Contents lists available at ScienceDirect

Journal of Computational Physics

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

Towards chemical accuracy using a multi-mesh adaptive finite element method in all-electron density functional theory

Yang Kuang [a], Yedan Shen [a,b], Guanghui Hu [c,d,*]

[a] School of Mathematics and Statistics & Center for Mathematics and Interdisciplinary Science (CMIS), Guangdong University of Technology, China
[b] Hunan Key Laboratory for Computation and Simulation in Science and Engineering, Xiangtan University, China
[c] State Key Laboratory of Internet of Things for Smart City and Department of Mathematics, University of Macau, Macao SAR, China
[d] Zhuhai UM Science and Technology Research Institute, Zhuhai, China

ARTICLE INFO

Keywords:
Chemical accuracy
Kohn–Sham equation
Adaptive finite element method
Multi-mesh method

ABSTRACT

Chemical accuracy serves as an important metric for assessing the effectiveness of the numerical method in Kohn–Sham density functional theory. It is found that to achieve chemical accuracy, not only the Kohn–Sham wavefunctions but also the Hartree potential, should be approximated accurately. Under the adaptive finite element framework, this can be implemented by constructing the *a posteriori* error indicator based on approximations of the aforementioned two quantities. However, this way results in a large amount of computational cost. To reduce the computational cost, we propose a novel solver for the Kohn–Sham equation by using the multi-mesh adaptive technique, in which the Kohn–Sham equation and the Poisson equation are solved in two different meshes on the same computational domain, respectively. With the proposed method, chemical accuracy can be achieved with less computational consumption compared with the adaptive method on a single mesh, as demonstrated in a number of numerical experiments.

1. Introduction

The Kohn–Sham density functional theory (KSDFT) plays an important role in the area of quantum physics, condensed matter, and computational chemistry. For a quantum system containing N_{nuc} nuclei and N_{ele} electrons, the ground state can be obtained by solving the lowest N_{occ} eigenpairs $(\varepsilon_i, \psi_i)_{i=1,\dots,N_{occ}}$ from the Kohn–Sham equation

$$\left(-\frac{1}{2}\nabla^2 + V_{\mathrm{ext}}(\mathbf{r}) + V_{\mathrm{Har}}(\mathbf{r}) + V_{\mathrm{xc}}(\mathbf{r})\right)\psi_i(\mathbf{r}) = \varepsilon_i \psi_i(\mathbf{r}),\ i = 1,\dots,N_{occ}, \tag{1}$$

where $N_{occ} = N_{ele}/2$ if N_{ele} is even and $(N_{ele}+1)/2$ if N_{ele} is odd is the number of occupation number. $V_{\mathrm{ext}}(\mathbf{r}) = -\sum_I Z_I/|\mathbf{R}_I - \mathbf{r}|$ represents the external potential due to the nuclei located at $\{\mathbf{R}_I\}_{I=1,\dots,N_{nuc}}$ with charge $\{Z_I\}_{I=1,\dots,N_{nuc}}$. $V_{\mathrm{Har}}(\mathbf{r})$ stands for the Hartree potential which describes the interaction among electrons. $V_{\mathrm{xc}}(\mathbf{r})$ is the exchange-correlation potential used to absorb the difference between the Kohn–Sham one-body non-interactive system and the real many-body interactive system.

Chemical accuracy represents the level of precision required to make realistic chemical predictions, generally considered as errors below 1 kcal/mole ($\approx 1.59 \times 10^{-3}$ Hartree/particle) with respect to the total energy [27]. However, achieving this level of precision in KSDFT is challenging due to limitations in current exchange-correlation functionals, which typically restrict errors to within 2-

[*] Corresponding author at: State Key Laboratory of Internet of Things for Smart City and Department of Mathematics, University of Macau, Macao SAR, China.
E-mail address: garyhu@um.edu.mo (G. Hu).

https://doi.org/10.1016/j.jcp.2024.113312
Received 24 October 2023; Received in revised form 11 July 2024; Accepted 24 July 2024

Available online 2 August 2024

3 kcal/mole for many molecules[6,38]. Despite these inherent model errors, precise DFT calculations remain essential for attaining chemical accuracy [5,25]. Notably, even with the same functional, different DFT codes—each implementing the formalism uniquely—often yield inconsistent results for the same input structure [20]. This highlights the importance of new methods being able to replicate results from existing packages, as reproducibility is a key measure of their reliability.

In this work, we concentrate on developing efficient methods for a given model, and model error is beyond our scope. Therefore, we follow [5,28,25] to treat chemical accuracy as achieving total energy error within 1×10^{-3} Hartree/atom compared to state-of-the-art packages to verify the effectiveness and reproducibility of our methods.

Various numerical methods have been presented to solve the KSDFT [35,1,37,2,9,25,28,11,10]. Due to the rapid variations of the wavefunctions around the nuclear positions, a uniform discretization of the Kohn–Sham equation in the computational domain would result in an unaffordable implementation, especially in all-electron calculations. The adaptive finite element (AFE) methods which enable different mesh sizes are favored in solving the Kohn–Sham equation. Furthermore, the ability of handling the non-periodic boundary condition and complex computational domain makes AFE methods more and more attractive in electronic structure calculation in recent decades [31,19,2,15,25,8,22,11,24,29].

To achieve chemical accuracy [31,25,28,11], the Hartree potential plays a crucial role. The Hartree potential, a component of the Kohn-Sham effective potential, is typically derived from solving a Poisson equation. It is important to note that the Hartree potential and the Kohn–Sham wavefunctions exhibit different behaviors in the computational domain. For example, the Hartree potential decays as $1/r$ at infinity, while the wavefunctions decay exponentially [18,19,26]. Moreover, the variations around the singularities of the wavefunctions are more rapid than those of the Hartree potential. Therefore, the finite element space chosen for the Kohn-Sham equation is not suitable for the Hartree potential, emphasizing the need to select an appropriate finite element space for it [11].

Various strategies have been proposed to find an appropriate finite element space for the Hartree potential. These strategies include using a finer mesh for the Hartree potential while maintaining the same polynomial degree for the Kohn-Sham wavefunctions [31], doubling the polynomial degree while keeping the mesh for the Hartree potential unchanged [28,11], and employing a sufficiently large computational domain for the Poisson equation [25]. However, it should be pointed out that these approaches are far from perfect, due to the reason that the approximate space for the Hartree potential in each approach is inherited from the one for wavefunctions. This space is not ideally suit the Hartree potential's distinct behaviors in comparison to wavefunctions. A possible remedy for creating a suitable space accommodating both the Hartree potential and the wavefunctions is the use of an *a posteriori* error indicator [33], considering contributions from both the Poisson and Kohn-Sham equations. While this approach can yield accurate results, it also introduces significant computational complexity, resulting from the substantial number of degrees of freedom defined on a single mesh, driven by differences not only in decay behavior but also in the magnitude of the error indicator between the two quantities. In light of these observations, a practical solution, that is, the *multi-mesh adaptive method*, to simultaneously address accuracy and complexity concerns. This involves the adaptive design of separate approximate spaces for wavefunctions and the Hartree potential. With such an idea, the numerical accuracy issue can be handled well since the approximate space for each variable is tailored according to its own feature, while the complexity is effectively reduced by replacing a large system of linear equations with two smaller ones.

Distinct from the method that solves all the equations on the same mesh (hereafter we call such method *single-mesh method*), the multi-mesh adaptive method (abbreviated as *multi-mesh method*) is to solve different variables with different regularity on different meshes in the same computational domain. The quality of mesh associated with a certain variable can be improved from the mesh adaption process similar to the mesh adaption in the single-mesh method. In this study, the implementation of the multi-mesh method is based on the framework proposed in [21]. Within this framework, mesh adaptation is achieved by locally refining or coarsening the element, a process known as h-adaptation. A similar idea can be found in [14,30] where p-adaption of the finite element space is also considered. The multi-mesh method has been successfully applied to various fields and models, such as in simulating the two- and three-dimensional dendritic growth [17,12], the wetting and spreading problems [13], the photonic band structure optimization [36], the eigenvalue problems [16], etc. With the multi-mesh method, chemical accuracy is able to be achieved since both wavefunctions and the Hartree potential are solved on different but qualified spaces.

An anticipated advantage of the multi-mesh method, when compared to the single-mesh method, is the potential to reduce computational costs in achieving chemical accuracy. This can be explained by a sample as displayed in Fig. 1. The first mesh can be viewed as the mesh for solving the Kohn–Sham equation, with a dense grid distribution near the singularity and a sparse distribution for regions far from it. The middle mesh is used to solve the Hartree potential, and due to its smooth behavior and low decay speed, the mesh grid distribution remains uniform. Finally, the last mesh, formed without employing the multi-mesh strategy, should be able to capture both the singularity and the low decay of the Hartree potential. The number of mesh grids for these three meshes is 37, 49, and 57, respectively. Assume that the linear finite element method is adopted. In the multi-mesh method, a generalized eigenvalue problem $Ax = \lambda Bx$ with $A, B \in \mathbb{R}^{37\times 37}$ and a linear system $Sx = b$ with $S \in \mathbb{R}^{49\times 49}$. While for the finite element method using one mesh, the eigenvalue problem and the linear system are of the same size $A, B, S \in \mathbb{R}^{57\times 57}$. Apparently, the computational cost for solving these two problems in the multi-mesh method is much less than the single-mesh method in this sample.

Nevertheless, the application of the multi-mesh method is accompanied by several challenges. The first of these challenges is the effective management of mesh grids, requiring the capacity to perform local grid refinements or coarsening in response to specific problem demands. This management process must seamlessly facilitate the update of solutions from the previous mesh to the new one. The second challenge centers on the critical necessity for efficient and accurate multi-mesh communication, particularly when computing integrals. The establishment of smooth communication is indispensable for maintaining consistency and accuracy across different meshes. To address these challenges, we introduce the hierarchical geometry tree data structure. This tree-based approach facilitates the natural and efficient manner in the management of the mesh grids. Furthermore, in order to prevent any loss of

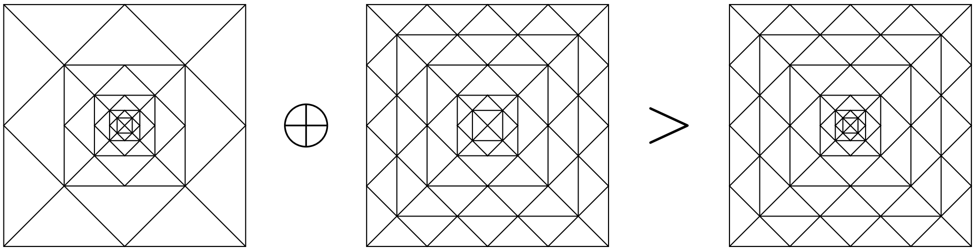

Fig. 1. A sample for multi-mesh method. The left two images show the meshes used in the multi-mesh method. The right one shows the mesh used in the single-mesh method. The number of the mesh grids for the three meshes are 37, 49, and 57, respectively.

accuracy during communications across various meshes, we adopt a strategy that maximizes the utilization of quadrature points in numerical integrals, relying solely on interpolation without the need for a projection process. This approach ensures the preservation of numerical accuracy throughout the communication process.

By employing the multi-mesh approach, we establish a multi-mesh adaptive finite element framework for the all-electron density functional theory. With the additional treat on the Hartree potential, systematical convergence of the total energy to chemical accuracy is observed and the computational cost is less compared with the single-mesh method given the same accuracy. The mesh adaption is based on the residual-based *a posteriori* error estimation of the Kohn–Sham equation and the Poisson equation whose solution is the Hartree potential. The ability to achieve chemical accuracy is verified by a series of numerical examples of atoms and molecules. By a detailed comparison of computational time between the single-mesh method and the multi-mesh method, the efficiency of the multi-mesh adaptive method is justified.

The organization of this paper is listed below. In Section 2, we introduce the Kohn–Sham equation, the Poisson equation, and the finite element discretizations of these two equations. Then in Section 3, the single-mesh adaptive finite element framework for Kohn–Sham density function theory is reviewed and tested on the Helium atom example. Followed by Section 4 in which the multi-mesh adaptive method is demonstrated and examined also on the Helium example. In Section 5, the efficiency and accuracy of the presented method are discussed in detail from numerical examples. Finally, this paper ends with the conclusion in Section 6.

2. The Kohn–Sham equation and finite element discretization

2.1. Kohn–Sham density functional theory

The Kohn–Sham equation for a p-electron system is read as the following eigenvalue problem

$$\begin{cases} \hat{H}\psi_l(\mathbf{r}) = \varepsilon_l \psi_l(\mathbf{r}), & l = 1, 2, \dots, p, \\ \displaystyle\int_{\mathbb{R}^3} \psi_l \psi_{l'} \, \mathrm{d}\mathbf{r} = \delta_{ll'}, & l, l' = 1, 2, \dots, p, \end{cases} \tag{2}$$

where (ε_l, ψ_l) is the l-th eigenpair, $\delta_{ll'}$ is the Kronecker delta, and $\hat{H}$ stands for the Hamiltonian operator. We denote $\rho(\mathbf{r}) = \sum_{l=1}^{p} |\psi_l|^2(\mathbf{r})$ as the electron density. Specifically, $\hat{H}$ consists of the following four terms

$$\hat{H}([\rho];\mathbf{r}) = -\frac{1}{2}\nabla_{\mathbf{r}}^2 + V_{\mathrm{ext}}(\mathbf{r}) + V_{\mathrm{Har}}([\rho];\mathbf{r}) + V_{\mathrm{xc}}([\rho];\mathbf{r}), \tag{3}$$

where the notation $V([\rho];\mathbf{r})$ implies that V is a functional of the electron density ρ. The first term $-\nabla^2/2$ in $\hat{H}$ is the kinetic operator. The second term in $\hat{H}$ describes the Coulomb external potential due to the nuclei which takes the form

$$V_{\mathrm{ext}}(\mathbf{r}) = -\sum_{j=1}^{M} \frac{Z_j}{|\mathbf{r} - \mathbf{R}_j|},$$

where M is the number of nuclei. The third term is the Hartree potential describing the Coulomb repulsion among the electrons

$$V_{\mathrm{Har}}([\rho];\mathbf{r}) = \int_{\mathbb{R}^3} \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} \, \mathrm{d}\mathbf{r}'. \tag{4}$$

The last term V_{xc} stands for the exchange-correlation potential, which is caused by the Pauli exclusion principle and other non-classical Coulomb interactions. Note that the analytical expression for the exchange-correlation term is unknown and therefore an approximation is needed. Specifically, the local density approximation (LDA) from the library Libxc [23] with the slater exchange potential and the Vosko-Wilk-Nusair (VWN4) [34] is adopted in this work.

Note that direct evaluation of the Hartree potential (4) requires computational cost $\mathcal{O}(N^2)$ with N being the number of grid points on the computational domain Ω. For simplicity, we denote $\phi = V_{\mathrm{Har}}(\mathbf{r})$ hereafter. In this paper, the Hartree potential is obtained by solving the Poisson equation

$$\begin{cases} -\nabla^2 \phi(\mathbf{r}) = 4\pi\rho(\mathbf{r}), \text{ in } \Omega, \\ \qquad \phi(\mathbf{r}) = \phi_{\partial\Omega}(\mathbf{r}), \text{ on } \partial\Omega. \end{cases} \tag{5}$$

The boundary value $\phi_{\partial\Omega}(\mathbf{r})$ is evaluated by the multipole expansion method. Specifically, the following approximation is used

$$\phi(\mathbf{r})|_{\mathbf{r}\in\partial\Omega} \approx \frac{1}{|\mathbf{r}-\mathbf{r}''|}\int\limits_{\Omega} \rho\left(\mathbf{r}'\right)\,\mathrm{d}\mathbf{r}' + \sum_{i=1,2,3} p_i \cdot \frac{r^i - r''^{,i}}{|\mathbf{r}-\mathbf{r}''|^3} + \sum_{i,j=1,2,3} q_{ij} \cdot \frac{3\left(r^i - r''^{,i}\right)\left(r^j - r''^{,j}\right) - \delta_{ij}\left|\mathbf{r}-\mathbf{r}''\right|^2}{|\mathbf{r}-\mathbf{r}''|^5},$$

where

$$p_i = \int\limits_{\Omega} \rho\left(\mathbf{r}'\right)\left(r'^{,i} - r''^{,i}\right)\mathrm{d}\mathbf{r}', \quad q_{ij} = \int\limits_{\Omega} \frac{1}{2}\rho\left(\mathbf{r}'\right)\left(r'^{,i} - r''^{,i}\right)\left(r'^{,j} - r''^{,j}\right)\mathrm{d}\mathbf{r}'.$$

In the above expressions, $\mathbf{r}''$ stands for an arbitrary point in Ω. In the simulations, we choose it to be

$$\mathbf{r}'' = \frac{\int \mathbf{r}\rho(\mathbf{r})\,d\mathbf{r}}{\int \rho(\mathbf{r})\,d\mathbf{r}}.$$

2.2. Finite element discretization

In practical simulations, a bounded polyhedral domain $\Omega \subset \mathbb{R}^3$ is served as the computational domain. Assume that the finite element space V_h is constructed on Ω and the finite element basis of V_h is denoted as $\{\varphi_1, \dots, \varphi_n\}$ with n being the dimension of the space. Then the wavefunction ψ_i can be approximated as ψ_i^h on V_h via

$$\psi_i^h(\mathbf{r}) = \sum_{k=1}^{n} X_{k,i}\varphi_k, \quad X_{k,i} = \psi_i\left(\mathbf{r}_k\right), \quad X \in \mathbb{R}^{n\times p}, \tag{6}$$

where $\mathbf{r}_k$ denotes the node corresponding to the i-th basis function. As a result, to find the approximation of the wavefunctions $\{\psi_i\}$ on V_h is to find X, i.e., the value of ψ_i at each node $\mathbf{r}_k$.

In the finite element space V_h, the variation form for the KS equation (2) becomes: find $(\psi_i^h, \varepsilon_i^h)_{i=1,\dots,p} \in V_h \times \mathbb{R}$, such that

$$\int\limits_{\Omega} \varphi \hat{H}\psi_i^h\,\mathrm{d}\mathbf{r} = \varepsilon_i^h \int\limits_{\Omega} \varphi\psi_i^h\,\mathrm{d}\mathbf{r}, \quad \forall\varphi \in V_h.$$

By letting φ be the finite element basis function and inserting (6) to the above variational form, we have the following discrete eigenvalue problem

$$AX = \varepsilon MY. \tag{7}$$

Here A and M are symmetric matrices with the entries

$$A_{i,j} = \int\limits_{\Omega} \frac{1}{2}\nabla\varphi_j \cdot \nabla\varphi_i + \left(V_{\mathrm{ext}} + \phi + V_{\mathrm{xc}}\right)\varphi_j\varphi_i\,\mathrm{d}\mathbf{r}, \tag{8}$$

$$M_{i,j} = \int\limits_{\Omega} \varphi_j\varphi_i\,\mathrm{d}\mathbf{r}. \tag{9}$$

Similarly, we can obtain the linear system for the Poisson equation (5) on V_h:

$$S\Phi = \mathbf{b}, \tag{10}$$

where S is the stiff matrix with the entry

$$S_{i,j} = \int\limits_{\Omega} \frac{1}{2}\nabla\varphi_j \cdot \nabla\varphi_i\,\mathrm{d}\mathbf{r}, \tag{11}$$

and the right hand side $\mathbf{b}$ and the discretized ϕ^h

$$\mathbf{b}_i = \int\limits_{\Omega} 4\pi\rho(\mathbf{r})\varphi_i\,\mathrm{d}\mathbf{r}, \quad \phi^h(\mathbf{r}) = \sum_{k=1}^{n} \Phi_k\varphi_k, \quad \Phi \in \mathbb{R}^n.$$

Owing to the singularity in the Hamiltonian (3), a uniform finite element discretization would lead to a large number of mesh grids to achieve chemical accuracy. Hence the adaptive mesh method is necessary to efficiently solve the Kohn–Sham equation, which will be discussed in the next section.

3. The adaptive finite element method

In this section, we will begin by introducing the adaptive finite element method that relies on the *a posteriori* error estimates [33]. Subsequently, we will present and compare two error indicators used to facilitate mesh adaptation for solving the Kohn–Sham equation (2). The sole distinction between these two error estimators is their utilization of the error involved in solving the Hartree potential.

3.1. The adaptive method based on the residual a posteriori *error estimation*

The adaptive mesh techniques offer enhanced numerical accuracy while requiring fewer mesh grids compared to uniform mesh methods. In this study, our primary focus lies on the h-adaptive methods, which involve local refinement and/or coarsening of mesh grids. An important aspect of h-adaptive methods is the determination of an error indicator. Generally speaking, error indicators identify regions within the domain that necessitate local refinement or coarsening, and they are typically derived from *a posteriori* error estimations [33]. When there is only one orbital in the system, it is natural to generate the indicator based on information from that specific orbital. However, when there are multiple orbitals in the system, generating the indicator solely from an individual orbital is no longer advisable. This is because every orbital in the system is expected to be well-resolved using the mesh grids after mesh adaptation. To address this, we adopt the strategy proposed in [2] for indicator generation. First, indicators are individually generated for each orbital using a specific method. Then, normalization is applied to each indicator. The final indicator is obtained by combining these normalized indicators.

Specifically, based on the *a posteriori* error estimates [33], the residual-based *a posteriori* error indicator for the KS equation (2) in the element $\mathcal{T}_K$ can be defined as

$$\eta_{K,\mathrm{KS}} = \left(h_K^2 \sum_{l=1}^{p} \left\| \mathcal{R}_{K,\mathrm{KS}}(\psi_l) \right\|_K^2 + \sum_{e\subset\partial\mathcal{T}_K} \frac{1}{2} h_e \sum_{l=1}^{p} \left\| \mathbb{J}_e(\psi_l) \right\|_e^2 \right)^{\frac{1}{2}}, \tag{12}$$

where h_K represents the largest length of the edges of the element $\mathcal{T}_K$, h_e stands for the largest length of the common face e of $\mathcal{T}_K$ and $\mathcal{T}_J$, $\mathcal{R}_{K,\mathrm{KS}}(\psi)$ and $\mathbb{J}_e(\psi)$ are the residual and jump term on the element $\mathcal{T}_K$, whose formulations are written as

$$\begin{cases} \mathcal{R}_{K,\mathrm{KS}}(\psi_l) = \hat{H}\psi_l - \varepsilon_l M \psi_l, \\ \mathbb{J}_e(\psi_l) = (\nabla\psi_l \Big|_{\mathcal{T}_K} - \nabla\psi_l \Big|_{\mathcal{T}_J}) \cdot \mathbf{n}_e, \end{cases}$$

where $\mathbf{n}_e$ is the out normal vector on the face e w.r.t. the element $\mathcal{T}_K$. The definition of the norms are

$$\|f(x)\|_K = \left(\int_K (f(x))^2 dx \right)^{\frac{1}{2}}, \quad \|f(x)\|_e = \left(\int_e (f(x))^2 dx \right)^{\frac{1}{2}}.$$

The indicator (12) involves the error arising from the Kohn–Sham equation (2) and is usually directly adopted to guide the mesh adaption.

An aspect that the indicator (12) possibly overlooks is the error associated with the Hartree potential. It is important to emphasize the accurate approximation of the Hartree potential, as it is a crucial component of the Hamiltonian (3) and contributes to the Hartree potential energy. As the Hartree potential is obtained by solving the Poisson equation (5), a similar approach can be employed to generate an error indicator specifically for the Hartree potential, just as done for the Kohn–Sham equation. Specifically, the error indicator for the Hartree potential can be defined as

$$\eta_{K,\mathrm{Har}} = \left(h_K^2 \left\| \mathcal{R}_{K,\mathrm{Har}}(\phi) \right\|_K^2 + \sum_{e\subset\partial\mathcal{T}_K} \frac{1}{2} h_e \left\| \mathbb{J}_e(\phi) \right\|_e^2 \right)^{\frac{1}{2}}, \tag{13}$$

where the residual part is defined as $\mathcal{R}_{K,\mathrm{Har}}(\phi) = \nabla^2\phi + 4\pi\rho$. Based on the analysis above, the second indicator for solving the Kohn–Sham equation can be designed as

$$\eta_{K,\mathrm{KS+Har}} = \sqrt{\eta_{K,\mathrm{Har}}^2 + \eta_{K,\mathrm{KS}}^2}. \tag{14}$$

The aim of using this error indicator is to generate a mesh on which both the wavefunctions and the Hartree potential are approximated well.

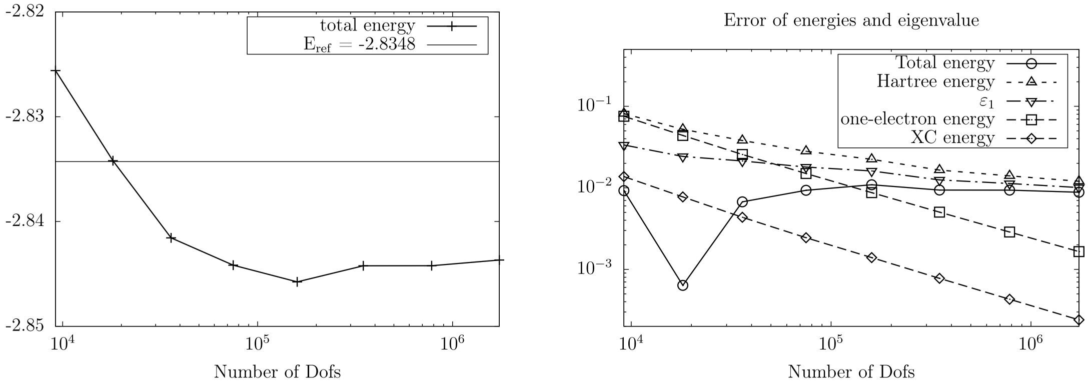

Fig. 2. Convergence history of the total energy (left), absolute errors of energies and eigenvalue (right) for the adaptive method using error indicator (12).

The mesh adaptation process, along with the solution of the Kohn–Sham equation, can be carried out using either the first (12) or the second error indicator (14). It is important to note that the adaptive algorithms utilizing these two different indicators are essentially identical, except for the choice of the indicator. Moreover, all simulations are performed on a single mesh simultaneously, i.e., the Poisson equation (5) for the Hartree potential is discretized and solved on the same finite element space for the Kohn–Sham equation. Consequently, we present the adaptive finite element method for solving the Kohn–Sham equation, outlined in Algorithm 1, and refer to this approach as the *single-mesh adaptive method*. Briefly, the adaptive algorithm consists of an outer iteration and an inner iteration. In the inner iteration, it is an SCF method for solving the nonlinear Kohn–Sham equation. In this work, we use a simple mixing scheme to update the electron density, and the mixing parameter is chosen as $\alpha = 0.618$. Meanwhile, it is the mesh adaption procedure in the outer iteration. In the next subsection, we would like to show an example using the single mesh adaptive method with the first indicator (12).

Algorithm 1: Single-mesh adaptive method.

```
Input: Initial mesh $\mathcal{T}^{(0)}$, initial electron density $\rho$, initial energy $E = 0$, energy tolerance $tol_1$, density tolerance $tol_2$.
while $|E - E_{old}| < tol_1$ do
    $E_{old} = E$;
    while $\|\rho - \rho_{old}\| < tol_2$ do
        $\rho_{old} = \rho$;
        Calculate the Hartree potential $\phi$;
        Generate the Hamiltonian matrix $A$;
        Solve the eigenvalue problem $AX = \varepsilon MX$;
        Update the electron density $\rho$;
    Calculate the ground state energy $E$;
    Mesh adaption based on the error indicator.
Output: Total ground state energy $E$
```

3.2. An issue for numerical solutions towards chemical accuracy

3.2.1. Single-mesh adaptive method using indicator (12)

To assess the convergence and behavior of the single mesh adaptive finite element method with the error indicator (12), we conduct a series of eight experiments for the helium atom, each involving a different number of mesh grids, controlled by varying adaption tolerance values. In this example, the referenced value is obtained from the state-of-art software `NWChem` using the *aug-cc-pv6z* basis set [32]. The hardware and software configurations of the experiment are introduced in the beginning of Section 5.

The results are displayed in the Fig. 2. As the tolerance for error indicator decreases, the number of degrees of freedom increases. The convergence of the total energy would serve as evidence of the superiority of the adaptive method compared to the method employing a uniform mesh partition. However, it is evident that the total energy does not converge to the reference result observed from the left of Fig. 2. Even in the mesh generated from the smallest tolerance, which consists of 1,737,569 mesh grids, a discrepancy of 0.01 Hartree in the energy is still observed, indicating that the adaptive method does not achieve the desired convergence.

As discussed earlier, the error in the total energy primarily stems from the inaccurate approximation of the Hartree energy. This is confirmed by comparing the Hartree energy with the reference value $E_{\text{Har,ref}} = 1.9961$ Hartree, as depicted on the right side of Fig. 2. It is observed that although the error in the Hartree energy diminishes with respect to the number of Dofs, on the finest mesh, the Hartree energy is found to be 0.01 Hartree lower than the reference value, resulting in a 0.01 Hartree error in the total energy. Furthermore, an imprecise approximation of the Hartree potential can also introduce a 0.01 Hartree error in the eigenvalue, as depicted on the right side of Fig. 2. As a comparison, it can be observed that the errors in the one-electron energy and the exchange-correlation energy systematically decrease as the number of Dofs increases.

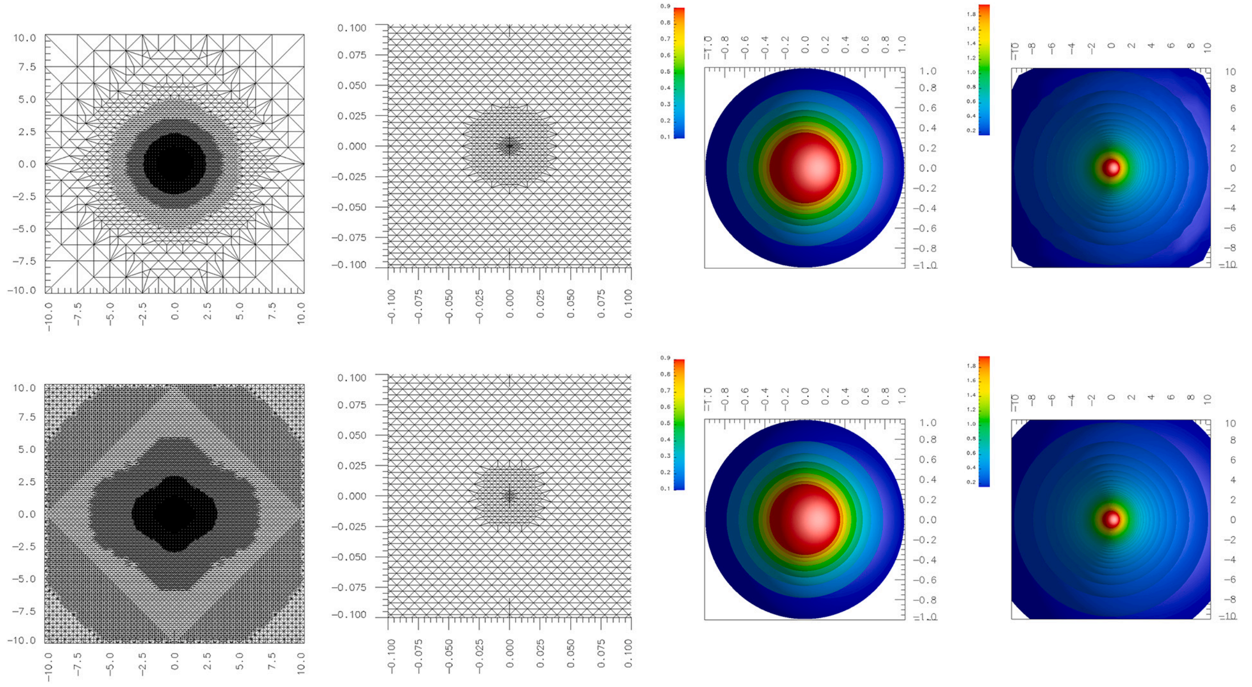

Fig. 3. Top row: results from a single mesh adaptive method using the indicator (12), with total 1,765,990 mesh grids, i.e., Global mesh and zoomed-in mesh on the sliced X-Y plane (left two), density profile (third one), and Hartree potential profile (fourth one). Bottom row: corresponding results using the indicator (14), with total 2,847,807 mesh grids. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

The reason behind the inaccurate approximation of the Hartree potential can be elucidated by referring to Fig. 3 (upper row), which displays the results obtained from the finest mesh consisting of 1,765,990 mesh grids. In the left two columns, the global and local mesh distributions are depicted. It is evident that the mesh grid density is considerably high around the singularity located at the origin, with the smallest mesh size being approximately 0.001 Bohr. Conversely, in regions distant from the singularity, the mesh grid distribution is sparse. Specifically, near the boundary, the largest mesh size can exceed 2 Bohr. Such a mesh distribution proves to be suitable for representing wavefunctions and electron density, as illustrated in the third column of Fig. 3 (upper row), since both quantities exhibit exponential decay, and the region where the density exceeds 0.1 is confined to a small interval of $[-1,1]^3$ for the helium atom. However, in the fourth column of Fig. 3 (upper row), it is apparent that the Hartree potential exhibits a much slower decay behavior, with potential values surpassing 0.1 throughout the entire computational domain of $[-10,10]^3$. Furthermore, the contours near the boundary oscillate and are not smooth. As a result, a mesh size as large as 2 Bohr is evidently insufficient for capturing the variations in the Hartree potential accurately. In order to enhance the numerical accuracy of the Kohn–Sham equation, it is essential to obtain a more precise approximation of the Hartree potential.

3.2.2. *Single mesh adaptive method using indicator* (14)

To achieve a more accurate Hartree potential, a direct approach is to incorporate the corresponding error indicator (13) with the indicator of the Kohn–Sham equation (12). This combination results in the second indicator (14). Consequently, we apply the single mesh adaptive method again to the Helium example, but this time utilizing the second indicator (14).

The convergence of energies and eigenvalues is depicted in Fig. 4. Notably, there is a systematic convergence observed in both the total energy and the Hartree potential energy. Moreover, the eigenvalue demonstrates convergence towards the reference value. On the finest mesh, which consists of 2,874,807 grid points, chemical accuracy is achieved. Sliced mesh representations and profiles of the electron density and Hartree potential are shown in Fig. 3 (lower row). A noticeable difference from the mesh in Fig. 3 (upper row) is the presence of smaller mesh sizes in regions far away from the origin. As a result, the Hartree potential is better captured, as illustrated in the fourth column of Fig. 3 (lower row), from which we can find smoother contour lines than that in Fig. 3 (upper row) using the first indicator (12).

While the Hartree potential is approximated accurately and chemical accuracy is obtained, the computational cost increases significantly. The number of mesh grids on the finest mesh reaches 2,874,807, surpassing the number of grids used in the previous section. This increment in the number of mesh grids is primarily caused by the need to capture the variations of the Hartree potential. Such an increase in the computational grid introduces additional complexity and computational overhead, leading to longer computation time and higher memory requirements. This becomes especially problematic when dealing with larger system sizes or when conducting complex simulations. To address this issue and reduce the computational cost, an alternative approach, known as the multi-mesh adaptive method, will be introduced in the next section.

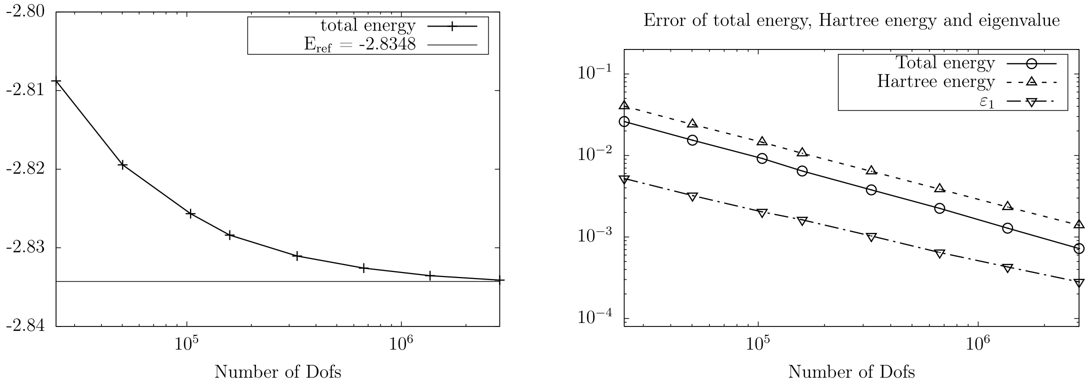

Fig. 4. Convergence history of the total energy (left), absolute errors of energies and eigenvalue (right) for the adaptive method using error indicator (14).

4. The multi-mesh adaptive method

In the preceding section, we explored the solution of the discretized Kohn–Sham equation on a single mesh, employing either the residual-based *a posteriori* error indicator (12) or (14). The results indicated that employing the first indicator (12) alone is possibly fall short of achieving chemical accuracy. On the other hand, the second indicator (14) demonstrated the ability to attain chemical accuracy; however, it came at the cost of significantly increased computational requirements.

To tackle this challenge, we introduce an approach known as the multi-mesh adaptive method [21]. The primary objective of this method is to strike a balance between achieving chemical accuracy and managing computational costs effectively. By utilizing multiple meshes instead of a single mesh, the multi-mesh adaptive method offers greater flexibility in adapting the mesh resolution to capture variations for different quantities of interest.

Specifically, in this method, we will utilize two distinct meshes. Both two meshes will be adapted during the simulation. The first mesh is specifically designed for solving the Kohn–Sham equation, with the primary objective of accurately capturing the variations in the wavefunctions. This mesh is tailored to ensure high resolution in regions where the wavefunctions exhibit significant changes. Conversely, the second mesh is dedicated to solving the Poisson equation (5). Its primary purpose is to capture the variations in the Hartree potential effectively. This second mesh is strategically designed to provide optimal resolution and precision in regions where the Hartree potential exhibits substantial changes. By employing two separate meshes with specific focus areas, we can ensure that each equation is solved with the appropriate level of accuracy and capture the variations unique to the wavefunctions and the Hartree potential, respectively.

To implement the multi-mesh method, careful handling of two components is crucial. The first component involves effectively managing the mesh grids, allowing for flexible local refinement or coarsening as needed. This management mechanism should also facilitate efficient solution updates from the old mesh to the new mesh. The second component focuses on ensuring efficient and accurate communication between different meshes, particularly in the calculation of integrals. Efficient communication protocols play a vital role in maintaining consistency and accuracy across the various meshes. These requirements can be fulfilled by the hierarchical geometry tree data structure, which will be introduced in detail in the following.

4.1. The hierarchical geometry tree

A well-designed data structure for the mesh grids is needed for an effective management mechanism. In the presented algorithm, the hierarchical geometry tree (HGT) [21,2,7] is utilized.

Firstly, the mesh structure is described hierarchically, which means that the mesh information is given from the lowest dimension (0-D, the points) to the highest dimension (3-D, the tetrahedron) hierarchically. An element such as a point for 0-D, an edge for 1-D, a triangle for 2-D, a tetrahedron for 3-D is called a geometry. In the hierarchical description of a tetrahedron, all geometries have a belonging-to relationship. For example, if an edge is one of the edges of a triangle, this edge belongs to this triangle. With this hierarchical structure, the geometry information of a tetrahedron can be referred to flexibly, and the refinement and coarsening of a mesh can also be implemented efficiently.

Secondly, the mesh is stored and managed by a tree data structure. The validity of using the tree data structure is due to the strategy of element refinement and coarsening. Specifically, for a tetrahedron element (the left of Fig. 5), the refinement of this tetrahedron is dividing it into eight equally small tetrahedrons via connecting the midpoints on each edge (the right of Fig. 5). As a result, a belonging-to relationship can be established, for example, any small tetrahedron that is called “child” belongs to the large “parent” tetrahedron. Meanwhile, the coarsening of any child tetrahedron in the right of Fig. 5 is releasing all the children to obtain the parent tetrahedron. By this refinement and coarsening strategy, the tree data structure is able to be established.

For a better understanding of the tree data structure, a two-dimensional illustration is presented. Similar to the 3D case, the refinement of a 2D element, i.e. the triangle, is to divide the triangle into four equal triangles, as displayed in the left two columns in Fig. 6. By refining a triangle $\mathcal{T}_0$, four sub-triangles $\{\mathcal{T}_{0,0}, \mathcal{T}_{0,1}, \mathcal{T}_{0,2}, \mathcal{T}_{0,3}\}$ are generated. Furthermore, via refining $\mathcal{T}_{0,0}$ the sub-triangles

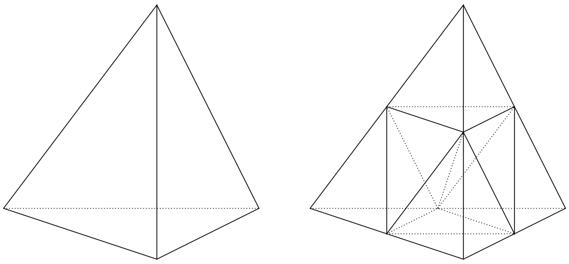

Fig. 5. The parent tetrahedron (left) and the eight child tetrahedrons (right).

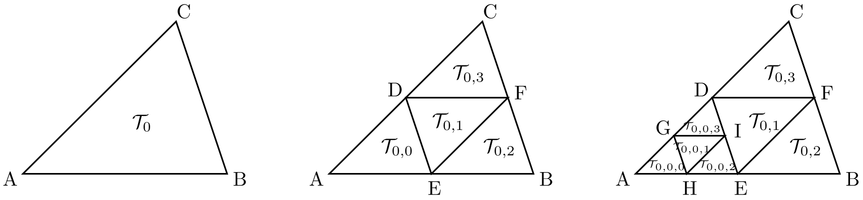

Fig. 6. The refinement of a triangle $\mathcal{T}_0$ and its sub-triangle $\mathcal{T}_{0,0}$.

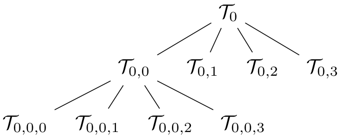

Fig. 7. The quadtree data structure for the mesh in the right of Fig. 6.

$\{\mathcal{T}_{0,0,0}, \mathcal{T}_{0,0,1}, \mathcal{T}_{0,0,2}, \mathcal{T}_{0,0,3}\}$ are obtained, which is demonstrated in Fig. 6. To manage this procedure, the quadtree data structure in which each internal node has exactly four children is utilized, as described in Fig. 7. When the local refinement and coarsening techniques are adopted, only some triangles are to refined, as indicated in the right of Fig. 6 and the bottom of Fig. 7 where only the triangle $\mathcal{T}_{0,0}$ is refined. In the quadtree, we call $\mathcal{T}_0$ the root node, and those nodes without further descents like $\mathcal{T}_{0,1}$ and $\mathcal{T}_{0,0,1}$ the leaf nodes. Suppose that there is a set of root nodes $\{\mathcal{T}_i\}, i = 0, 1, 2, \ldots$, which form a mesh for a domain Ω, then from the above definition we know that a set of all leaf nodes also form a mesh. For example, the set $\{\mathcal{T}_0\}$ forms the mesh in the left of Fig. 6, the set $\{\mathcal{T}_{0,0}, \mathcal{T}_{0,1}, \mathcal{T}_{0,2}, \mathcal{T}_{0,3}\}$ forms the middle mesh in Fig. 6, and the set $\{\mathcal{T}_{0,0,0}, \mathcal{T}_{0,0,1}, \mathcal{T}_{0,0,2}, \mathcal{T}_{0,0,3}, \mathcal{T}_{0,1}, \mathcal{T}_{0,2}, \mathcal{T}_{0,3}\}$ forms the right mesh in Fig. 6.

With the hierarchical description of the geometry and the tree data structure, the mesh is effectively managed by the HGT. However, building the finite element space directly on a mesh results in non-conforming finite element because of the hanging points in the direct neighbors of those refined triangles. These hanging points can be handled mainly in two ways. If there are more than one hanging point on a triangle, this triangle will be refined. This process is called the semi-regularization procedure [21]. If there exists only one hanging point on a triangle, for example, the triangle $\triangle DEF$ in the right of Fig. 6 has a hanging point I arose in the refinement of $\mathcal{T}_{0,0}$. To deal with these kind of hanging points, the twin-triangle geometry is introduced, which is demonstrated in Fig. 8. The twin-triangle actually consists of two triangles $\triangle FDI$ and $\triangle FIE$. Meanwhile, the degrees of freedom and the interpolation information of the twin-triangle $\triangle DEF$ inherit from the two triangles $\triangle FDI$ and $\triangle FIE$. To make the finite element space conforming, the following strategy is adopted to construct the basis functions in the twin-triangle geometry: For each basis function, its value is 1 at its corresponding interpolation point and 0 at other interpolation points. The support of the basis function whose interpolation point is a common point of two sub-triangles in a twin-triangle geometry such as F and I in Fig. 8, is the whole twin-triangle geometry. For the non-common points like D, the support of its basis function is only the triangle $\triangle FDI$. With such a strategy, a conforming finite element space can be built in a mesh where the local refinement is implemented.

In virtue of the tree data structure, multiple meshes are allowed to be described by the same HGT. Naturally, a problem arises: is it possible to make the information communicates among these meshes without any loss? The answer is affirmative, and the reason relies on the belonging-to relationship between any two nodes in the HGT. Briefly speaking, there exist only three kinds of relationship between any two nodes: equal, belonging-to, and no-overlap. The communication among equal or no-overlap elements is trivial. Consequently, we only need to take care of the second kind of relationship. We introduce the implementation details in the following.

In the presented method, the Hamiltonian matrix and energy require information from both the KS mesh and the Hartree mesh, which mainly involve numerical integration. We take the evaluation of the Hartree potential energy for example, which could be written as

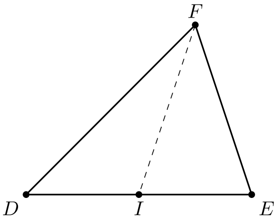

Fig. 8. Twin-triangle.

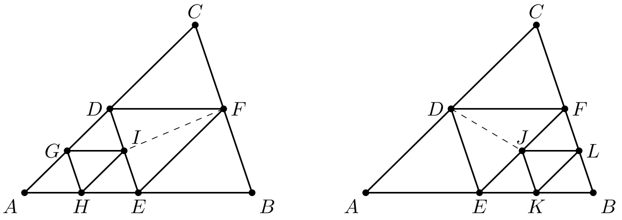

Fig. 9. Left: $\mathcal{T}_{\text{KS}}$. Right: $\mathcal{T}_{\text{Har}}$.

$$E_{\text{Har}} = \frac{1}{2}\int\limits_{\Omega} \phi(\mathbf{r})\left(\sum_{l=1}^{p}\psi_l(\mathbf{r})^2\right)\mathrm{d}\mathbf{r}. \tag{15}$$

The integration should be carefully treated since the Hartree potential $\phi(\mathbf{r})$ and the wavefunction $\psi_l(\mathbf{r})$ belong to different finite element spaces built on different meshes. An intuitive illustration is given to show how the presented method calculates the integral (15). Assume the finite element space $V_{\mathcal{T}_{\text{KS}}}$ for the wavefunction is built on the mesh $\mathcal{T}_{\text{KS}}$, and the space $V_{\mathcal{T}_{\text{Har}}}$ is built on the mesh $V_{\mathcal{T}_{\text{Har}}}$, see Fig. 9. For the common elements such as $\triangle CDF$ which belongs to both $\mathcal{T}_{\text{KS}}$ and $\mathcal{T}_{\text{Har}}$, the numerical integration can be calculated directly. While for the remained elements, for example, in $\mathcal{T}_{\text{KS}}$, the triangle $\triangle DAE$ is refined, while in $\mathcal{T}_{\text{Har}}$ this triangle is kept, and a similar case for the triangle $\triangle FEB$, special treatment is needed to avoid the loss of accuracy.

In order to prevent the loss of accuracy, we employ a strategy that maximizes the utilization of quadrature points in numerical integrals. Specifically, the numerical integration on the element $\triangle ADE$ is divided into the integrations on its four refined sub-triangles,

$$\frac{1}{2}\int\limits_{\triangle ADE} \phi(\mathbf{r})\left(\sum_{l=1}^{p}\psi_l(\mathbf{f})^2\right)\mathrm{d}\mathbf{r} \approx \sum_{K=1}^{4}\sum_{j=1}^{q}\text{area}(K)J_j^K w_j^K \phi(\mathbf{r}_j^K)\left(\sum_{l=1}^{p}\psi_l(\mathbf{r}_j^K)^2\right),$$

where the element K represents the four sub-triangles of $\triangle ADE$, $\mathbf{r}_j^K$ is the j-th quadrature point of K, J_j^K is the jacobian at $\mathbf{r}_j^K$, and w_l^K is the associated weight in the numerical quadrature. The values of the Hartree potential on these quadrature points can be obtained by numerical interpolation. Similarly, the integral on the triangle $\triangle FEB$ is evaluated by doing the numerical integral on its sub-triangles. In this way, the accuracy of the integral will not be affected in the communication of different meshes.

With the HGT, the solution update from the old mesh to the new mesh after the local refinement and coarsening can also be implemented efficiently. As we mentioned before, a set of all leaf nodes forms a mesh. Although different meshes correspond to different sets of leaf nodes, all meshes are from the same set of root nodes. Then the belonging-to relationship of the geometries between two meshes can be analyzed easily. As a result, the solution update can be implemented efficiently according to the relationship.

4.2. The multi-mesh adaptive algorithm

In virtue of the HGT data structure, the multi-mesh method can be efficiently implemented. To solve the Kohn–Sham equation, we propose the multi-mesh adaptive algorithm illustrated in the flowchart presented in Fig. 10. The Kohn–Sham equation (2) is discretized and solved on the finite element space built on the mesh $\mathcal{T}_{\text{KS}}$, and the Hartree potential (5) is solved on the mesh $\mathcal{T}_{\text{Har}}$. The KS mesh $\mathcal{T}_{\text{KS}}$ is adapted using the error indicator (12) and the Har mesh $\mathcal{T}_{\text{Har}}$ is adapted using the error indicator (13). Both of these two mesh adaptions occur after the end of the SCF iteration. The mesh adaptations continue until the final total energy reaches convergence, at which point the mesh adaptation ceases, and the results are outputted.

In order to assess the efficiency and make a comparison between the multi-mesh algorithm Fig. 10 and the single mesh algorithm Algorithm 1, we categorize the calculations into three distinct types. Namely, the calculations on the Kohn–Sham mesh $\mathcal{T}_{\text{KS}}$ with a white background in Fig. 10, the calculations conducted in the Hartree mesh $\mathcal{T}_{\text{Har}}$ with a blue background, and the calculations that require information from both $\mathcal{T}_{\text{KS}}$ and $\mathcal{T}_{\text{Har}}$ with a red background. When comparing the multi-mesh algorithm to the single mesh algorithm, it becomes evident that additional calculations fall into the third category. These calculations necessitate communication between the Kohn–Sham mesh and the Hartree mesh. Specifically, such communication is required during the generation of the

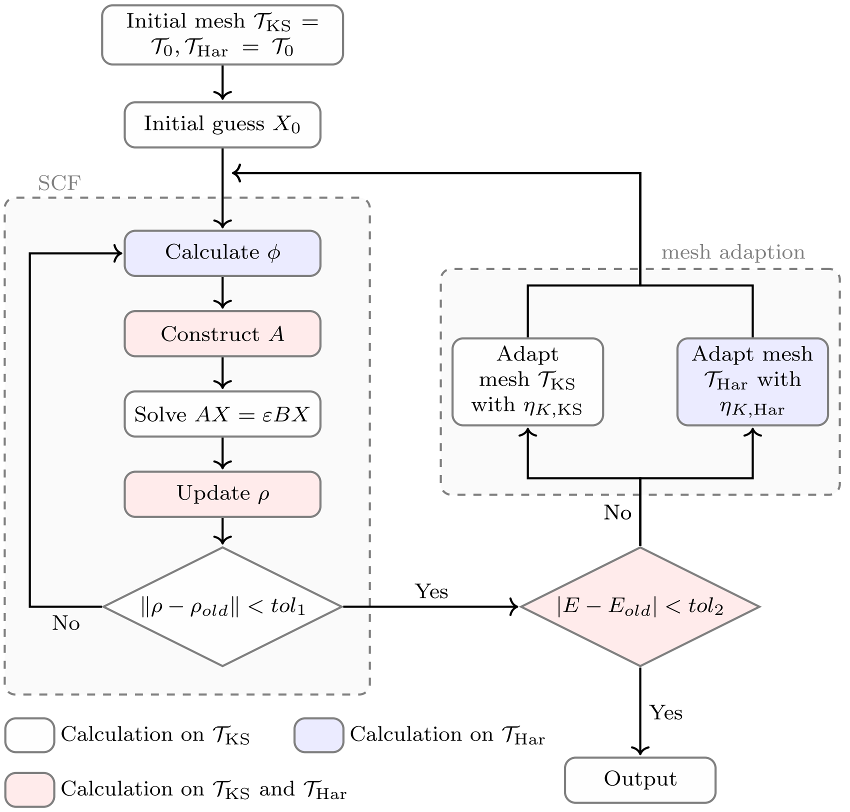
Initial mesh $\mathcal{T}_{\text{KS}} = \mathcal{T}_0, \mathcal{T}_{\text{Har}} = \mathcal{T}_0$
Initial guess X_0
SCF
Calculate ϕ
Construct A
Solve $AX = \varepsilon BX$
Update ρ
$\|\rho - \rho_{old}\| < tol_1$
No
Yes
mesh adaption
Adapt mesh $\mathcal{T}_{\text{KS}}$ with $\eta_{K,\text{KS}}$
Adapt mesh $\mathcal{T}_{\text{Har}}$ with $\eta_{K,\text{Har}}$
No
$|E - E_{old}| < tol_2$
Yes
Output
Calculation on $\mathcal{T}_{\text{KS}}$
Calculation on $\mathcal{T}_{\text{Har}}$
Calculation on $\mathcal{T}_{\text{KS}}$ and $\mathcal{T}_{\text{Har}}$

Fig. 10. Flowchart of the multi-mesh adaptive algorithm for the KS equation.

Hamiltonian, the updating of the electron density (which forms the right-hand side of the Poisson equation (5) for the Hartree potential), and the computation of the total energy. We will conduct a detailed comparison of the time consumed by these individual parts to evaluate the overall performance of the algorithms in the next section.

By employing the multi-mesh method and incorporating adaptive algorithms guided by appropriate error indicators, we can achieve accurate and efficient solutions for the Kohn–Sham equation. The flexibility of mesh adaptation ensures that the mesh resolution aligns with the specific requirements of each equation, ultimately leading to improved convergence and reliable results. This can be verified by the Helium example.

5. Numerical experiments

In this section, we examine the convergence and efficiency of the multi-mesh adaptive method through a series of numerical examples. All the simulations are performed on a workstation “Moss” with two AMD EPYC 7713 64-Core Processors (at 2.0 GHz×64, 512M cache) and 900 GB of RAM, and the total number of cores is 128. The software is the C++ library `AFEABIC` [2–4] under Ubuntu 20.04.

5.1. Case study: for chemical accuracy and computational efficiency

The multi-mesh adaptive method Fig. 10 is first tested for three cases: the helium atom, the LiH molecule, and the BeH_2 molecule. Our examination unfolds in three phases: firstly, we assess the method’s systematic convergence in the context of the helium atom example; next, we determine whether it can attain chemical accuracy across all three cases; and lastly, we undertake a comparative analysis of the computational costs required to achieve chemical accuracy between the multi-mesh method and the single-mesh method.

5.1.1. Helium atom

Similar to the adaptive method using error indicator (14), the systematical convergence of the energies and eigenvalue are observed in Fig. 11, and the chemical accuracy is obtained when the number of degrees of freedom in $\mathcal{T}_{\text{KS}}$ achieves 1,262,599, which is far less than that requires in the single-mesh algorithm. Furthermore, there are 2,185,920 degrees of freedom in the Hartree mesh $\mathcal{T}_{\text{KS}}$, which is also less than the number of mesh grids used to achieve the chemical accuracy using the single mesh adaptive method.

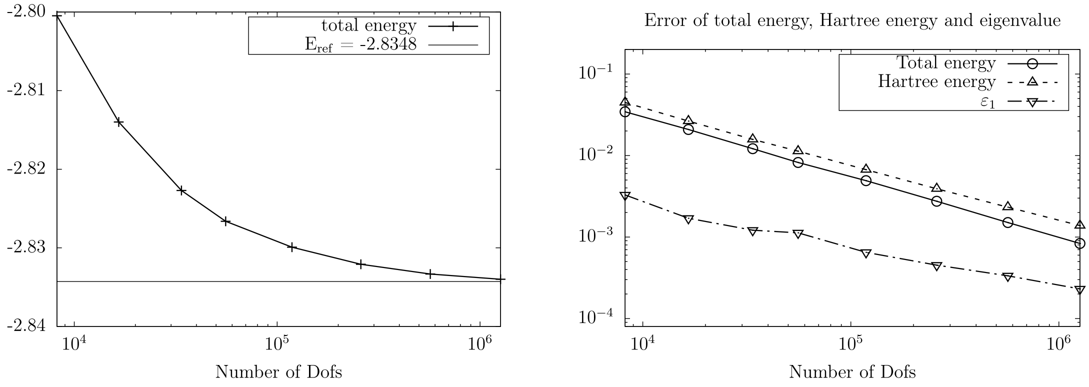

Fig. 11. Convergence history of the total energy (left), absolute errors of energies, and eigenvalue (right) for the multi-mesh adaptive method.

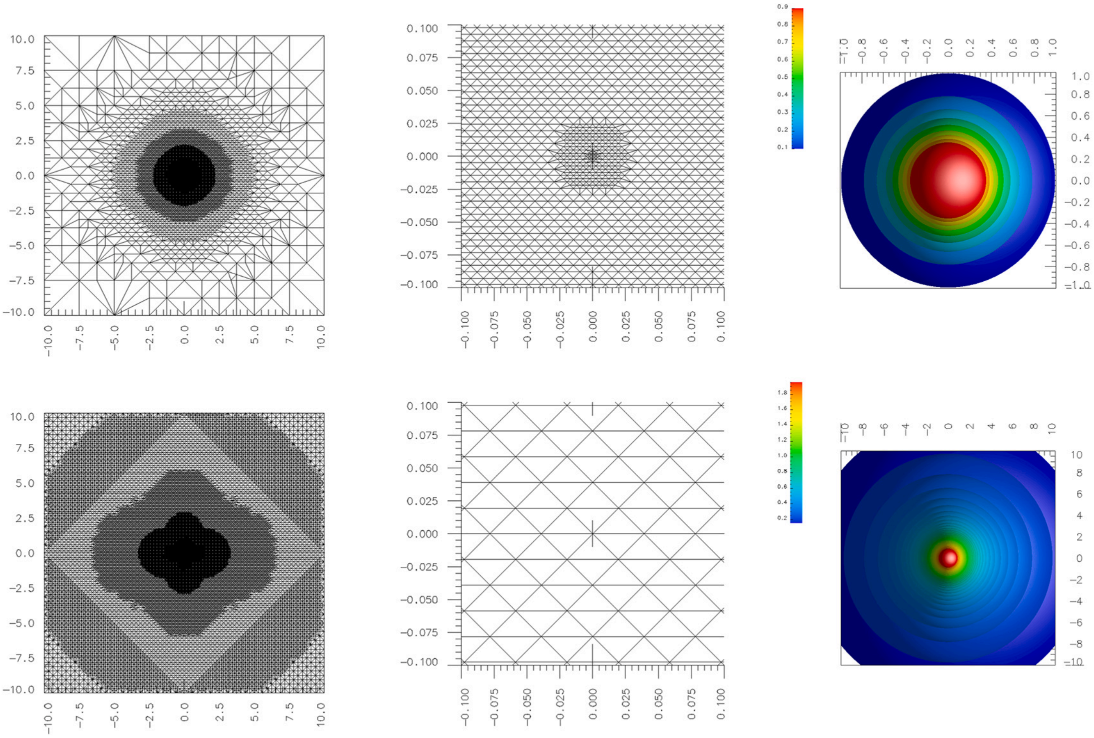

Fig. 12. Global and zoomed-in meshes on the sliced X-Y plane (top: $\mathcal{T}_{\mathrm{KS}}$ (1,2262,599 mesh grids); bottom: $\mathcal{T}_{\mathrm{Har}}$ (2,185,920 mesh grids)), density profile (top right), and Hartree potential profile (bottom right) for the multi-mesh adaptive method.

The final meshes are displayed in Fig. 12. In the top of Fig. 12 is the sliced mesh of $\mathcal{T}_{\mathrm{KS}}$ which is quite similar with Fig. 3 (upper row). The meshes of $\mathcal{T}_{\mathrm{Har}}$ are demonstrated in the bottom of Fig. 12, which has a similar mesh grid distribution as that in Fig. 3 (lower row) as shown in a global vision, while it has a sparser grid distribution comparing with Fig. 3 (lower row) since the Hartree potential behaves less singular than the Kohn–Sham wavefunctions. Although there is a larger number of mesh grids in $\mathcal{T}_{\mathrm{Har}}$ than in $\mathcal{T}_{\mathrm{KS}}$, it should be mentioned that on $\mathcal{T}_{\mathrm{Har}}$ we only need to solve a linear system, while in $\mathcal{T}_{\mathrm{KS}}$, a nonlinear eigenvalue problem should be solved. Consequently, the major computational cost is from the calculations on $\mathcal{T}_{\mathrm{KS}}$, and the increment of the computational cost due to the large number of mesh grids on $\mathcal{T}_{\mathrm{Har}}$ is slight. As a result, the computational cost of the multi-mesh method is of the same order of magnitude of the adaptive method, while it can achieve chemical accuracy.

As discussed in previous sections, both the single-mesh and multi-mesh methods are able to achieve chemical accuracy when the Hartree potential is considered in constructing the error indicator. To demonstrate the efficiency of the multi-mesh method, we compare the serial time for these two methods to achieve chemical accuracy. For the sake of fairness, we start the two algorithms from the same mesh and initial guess, and the tolerance for guiding the mesh adaption is also the same. With these settings, it is found that the convergence in SCF is quite similar and the number of mesh adaptions is the same, as displayed in Fig. 13. From the

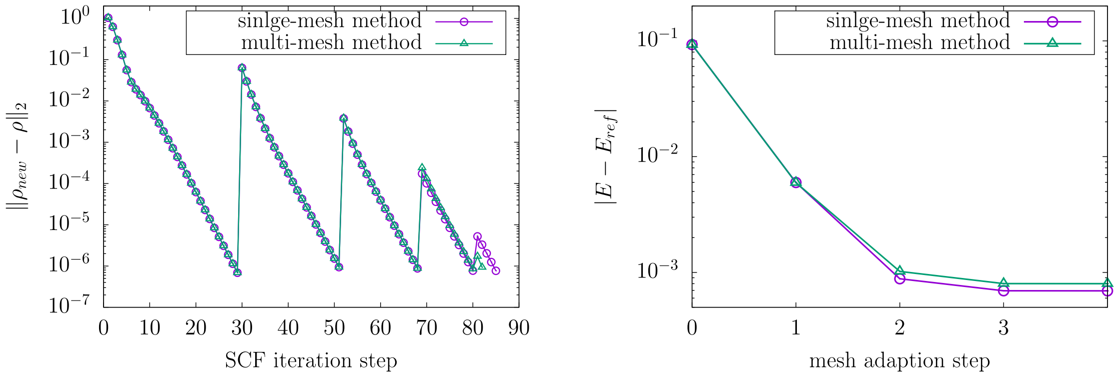

Fig. 13. SCF convergence history (left) and mesh adaption history (right) for Helium.

Table 1
Comparison on He with respect to results and serial computational time. t_{all} represents the total CPU time. t_{SCF} stands for the time for SCF and t_{MA} represents the time for mesh adaption.

	E_{tot}	ΔE	N_{KS}	N_{Har}	t_{all}	t_{SCF}	t_{MA}
single-mesh	-2.8341	0.0007	2,874,807	2,874,807	10290.45	7585.62	2704.83
multi-mesh	-2.8340	0.0008	1,262,599	2,185,920	7063.97	5135.69	1928.28

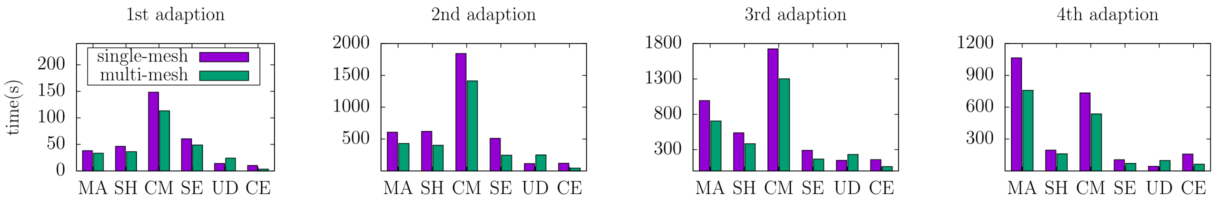

Fig. 14. CPU time for single-mesh method and multi-mesh method on six parts: **MA** (mesh adaption), **SH** (solve Hartree potential), **CM** (construct matrix), **SE** (solve eigenvalue problem), **UD** (update electron density) and **CE** (calculate energy). Four mesh adaptations are needed to achieve chemical accuracy in both methods starting from the same initial setup.

Table 2
Comparison on LiH with respect to results and serial computational time. The referenced total energy is $E_{\text{ref}} = -7.9195$ Hartree.

	E_{tot}	$\Delta E/2$	N_{KS}	N_{Har}	t_{all}	t_{SCF}	t_{MA}
single-mesh	-7.9178	0.0008	6,029,109	6,029,109	18613.60	13279.79	5333.82
multi-mesh	-7.9181	0.0007	3,786,064	3,665,015	14025.13	10245.13	4262.25

figure, it suggests that the introduction of Hartree mesh will not affect the convergence of the SCF procedure in KS mesh. Therefore, the comparison of the CPU time is fair and effective.

A summary of the comparison is illustrated in Table 1. Both methods achieved chemical accuracy, and the number of mesh grids in the single-mesh method is the largest among all meshes. The CPU time is compared with respect to two parts: the SCF part and the mesh adaption part. From Table 1, the multi-mesh method is proved to be more efficient than the single-mesh method in achieving the chemical accuracy of the helium atom.

A more detailed comparison is presented in Fig. 14. The method can be divided into six parts as indicated in Fig. 14. The four images represent CPU time on four meshes during the mesh adaption process. Obviously, the cost for solving the Hartree potential and the eigenvalue value problem is less in the multi-mesh method, as expected. Furthermore, the cost of constructing a matrix in the multi-mesh method is also less than that in the single-mesh method on the finest mesh. It is noted that this part contributes the largest portion of the total cost since in the helium example the eigensolver only needs to solve one eigenpair. In larger systems, solving the eigenvalue problem will be the most time-consuming part.

5.1.2. LiH molecule

A similar comparison for the LiH molecule is employed. The referenced total energy is $E_{\text{ref}} = -7.9195$ Hartree. The results are displayed in Table 2, revealing the attainment of chemical accuracy. From this table, the multi-mesh approach demonstrates its efficiency compared to the single-mesh method.

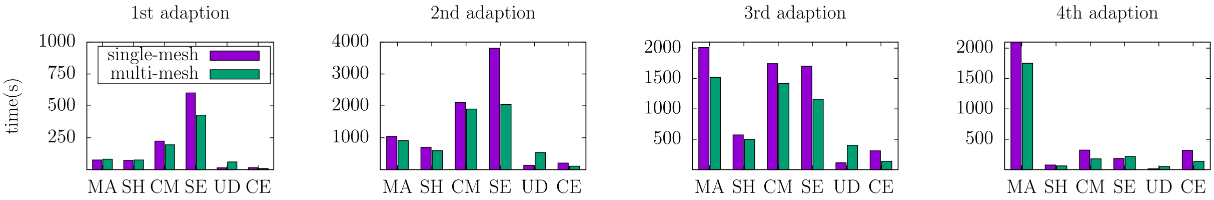

Fig. 15. CPU time for single-mesh method and multi-mesh method for LiH molecule. Four mesh adaptations are needed to achieve chemical accuracy in both methods starting from the same initial setup.

Table 3
Comparison on BeH_2 with respect to results and serial computational time. The referenced value is $E_{\text{ref}} = -15.6606$ Hartree.

	E_{tot}	$\Delta E/3$	N_{KS}	N_{Har}	t_{all}	t_{SCF}	t_{MA}
single-mesh	-15.6573	0.0011	7,338,638	7,338,638	42445.26	32331.87	10113.40
multi-mesh	-15.6580	0.0009	4,852,528	4,276,863	32133.92	23925.93	8207.99

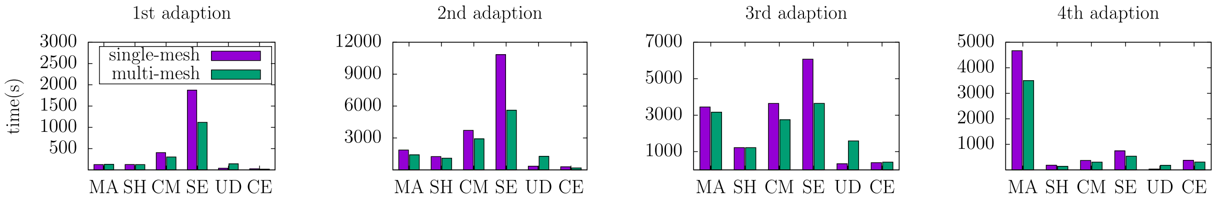

Fig. 16. CPU time for single-mesh method and multi-mesh method for BeH_2 molecule. Four mesh adaptations are needed to achieve chemical accuracy in both methods starting from a same initial setup.

The detailed comparison regarding CPU time is illustrated in Fig. 15. As expected, in the LiH example, the solution of the eigenvalue problem significantly contributes to the CPU time due to the increased number of required eigenvalues. Notably, the cost of the third mesh is higher than that of the final mesh. This is because, during the third mesh adaptation step, the mesh tends to stabilize with minimal changes, leading to fewer iteration steps in the final mesh, as shown in Fig. 13. Consequently, the CPU time on the third mesh becomes the most significant part contributing to the total CPU time. Moreover, since the number of Dofs in the multi-mesh method is smaller than that in the single-mesh method, the SCF time for the multi-mesh approach is also reduced. Overall, the multi-mesh method proves to be more efficient in terms of computational time than the single-mesh method in this case.

5.1.3. BeH_2 molecule

The comparison between the multi-mesh method and the single-mesh method is also presented in the BeH_2 example, as displayed in Table 3 and Fig. 16. A similar conclusion to the previous examples on the accuracy and efficiency can be delivered from the results. In this case, given the same mesh adaption tolerance, the multi-mesh method succeeds in achieving the chemical accuracy, while the single-mesh method fails. This observation illustrates the superior accuracy through the multi-mesh method compared to the single-mesh method.

5.2. Collection of results for more general electronic structures

In this subsection, we list the results of atoms and molecules using multi-mesh adaptive methods. The referenced values are generated from the state-of-art software `NWChem` using *aug-cc-pv5z* basis sets except for the helium atom in which the *aug-cc-pv6z* basis set is adopted. According to Table 4, we observe that chemical accuracy is achieved in the first four examples. However, as the number of atoms increases, the accuracy diminishes. This degradation can be attributed to the limitations of computational resources. In such scenarios, we try to fully utilize our computational resources for simulations using both the single-mesh and multi-mesh adaptive methods, which can be implemented by dynamically adjusting the mesh adaption tolerance. Hence a comparative assessment between the two approaches can be delivered.

The results are summarized in Table 4 and we display several meshes and isosurfaces for these examples in Fig. 17. From the table and the figure, we can see that

- For the last four molecules, besides a similar observation for the CPU time in Figs. 14, 15, and 16, another observation can be made, that with comparable computational resource, our multi-mesh method can generally produce a much better result. It means that the multi-mesh method would be an answer to the question of how to generate the most accurate result, by fully using the given computational resource.

Table 4
List of examples. The referenced value is obtained from `NWChem`. $N_{s,\mathrm{KS}}$ represents the number of mesh grids for the single mesh method. $E_{s,\mathrm{KS}}$ stands for the energy obtained from the single-mesh method. The last row refers to the comparison with chemical accuracy.

molecule	He	LiH	BeH_2	CH_4	CH_3OH	CH_3CH_2OH	C_6H_6
$N_{s,\mathrm{KS}}$	2,874,807	6,029,109	7,338,638	8,121,252	7,552,976	7,720,132	7,210,851
N_{KS}	1,262,599	3,786,064	4,852,528	5,440,384	6,226,903	6,864,714	6,036,918
N_{Har}	2,185,920	3,665,015	4,276,863	5,322,827	4,063,275	3,593,622	4,798,747
E_{ref}	-2.8348	-7.9196	-15.6606	-40.1198	-114.8503	-153.8158	-230.1916
$E_{s,\mathrm{KS}}$	-2.8341	-7.9178	-15.6573	-40.1105	-114.8033	-153.7418	-229.9824
E_{KS}	-2.8340	-7.9181	-15.6580	-40.1147	-114.8205	-153.7734	-230.0663
$\|\Delta E_{s,\mathrm{KS}}/E_{\mathrm{ref}}\|$	0.0002	0.0002	0.0002	0.0002	0.0004	0.0005	0.0009
$\|\Delta E_{\mathrm{KS}}/E_{\mathrm{ref}}\|$	0.0003	0.0002	0.0002	0.0001	0.0003	0.0003	0.0005
$\|\Delta E_{s,\mathrm{KS}}/n_{atom}\|$	**0.0007**	**0.0009**	0.0011	0.0019	0.0078	0.0082	0.0174
$\|\Delta E_{\mathrm{KS}}/n_{atom}\|$	**0.0008**	**0.0008**	**0.0009**	**0.0010**	0.0050	0.0047	0.0104

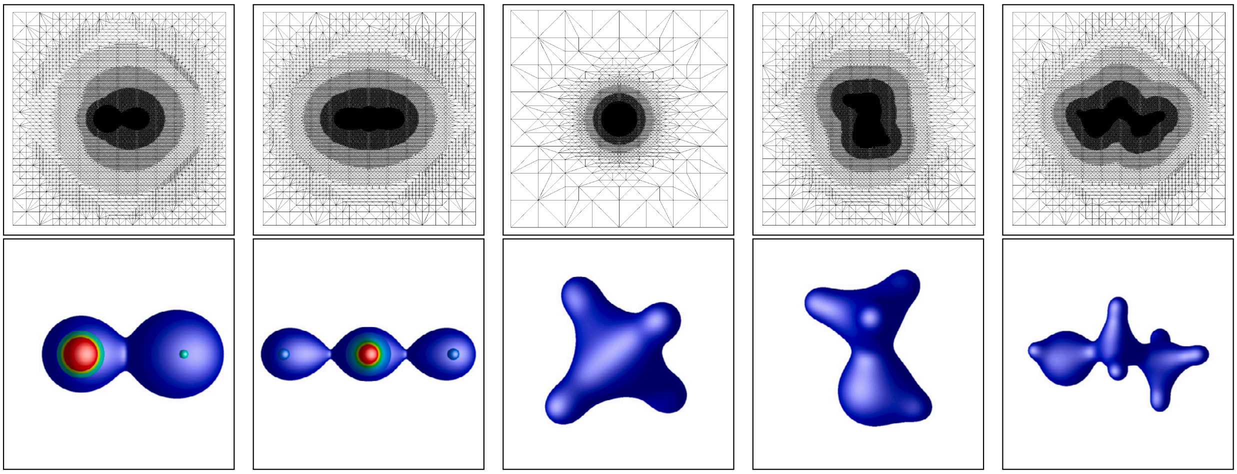

Fig. 17. Sliced KS meshes (top) and isosurfaces (bottom) for the following molecules: LiH, BeH_2, CH_4, CH_3OH, and CH_3CH_2OH. Contours are depicted for the two-dimensional structures (LiH and BeH_2). The computational domain is set as $[-10, 10]^3$, with the exception of the CH_4 example, for which it is extended to $[-20, 20]^3$.

- When maximizing our computational resources, the multi-mesh approach yields more precise results. To be specific, the multi-mesh method attains chemical accuracy for the first four examples, as indicated in the last row of Table 4. In contrast, the single-mesh method only achieves chemical accuracy in the first two examples.
- In the multi-mesh method, the number of mesh grids for the Hartree mesh exceeds the number for the KS mesh only in the case of helium. As the system size scales up, the number of Hartree mesh grids becomes progressively less than the number of KS mesh grids. Consequently, the computational time needed for the Hartree potential becomes less significant as the system size increases. There is an exception in the cases of CH_4 and C_6H_6, which arises because we opted for a larger computational domain, specifically, $[-20, 20]^3$.

6. Conclusion

In this paper, we introduce a multi-mesh adaptive finite element framework for Kohn-Sham density functional theory, aiming to achieve chemical accuracy. We investigate the impact of the Hartree potential approximation on total energy, finding that chemical accuracy cannot be attained without a well-approximated Hartree potential. While the single-mesh adaptive method, considering the Hartree potential, achieves chemical accuracy, it comes with significant computational costs. To address this, we propose the multi-mesh adaptive method, which offers a more efficient route to achieving chemical accuracy.

We demonstrate the effectiveness and accuracy of the multi-mesh adaptive method through various numerical examples. However, we observe that even for the helium atom, achieving chemical accuracy demands a substantial number of mesh grids, mainly due to the use of linear Lagrange finite elements. One strategy is to adopt higher-order elements under the multi-mesh framework. Additionally, we consider future work on conducting numerical analysis for the multi-mesh adaptive method. Moreover, we recognize the need for accelerating the multi-mesh adaptive approach, particularly through parallel implementation and the development of more effective error indicators for mesh adaptation methods. These enhancements will further refine the efficiency and accuracy of the multi-mesh adaptive framework.

CRediT authorship contribution statement

Yang Kuang: Writing – original draft, Validation, Methodology. **Yedan Shen:** Methodology. **Guanghui Hu:** Writing – review & editing, Validation, Supervision, Methodology, Investigation, Conceptualization.

Declaration of competing interest

The authors declare no competing interests.

Data availability

No data was used for the research described in the article.

Acknowledgement

The authors would like to thank the anonymous reviewers for their professional comments on chemical accuracy. The work of Y. Kuang was supported in part by the National Natural Science Foundation of China (No. 12201130) and the Guangzhou Municipal Science and Technology Bureau (No. 2023A04J1321). The work of Y. Shen was supported by Hunan Key Laboratory for Computation and Simulation in Science and Engineering (No. LCSSE202307). The work of G. Hu was supported by the Science and Technology Development Fund, Macao SAR (No. 0082/2020/A2), National Natural Science Foundation of China (Nos. 11922120, 11871489), MYRG of University of Macau (No. MYRG-GRG2023-00157-FST-UMDF).

References

[1] Reinhart Ahlrichs, Michael Bär, Marco Häser, Hans Horn, Christoph Kölmel, Electronic structure calculations on workstation computers: the program system turbomole, Chem. Phys. Lett. 162 (3) (1989) 165–169.
[2] Gang Bao, Guanghui Hu, Di Liu, An h-adaptive finite element solver for the calculations of the electronic structures, J. Comput. Phys. 231 (14) (2012) 4967–4979.
[3] Gang Bao, Guanghui Hu, Di Liu, Real-time adaptive finite element solution of time-dependent Kohn–Sham equation, J. Comput. Phys. 281 () (2015) 743–758.
[4] Gang Bao, Guanghui Hu, Di Liu, Towards translational invariance of total energy with finite element methods for Kohn–Sham equation, Commun. Comput. Phys. 19 (1) (2016) 1–23.
[5] P.F. Batcho, Computational method for general multicenter electronic structure calculations, Phys. Rev. E 61 (6) (2000) 7169.
[6] Mihail Bogojeski, Leslie Vogt-Maranto, Mark E Tuckerman, Klaus-Robert Muller, Kieron Burke, Quantum chemial accuracy from density functional theory approximations via machine learning, Nat. Commun. 11 (1) (2020) 5223.
[7] Zhenning Cai, Yun Chen, Yana Di, Guanghui Hu, Ruo Li, Wenbin Liu, Heyu Wang, Fanyi Yang, Chengbao Yao, Hongfei Zhan, AFEPack: a general-purpose C++ library for numerical solutions of partial differential equations, Commun. Comput. Phys. 36 (1) (2024) 274–318.
[8] Huajie Chen, Xiaoying Dai, Xingao Gong, Lianhua He, Aihui Zhou, Adaptive finite element approximations for Kohn–Sham models, Multiscale Model. Simul. 12 (4) (2014) 1828–1869.
[9] Or Cohen, Leeor Kronik, Achi Brandt, Locally refined multigrid solution of the all-electron Kohn–Sham equation, J. Chem. Theory Comput. 9 (11) (2013) 4744–4760.
[10] Xiaoying Dai, Liwen Zhang, Aihui Zhou, Convergent and orthogonality preserving schemes for approximating the Kohn–Sham orbitals, Numer. Math. Theor. Meth. Appl. 16 (1) (2023) 1–25.
[11] Denis Davydov, Toby D. Young, Paul Steinmann, On the adaptive finite element analysis of the Kohn–Sham equations: methods, algorithms, and implementation, Int. J. Numer. Methods Eng. 106 (11) (2016) 863–888.
[12] Yana Di, Ruo Li, Computation of dendritic growth with level set model using a multi-mesh adaptive finite element method, J. Sci. Comput. 39 (3) (2009) 441–453.
[13] Yana Di, Xiao-Ping Wang, Precursor simulations in spreading using a multi-mesh adaptive finite element method, J. Comput. Phys. 228 (5) (2009) 1380–1390.
[14] Lenka Dubcova, Pavel Solín, J. Cerveny, P. Kus, Space and time adaptive two-mesh hp-finite element method for transient microwave heating problems, Electromagnetics 30 (1–2) (2010) 23–40.
[15] Jun Fang, Xingyu Gao, Aihui Zhou, A Kohn–Sham equation solver based on hexahedral finite elements, J. Comput. Phys. 231 (8) (2012) 3166–3180.
[16] Stefano Giani, Pavel Solín, Solving elliptic eigenproblems with adaptive multimesh hp-FEM, J. Comput. Appl. Math. 394 (2021) 113528.
[17] Xianliang Hu, Ruo Li, Tao Tang, A multi-mesh adaptive finite element approximation to phase field models, Commun. Comput. Phys. 5 (5) (2009) 1012–1029.
[18] John David Jackson, Classical Electrodynamics, American Association of Physics Teachers, 1999.
[19] Lauri Lehtovaara, Ville Havu, Martti Puska, All-electron density functional theory and time-dependent density functional theory with high-order finite elements, J. Chem. Phys. 131 (5) (2009) 054103.
[20] Kurt Lejaeghere, Gustav Bihlmayer, Torbjörn Björkman, Peter Blaha, Stefan Blügel, Volker Blum, Damien Caliste, Ivano E. Castelli, Stewart J. Clark, Andrea Dal Corso, et al., Reproducibility in density functional theory calculations of solids, Science 351 (6280) (2016) aad3000.
[21] Ruo Li, On multi-mesh h-adaptive methods, J. Sci. Comput. 24 (3) (2005) 321–341.
[22] Yvon Maday, h-P finite element approximation for full-potential electronic structure calculations, in: Partial Differential Equations: Theory, Control and Approximation, Springer, 2014, pp. 349–377.
[23] Miguel A.L. Marques, Micael J.T. Oliveira, Tobias Burnus, Libxc: a library of exchange and correlation functionals for density functional theory, Comput. Phys. Commun. 183 (10) (2012) 2272–2281.
[24] Phani Motamarri, Sambit Das, Shiva Rudraraju, Krishnendu Ghosh, Denis Davydov, Vikram Gavini, DFT-FE–A massively parallel adaptive finite-element code for large-scale density functional theory calculations, Comput. Phys. Commun. 246 (2020) 106853.
[25] Phani Motamarri, Michael R. Nowak, Kenneth Leiter, Jaroslaw Knap, Vikram Gavini, Higher-order adaptive finite-element methods for Kohn–Sham density functional theory, J. Comput. Phys. 253 (2013) 308–343.
[26] J.E. Pask, N. Sukumar, S.E. Mousavi, Linear scaling solution of the all-electron Coulomb problem insolids, Int. J. Multiscale Comput. Eng. 10 (1) (2012).
[27] John A. Pople, Nobel lecture: quantum chemical models, Rev. Mod. Phys. 71 (5) (1999) 1267.
[28] Volker Schauer, Christian Linder, All-electron Kohn–Sham density functional theory on hierarchic finite element spaces, J. Comput. Phys. 250 (2013) 644–664.
[29] Yedan Shen, Ting Wang, Jie Zhou, Guanghui Hu, A convergence analysis of a structure-preserving gradient flow method for the all-electron Kohn–Sham model, Numer. Math. Theor. Meth. Appl. 16 (3) (2023) 597–621.

[30] Pavel Solín, J. Cerveny, Lenka Dubcova, David Andrs, Monolithic discretization of linear thermoelasticity problems via adaptive multimesh hp-FEM, J. Comput. Appl. Math. 234 (7) (2010) 2350–2357.

[31] Eiji Tsuchida, Masaru Tsukada, Electronic-structure calculations based on the finite-element method, Phys. Rev. B 52 (8) (1995) 5573.

[32] Marat Valiev, Eric J. Bylaska, et al., NWChem: a comprehensive and scalable open-source solution for large scale molecular simulations, Comput. Phys. Commun. 181 (9) (2010) 1477–1489.

[33] Rüdiger Verfürth, A Posteriori Error Estimation Techniques for Finite Element Methods, OUP, Oxford, 2013.

[34] Seymour H. Vosko, Leslie Wilk, Marwan Nusair, Accurate spin-dependent electron liquid correlation energies for local spin density calculations: a critical analysis, Can. J. Phys. 58 (8) (1980) 1200–1211.

[35] Erich Wimmer, Henry Krakauer, Michael Weinert, A.J. Freeman, Full-potential self-consistent linearized-augmented-plane-wave method for calculating the electronic structure of molecules and surfaces: O_2 molecule, Phys. Rev. B 24 (2) (1981) 864.

[36] Shengyang Wu, Xianliang Hu, Shengfeng Zhu, A multi-mesh finite element method for phase-field based photonic band structure optimization, J. Comput. Phys. 357 (2018) 324–337.

[37] Chao Yang, Juan C. Meza, Lin-Wang Wang, A trust region direct constrained minimization algorithm for the Kohn–Sham equation, SIAM J. Sci. Comput. 29 (5) (2007) 1854–1875.

[38] Yubo Zhang, Daniil A Kitchaev, Julia Yang, Tina Chen, Stephen T Dacek, Rafael A Sarmiento-Perez, Maguel AL Marques, Haowei Peng, Gerbrand Ceder, John P Perdew, Jianwei Sun, Efficient first-principles prediction of solid stability: towards chemical accuracy, Npj Comput. Mater. 4 (1) (2018) 9.