibXC [42], can also be easily interfaced with ARES as required.

The ionic potential is used to described the ion–electron interaction by the norm-conserving pseudopotential [43], which is given by

$$\hat{V}_{ion} = \hat{V}_{loc} + \hat{V}_{nl}, \tag{13}$$

where V_{loc} and V_{nl} are the local and nonlocal parts of the pseudopotential, respectively. The Kleinman–Bylander form [44] used for the nonlocal part is given by

$$\hat{V}_{nl} = \sum_{a=1}^{N_a} \sum_{lm} \frac{1}{\overline{V}_{lm}^{a}} \left| \chi_{lm}^{a} \right\rangle \left\langle \chi_{lm}^{a} \right|, \tag{14}$$

where $\left| \chi_{lm}^{a} \right\rangle \left\langle \chi_{lm}^{a} \right|$ is the non-local projectors corresponding to the angular momentum number, lm, of the *a-th* atom. The norm-conserving Troullier–Martins pseudopotential [45] in the Kleinman–Bylander form has been implemented in the ARES package. The local part of the pseudopotential can be obtained by

$$V_{loc}(\vec{r}) = FFT'\left[\sum_{t=1}^{N_{type}} S^{t}(G)V_{loc}^{t}(|G|)\right], \tag{15}$$

where $FFT'[\cdot]$ denotes the reverse Fourier transform. N_{type} is the number of atomic species, and $S^{t}(G)$ and $V_{loc}^{t}(|G|)$ are the structure factor and 1D Fourier component of the local pseudopotential of the *t-th* type atom, respectively. More details about the local part of the pseudopotential can be found in Ref. [8]. The nonlocal part of the pseudopotential is given by

$$\hat{V}_{nl}^{\vec{k}}u_{n,\vec{k}} = e^{-i\vec{k}\cdot\vec{r}}\sum_{a=1}^{N_a}\sum_{lm}\frac{\chi_{lm}^{a}(\vec{r})}{\bar{V}_{lm}^{a}}\int\chi_{lm}^{a}(\vec{r}_a{}')e^{i\vec{k}\cdot\vec{r}'}u_{n,\vec{k}}(\vec{r}')d^3r'\cdot \tag{16}$$

Where $\vec{r}_a = \vec{r} - \vec{R}_a$. In addition, the local pseudopotentials [46–48] with high accuracy and transferability for elements can also be used in ARES.

2.3 CheFSI Method

The electronic structure calculations in DFT a large number of eigenpairs to be obtained for the KS-DFT equation. However, only the eigenpairs with energies within a small window inside the spectrum of $H_{\vec{k}}$ are needed. Chebyshev polynomials efficiently extract the subspace projection onto the target space of wanted eigenvectors associated with the occupied states [28,49]. Recently, the well-established CheFSI method has been used to solve the KS-DFT equation with 5-10 times faster SCF iteration than the eigensolver-based method [28]. Therefore, this method was also employed in ARES package.

In our implementation, the first type of Chebyshev polynomial is used to extract the required invariant subspace. The Chebyshev polynomials with *m*-degree, denoted as p^{m}, are defined by the three-term recurrence,

$$p^{m+1}(x) = 2xp^{m}(x) - p^{m-1}(x),\quad x \in R\cdot \tag{17}$$

Note that $p^0(x)=0,\ p^1(x)=x$.

A remarkable property of Chebyshev polynomials is rapid growth outside the interval $[-1,\ 1]$. Note that the detailed descriptions of CheFSI method can be found in Ref. [28]. Generally, it is assumed that the full spectrum of the Hamiltonian ($\sigma(H_{\vec{k}})$) is located in $[a_0,\ b]$, whereas the spectrum of the wanted eigenvectors is bounded within $[a_0,\ a]$. a should be larger than a_0 but smaller than b, whereas b should be greater than the upper bound of $\sigma(H_{\vec{k}})$. In the CheFSI method, the interval of the spectrum $[a,\ b]$ is dampened by affine mapping of $[a,\ b]$ into $[-1,\ 1]$,

$$H'_{\vec{k}}=\frac{H_{\vec{k}}-c}{e};\ \ c=\frac{a+b}{2},\ e=\frac{b-a}{2}\cdot \tag{18}$$

Then, the new filter is denoted by

$$P^m(H_{\vec{k}})=p^m(H'_{\vec{k}})\cdot \tag{19}$$

Lower bound a and upper bound b of the unwanted spectra play a pivotal role in this method. In our implementation, the upper bound can be estimated by a few steps of the Lanczos algorithm[50], and the largest Rayleigh–Ritz value of the previous iteration is used as the lower bound.

The details of the filtering processes are as follows.

1) The desired subspace, which corresponds to occupied states, is constructed by Chebyshev polynomial filtering, where the components of the wanted spectrum are assuredly magnified with respect to the components of the unwanted spectrum,

$$\tilde{U}^F_{\vec{k}}=P^m(H_{\vec{k}})\tilde{U}_{\vec{k}}, \tag{20}$$

where the matrix $\tilde{U}_{\vec{k}}\in\mathbb{C}^{N_b\times N_s}$ contains N_s column discrete vectors $\tilde{u}_{n,\vec{k}}$. $\tilde{U}^F_{\vec{k}}\in\mathbb{C}^{N_b\times N_s}$ denotes the filtered basis of the new subspace.

2) The Ritz pairs approximating the exact diagonalization solutions are evaluated by subspace diagonalization using the generalized Rayleigh–Ritz approach, $(\tilde{U}_{\vec{k}},\ \varepsilon_{\vec{k}})=Rayleigh\_Ritz(H_{\vec{k}},\ \tilde{U}^F_{\vec{k}})$. The detailed procedure is shown in Algorithm 1

[27].

Algorithm 1: Procedure for the Rayleigh-Ritz scheme

```
Rayleigh_Ritz( $H$, $\tilde{U}$)
    Compute the projected Hamiltonian $\bar{H} = \tilde{U}^{\dagger} H \tilde{U}$
    IF $\tilde{U}^{\dagger}\tilde{U} \neq I$ THEN
        Compute the overlap matrix $\tilde{S} = \tilde{U}^{\dagger}\tilde{U}$
        Diagonalize $\bar{H}Q = \tilde{S}Q\Sigma$, where Q contains the eigenvectors of $\bar{H}$,
        diagonal part of matrix $\Sigma$ contains the Ritz values of $H$, denote as
        $diag(\Sigma)$.
    ELSE
        Diagonalize $\bar{H}Q = Q\Sigma$
    END IF
    Rotate the basis $\tilde{U} = \tilde{U}Q$
Return the basis and eigenvalues $\tilde{U}$, $diag(\Sigma)$.
```

2.4 Implementation of ARES

2.4.1 Solving the KS-DFT equation using Chebyshev filtering

The flow chart of the ARES process used to solve the KS-DFT equation using Chebyshev filtering is shown in Fig. 1. The process comprises five main steps. First, the initial estimated electron density and the subspace are generated and used to estimate the Rayleigh–Ritz values and the initial lower bounds of the unwanted spectrums. Second, Chebyshev polynomial filtering is performed to obtain the desired subspace based on the previous subspace. Third, the Rayleigh–Ritz step is performed to evaluate the approximate eigenpairs. Fourth, the new Hamiltonian is constructed by updating the charge density. Note that the Pulay mixing [51] with Kerker preconditioning [51, 52] scheme for mixing the charge density is used to accelerate the SCF convergence. Fifth, the lower and upper bounds of the unwanted spectrum are determined by maximum Ritz value and the Lanczos algorithm. The self-consistent iteration step progressively approximates the wanted eigensubspace of the Hamiltonian and is repeated iteratively until a termination criterion, such as a prescribed error threshold for the charge density and total energy, is attained.

Estimating the initial subspace is a critical step to determine the numerical stability and convergence rate for solving the KS-DFT equation using the CheFSI method. The initial subspace generation scheme originally proposed by Zhou et al. requires solving for the eigenvectors of the initial Hamiltonian[28], which is a major problem for large-scale simulations. It was later demonstrated that a random initial subspace is more efficient [49]. Recently, initial subspace generation schemes using the extended Hückel (EH) method [54] or numerical atomic orbitals (NAOs) with a one-shot Rayleigh–Ritz step [27] have also been proposed.

Compared with the EH method, the method combining NAOs with the Rayleigh–Ritz scheme is easy to implement. Therefore, we also employed the NAO method to generate the initial subspace in ARES. However, this technique is very memory consuming for systems containing thousands of atoms because the number of NAOs used to construct the subspace is usually more than ten times the number of atoms. Furthermore, the technique is inefficient for large systems because the Rayleigh–Ritz scheme requires diagonalization of a large matrix, whose dimensions are the same as the number of NAOs. Although the direct truncation scheme of NAOs (DTNAO) provides a possible solution to overcome the shortcomings of the NAO method, the operations usually miss a part of the energy spectrum, leading to slow convergence (Fig. 2).

Here, a new scheme called Slater-type orbital combinations (STOC) is proposed to generate the initial subspace for large-scale systems. In this scheme, all the Slater-type orbitals (STOs) as one of NAOs are selected and linearly combined to be the basis of the subspace. The combination operation reduces the dimension of the subspace (N_s). Note that the number of STOs (N_{STO}) is greater than N_s, therefore, the STOC scheme is less memory consuming than the previous NAOs [27]. Furthermore, the new scheme reduces the spectrum loss because all the STOs are included.

The details of the generation of the initial subspace using the STOC are presented in Algorithm 2. The N_{STO}STOs are mapped to the basis of the subspace one by one; thus, the interpolation index of the basis periodically returns to the first basis vector when the index exceeds N_s, which is meaningful for describing orbital hybridization. We select a large system containing 1024 Si atoms to evaluate the effectiveness of STOC. The number of steps for convergence as a function of various initialized subspace generation schemes and the Chebyshev filter degree are shown in Fig. 2. The STOC and NAO schemes are more efficient than DTNAO and the number of steps for convergence for the STOC scheme is comparable to that for the NAO method when the Chebyshev polynomial degree is larger than 16. Since the STOC scheme is less memory consuming, it is suitable for large-scale simulations.

Algorithm 2: Initialize subspace by STOC

```
Initilize_subspace ( $H_{\text{int}}$)
    Define a $\mathbb{C}^{N_b \times N_s}$ matrix $U = [u_1,\ u_2,\ \cdots,\ u_{N_s}] = 0$
    $i = 1, j = 1$
    WHILE $i <= N_{STO}$ DO
        IF ( $j > N_s$)  set $j = 1$
        $u_j = u_j + \phi_i^{STO}$, combination of orbitals, where $\phi_i^{STO}$ is the $i$-th STO.
        $j = j + 1,\quad i = i + 1$
    ENDDO
    Rayleigh-Ritz step $(U_{\text{int}},\ e) = Rayleigh\_Ritz(H_{\text{int}},\ U)$ , where $U_{\text{int}}$ is
    the initial subspace and $e$ contains the Ritz values of $H_{\text{int}}$.
    Lower bound $a = \max\{e\}$ and upper bound $b$ is evaluated the Lanczos
    algorithm [50].
Return $U_{\text{int}},\ a,\ b$
```

2.4.2 Structural geometry relaxation

One major application of theoretical simulations is geometrical relaxation, which determine the atomic configuration with minimum energy on the Born–Oppenheimer energy surface. Generally, the force on each atom in relaxed structures should be close to zero. The forces on nuclei are related to the first derivatives of the total energy with respect to the nuclear coordinates. Generally, the total energy functional explicitly depends on the atomic positions and lattice matrix on the Born–Oppenheimer surface,

$$E_{tot}[R] = T_s[U] + E_H[\rho] + E_{xc}[\rho] + E_{i-e}^{loc}[\rho, R] + E_{i-e}^{nl}[U, R] + E_{i-i}[R], \qquad (21)$$

where $T_s[U]$, $E_H[\rho]$, $E_{xc}[\rho]$, $E_{i-e}^{loc}[\rho, R]$ and $E_{i-e}^{nl}[U, R]$ are the non-interacting kinetic energy, the Hartree potential energy, the exchange correlation energy, the local-

and nonlocal- part of ion-electron interaction energy, respectively. $E_{i-i}[R]$ is the ion-ion interaction energy, which can be calculated by Ewald summation[55–57].

The Hellmann-Feynman force on the *a*-th atom is given by

$$\vec{F}_a = -\frac{\partial E_{tot}[R]}{\partial \vec{R}_a} = -\frac{\partial E_{i-e}^{loc}[\rho, R]}{\partial \vec{R}_a} - \frac{\partial E_{i-e}^{nl}[U, R]}{\partial \vec{R}_a} - \frac{\partial E_{Ewald}[R]}{\partial \vec{R}_a}$$

$$= \vec{F}_a^{loc} + \vec{F}_a^{nl} + \vec{F}_a^{Ewald} \qquad (22)$$

where $\vec{F}_a^{local}$ and $\vec{F}_a^{Ewald}$ are the contribution from the local ionic potential and Ewald force[57], respectively. The nonlocal force expression of $\vec{F}_a^{nl}$ is given by

$$\vec{F}_{a,i-e}^{nl} = -\frac{\partial E_{i-e}^{nl}[U, R]}{\partial \vec{R}_a} = 2\mathrm{Re}\left\{ \sum_{n,\vec{k}} f_{n,\vec{k}} \sum_{lm} \frac{1}{\bar{V}_{lm}^a} G_{n\vec{k},alm}^* \int \frac{d\chi_{lm}^a(\vec{r}_a)}{d\vec{r}_a} \psi_{n,\vec{k}}(\vec{r}) d^3r \right\}, \qquad (23)$$

where $G_{n\vec{k},alm} = \int \chi_{lm}^a(\vec{r}_a) \psi_{n,\vec{k}}(\vec{r}) d^3r \cdot$

Currently, several well-established local optimization algorithms (e.g., steepest descent, conjugate gradient, quasi-Newton, and Fast Inertial Relaxation Engine) are available. Our previous study demonstrated that the improved limited-memory quasi-Newton (L-BFGS) method only requires limited computer memory and yields significantly faster convergence for large-scale geometrical structure relaxation[58]. Thus, L-BFGS was also used in the ARES package.

2.4.3 Parallel implementation

ARES is intended to simulate large-scale systems. Therefore, it is highly desirable to implement a parallel scheme in ARES to take full advantage of the massive parallelization. In our parallelization scheme, the parallel mode uses the standard message passing interface library for communication, and all the terms of the Hamilton on the real-space grids are implemented using the spatial decomposition, where the 3D domain is divided into a 2D block distribution. We expect ARES to be highly efficient because less communication is required between the processors owing to the short-range operations referring to real-space finite difference expansion. Furthermore, the ScaLAPACK interface is also used to harness the computational power of the cores efficiently.

3 Numerical results

To evaluate the accuracy, computational efficiency, and parallel scaling of ARES, we use it to simulate a wide variety of materials, including metals, semiconductors, and insulators. All the calculations use the LDA for electron exchange and correlation as parameterized by Perdew and Zunger [40]. The electron-ion interaction of Al, Si, Mg, Ga and As was described by local pseudopotentials available from the website[59]. The electron-ion interaction of B, C, N, O and Zn was described by norm-conserving Troullier–Martins pseudopotentials with $2s^2 2p^1$, $2s^2 2p^2$, $2s^2 2p^3$, $2s^2 2p^4$ and $3d^{10} 4s^2$ configurations treated as the valence electrons. The B, C, N and O have the core cutoff radii of 1.39, 1.60, 1.50 and 1.30 bohr for *s*- and *p*-channels, respectively. The Zn has the core cutoff radii of 2.28 bohr for *s*-, *p*- and *d*-channels. All the Troullier–Martins pseudopotentials are available from the website[60].

3.1. Tests of ARES convergence

The order of finite-difference expansion and the grid spacing are the controllable real-space finite-difference parameters in ARES that critically affect the accuracy of the calculations. These parameters are selected depending on the convergence test of the total energies of the systems. Here, we describe how to select the values of these parameters in practice. We run the ARES code to calculate the total energy for crystalline Al, Si and C. Just as shown in Fig. 3, the 16th-, 16th- and 10th-order finite-difference expansion with the grid spacings of 0.32, 0.22 and 0.12 Å are sufficient for a well-converged total energy (less than 1.0 meV/atom) for Al(FCC), Si(FCC) and C(CD), respectively.

The Chebyshev polynomial degree (*m*) and the number of computed eigenstates (N_s) are the two critical parameters for the CheFSI method implemented in ARES to determine the convergence rate. Here, crystalline silicon containing 1024 atoms is used to assess the convergence rate of these parameters. The density residual depending on the Chebyshev polynomial degree is shown in Fig. 4(a). The Chebyshev filter degree of 16 gives good convergence for crystalline silicon. Generally, N_s should be greater than the number of occupied states, N_{occ}, to avoid missing occupied eigenstates during the filtering [27, 28]. Here, we also test convergence rate depending on the number of computed eigenstates using crystalline silicon containing 1024 atoms. Fig. 4(b) shows the number of iterations required for convergence of the density as a function of the number of extra eigenstates N_{extra} relative to the number of atoms. The addition of 10% extra unoccupied states results in a fast convergence (14 steps) for crystalline silicon.

3.2 Computational accuracy

We verify the computational accuracy of ARES by comparing the crystalline bulk properties of several elements and binary and ternary compounds with those determined using the CASTEP package. The total energy versus volume equation of states of Al with a face-centered-cubic (FCC) structure and Si, C with a cubic diamond (CD) structure are obtained using the ARES and CASTEP software with the same pseudopotentials. In Figs. 5(a–c), we plot the energy as a function of the volume. There is excellent agreement between ARES and CASTEP, with the curves being practically indistinguishable. We find that the calculated equilibrium volume (V_0), equilibrium energy (E_0), and bulk modulus (B_0) obtained using ARES are consistent with the CASTEP data, and their differences are within 0.05%, 0.5 meV/atom, and 1.0%, respectively. The theoretical V_0 and B_0 are estimated by fitting the total energies as a function of volume to the Murnaghan equation of states[61]. Further validation of the numerical stability of ARES is also given by comparing the total energies of ten GaAs structures calculated using ARES and CASTEP. These structures are randomly generated by the CALYPSO package[62, 63]. The results in Fig. 5(d) show identical energy differences for the two sets, validating the numerical stability of ARES further.

The binary and ternary compounds of BN, ZnO, and $MgSiO_3$ with complex structures are used to evaluate the computational accuracy of the ARES software further. The calculated V_0, E_0, and B_0 values using ARES and CASTEP are presented in Table 1. Our results are in excellent agreement with that obtained by CASTEP, which supports the validity of the ARES software. Note that the structural details of BN, ZnO, and $MgSiO_3$ are listed in Table 2.

We compare the band structures of Al, Si, C, and GaAs calculated using ARES and

CASTEP. We use the W – L – Γ – X – K bands, whose reciprocal lattice vector coordinates are [0.5 0.25 0.75], [0.5 0.5 0.5], [0.0 0.0 0.0], [0.5 0.0 0.5], and [0.375 0.375 0.75], respectively. We discretize each line to a set of *k*-points and determine the band structures using ARES. Fig. 6 shows the comparisons of the calculated band structures by ARES and CASTEP. Obviously, the two calculations give nearly identical band structures, HOMO eigenvalues, LUMO eigenvalues, and bandgaps, demonstrating the accuracy of ARES.

3.3 Computational efficiency

To evaluate the computational efficiency of the parallel ARES package, static simulations of the supercell structures of Al containing different numbers of atoms are performed using 16, 64, and 256 processors. The total wall time per self-consistent step of supercell Al as a function of the number of atoms is presented in Fig. 7(a). The scaling with respect to the number of atoms for ARES is approximately quadratic O($N^{2.22}$). The calculations are performed using the high-performance Tianhe-2 supercomputer at the National Supercomputer Center of Guangzhou, where each node contains two 12-core Intel Xeon E5-2692 v2 CPUs with 128-GB memory with a maximum interconnect speed of 160 GBps.

To illustrate the parallel scalability of ARES, we perform a static calculation of a supercell of the diamond structure containing 2048 Si atoms using various cores. The speedup ratio and parallel efficiency as a function of the number of cores with respect to 32 cores are plotted in Fig. 7(b). The parallel efficiency reaches 75% for 256 processors, demonstrating the strong parallel scalability of ARES for simulating large-scale systems.

The total energies for large supercell of Si, C, and Al containing 2048, 3072, 2304, 3072, 4096, 6912, and 10192 atoms (corresponding to 8192, 12288, 9216, 12288, 12288, 20736 and 30576 electrons, respectively) calculated using ARES are listed in Table 3. The total energy differences between ARES and CASTEP software for Si, C and Al are 5.0, 12.0 and 4.0 meV/atom, respectively. Note that the k-space representation equivalent direct simulation BvK supercells is employed in CASTEP and the plane-wave basis kinetic energy cutoff of 940, 2600 and 940 eV are chosen for

Si, C and Al to ensure that energy can converge to better than 1.0 meV/atom. The maximum iteration steps to convergence are 18, 19, and 13 for Si, C, and Al, respectively. It is worth mentioning that a large-scale simulation of a supercell Al structure containing 10,192 atoms not accessible using eigenvector-based methods was performed by ARES and the total wall time for convergence of the electron density using 256 cores is 9.76 hours. These results illustrate the superior performance of ARES for simulating large-scale systems.

To test the convergence of the geometrical structure relaxation of ARES for a large system, a crystalline structure containing 108 Al atoms is used as a benchmark. The total energy and maximum force as a function of the number of L-BFGS steps during the geometrical structure relaxation are presented in Figs. 8(a) and (b), respectively. Only about 13 steps are required for the total energy and maximal force to converge to within 1 meV/atom and 0.01 eV/Å, respectively. Thus, ARES yields fast convergence of geometrical structure relaxation for complex structures.

4. Conclusion

The real-space finite-difference method combined with the Chebyshev filter subspace iteration was employed to solve the Kohn–Sham equation and was implemented in our ARES software package. The performance of ARES was thoroughly tested using static simulations of a wide variety of material systems containing thousands of atoms on a modest computer cluster. The high efficiency and scalability of the paralleled ARES software make it an efficient, portable, massively parallel computational tool for large-scale simulations of a wide range of materials.

ACKNOWLEDGEMENTS

The authors would like to acknowledge funding support received from the National Natural Science Foundation of China under Grant Nos. 11822404, 11774127, and 11534003; the National Key Research and Development Program of China under Grant Nos. 2016YFB0201200, 2016YFB0201201, and 2016YFB0201204; the Program for JLU Science and Technology Innovative Research Team (JLUSTIRT); and the Science Challenge Project No. TZ2016001. Parts of the calculation were performed at the high-performance computing center of Jilin University and at Tianhe2-JK at the Beijing Computational Science Research Center.

References

[1] P. Hohenberg, W. Kohn, Phys. Rev. 136 (1964) B864–B871.

[2] W. Kohn, L.J. Sham, Phys. Rev. 140 (1965) A1133–A1138.

[3] G. Kresse, J. Furthmüller, Phys. Rev. B 54 (1996) 11169–11186.

[4] G. Kresse, J. Furthmüller, Comput. Mater. Sci. 6 (1996) 15–50.

[5] X. Gonze, B. Amadon, et al., Comput. Phys. Commun. 180 (2009) 2582–2615.

[6] M.D. Segall, J. Phys. Condens. Matter 14 (2002) 2717.

[7] P. Giannozzi, S. Baroni, et al., J. Phys. Condens. Matter 21 (2009) 395502.

[8] M.C. Payne, M.P. Teter, D.C. Allan, T.A. Arias, J.D. Joannopoulos, Rev. Mod. Phys. 64 (1992) 1045–1097.

[9] Weitao Yang, Phys. Rev. Lett. 66 (1991) 1438–1441.

[10] W. Yang, Phys. Rev. A 44 (1991) 7823–7826.

[11] C. Lee, W. Yang, J. Chem. Phys. 96 (1992) 2408–2411.

[12] F. Shimojo, R.K. Kalia, A. Nakano, P. Vashishta, Comput. Phys. Commun. 167 (2005) 151–164.

[13] F. Shimojo, R.K. Kalia, A. Nakano, P. Vashishta, Phys. Rev. B 77 (2008) 085103.

[14] S. Goedecker, Rev. Mod. Phys. 71 (1999) 1085–1123.

[15] C.-K. Skylaris, P.D. Haynes, A.A. Mostofi, M.C. Payne, J. Chem. Phys. 122 (2005) 084119.

[16] P.D. Haynes, C.K. Skylaris, A.A. Mostofi, M.C. Payne, Phys. Status Solidi Basic Res. 243 (2006) 2489–2499.

[17] D.R. Bowler, T. Miyazaki, M.J. Gillan, J. Phys. Condens. Matter 14 (2002) 2781–2798.

[18] D.R. Bowler, R. Choudhury, M.J. Gillan, T. Miyazaki, Phys. Status Solidi Basic Res. 243 (2006) 989–1000.

[19] J. Vandevondele, M. Krack, F. Mohamed, M. Parrinello, T. Chassaing, J. Hutter, Comput. Phys. Commun. 167 (2005) 103–128.

[20] S. Mohr, L.E. Ratcliff, P. Boulanger, L. Genovese, D. Caliste, T. Deutsch, S. Goedecker, J. Chem. Phys. 140 (2014) 204110.

[21] S. Mohr, L.E. Ratcliff, L. Genovese, D. Caliste, P. Boulanger, S. Goedecker, T. Deutsch, Phys. Chem. Chem. Phys. 17 (2015) 31360–31370.

[22] D.R. Bowler, T. Miyazaki, Reports Prog. Phys. 75 (2012) 036503.

[23] L. Lin, J. Lu, R. Car, W. E, Phys. Rev. B 79 (2009) 115133.

[24] L. Lin, M. Chen, C. Yang, L. He, J. Phys. Condens. Matter 25 (2013) 295501.

[25] L. Lin, A. García, G. Huhs, C. Yang, J. Phys. Condens. Matter 26 (2014) 305503.

[26] P. Li, X. Liu, M. Chen, P. Lin, X. Ren, L. Lin, C. Yang, L. He, 112 (2016) 503–517.

[27] V. Michaud-rioux, L. Zhang, H. Guo, J. Comput. Phys. 307 (2016) 593–613.

[28] Y. Zhou, Y. Saad, M.L. Tiago, J.R. Chelikowsky, J. Comput. Phys. 219 (2006) 172–184.

[29] Y. Zhou, Y. Saad, M.L. Tiago, J.R. Chelikowsky, Phys. Rev. E 74 (2006) 066704.

[30] S. Ghosh, P. Suryanarayana, Comput. Phys. Commun. 212 (2017) 189–204.

[31] S. Ghosh, P. Suryanarayana, Comput. Phys. Commun. 216 (2017) 109–125.

[32] J.R. Chelikowsky, N. Troullier, Y. Saad, Phys. Rev. Lett. 72 (1994) 1240–1243.

[33] A. Natan, A. Benjamini, D. Naveh, L. Kronik, M.L. Tiago, S.P. Beckman, J.R. Chelikowsky, Phys. Rev. B 78 (2008) 075109.

[34] Y. Saad, J.R. Chelikowsky, S.M. Shontz, J. Soc. Ind. Appl. Math. 52 (2010) 3–54.

[35] T.L. Beck, Rev. Mod. Phys. 72 (2000) 1041–1080.

[36] B. Fornberg, SIAM Rev. 40 (1998) 685–691.

[37] D. Ninno, G. Cantele, F. Trani, J. Comput. Chem. 39 (2018) 1406–1412.

[38] W. Mi, X. Shao, C. Su, Y. Zhou, S. Zhang, Q. Li, H. Wang, L. Zhang, M. Miao, Y. Wang, Y. Ma, Comput. Phys. Commun. 200 (2016) 87–95.

[39] A. Gholami, D. Malhotra, H. Sundar, G. Biros, SIAM J. Sci. Comput. 38 (2016) C280–C306.

[40] J.P. Perdew, A. Zunger, Phys. Rev. B 23 (1981) 5048–5079.

[41] J.P. Perdew, K. Burke, M. Ernzerhof, Phys. Rev. Lett. 77 (1996) 3865–3868.

[42] M.A.L. Marques, M.J.T. Oliveira, T. Burnus, Comput. Phys. Commun. 183 (2012) 2227–2281.

[43] D.R. Hamann, M. Schlüter, C. Chiang, Phys. Rev. Lett. 43 (1979) 1494–1497.

[44] L. Kleinman, D.M. Bylander, Phys. Rev. Lett. 48 (1982) 1425–1428.

[45] N. Troullier, J.L. Martins, Phys. Rev. B 43 (1991) 1993–2006.

[46] B. Zhou, Y. Alexander Wang, E.A. Carter, Phys. Rev. B 69 (2004) 125109.

[47] C. Huang, E.A. Carter, Phys. Chem. Chem. Phys. 10 (2008) 7109–7120.

[48] W. Mi, S. Zhang, Y. Wang, Y. Ma, M. Miao, J. Chem. Phys. 144 (2016) 134108.

[49] Y. Zhou, J.R. Chelikowsky, Y. Saad, J. Comput. Phys. 274 (2014) 770–782.

[50] Y. Zhou, R.C. Li, Linear Algebra Appl. 435 (2011) 480–493.

[51] P. Pulay, Chem. Phys. Lett. 73 (1980) 393–398.

[52] G.P. Kerker, Phys. Rev. B 23 (1981) 3082–3084.

[53] Y. Zhou, H. Wang, Y. Liu, X. Gao, H. Song, Phys. Rev. E 97 (2018) 1–12.

[54] M. Lee, K. Leiter, C. Eisner, J. Crone, J. Knap, Comput. Theor. Chem. 1062 (2015) 24–29.

[55] P. Ewald, Ann. Phys. (1390) 253–287.

[56] A.Y. Toukmaji, J.A. Board, Comput. Phys. Commun. 95 (1996) 73–92.

[57] G.S. Ho, V.L. Lignères, E.A. Carter, Comput. Phys. Commun. 179 (2008) 839–854.

[58] X. Shao, Q. Xu, S. Wang, J. Lv, Y. Wang, Y. Ma, Comput. Phys. Commun. 233 (2018) 78–83.

[59] https://Github.Com/PrincetonUniversity/BLPSLibrary

[60] https://Parsec.Oden.Utexas.Edu/Styled-2/

[61] F.D. Murnaghan, Proc. Natl. Acad. Sci. U. S. A. 30 (1923) 244–247.

[62] Y. Wang, J. Lv, L. Zhu, Y. Ma, Phys. Rev. B 82 (2010) 094116.

[63] Y. Wang, J. Lv, L. Zhu, Y. Ma, Comput. Phys. Commun. 183 (2012) 2063–2070.

Table 1: Comparison of bulk properties of BN, ZnO, and $MgSiO_3$ obtained using CASTEP and ARES software.

Systems	software	V_0(Å^3/Cell)	E_0(eV/atom)	B_0(GPa)
BN	CASTEP	69.771	-175.547	250.0
	ARES	69.848	-175.549	248.8
ZnO	CASTEP	93.561	-974.118	188.6
	ARES	93.455	-974.117	185.9
$MgSiO_3$	CASTEP	394.124	-290.001	226.7
	ARES	393.530	-290.000	218.2

Table 2: Structural details of BN, ZnO and $MgSiO_3$.

Systems	Space group (Number)	Lattice parameters	Element	Wykoff position
BN	*Cmcm* (63)	a=2.48600 Å	B	4c 0.50000 0.83330 0.75000
		b=4.30588 Å	N	4c 0.50000 0.16670 0.75000
		c=6.51600 Å		
ZnO	*Cmc2₁* (36)	a=3.24986 Å	O	4a 0.50000 0.83330 0.61750
		b=5.62892 Å	Zn	4a 0.50000 0.83330 0.00000
		c=5.20662 Å		
$MgSiO_3$	*P2₁/c* (14)	a=9.38447 Å	Mg	4e 0.75800 0.01400 0.56500
		b=8.82500 Å		4e 0.75800 0.65300 0.53300
		c=5.18800 Å	O	4e 0.89500 0.18700 0.84400
		β=103.3233°		4e 0.75800 0.65300 0.53300
				4e 0.87600 0.65900 0.26400
				4e 0.63800 0.65600 0.81800
				4e 0.62300 0.50200 0.32500
				4e 0.60500 0.22500 0.50400
			Si	4e 0.94700 0.16100 0.18300
				4e 0.54300 0.15800 0.74900

Table 3. Total wall time for ARES calculations on large Si, C, and Al supercell systems. Note that only the gamma point is used in ARES. The number of atoms and electrons in the systems are denoted by N_{atom} and N_e, respectively. (N_x, N_y, N_z) is the size of the real-space grid. The subspace dimension is denoted by N_s. The grid spacings are 0.30, 0.15 and 0.37 Å for Si, C and Al systems, respectively.

System	$N_{atom}(N_e)$	N_x	N_y	N_z	N_s	steps	cores	Time(h)
Si	2048(8192)	104	104	104	4300	18	64	1.67
Si	3072(12288)	104	104	160	6451	17	64	4.79
C	2304(9216)	144	144	192	4618	18	256	2.33
C	3072(12288)	144	192	192	6154	19	256	5.07
Al	4096(12288)	175	96	96	6553	13	256	1.14
Al	6912(20736)	132	144	144	11059	13	256	3.95
Al	10192(30576)	154	160	144	16307	12	256	9.76

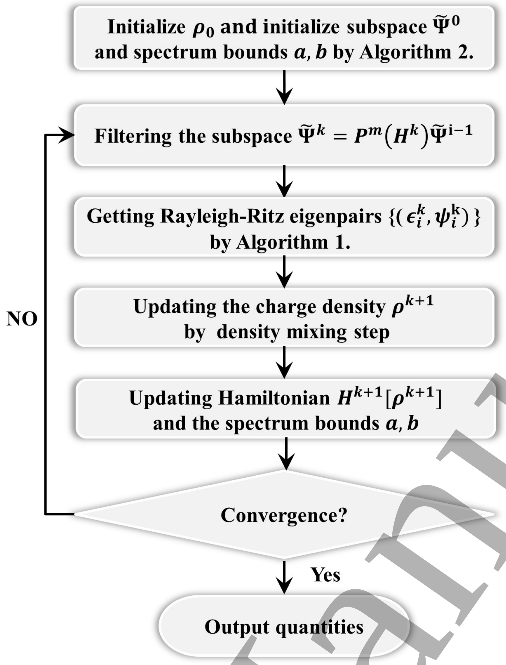
Initialize ρ_0 and initialize subspace $\widetilde{\Psi}^0$ and spectrum bounds a, b by Algorithm 2.
Filtering the subspace $\widetilde{\Psi}^k = P^m(H^k)\widetilde{\Psi}^{i-1}$
Getting Rayleigh-Ritz eigenpairs $\{(\epsilon_i^k, \psi_i^k)\}$ by Algorithm 1.
Updating the charge density ρ^{k+1} by density mixing step
Updating Hamiltonian $H^{k+1}[\rho^{k+1}]$ and the spectrum bounds a, b
NO
Convergence?
Yes
Output quantities

Fig. 1 Flow chart of the SCF calculation in ARES.

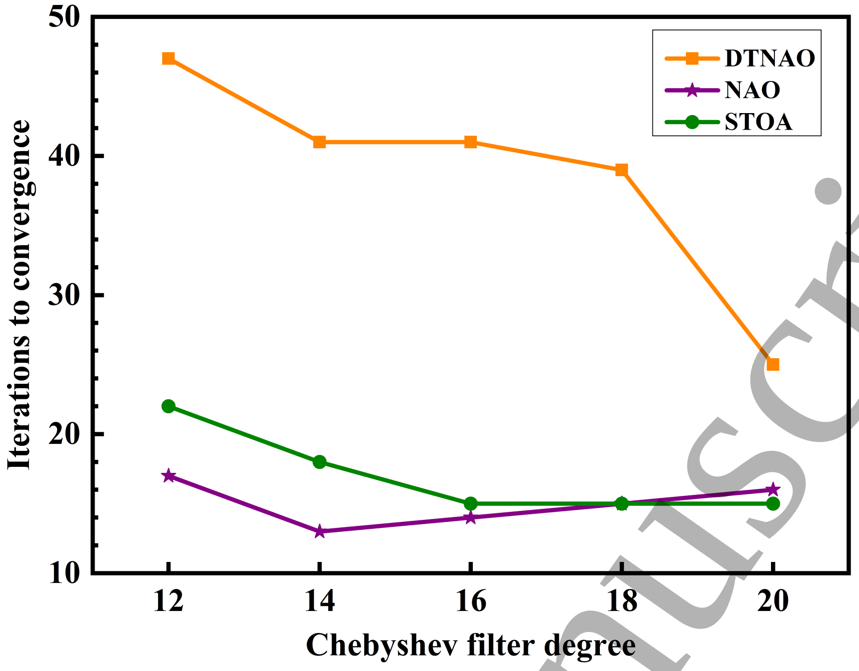

Fig. 2. Number of iterations required to converge the total energy of a unit cell containing 1024 Si atoms as a function of various initialized subspace generation schemes and Chebyshev filtering degree.

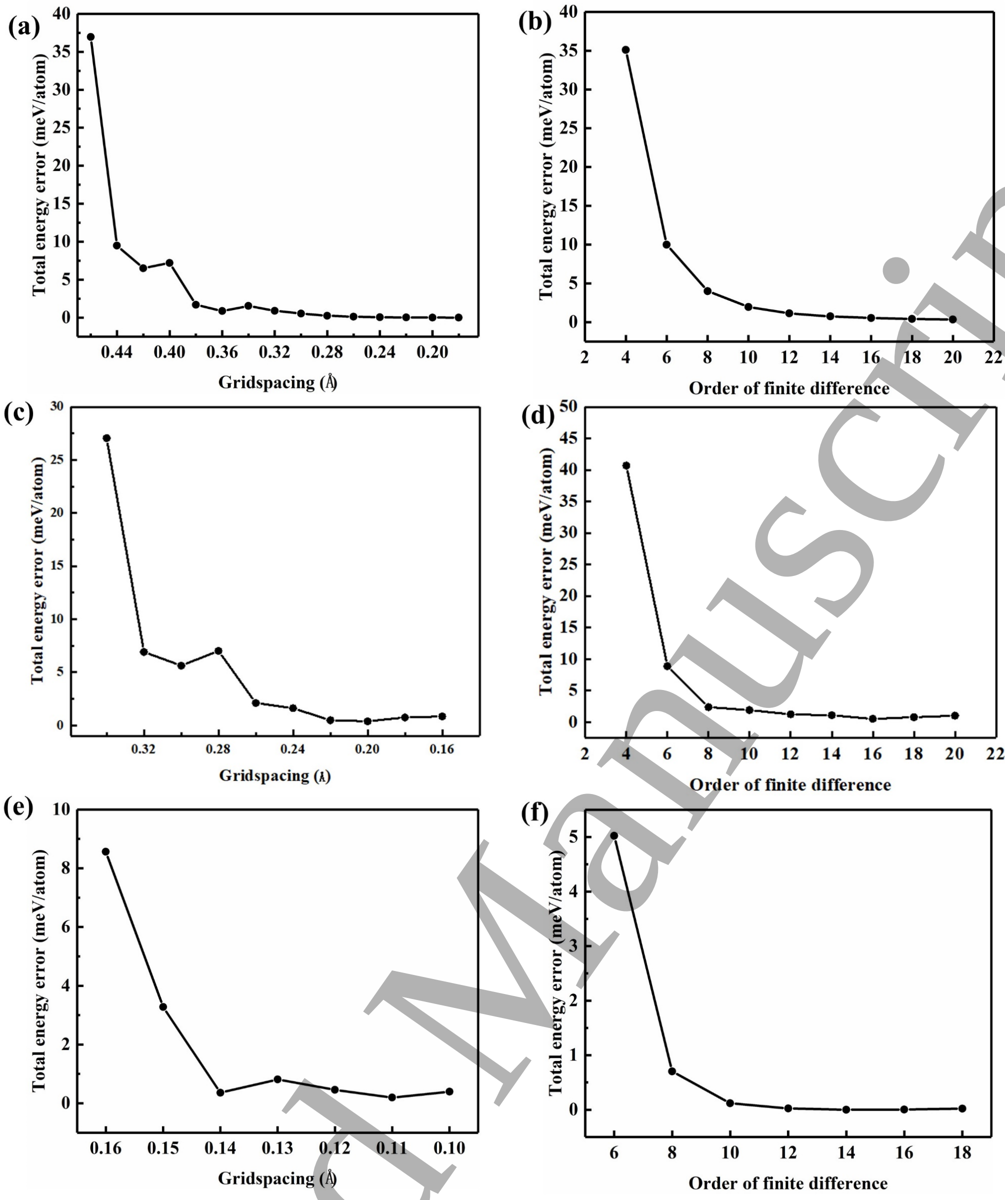

Fig. 3 Convergence test for ARES. Effect of grid spacing h and the order of finite-difference approximation N_{ord} on the total energy of supercell FCC Al containing 64 atoms (a, b), supercell FCC Si containing 256 atoms (c, d) and C with CD structure containing 2 atoms (e, f). Note that only the gamma point is used for simulations of Al and Si, while $8\times8\times8$ k-meshes are employed for simulation of C.

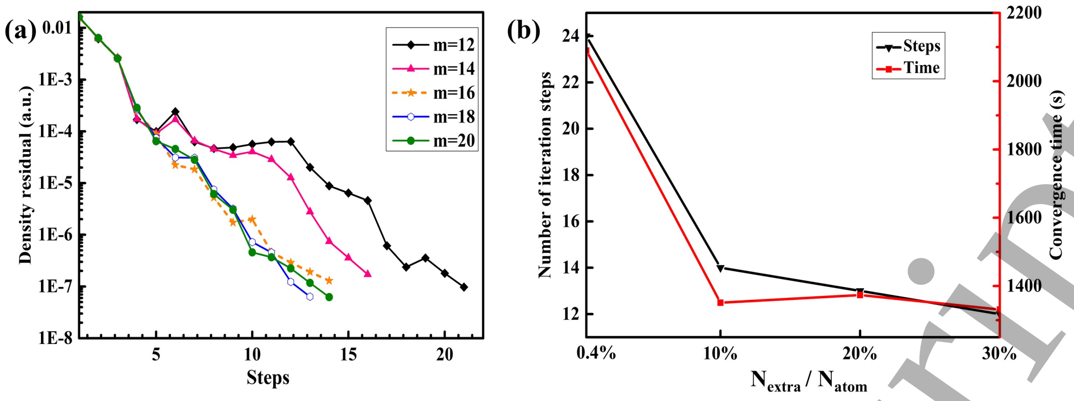

Fig. 4 (a) Dependence of density residual on Chebyshev polynomial degree(m). (b) Dependence of number of iteration steps on the number of extra eigenstates relative to the number of atoms.

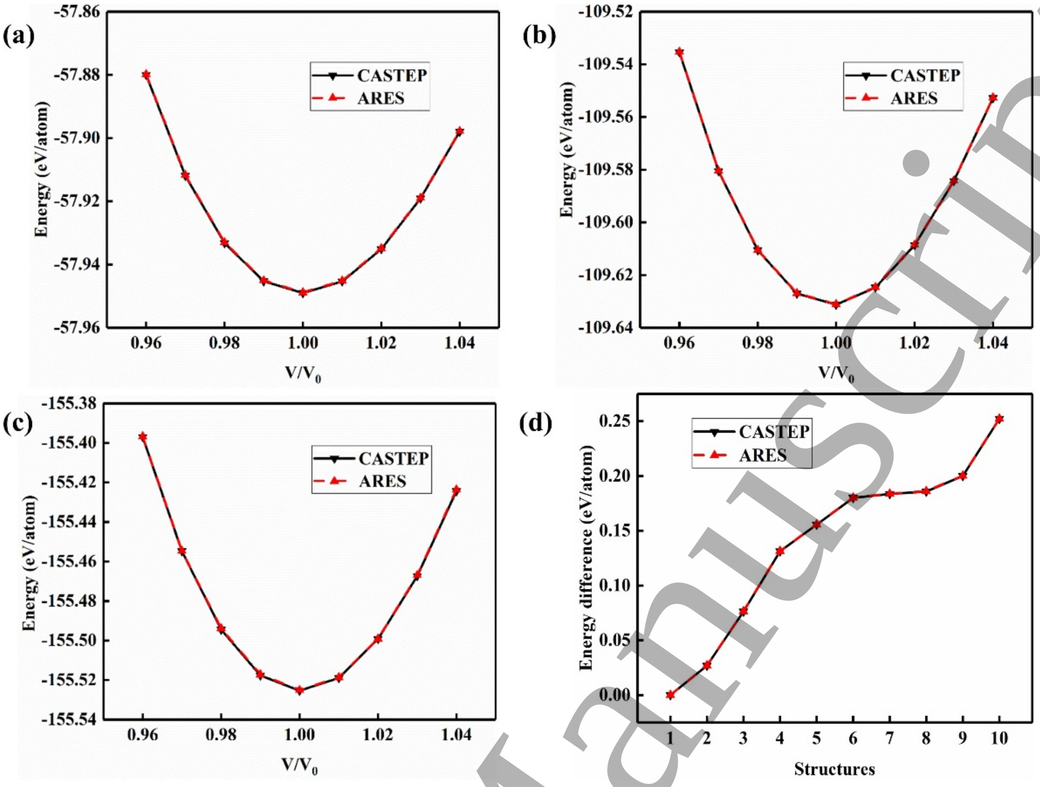

Fig. 5 Comparison of the equation of states for (a) Al FCC, (b) Si CD, and (c) C CD obtained using ARES and CASTEP. (d) Relative energy differences between the structure with the lowest energy for all the structures considered and another nine structures of GaAs generated by CALYPSO. Note that the $18\times18\times18$, $8\times8\times8$, and $8\times8\times8$ *k*-meshes used for Al, Si, and C give energy convergences of less than 1.0 meV/atom. The grid spacings are 0.20, 0.20, 0.10, and 0.20 Å in ARES and the kinetic cutoff energies are 940, 940, 2600, and 940 eV in CASTEP for Al, Si, C, and GaAs, respectively.

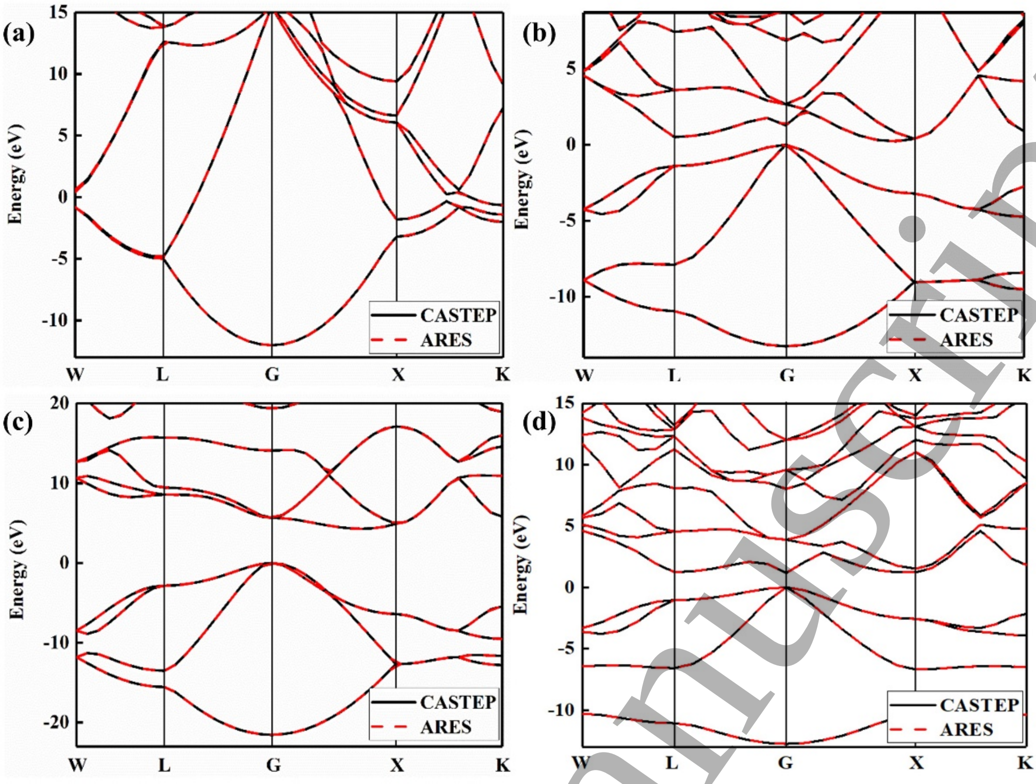

Fig. 6. Band structures of (a) Al, (b) Si, (c) C, and (d) GaAs.

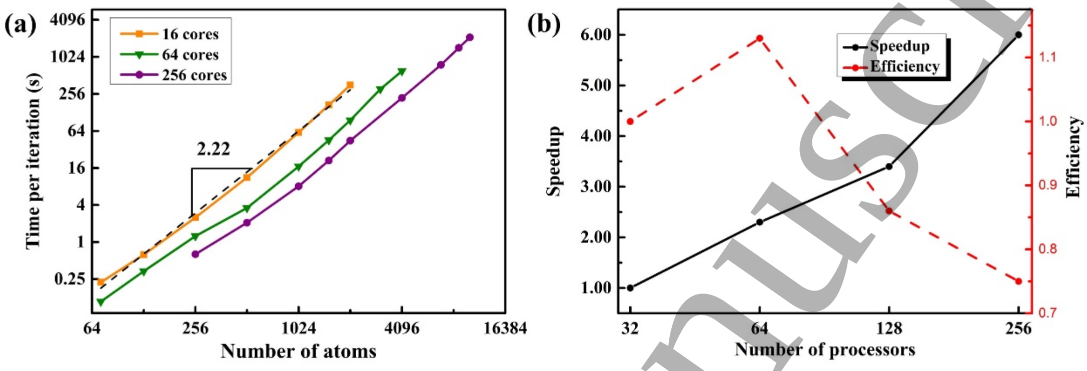

Fig. 7. (a) Wall time per self-consistent step using the various cores as a function of the number of Al atoms. (b) Speedup ratio (black solid line) and parallel efficiency (red dashed line) as a function of the number of processors for Si supercell containing 2048 atoms.

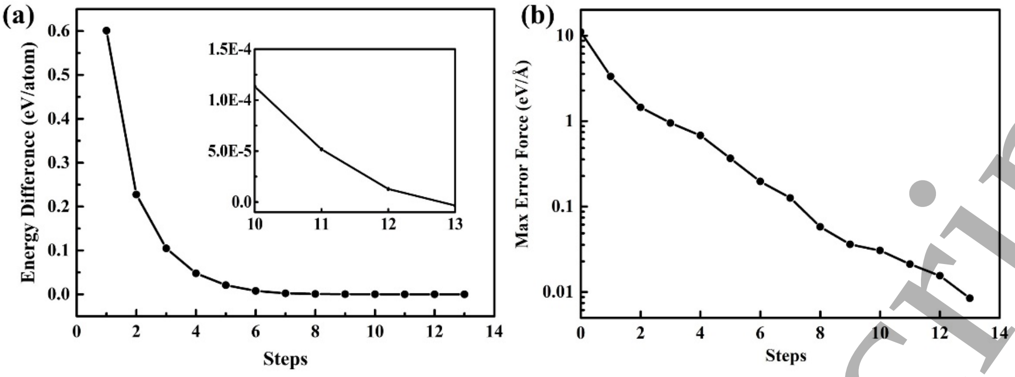

Fig. 8. Evolution of the energy difference (a) and maximum force (b) as functions of the number of L-BFGS steps during geometry relaxation for crystalline Al containing 108 atoms. The inset of (a) shows the evolution of the energy difference for the last few steps.