

# A Nonnested Augmented Subspace Method for Kohn-Sham Equation

Guanghai Hu<sup>1,2</sup>, Hehu Xie<sup>3,4</sup>, Fei Xu<sup>5</sup>, and Gang Zhao<sup>3</sup>

<sup>1</sup>Department of Mathematics, University of Macau, Macao S.A.R., China

<sup>2</sup>UM Zhuhai Research Institute, Zhuhai, Guangdong, China

<sup>3</sup>LSEC, ICMSEC, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China

<sup>4</sup>School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing, 100049, China

<sup>5</sup>Faculty of Science, Beijing University of Technology, Beijing 100124, China

## Abstract

In this paper, a novel adaptive finite element method is proposed to solve the Kohn-Sham equation based on the moving mesh (nonnested mesh) adaptive technique and the augmented subspace method. Different from the classical self-consistent field iterative algorithm which requires to solve the Kohn-Sham equation directly in each adaptive finite element space, our algorithm transforms the Kohn-Sham equation into some linear boundary value problems of the same scale in each adaptive finite element space, and then the wavefunctions derived from the linear boundary value problems are corrected by solving a small-scale Kohn-Sham equation defined in a low-dimensional augmented subspace. Since the new algorithm avoids solving large-scale Kohn-Sham equation directly, a significant improvement for the solving efficiency can be obtained. In addition, the adaptive moving mesh technique is used to generate the nonnested adaptive mesh for the nonnested augmented subspace method according to the singularity of the approximate wavefunctions. The modified Hessian matrix of the approximate wavefunctions is used as the metric matrix to redistribute the mesh. Through the moving mesh adaptive technique, the redistributed mesh is almost optimal. A number of numerical experiments are carried out to verify the efficiency and the accuracy of the proposed algorithm.

**Keywords.** Density functional theory, Kohn-Sham equation, nonnested mesh, augmented subspace method.

**AMS subject classifications.** 65N30, 65N25, 65L15, 65B99.

## 1 Introduction

Density functional theory (DFT) is one of the major breakthroughs in quantum physics and quantum chemistry. In the framework of density functional theory, the many-body problem can be simplified as the motion of electrons without interaction in the effective potential field, which includes the Coulomb potential of the external potential field, the Hartree potential generated by the interaction between electrons, and the exchange-correlation potential for the nonclassical Coulomb interaction. For the exchange-correlation potential, it has always been a difficulty in density functional theory. At present, there is no exact analytical expression for the exchange-correlation potential. Generally, it is approximately described by the local density approximation (LDA), the local spin density approximation (LSDA) and the generalized gradient approximation (GGA), etc. The Kohn-Sham equation is one of the most important models in the calculation of electronic structure, which transforms the wavefunctions (in  $\mathbb{R}^{3N}$  space) describing  $N$  particle states into density function (in  $\mathbb{R}^3$  space), so as to reduce the degrees of freedom of equation and reduce the computational work of numerical simulation. The idea of Kohn-Sham equation can be traced back to Thomas-Fermi model in 1927. The model gave a primary description of the electronic structure of atoms. The strict theoretical analysis was described by Hohenberg and Kohn in [26], which proved the correctness and feasibility of using single electron orbit to replace the wavefunctions describing multiple electrons.

So far, lots of numerical methods for solving Kohn-Sham equation have been developed. For instance, plane-wave method [10, 35] is the most popular method in the computational quantum chemistry community. Since the basis

function is independent from the ionic position, plane-wave method has advantage in calculating intermolecular force. Combined with the pseudopotential method, plane-wave method plays an important role in the study of the ground and excited states calculations, and geometry optimization of the electronic structures. Although the plane-wave method is the most popular one in the computational quantum chemistry community, it is inefficient in solving non-periodic systems like molecules, nano-clusters, or materials systems with defects, etc. Furthermore, the plane-wave method uses the global basis which significantly affect the scalability on parallel computing platforms. The atomic-orbital-type basis sets [21, 22] are also widely used for simulating materials systems such as molecules and clusters. However, they are well-suited only for isolated systems with special boundary conditions. It is difficult to develop a systematic basis-set for all materials systems. Thus over the past decades, more and more attentions are attracted to develop efficient and scalable grid-based methods such as the finite element method for electronic structure calculations. The advantages of grid-based methods include that it can use unstructured meshes and local basis sets, hence owing high scalability on parallel computing platforms. So far, the applications of the finite element method in solving Kohn-Sham equation have been studied systematically. We refer to [6, 8, 11, 33, 36, 39, 40, 42, 43, 47–49] and references therein for a comprehensive overview.

In this paper, we propose a nonnested augmented subspace method to solve the Kohn-Sham equation based on the moving mesh adaptive technique and the augmented subspace method. Since the Coulomb potential and Hartree potential have strong singularities, the adaptive mesh refinement is a competitive strategy to improve solving efficiency. Adaptive methods mainly include  $h$ -adaptive method,  $p$ -adaptive method and  $r$ -adaptive method. The  $h$ -adaptive method uses *a posteriori* error indicator for local refinement,  $p$ -adaptive method uses higher-order polynomials on local mesh elements,  $r$ -adaptive method (moving mesh method) uses control function or metric tensor (also known as metric matrix) for mesh redistribution. In this paper, we adopt the  $r$ -adaptive method to generate a series of nonnested adaptive finite element spaces. The  $r$ -adaptive method uses the control function or metric matrix to guide the mesh movement, which can improve the accuracy by moving the grid nodes to the area with large errors. This method can be traced back to Alexander’s work in [3], and then Miller’s work [37, 38] promoted the development of  $r$ -adaptive method. In 2001, Li and Tang etc [31, 32] proposed the moving mesh method based on harmonic mapping. Later, Di and Li etc [18] applied the moving mesh method based on harmonic mapping to solve the incompressible Navier-Stokes equation. More results about moving mesh method can be found in [19, 20, 53, 54] and the references therein.

Another technique adopted in this paper is the multilevel correction method (augmented subspace method) [27, 34, 51, 52]. The traditional multilevel correction method can only be performed on the nested multilevel space sequence. In this paper, we develop a nonnested multilevel correction technique for Kohn-Sham equation on a nonnested mesh sequence generated by the moving mesh technique. Different from the classical self-consistent field iterative algorithm which requires to solve the Kohn-Sham equation directly in each adaptive finite element space, our algorithm transforms the Kohn-Sham equation into some linear boundary value problems of the same scale on each level of the adaptive refinement mesh sequence, and then the wavefunctions derived from the linear boundary value problems are corrected by solving a small-scale Kohn-Sham equation defined in a low-dimensional augmented subspace. Since the new algorithm avoids solving large-scale Kohn-Sham equation directly, a significant improvement for the efficiency can be obtained. In addition, the adaptive mesh is produced using the moving mesh technique according to the singularity of the wavefunctions, which can dramatically generate a high accuracy with less computational work. In our algorithm, the approximate wavefunctions are used as the adaptive function, and the modified Hessian matrix of the density function is used as the metric matrix to redistribute the mesh. Through the moving mesh adaptive technique, the redistributed mesh is almost optimal. By combining the moving mesh technique and the nonnested augmented subspace method, the solving efficiency for Kohn-Sham equation can be significantly improved.

The remainder of this paper is as follows. Section 2 introduces Kohn-Sham equation and its finite element discretization method. Section 3 introduces the nonnested augmented subspace method for Kohn-Sham equation by using the moving mesh technique and the multilevel correction method. In Section 4, we propose the convergence analysis and the estimate of computational work for the presented algorithm. In Section 5, a number of numerical examples are demonstrated to verify the efficiency and accuracy of the proposed method.

## 2 Finite element method for Kohn-Sham equation

To describe the Kohn-Sham equation and its finite element discretization, we first introduce some notation. Following [1], we use  $W^{s,p}(\Omega)$  to denote Sobolev spaces, and use  $\|\cdot\|_{s,p,\Omega}$  and  $|\cdot|_{s,p,\Omega}$  to denote the associated norms and seminorms, respectively. In case  $p = 2$ , we denote  $H^s(\Omega) = W^{s,2}(\Omega)$  and  $H_0^1(\Omega) = \{v \in H^1(\Omega) : v|_{\partial\Omega} = 0\}$ , where  $v|_{\partial\Omega} = 0$  is in the sense of trace, and denote  $\|\cdot\|_{s,\Omega} = \|\cdot\|_{s,2,\Omega}$ . For convenience, the symbols  $x_1 \lesssim y_1$ ,  $x_2 \gtrsim y_2$  and  $x_3 \approx y_3$  are used to represent  $x_1 \leq C_1 y_1$ ,  $x_2 \geq c_2 y_2$ , and  $c_3 x_3 \leq y_3 \leq C_3 y_3$ , respectively, where  $C_1$ ,  $c_2$ ,  $c_3$ , and  $C_3$  denote some mesh-independent constants. In this paper,  $\Omega$  is dropped from the subscript of the norm for simplicity.

Let  $\mathcal{H} := (H_0^1(\Omega))^N$  be the Hilbert space with the inner product

$$(\Phi, \Psi) = \sum_{i=1}^N \int_{\Omega} \phi_i \psi_i dx, \quad \forall \Phi = (\phi_1, \dots, \phi_N), \quad \Psi = (\psi_1, \dots, \psi_N) \in \mathcal{H}, \quad (2.1)$$

where  $\Omega \subset \mathbb{R}^3$ . For any  $\Psi \in \mathcal{H}$  and a subdomain  $\omega \subset \Omega$ , we define  $\rho_{\Psi} = \sum_{i=1}^N |\psi_i|^2$  and

$$\|\Psi\|_{s,\omega} = \left( \sum_{i=1}^N \|\psi_i\|_{s,\omega}^2 \right)^{1/2}, \quad s = 0, 1.$$

Let  $\mathcal{Q}$  be a subspace of  $\mathcal{H}$  with orthonormality constraints:

$$\mathcal{Q} = \{\Psi \in \mathcal{H} : \Psi^T \Psi = I^{N \times N}\}, \quad (2.2)$$

where  $\Phi^T \Psi = \left( \int_{\Omega} \phi_i \psi_j dx \right)_{i,j=1}^N \in \mathbb{R}^{N \times N}$ .

We consider a molecular system consisting of  $M$  nuclei with charges  $\{Z_1, \dots, Z_M\}$  and locations  $\{R_1, \dots, R_M\}$ , respectively, and  $N$  electrons in the non-relativistic and spin-unpolarized setting. The general form of Kohn-Sham energy functional can be demonstrated as follows

$$E(\Psi) = \int_{\Omega} \left( \frac{1}{2} \sum_{i=1}^N |\nabla \psi_i|^2 + V_{ext}(x) \rho_{\Psi} + E_{xc}(\rho_{\Psi}) \right) dx + \frac{1}{2} D(\rho_{\Psi}, \rho_{\Psi}), \quad (2.3)$$

for  $\Psi = (\psi_1, \psi_2, \dots, \psi_N) \in \mathcal{H}$ . Here,  $V_{ext}$  is the Coulomb potential defined by  $V_{ext} = - \sum_{k=1}^M Z_k / |x - R_k|$ ,  $D(\rho_{\Psi}, \rho_{\Psi})$  is the Hartree energy defined by

$$D(f, g) = \int_{\Omega} f(g * r^{-1}) dx = \int_{\Omega} \int_{\Omega} f(x) g(y) \frac{1}{|x - y|} dx dy, \quad (2.4)$$

and  $E_{xc}(t)$  is some real function over  $[0, \infty)$  denoting the exchange-correlation energy.

The ground state of the system is obtained by solving the minimization problem:

$$\inf \{E(\Psi) : \Psi \in \mathcal{Q}\}, \quad (2.5)$$

and we refer to [10, 16] for the existence of a minimizer under some conditions.

The Euler-Lagrange equation corresponding to the minimization problem (2.5) is the well-known Kohn-Sham equation: Find  $(\Lambda, \Psi) \in \mathbb{R}^N \times \mathcal{H}$  such that

$$\begin{cases} H_{\Psi} \psi_i = \lambda_i \psi_i & \text{in } \Omega, \quad i = 1, 2, \dots, N, \\ \int_{\Omega} \psi_i \psi_j dx = \delta_{ij}, \end{cases} \quad (2.6)$$

where  $H_{\Psi}$  is the Kohn-Sham Hamiltonian operator defined by

$$H_{\Psi} = -\frac{1}{2} \Delta + V_{ext} + V_{Har}(\rho_{\Psi}) + V_{xc}(\rho_{\Psi}) \quad (2.7)$$

with  $V_{Har}(\rho_\Psi) = r^{-1} * \rho_\Psi$ ,  $V_{xc}(\rho_\Psi) = E'_{xc}(\rho_\Psi)$ ,  $\Lambda = (\lambda_1, \dots, \lambda_N)$  and  $\lambda_i = (H_\Psi \psi_i, \psi_i)$ . The variational form of the Kohn-Sham equation can be described as follows: Find  $(\Lambda, \Psi) \in \mathbb{R}^N \times \mathcal{H}$  such that

$$\begin{cases} (H_\Psi \psi_i, v) = \lambda_i(\psi_i, v), & \forall v \in H_0^1(\Omega), \quad i = 1, 2, \dots, N, \\ \int_{\Omega} \psi_i \psi_j dx = \delta_{ij}. \end{cases} \quad (2.8)$$

Now, let us define the finite element discretization of (2.8). First we generate a shape regular decomposition  $\mathcal{T}_h$  of the computing domain  $\Omega$ . Then, based on the mesh  $\mathcal{T}_h$ , we construct the linear finite element space denoted by  $S_h \subset H_0^1(\Omega)$ . Define  $V_h = (S_h)^N \subset \mathcal{H}$ . Then the discrete form of (2.8) can be described as follows: Find  $(\bar{\Lambda}_h, \bar{\Phi}_h) \in \mathbb{R}^N \times V_h$  such that

$$\begin{cases} (H_{\bar{\Psi}_h} \bar{\psi}_{i,h}, v) = \bar{\lambda}_{i,h}(\bar{\psi}_{i,h}, v), & \forall v \in S_h, \quad i = 1, \dots, N, \\ \int_{\Omega} \bar{\psi}_{i,h} \bar{\psi}_{j,h} dx = \delta_{ij}, \end{cases} \quad (2.9)$$

with  $\bar{\lambda}_{i,h} = (H_{\bar{\Psi}_h} \bar{\psi}_{i,h}, \bar{\psi}_{i,h})$ .

For simplicity and generality, some assumptions are given for the error analysis of the proposed algorithm in this paper. We would like to mention that the following assumptions are satisfied by many practical physical models. For detail of the assumptions, please refer to [9, 10, 14, 15].

- Assumption A:  $|V_{xc}(t)| + |tV'_{xc}(t)| \in \mathcal{P}(p_1, (c_1, c_2))$  for some  $p_1 \in [0, 2]$ , where  $\mathcal{P}(p, (c_1, c_2))$  denotes the following function set:

$$\mathcal{P}(p, (c_1, c_2)) = \left\{ f : \exists a_1, a_2 \in \mathbb{R} \text{ such that } c_1 t^p + a_1 \leq f(t) \leq c_2 t^p + a_2, \quad \forall t \geq 0 \right\} \quad (2.10)$$

with  $c_1 \in \mathbb{R}$  and  $c_2, p \in [0, \infty)$ .

- Assumption B: There exists a constant  $\alpha \in (0, 1]$  such that  $|V'_{xc}(t)| + |tV''_{xc}(t)| \lesssim 1 + t^{\alpha-1}$ .
- Assumption C: Let  $(\Lambda, \Psi)$  be a solution of (2.8). When the mesh size  $h$  is small enough, the discrete solution  $(\bar{\Lambda}_h, \bar{\Psi}_h) \in \mathbb{R}^N \times V_h$  satisfies the following estimate

$$\|\Psi - \bar{\Psi}_h\|_1 \lesssim \delta_h(\Psi), \quad (2.11)$$

$$\|\Psi - \bar{\Psi}_h\|_0 + |\Lambda - \bar{\Lambda}_h| \lesssim r(V_h) \|\Psi - \bar{\Psi}_h\|_1, \quad (2.12)$$

where  $r(V_h) \ll 1$  and

$$\delta_h(\Psi) := \inf_{\Phi \in V_h} \|\Psi - \Phi\|_1.$$

### 3 Nonnested augmented subspace method for Kohn-Sham equation

In this section, we design the nonnested augmented subspace method for Kohn-Sham equation based on the moving mesh adaptive technique and the augmented subspace method. In the first two subsections, we introduce some computing techniques including the solving process for exchange-correlation potential and Hartree potential, and the detailed process of moving mesh technique. Then next, we combine these techniques to generate the nonnested augmented subspace method for Kohn-Sham equation.

#### 3.1 The calculation of exchange-correlation potential and Hartree potential

The Kohn-Sham equation contains two nonlinear terms: the Hartree potential and the exchange-correlation potential. In this subsection, we introduce the detailed form for exchange-correlation potential and the solving process for Hartree potential.

Since there is no exact expression for the exchange-correlation potential, we use the local density approximation (LDA) in our numerical simulations. The exchange-correlation potential can be treated as the variational derivative

of the exchange-correlation energy to the density function  $V_{xc} = \delta E_{xc}^{LDA} / \delta \rho$ . The exchange-correlation energy  $E_{xc}^{LDA}$  is divided into two parts: exchange energy  $E_x^{LDA}$  and correlation energy  $E_c^{LDA}$ , that is  $E_{xc}^{LDA} = E_x^{LDA} + E_c^{LDA}$ . For the exchange energy, we use the following approximate form given by Kohn and Sham in [30]:

$$E_x^{LDA}(\rho) = -\frac{3}{4} \left( \frac{3}{\pi} \right)^{1/3} \int_{\Omega} \rho(x)^{4/3} d\Omega, \quad (3.1)$$

with the exchange-correlation potential

$$V_x^{LDA}(\rho) = - \left( \frac{3\rho}{\pi} \right)^{1/3}. \quad (3.2)$$

For the relatively complex correlation energy and correlation potential, we use the expression proposed by Perdew and Zunger in [44]. The correlation energy per electron is:

$$\varepsilon_c^{LDA} = \begin{cases} A \ln r_s + B + C r_s \ln r_s + D r_s, & \text{if } r_s < 1, \\ \frac{\gamma}{(1 + \beta_1 \sqrt{r_s} + \beta_2 r_s)}, & \text{if } r_s \geq 1. \end{cases}$$

The corresponding correlation potential is

$$V_c^{LDA}(\rho) = \begin{cases} A \ln r_s + (B - \frac{1}{3}A) + \frac{2}{3}C r_s \ln r_s + \frac{1}{3}(2D - C)r_s, & \text{if } r_s < 1, \\ \varepsilon_c^{LDA} \frac{1 + \frac{7}{6}\beta_1 \sqrt{r_s} + \frac{4}{3}\beta_2 r_s}{(1 + \beta_1 \sqrt{r_s} + \beta_2 r_s)}, & \text{if } r_s \geq 1 \end{cases} \quad (3.3)$$

with  $r_s = (3/(4\pi\rho))^{1/3}$ ,  $A = 0.0311$ ,  $B = -0.048$ ,  $\gamma = -0.1423$ ,  $\beta_1 = 1.0529$ ,  $\beta_2 = 0.3334$ ,  $C = 0.0020$ ,  $D = -0.0116$ .

The ground state energy of the molecular system can be calculated through:

$$E = \sum_{i=1}^N \lambda_i - \int_{\Omega} \left( \frac{1}{2} V_{Hartree} + V_{xc} \right) \rho(x) d\Omega + E_{xc} + E_{nn}. \quad (3.4)$$

The last term  $E_{nn}$  accounts for the interactions between the nuclei:

$$E_{nn} = \sum_{1 \leq k < j \leq M} \frac{Z_j Z_k}{|x_j - x_k|}. \quad (3.5)$$

Next, we introduce the solving process for Hartree potential  $V_{Har}$ , which is computed by solving the following Poisson equation:

$$\begin{cases} -\Delta V_{Har} = 4\pi\rho, & \text{in } \Omega, \\ V_{Har} = w_D, & \text{on } \partial\Omega. \end{cases} \quad (3.6)$$

Because the Hartree potential decays with the rate  $N/r$ , where  $N$  is the electron number, the simple use of zero Dirichlet boundary condition will introduce large truncation error on the boundary. To give the evaluation of the Hartree potential on the boundary, a multipole expansion approximation is employed for the boundary values. In our numerical simulations, the following approximation is used:

$$w_D = w|_{\partial\Omega} \approx \frac{1}{|\mathbf{x} - \mathbf{x}''|} \int_{\Omega} \rho(\mathbf{x}') d\mathbf{x}' + \sum_{i=1,2,3} p_i \cdot \frac{x^i - x''^i}{|\mathbf{x} - \mathbf{x}''|^3} + \sum_{i,j=1,2,3} q_{i,j} \frac{3(x^i - x''^i)(x^j - x''^j) - \delta_{i,j}|\mathbf{x} - \mathbf{x}''|^2}{|\mathbf{x} - \mathbf{x}''|^5}, \quad (3.7)$$

where

$$p_i = \int_{\Omega} \rho(\mathbf{x}') (x'^i - x''^i) d\mathbf{x}', \quad q_{i,j} = \int_{\Omega} \rho(\mathbf{x}') (x'^i - x''^i)(x'^j - x''^j) d\mathbf{x}'.$$

In the above expressions,  $\mathbf{x}''$  is chosen as  $\mathbf{x}'' = \int_{\Omega} \mathbf{x} \rho(\mathbf{x}) d\mathbf{x} / \int_{\Omega} \rho(\mathbf{x}) d\mathbf{x}$ .

### 3.2 Moving mesh adaptive technique

The adaptive methods mainly include  $h$ -adaptive method,  $p$ -adaptive method and  $r$ -adaptive method. The  $h$ -adaptive method uses *a posteriori* error indicator for local refinement,  $p$ -adaptive method uses higher-order polynomials on local mesh elements,  $r$ -adaptive method uses control function or metric tensor (also known as metric matrix) for mesh redistribution. In this subsection, we introduce the detailed process for the  $r$ -adaptive method which is used for mesh adaptive refinement in this paper.

The moving mesh adaptive ( $r$ -adaptive) software packages mostly control mesh movement based on the control function or metric tensor. Some well-known finite element software packages such as AFEPack [18, 31, 32] are based on the control function. Other software packages such as BAMG [24], Mshmet and Mmg [17, 20] are based on metric tensor. Essentially, the control function and the measurement matrix are consistent. From mathematical point of view, they differ only by a constant factor. From the way of controlling the movement of the grid points, the control function is based on the error equal distribution principle, and the measurement matrix is based on the unit volume principle (BAMG) or the unit edge principle (mshmet and Mmg). Here, we use the three-dimensional anisotropic adaptive software Mshmet and Mmg3d (three-dimensional module in Mmg) based on the  $M$ -unit side length principle, in which the measurement used is the correction of the Hessian matrix  $H_u(z)$  of the given adaptive function  $u$  at the grid node  $z$ . The detailed form is as follows:

$$\mathcal{M}(z) = R\tilde{\Lambda}R^{-1}, \quad (3.8)$$

where  $z$  denotes the location of the grid node,  $R$  denotes an orthogonal matrix composed of the orthogonal eigenfunctions of the Hessian matrix  $H_u(z)$ ,  $\tilde{\Lambda}$  is defined by

$$\tilde{\Lambda} := \begin{pmatrix} \tilde{\lambda}_1 & & & \\ & \tilde{\lambda}_2 & & \\ & & \ddots & \\ & & & \tilde{\lambda}_n \end{pmatrix}. \quad (3.9)$$

To guarantee the positive definite of the measurement matrix,  $\tilde{\lambda}_i$  is set to be

$$\tilde{\lambda}_i = \min \left( \max \left( \frac{C_d |\lambda_i|}{\varepsilon}, \frac{1}{h_{\max}^2} \right), \frac{1}{h_{\min}^2} \right), \quad (3.10)$$

where  $\varepsilon$  is used to control the  $P_1$ -interpolation error in the sense of  $L^\infty$ -norm,  $\lambda_i$  denotes the eigenvalue of the Hessian matrix  $H_u(z)$ ,  $h_{\max}$  and  $h_{\min}$  represent the upper limit of the longest edge scale and the lower limit of the shortest edge scale, respectively. In addition, the coefficient  $C_d$  is selected based on the following  $L^\infty$ -error estimate (see [2]):

**Lemma 3.1.** *Let  $K$  be a mesh element ( $d$ -dimensional simplex) of mesh  $\mathcal{T}_h$ ,  $E_K$  is a set of all edges on  $K$ , and  $u$  is a quadratic differentiable function. Then, we have the following  $L^\infty$ -interpolation error estimate*

$$\|u - \Pi_h u\|_{\infty, K} \leq C_d \max_{\mathbf{y} \in K} \max_{\mathbf{e} \in E_k} \mathbf{e}^T |H_u(\mathbf{y})| \mathbf{e}, \quad \forall K \in \mathcal{T}_h, \quad (3.11)$$

where

$$C_d = \frac{1}{2} \left( \frac{d}{d+1} \right)^2, \quad |H_u| := R|\Lambda|R^{-1}, \quad (3.12)$$

with  $|\Lambda| := \text{diag}\{|\lambda_1|, |\lambda_2|, \dots, |\lambda_n|\}$ .

The computation of the right term of (3.11) is nontrivial. Thus, we first assume that there exists a measurement  $\overline{\mathcal{M}}(K)$ , such that

$$\max_{\mathbf{y} \in K} \mathbf{e}^T |H_u(\mathbf{y})| \mathbf{e} \leq \mathbf{e}^T \overline{\mathcal{M}}(K) \mathbf{e}, \quad \forall \mathbf{e} \in E_k. \quad (3.13)$$

To derive  $\|u - \Pi_h u\|_{\infty, K} \leq \varepsilon$ , it is sufficient to require that

$$C_d \mathbf{e}^T \overline{\mathcal{M}}(K) \mathbf{e} = \varepsilon, \quad \forall \mathbf{e} \in E_k, \quad \text{or} \quad C_d \max_{\mathbf{e} \in E_k} \mathbf{e}^T \overline{\mathcal{M}}(K) \mathbf{e} = \varepsilon. \quad (3.14)$$

By defining the measurement  $\mathcal{M} := (C_d/\varepsilon)\overline{\mathcal{M}}$ , and based on (3.14), we know the measurement satisfies the following  $\mathcal{M}$ -unit edge length principle

$$\mathbf{e}^T \mathcal{M}(K) \mathbf{e} = 1, \quad \forall \mathbf{e} \in E_k. \quad (3.15)$$

That is, the length of side  $\mathbf{e}$  of element  $K$  under  $\mathcal{M}$  is 1, which is denoted by

$$\ell_{\mathcal{M}(K)}(\mathbf{e}) = 1, \quad \forall \mathbf{e} \in E_k. \quad (3.16)$$

In the actual calculation,  $\mathcal{M}$  can be determined by (3.8)–(3.10). In fact, in the software package Mshmet, we only need to adjust the parameters  $\text{err}(\varepsilon)$ ,  $\text{hmin}(h_{\min})$ ,  $\text{hmax}(h_{\max})$  to control the edge length of mesh elements. For more detailed introduction to the metric based adaptive mesh using the  $M$ -unit edge length principle, please refer to [2, 20]. For more discussion on metric matrices and control functions, please refer to [19, 53, 54].

Next, we introduce a fast interpolation algorithm between two nonnested meshes. Let  $\Omega_0$  and  $\Omega_1$  be two different computing domains,  $\mathcal{T}_h^{(0)} = \cup_i K_i^{(0)}$  and  $\mathcal{T}_h^{(1)} = \cup_i K_i^{(1)}$  be the corresponding mesh decompositions on these two domains, respectively. The two decompositions are nonnested. Then, we briefly introduce the fast interpolation algorithm proposed by the FreeFEM team to deal with nonnested meshes [23].

Let  $V_h^{(\ell)} := \{v \in C^0(\mathcal{T}_h^{(\ell)}) \mid v|_{K_i^{(\ell)}} \in P_1\}$ ,  $\ell = 0, 1$ , be the finite element space defined on  $\mathcal{T}_h^{(0)}$  and  $\mathcal{T}_h^{(1)}$ . For  $f \in V_h^{(0)}$ , the problem is to find  $g \in V_h^{(1)}$  such that:

$$g(q) = f(q), \quad \forall \text{ vertex } q \text{ of } \mathcal{T}_h^{(1)}. \quad (3.17)$$

A fast interpolation algorithm is proposed in [23] which is of complexity  $\mathcal{O}(N_1 \ln N_0)$ , where  $N_0$  is the number of vertices of  $\mathcal{T}_h^{(0)}$ ,  $N_1$  is the number of vertices of  $\mathcal{T}_h^{(1)}$ . The detailed process is presented in Algorithm 1.

---

**Algorithm 1:** Fast interpolation algorithm in FreeFEM

---

**Data:** First a quadtree is built containing all the vertices of the mesh  $\mathcal{T}_h^{(0)}$  such that in each terminal element there are at least one, and at most  $2^d$  vertices.

**Result:**  $g(q^1)$

```

1 for each vertex  $q^1$  in  $\mathcal{T}_h^{(1)}$  do
2   Find the terminal cell of the quadtree containing  $q^1$ ;
3   Find the nearest vertex  $q_j^0$  to  $q^1$  in that cell;
4   Choose one triangle  $K_i^{(0)} \in \mathcal{T}_h^{(0)}$  which has  $q_j^0$  for vertex;
5   Compute the barycentric coordinates  $\{\lambda_j\}_{j=1}^{d+1}$  of  $q^1$  in  $K_i^{(0)}$ ;
6   while Not all barycentric coordinates are positive do
7     if One barycentric coordinate  $\lambda_i$  is negative then
8       replace  $K_i^{(0)}$  by the adjacent triangle opposite  $q_i^0$ 
9     else if two barycentric coordinates are negative then
10      take one of the two randomly and replace  $K_i^{(0)}$  by the adjacent triangle as above
11   end
12   Compute the barycentric coordinates  $\{\lambda_j\}_{j=1}^{d+1}$  of  $q^1$  in  $K_i^{(0)}$ ;
13 end
14 Calculate  $g(q^1)$  on  $K_i^{(0)}$  by linear interpolation of  $f$ :  $g(q^1) = \sum_{j=1}^{d+1} \lambda_j f(q_j^{(0)})$ ;
15 end

```

---

### 3.3 Nonnested augmented subspace method for Kohn-Sham equation

In this subsection, we introduce the nonnested augmented subspace algorithm for Kohn-Sham equation. To simplify the description, we define a nonlinear operator which is composed of the Kohn-Sham Hamiltonian operator and a positive constant:

$$\tilde{H}_\Psi = -\frac{1}{2}\Delta + V_{ext} + \mu + V_{Har}(\rho_\Psi) + V_{xc}(\rho_\Psi), \quad \forall \Psi \in V, \quad (3.18)$$



where  $\mu$  is a positive constant that can be used to guarantee the coercive property.

The essence of the nonnested augmented method is to transform the solution of the Kohn-Sham equation on the fine mesh into solving the linear boundary value problems of the same scale and solving a small-scale Kohn-Sham equation in a low-dimensional subspace. In order to describe the nonnested augmented subspace algorithm, we generate a coarse mesh  $\mathcal{T}_H$  and corresponding linear finite element space  $S_H$ . The traditional multilevel correction method requires the multilevel finite element space sequence satisfies the nested relationship

$$\mathcal{T}_H \subset \mathcal{T}_{h_1} \subset \cdots \subset \mathcal{T}_{h_N}, \text{ and } S_H \subset S_{h_1} \subset \cdots \subset S_{h_N}. \quad (3.19)$$

The nonnested augmented subspace method presented in this paper does not require the mesh and finite element space to meet the above nested property, which guarantees that the augmented subspace method can be used in moving mesh sequence. In Algorithm 2, the flowchart of the moving mesh adaptive algorithm for the Kohn-Sham equation is given.

---

**Algorithm 2:** Moving mesh adaptive algorithm for Kohn-Sham equation

---

**Data:** A given electronic structure, and the initial guess of the ground state

**Result:** Ground state of the given electronic structure

- 1 Give  $K_{asm}, K_{max} \in \mathbb{N}$  with  $K_{asm} < K_{max}$ ,  $tol \in \mathbb{R}^+$ , and let  $k = 0$ ;
- 2 **repeat**
- 3   Using the standard self-consistent field iteration method to solve the problem: Find  $(\Lambda_{h_k}, \Psi_{h_k}) \in \mathbb{R} \times V_{h_k}$ , such that

$$\begin{cases} (H_{\Psi_{h_k}} \psi_{i,h_k}, v_{h_k}) &= \lambda_{i,h_k}(\psi_{i,h_k}, v_{h_k}), \quad \forall v_{h_k} \in S_{h_k}, \quad i = 1, \dots, N, \\ \int_{\Omega} \psi_{i,h_k} \psi_{j,h_k} dx &= \delta_{ij}. \end{cases} \quad (3.20)$$

- 4   Move the mesh grids according to the measurement (3.8) by using  $\sqrt{\rho_{h_k}}$  as the adaptive function;
  - 5   Update the mesh from  $\mathcal{T}_{h_k}$  to  $\mathcal{T}_{h_{k+1}}$  and interpolate the wavefunctions into the new mesh;
  - 6    $k = k + 1$ ;
  - 7 **until**  $k > K_{asm}$  or  $\Delta E/E < tol$ ;
  - 8 **while**  $k < K_{max}$  and  $\Delta E/E > tol$  **do**
  - 9   Using Algorithm 3 to solve the above problem (3.20);
  - 10   Move the mesh grids according to the measurement (3.8) by using  $\sqrt{\rho_{h_k}}$  as the adaptive function;
  - 11   Update the mesh from  $\mathcal{T}_{h_k}$  to  $\mathcal{T}_{h_{k+1}}$  and interpolate the wavefunctions into the new mesh;
  - 12    $k = k + 1$ ;
  - 13 **end**
- 

Next, we introduce the an augmented subspace method to solve the Kohn-Sham equation (3.20), which transforms the large-scale nonlinear eigenvalue problem into some linear boundary value problems of the same scale and small-scale Kohn-Sham equations defined in a low-dimensional augmented subspace. Through the augmented subspace method, the total computational work is asymptotically optimal since the dimension of the augmented subspace is small and fixed. The detailed process is described in Algorithm 3.

From Algorithm 3, we can find that the Kohn-Sham equation (3.22) is defined in a low-dimensional space  $S_{H,h_k}$ , thus a small-scale linear eigenvalue problem is needed to be solved in each nonlinear iteration step, which requires little computational work. But the augmented subspace  $S_{H,h_k}$  includes the basis functions  $\hat{\Psi}_{h_k}^{(\ell+1)}$  derived from the fine mesh  $\mathcal{T}_{h_k}$ , so the matrices assembling should be performed on the fine mesh to keep the accuracy.

Next, we introduce the detailed process for solving the small-scale Kohn-Sham equation (3.22) in  $S_{H,h_k}$ . For simplicity, we use  $h, V_h, \hat{\psi}_{i,h}$  to denote  $h_k, V_{h_k}, \hat{\psi}_{i,h_k}^{(\ell+1)}$ , respectively. Define  $N_H := \dim S_H$  and  $N_k := \dim S_{h_k}$ . Let  $\{\psi_{j,H}\}_{j=1}^{N_H}$  be the basis functions for  $S_H$ . When solving (3.22) by iteration method, a linear eigenvalue problem as follows is required to be solved in each iteration step:

$$F_A U_H = \lambda_h F_B U_H, \quad (3.23)$$

where

$$F_A := \begin{pmatrix} A_H & \mathbf{b}_H \\ \mathbf{b}_H^T & \beta \end{pmatrix}, \quad F_B := \begin{pmatrix} B_H & \mathbf{c}_H \\ \mathbf{c}_H^T & \gamma \end{pmatrix}, \quad U_H = \begin{pmatrix} \mathbf{u}_H \\ \alpha \end{pmatrix}, \quad (3.24)$$



---

**Algorithm 3:** Augmented subspace method for Kohn-Sham equation

---

**Data:** The initial value  $(\Lambda_{h_k}^{(0)}, \Psi_{h_k}^{(0)})$

**Result:** The ground state  $(\Lambda_{h_k}^{(\ell)}, \Psi_{h_k}^{(\ell)})$

1 Give  $K_{max} \in \mathbb{N}$ , and  $tol \in \mathbb{R}^+$ , and let  $\ell = 0$ ;

2 **repeat**

3 Solve the following  $N$  problems: Find  $\widehat{\psi}_{i,h_k}^{(\ell+1)} \in S_{h_k}$  such that

$$\left( \widetilde{H}_{\Psi_{h_k}^{(\ell)}} \widehat{\psi}_{i,h_k}^{(\ell+1)}, v_{h_k} \right) = \left( \lambda_{i,h_k}^{(\ell)} + \mu \right) (\psi_{i,h_k}^{(\ell)}, v_{h_k}), \quad \forall v_{h_k} \in S_{h_k}, \quad i = 1, \dots, N. \quad (3.21)$$

4 Solve the following small-scale Kohn-Sham equation in an augmented subspace

$S_{H,h_k} = S_H + \text{span}\{\widehat{\Psi}_{h_k}^{(\ell+1)}\}$ : Find  $(\lambda_{i,h_k}^{(\ell+1)}, \psi_{i,h_k}^{(\ell+1)}) \in \mathbb{R} \times S_{H,h_k}$  such that

$$\left( H_{\Psi_{h_k}^{(\ell+1)}} \psi_{i,h_k}^{(\ell+1)}, v_{h_k} \right) = \lambda_{i,h_k}^{(\ell+1)} (\psi_{i,h_k}^{(\ell+1)}, v_{h_k}), \quad \forall v_{h_k} \in S_{h_k}, \quad i = 1, \dots, N, \quad (3.22)$$

to get the new approximate eigenpair  $(\Lambda_{h_k}^{(\ell+1)}, \Psi_{h_k}^{(\ell+1)})$ ;

5 Update the density function  $\rho_{h_k}^{(\ell+1)} = \sum_{i=1}^N \psi_{i,h_k}^{(\ell+1)}$ ;

6 Let  $\ell = \ell + 1$ ;

7 **until**  $\|\rho_{h_k}^{(\ell)} - \rho_{h_k}^{(\ell-1)}\|_0 < tol$  or  $k > K_{max}$ ;

---

with  $A_H, B_H \in \mathbb{R}^{N_H \times N_H}$ ,  $\mathbf{b}_H, \mathbf{c}_H \in \mathbb{R}^{N_H \times N}$ ,  $\beta, \gamma \in \mathbb{R}^{N \times N}$ . Here,  $U_H = (\mathbf{u}_H, \alpha)^T$  is the eigenvector to be solved in each iteration step, and  $\dim U_H = \dim V_H + N$ ,  $N$  is the number of eigenpair to be solved. The mass matrix  $B_H = ((\phi_{i,H}, \phi_{j,H})_{\mathcal{T}_H})_{1 \leq i, j \leq N_H}$  remains unchanged during the nonlinear iteration, which can be assembled once at the beginning of Algorithm 3. Hereafter,  $(\cdot, \cdot)_{\mathcal{T}}$  denotes the  $L^2$ -inner product on the mesh  $\mathcal{T}$ .

For the matrices  $c_H = ((\widehat{\psi}_{j,h}, \psi_{i,H})_{\mathcal{T}_h})_{1 \leq i \leq N_H, 1 \leq j \leq N}$  and  $\gamma = ((\widehat{\psi}_{j,h}, \widehat{\psi}_{i,h})_{\mathcal{T}_h})_{1 \leq i, j \leq N}$ , the involved basis functions  $\{\widehat{\psi}_{j,h}\}_{1 \leq j \leq N}$  are defined on the fine mesh  $\mathcal{T}_h$ . Thus, to keep the accuracy, these matrices assembling should be performed on the fine mesh  $\mathcal{T}_h$ . As we can see, the basis functions  $\{\widehat{\psi}_{j,h}\}_{1 \leq j \leq N}$  remain unchanged during the nonlinear iteration step, thus the matrices  $c_H$  and  $\gamma$  also can be assembled once at the beginning of Algorithm 3.

It is noted that, based on the above discussion, the mass matrix  $F_B$  only need to be assembled once at the beginning of the nonlinear iteration in Algorithm 3. No updates are required during the iteration.

Next, we discuss the stiffness matrix  $F_A$ . Let us divide the stiffness matrix into linear and nonlinear parts. The linear part  $A_h^L$  is defined by:

$$(A_h^L)_{i,j} = \frac{1}{2} (\nabla \psi_{i,h}, \nabla \psi_{j,h})_{\mathcal{T}_h} + (V_{ext} \psi_{i,h}, \psi_{j,h})_{\mathcal{T}_h} + (\mu \psi_{i,h}, \psi_{j,h})_{\mathcal{T}_h}. \quad (3.25)$$

The nonlinear part  $A_h^{NL}$  consisting of the Hartree potential and exchange-correlation potential is defined by:

$$(A_h^{NL})_{i,j} = ((V_{Har} + V_{xc}) \psi_{i,h}, \psi_{j,h})_{\mathcal{T}_h}, \quad (3.26)$$

where  $V_{Har}$  and  $V_{xc}$  need to be updated in each nonlinear iteration because they depends on the density function that will be updated after each iteration. Let us define  $A_h := A_h^L + A_h^{NL}$ . Using the interpolation algorithm defined in Algorithm 1. The matrix  $A_H$  involved in  $F_A$  can be calculated in the following way

$$A_H = (I_H^h)^T A_h I_H^h, \quad (3.27)$$

where  $I_H^h$  denotes the interpolate operator from  $S_H$  to  $S_h$  based on Algorithm 1.

The remaining parts  $b_H$  and  $\beta$  can be calculated by:

$$b_H = (I_H^h)^T A_h \widehat{\Psi}_h \quad (3.28)$$

and

$$\beta = \widehat{\Psi}_h^T A_h \widehat{\Psi}_h. \quad (3.29)$$

**Remark 3.1.** Based on the assembling process (3.23)–(3.29), the main computational work is spent on (3.26) in each nonlinear iteration. Fortunately, the matrix assembling can be performed in parallel since there is no data transfer, and meanwhile the software package used in this paper has an excellent parallel ability [29, 41], thus we assemble the matrix using parallel computing technique.

### 3.4 Convergence analysis and computational work estimate

In this subsection, we give the convergence analysis and computational complexity estimate for Algorithms 2 and 3. First, we can obtain the following theorem to guarantee the well-posedness of the linear boundary value problems.

**Theorem 3.1.** *With sufficiently small mesh size, there exists a positive constant  $\mu$  such that the following coercive property holds true*

$$a_\mu(\phi, \phi) + (V_{Har}(\rho_{\Psi_h})\phi, \phi) + (V_{xc}(\rho_{\Psi_h})\phi, \phi) \gtrsim \|\nabla\phi\|_0^2, \quad \forall \phi \in H_0^1(\Omega), \quad (3.30)$$

where

$$a_\mu(\phi, \phi) = \frac{1}{2}(\nabla\phi, \nabla\phi) + (V_{ext}\phi, \phi) + \mu(\phi, \phi). \quad (3.31)$$

*Proof.* For the Coulomb potential  $V_{ext} = -\sum_{k=1}^M Z_k/|x - R_k|$ , using the following uncertainty principle lemma [45]:

$$\int_{\Omega} \frac{w^2(x)}{|x|^2} dx \leq 4 \int_{\Omega} |\nabla w|^2 dx, \quad \forall w \in H_0^1(\Omega), \quad (3.32)$$

we obtain

$$\begin{aligned} (V_{ext}\phi, \phi) &\leq \left( \int_{\Omega} \left( \sum_{k=1}^M \frac{Z_k \phi}{|x - R_k|} \right)^2 dx \right)^{1/2} \|\phi\|_0 \\ &\leq \left( \int_{\Omega} \left( \sum_{k=1}^M Z_K^2 \right) \left( \sum_{k=1}^M \frac{\phi^2}{|x - R_k|^2} \right) dx \right)^{1/2} \|\phi\|_0 \\ &\leq 2\sqrt{M} \left( \sum_{k=1}^M Z_K^2 \right)^{1/2} \|\nabla\phi\|_0 \|\phi\|_0 \\ &\leq \frac{1}{8} \|\nabla\phi\|_0^2 + 8M \left( \sum_{k=1}^M Z_K^2 \right) \|\phi\|_0^2. \end{aligned} \quad (3.33)$$

For the Hartree potential, using (3.32) and Hölder inequality, we have

$$\begin{aligned} (V_{Har}(\rho_{\Psi_h})\psi, \psi) &\leq \|r^{-1} * \rho_{\Psi_h}\|_{0,\infty} \|\phi\|_0 \|\phi\|_0 \\ &\leq 2 \|\nabla\Psi_h\|_0 \|\Psi_h\|_0 \|\nabla\phi\|_0 \|\phi\|_0 \\ &\leq \frac{1}{8} \|\nabla\phi\|_0^2 + 8 \|\nabla\Psi_h\|_0^2 \|\Psi_h\|_0^2 \|\phi\|_0^2. \end{aligned} \quad (3.34)$$

For the exchange-correlation potential, we have

$$(V_{xc}(\rho_{\Psi_h})\phi, \phi) = ((V_{xc}(\rho_{\Psi_h}) - V_{xc}(0))\phi, \phi) + (V_{xc}(0)\phi, \phi). \quad (3.35)$$

From Assumption A, we have  $|V_{xc}(0)| \leq a_2$ . For the first part of (3.35), the following estimate holds

$$V_{xc}(\rho_{\Psi_h}) - V_{xc}(0) = 2 \sum_{i=1}^N V'_{xc}(\rho_{\Psi_\varepsilon}) \psi_{\varepsilon,i} \psi_i \quad (3.36)$$

with  $\psi_{\varepsilon,i} = \theta\psi_{i,h} + (1-\theta)0 = \theta\psi_{i,h}$  and  $\theta \in [0, 1]$ . Using Assumption B, we can further derive

$$\begin{aligned} V_{xc}(\rho_{\Psi_h}) - V_{xc}(0) &\leq 2 \left( 1 + \left( \sum_{i=1}^N \psi_{\varepsilon,i}^2 \right)^{\alpha-1} \right) \sum_{i=1}^N \psi_{\varepsilon,i} \psi_{i,h} \\ &= 2 \left( 1 + \left( \sum_{i=1}^N (\theta\psi_{i,h})^2 \right)^{\alpha-1} \right) \sum_{i=1}^N \theta\psi_{i,h}^2 \leq 2\rho_{\Psi_h} + 2\rho_{\Psi_h}^\alpha. \end{aligned} \quad (3.37)$$

Inserting (3.37) into (3.35) and using Holder inequality, we can derive

$$\begin{aligned} (V_{xc}(\rho_{\Psi_h})\phi, \phi) &\leq 2(\rho_{\Psi_h}\phi, \phi) + 2(\rho_{\Psi_h}^\alpha\phi, \phi) + a_2\|\phi\|_0^2 \\ &\leq 2\|\rho_{\Psi_h}\|_{0,3}\|\phi\|_0\|\phi\|_{0,6} + 2\|\rho_{\Psi_h}^\alpha\|_{0,3/\alpha}\|\phi\|_0\|\phi\|_{0,6/(3-2\alpha)} + a_2\|\phi\|_0^2 \\ &\leq 2C_{em}(\|\nabla\Psi_h\|_0^2 + \|\nabla\Psi_h\|_0^{2\alpha})\|\phi\|_0\|\nabla\phi\|_0 + a_2\|\phi\|_0^2, \\ &\leq \frac{1}{8}\|\nabla\phi\|_0^2 + 8C_{em}^2(\|\nabla\Psi_h\|_0^2 + \|\nabla\Psi_h\|_0^{2\alpha})^2\|\phi\|_0^2 + a_2\|\phi\|_0^2, \end{aligned} \quad (3.38)$$

where  $C_{em}$  is the mesh-independent constant involved in embedding theorem  $H_0^1(\Omega) \hookrightarrow L^6(\Omega)$ , when  $\Omega \subset \mathbb{R}^3$ .

Combining (3.33), (3.34) and (3.38), the following inequality holds

$$\begin{aligned} a_\mu(\phi, \phi) + (V_{Har}(\rho_{\Psi_h})\phi, \phi) + (V_{xc}(\rho_{\Psi_h})\phi, \phi) \\ \geq \frac{1}{2}\|\nabla\phi\|_0^2 + \mu\|\phi\|_0^2 - \frac{1}{8}\|\nabla\phi\|_0^2 - \frac{1}{8}\|\nabla\phi\|_0^2 - \frac{1}{8}\|\nabla\phi\|_0^2 - 8M \left( \sum_{k=1}^M Z_K^2 \right) \|\phi\|_0^2 \\ - 8\|\nabla\Psi_h\|_0^2\|\Psi_h\|_0^2\|\phi\|_0^2 - 8C_{em}^2(\|\nabla\Psi_h\|_0^2 + \|\nabla\Psi_h\|_0^{2\alpha})^2\|\phi\|_0^2 - a_2\|\phi\|_0^2. \end{aligned} \quad (3.39)$$

Then we derive the desired result (3.30) by choosing

$$\mu = 8M \left( \sum_{k=1}^M Z_K^2 \right) + 8\|\nabla\Psi_h\|_0^2\|\Psi_h\|_0^2 + 8C_{em}^2(\|\nabla\Psi_h\|_0^2 + \|\nabla\Psi_h\|_0^{2\alpha})^2 + a_2,$$

and the proof is completed.  $\square$

**Theorem 3.2.** Under Assumption A-C, the eigenpair approximation  $(\Lambda_{h_k}^{(\ell+1)}, \Psi_{h_k}^{(\ell+1)})$  derived by by Algorithm 3 has following error estimates

$$\|\Psi - \Psi_{h_k}^{(\ell+1)}\|_1 \lesssim |\Lambda - \Lambda_{h_k}^{(\ell)}| + \|\Psi - \Psi_{h_k}^{(\ell)}\|_0 + \|\Psi - \mathbb{P}_{h_k}\Psi\|_1, \quad (3.40)$$

$$\|\Psi - \Psi_{h_k}^{(\ell+1)}\|_0 + |\Lambda - \Lambda_{h_k}^{(\ell+1)}| \lesssim r(V_{h_k})\|\Psi - \Psi_{h_k}^{(\ell+1)}\|_1, \quad (3.41)$$

where the projection operator  $\mathbb{P}_{h_k} : \mathcal{H} \rightarrow V_{h_k}$  is defined by:  $\mathbb{P}_{h_k} = (P_{h_k})^N$  and

$$a_\mu(P_{h_k}u, v_{h_k}) = a_\mu(u, v_{h_k}), \quad \forall v_{h_k} \in S_{h_k}, \quad \forall u \in H_0^1(\Omega). \quad (3.42)$$

*Proof.* From (2.8) and (3.42), the following equation holds true for  $i = 1, \dots, N$ :

$$a_\mu(P_{h_k}\psi_i, v_{h_k}) = a_\mu(\psi_i, v_{h_k}) = ((\lambda_i + \mu)\psi_i, v_{h_k}) - (V_{Har}(\rho_\Psi)\psi_i, v_{h_k}) - (V_{xc}(\rho_\Psi)\psi_i, v_{h_k}), \quad \forall v_{h_k} \in S_{h_k}. \quad (3.43)$$

Then from (3.21) and (3.43), we obtain

$$\begin{aligned} a_\mu(P_{h_k}\psi_i - \widehat{\psi}_{i,h_k}^{(\ell+1)}, v_{h_k}) + \left( (V_{Har}(\rho_{\Psi_{h_k}^{(\ell)}}) + V_{xc}(\rho_{\Psi_{h_k}^{(\ell)}}))(P_{h_k}\psi_i - \widehat{\psi}_{i,h_k}^{(\ell+1)}), v_{h_k} \right) \\ = ((\lambda_i + \mu)\psi_i - (\lambda_{i,h_k}^{(\ell)} + \mu)\psi_{i,h_k}^{(\ell)}, v_{h_k}) + (V_{Har}(\rho_{\Psi_{h_k}^{(\ell)}})P_{h_k}\psi_i, v_{h_k}) - (V_{Har}(\rho_\Psi)\psi_i, v_{h_k}) \\ + (V_{xc}(\rho_{\Psi_{h_k}^{(\ell)}})P_{h_k}\psi_i, v_{h_k}) - (V_{xc}(\rho_\Psi)\psi_i, v_{h_k}), \quad \forall v_{h_k} \in S_{h_k}. \end{aligned} \quad (3.44)$$

Now let us estimate the equation (3.44) termwise. We deduce from triangle inequality that

$$((\lambda_i + \mu)\psi_i - (\lambda_{i,h_k}^{(\ell)} + \mu)\psi_{i,h_k}^{(\ell)}, v_{h_k}) \lesssim |\lambda_i - \lambda_{i,h_k}^{(\ell)}| \|\psi_i\|_0 \|v_{h_k}\|_0 + (\lambda_{i,h_k}^{(\ell)} + \mu) \|\psi_i - \psi_{i,h_k}^{(\ell)}\|_0 \|v_{h_k}\|_0.$$

For the Hartree potential, using the uncertain principle lemma, the following inequalities hold

$$\begin{aligned} & \left( V_{Har}(\rho_{\Psi_{h_k}^{(\ell)}}) P_{h_k} \psi_i, v_{h_k} \right) - \left( V_{Har}(\rho_{\Psi}) \psi_i, v_{h_k} \right) \\ &= \left( V_{Har}(\rho_{\Psi_{h_k}^{(\ell)}}) (P_{h_k} \psi_i - \psi_i), v_{h_k} \right) + \left( (V_{Har}(\rho_{\Psi_{h_k}^{(\ell)}}) - V_{Har}(\rho_{\Psi})) \psi_i, v_{h_k} \right) \\ &= (r^{-1} * \rho_{\Psi_{h_k}^{(\ell)}}) (P_{h_k} \psi_i - \psi_i) v_{h_k} + (r^{-1} * (\rho_{\Psi_{h_k}^{(\ell)}} - \rho_{\Psi}), \psi_i v_{h_k}) \\ &\lesssim \|r^{-1} * \rho_{\Psi_{h_k}^{(\ell)}}\|_{0,\infty} \|P_{h_k} \psi_i - \psi_i\|_0 \|v_{h_k}\|_0 + \|r^{-1} * (\rho_{\Psi_{h_k}^{(\ell)}} - \rho_{\Psi})\|_{0,\infty} \|\psi_i\|_0 \|v_{h_k}\|_0 \\ &\lesssim \|\Psi_{h_k}^{(\ell)}\|_0 \|\nabla \Psi_{h_k}^{(\ell)}\|_0 \|P_{h_k} \psi_i - \psi_i\|_0 \|v_{h_k}\|_0 + \|\Psi_{h_k}^{(\ell)} - \Psi\|_0 \|\nabla(\Psi_{h_k}^{(\ell)} + \Psi)\|_0 \|\psi_i\|_0 \|v_{h_k}\|_0. \end{aligned} \quad (3.45)$$

For any  $\Gamma \in \mathcal{H}$ , based on Assumption B, the exchange-correlation potential can be estimated by

$$\begin{aligned} & \left( V_{xc}(\rho_{\Psi_{h_k}^{(\ell)}}) \mathbb{P}_{h_k} \Psi, \Gamma \right) - \left( V_{xc}(\rho_{\Psi}) \Psi, \Gamma \right) \\ &= (V_{xc}(\rho_{\Psi_{h_k}^{(\ell)}}) \Psi_{h_k}^{(\ell)}, \Gamma) - (V_{xc}(\rho_{\Psi}) \Psi, \Gamma) + (V_{xc}(\rho_{\Psi_{h_k}^{(\ell)}}) (\mathbb{P}_{h_k} \Psi - \Psi_{h_k}^{(\ell)}), \Gamma) \\ &= 2 \left( V'_{xc}(\rho_{\Psi_{\xi}}) \sum_{i=1}^N \psi_{\xi,i} (\psi_{i,h_k}^{(\ell)} - \psi_i), \sum_{j=1}^N \psi_{\xi,j} \gamma_j \right) + (V_{xc}(\rho_{\Psi_{\xi}}) (\Psi_{h_k}^{(\ell)} - \Psi), \Gamma) \\ &\quad + (V_{xc}(\rho_{\Psi_{h_k}^{(\ell)}}) (\mathbb{P}_{h_k} \Psi - \Psi_{h_k}^{(\ell)}), \Gamma) \\ &\lesssim \left( V'_{xc}(\rho_{\Psi_{\xi}}) \rho_{\Psi_{\xi}}^{1/2} \left( \sum_{i=1}^N (\psi_{i,h_k}^{(\ell)} - \psi_i)^2 \right)^{1/2}, \rho_{\Psi_{\xi}}^{1/2} \left( \sum_{j=1}^N \gamma_j^2 \right)^{1/2} \right) + (V_{xc}(\rho_{\Psi_{\xi}}) (\Psi_{h_k}^{(\ell)} - \Psi), \Gamma) \\ &\quad + (V_{xc}(\rho_{\Psi_{h_k}^{(\ell)}}) (\mathbb{P}_{h_k} \Psi - \Psi_{h_k}^{(\ell)}), \Gamma) \\ &\lesssim \left( V'_{xc}(\rho_{\Psi_{\xi}}) \rho_{\Psi_{\xi}} \left( \sum_{i=1}^N (\psi_{i,h_k}^{(\ell)} - \psi_i)^2 \right)^{1/2}, \left( \sum_{j=1}^N \gamma_j^2 \right)^{1/2} \right) + (V_{xc}(\rho_{\Psi_{\xi}}) (\Psi_{h_k}^{(\ell)} - \Psi), \Gamma) \\ &\quad + (V_{xc}(\rho_{\Psi_{h_k}^{(\ell)}}) (\mathbb{P}_{h_k} \Psi - \Psi_{h_k}^{(\ell)}), \Gamma) \\ &\lesssim \left( (\rho_{\Psi_{\xi}} + \rho_{\Psi_{\xi}}^{\alpha}) \left( \sum_{i=1}^N (\psi_{i,h_k}^{(\ell)} - \psi_i)^2 \right)^{1/2}, \left( \sum_{j=1}^N \gamma_j^2 \right)^{1/2} \right) + (V_{xc}(\rho_{\Psi_{\xi}}) (\Psi_{h_k}^{(\ell)} - \Psi), \Gamma) \\ &\quad + (V_{xc}(\rho_{\Psi_{h_k}^{(\ell)}}) (\mathbb{P}_{h_k} \Psi - \Psi_{h_k}^{(\ell)}), \Gamma), \end{aligned} \quad (3.46)$$

where  $\Psi_{\xi} = \xi \Psi_{h_k}^{(\ell)} + (1 - \xi) \Psi$  with  $\xi \in [0, 1]$ .

For the first part of (3.46), we have

$$\begin{aligned} & \left( (\rho_{\Psi_{\xi}} + \rho_{\Psi_{\xi}}^{\alpha}) \left( \sum_{i=1}^N (\psi_{i,h_k}^{(\ell)} - \psi_i)^2 \right)^{1/2}, \left( \sum_{j=1}^N \gamma_j^2 \right)^{1/2} \right) \\ &\lesssim \left( (\rho_{\Psi_{\xi}} + \rho_{\Psi_{\xi}}^{\alpha}) \left( \sum_{i=1}^N (\psi_{i,h_k}^{(\ell)} - \psi_i)^2 \right)^{1/2}, \sum_{j=1}^N |\gamma_j| \right) \\ &\lesssim \sum_{j=1}^N \|\rho_{\Psi_{\xi}}^{\alpha}\|_{0,3/\alpha} \left\| \left( \sum_{i=1}^N (\psi_{i,h_k}^{(\ell)} - \psi_i)^2 \right)^{1/2} \right\|_{0,\Omega} \|\gamma_j\|_{0,6/(3-2\alpha)} \\ &\quad + \sum_{j=1}^N \|\rho_{\Psi_{\xi}}\|_{0,3} \left\| \left( \sum_{i=1}^N (\psi_{i,h_k}^{(\ell)} - \psi_i)^2 \right)^{1/2} \right\|_{0,\Omega} \|\gamma_j\|_{0,6} \\ &\lesssim \|\Psi_{h_k}^{(\ell)} - \Psi\|_{0,\Omega} \|\Gamma\|_{1,\Omega}. \end{aligned} \quad (3.47)$$

For the second and third parts of (3.46), using the proof technique for (3.38), the following estimate holds

$$(V_{xc}(\rho_{\Psi_{\xi}}) (\Psi_{h_k}^{(\ell)} - \Psi), \Gamma) + (V_{xc}(\rho_{\Psi_{h_k}^{(\ell)}}) (\mathbb{P}_{h_k} \Psi - \Psi_{h_k}^{(\ell)}), \Gamma) \lesssim (\|\Psi - \Psi_{h_k}^{(\ell)}\|_{0,\Omega} + \|\Psi - \mathbb{P}_{h_k} \Psi\|_{0,\Omega}) \|\Gamma\|_{1,\Omega}. \quad (3.48)$$

Using (3.46), (3.47) and (3.48), there holds

$$\left( V_{xc}(\rho_{\Psi_{h_k}^{(\ell)}}) \mathbb{P}_{h_k} \Psi, \Gamma \right) - \left( V_{xc}(\rho_{\Psi}) \Psi, \Gamma \right) \lesssim \left( \|\Psi - \Psi_{h_k}^{(\ell)}\|_{0,\Omega} + \|\Psi - \mathbb{P}_{h_k} \Psi\|_{0,\Omega} \right) \|\Gamma\|_{1,\Omega}. \quad (3.49)$$

Taking  $v_{h_k} = P_{h_k} \psi_i - \widehat{\psi}_{i,h_k}^{(\ell+1)}$  in (3.44), due to Lemma 3.1, (3.45), (3.46) and (3.49), we derive

$$\begin{aligned} \|\mathbb{P}_{h_k} \Psi - \widehat{\Psi}_{h_k}^{(\ell+1)}\|_1^2 &= \sum_{i=1}^N \|P_{h_k} \psi_i - \widehat{\psi}_{i,h_k}^{(\ell+1)}\|_1^2 \\ &\lesssim \sum_{i=1}^N \left\{ a_{\mu}(P_{h_k} \psi_i - \widehat{\psi}_{i,h_k}^{(\ell+1)}, P_{h_k} \psi_i - \widehat{\psi}_{i,h_k}^{(\ell+1)}) \right. \\ &\quad \left. + (V_{Har}(\rho_{\Psi_{h_k}^{(\ell)}})(P_{h_k} \psi_i - \widehat{\psi}_{i,h_k}^{(\ell+1)}), P_{h_k} \psi_i - \widehat{\psi}_{i,h_k}^{(\ell+1)}) \right. \\ &\quad \left. + (V_{xc}(\rho_{\Psi_{h_k}^{(\ell)}})(P_{h_k} \psi_i - \widehat{\psi}_{i,h_k}^{(\ell+1)}), P_{h_k} \psi_i - \widehat{\psi}_{i,h_k}^{(\ell+1)}) \right\} \\ &\lesssim \left( |\Lambda - \Lambda_{h_k}^{(\ell)}| + \|\Psi - \Psi_{h_k}^{(\ell)}\|_0 + \|\Psi - P_{h_k} \Psi\|_0 \right) \|\mathbb{P}_{h_k} \Psi - \widehat{\Psi}_{h_k}^{(\ell+1)}\|_1, \end{aligned} \quad (3.50)$$

which yields

$$\begin{aligned} \|\Psi - \widehat{\Psi}_{h_k}^{(\ell+1)}\|_1 &\lesssim \|\Psi - \mathbb{P}_{h_k} \Psi\|_1 + \|\mathbb{P}_{h_k} \Psi - \widehat{\Psi}_{h_k}^{(\ell+1)}\|_1 \\ &\lesssim \|\Psi - \mathbb{P}_{h_k} \Psi\|_1 + |\Lambda - \Lambda_{h_k}^{(\ell)}| + \|\Psi - \Psi_{h_k}^{(\ell)}\|_0. \end{aligned} \quad (3.51)$$

Since  $S_{H,h}$  is a subspace of  $S_{h_k}$ , the Kohn-Sham equation (3.22) can be regarded as a subspace approximation for (2.9). From Assumption C and (3.51), we have

$$\|\Psi - \Psi_{h_k}^{(\ell+1)}\|_1 \lesssim \|\Psi - \widehat{\Psi}_{h_k}^{(\ell+1)}\|_1 \lesssim \|\Psi - \mathbb{P}_{h_k} \Psi\|_1 + |\Lambda - \Lambda_{h_k}^{(\ell)}| + \|\Psi - \Psi_{h_k}^{(\ell)}\|_0.$$

This is the desired result (3.40). The estimate (3.41) can be obtained by Assumption C, and the proof is completed.  $\square$

Theorem 3.2 shows that the augmented subspace method defined by Algorithm 3 can really improve the accuracy after each iteration, which is the base for designing the nonnested augmented subspace method for Kohn-Sham equation.

Next, we estimate the computational work of the nonnested augmented subspace method. In this paper, we use Mshmet and Mmg3d [17, 20] to realize the moving mesh process. Therefore, the unit mesh principle shall be met [17, 20]. That is, for any  $K \in \mathcal{T}_h$ , there holds

$$e^T \mathcal{M}(k) e = 1, \quad \forall e \in E_K, \quad (3.52)$$

where  $\mathcal{M}(k)$  is the metric on the mesh element  $K$ ,  $E_K$  is the union of the boundaries of  $K$ . Based on the definition of  $\mathcal{M}$  and the unit mesh principle (3.52), we can change the length of each edge through adjusting the value of  $\varepsilon$ , which then leads to the following relationship for the number of degrees of freedom:

$$N_k \approx \eta^{n-k} N_n, \quad k = 1, 2, \dots, n, \quad (3.53)$$

where  $\eta$  denotes the coarsening rate between two consecutive meshes.

Let  $W_k$  denote the computational work in the  $k$ -th finite element space,  $\mathcal{O}(M_{k_0})$  denote the computational work for solving the Kohn-Sham equation in the first  $k_0$  adaptive spaces,  $n$  denote the number of mesh levels,  $\varpi$  denote the nonlinear iteration times,  $\vartheta$  denote the number of processes. We use parallel computing technique to assemble the matrix in step 3 of Algorithm 3, and the resultant algebraic system is solved by the parallel algebraic multigrid method. Thus the computational work for the  $N$  linear boundary value problems is  $\mathcal{O}(NN_k/\vartheta)$ . For the small-scale nonlinear eigenvalue problem involved in step 4 of Algorithm 3, the matrices are assembled through parallel computing technique, and the small-scale algebraic eigenvalue problem is solved in serial computing with computational work  $\mathcal{O}(M_H)$ . Then the total computational work in the  $k$ -th ( $k > k_0$ ) finite element space is

$$W_k = \mathcal{O}\left(N \frac{N_k}{\vartheta} + \varpi \left( \frac{N_k}{\vartheta} + M_H \right)\right). \quad (3.54)$$

**Theorem 3.3.** Assume that the number of degrees of freedom on two consecutive meshes satisfies the proportional relationship (3.53). Then the total computational work of Algorithm 3 can be estimated as follows

$$W_{total} = \mathcal{O}\left(\frac{N + \varpi}{\vartheta} N_n + \varpi \log N_n M_H + M_{k_0}\right). \quad (3.55)$$

*Proof.* Based on the computational work (3.54) in one finite element space and the proportional relationship (3.53), we have

$$\begin{aligned} W_{total} &= \sum_{k=1}^N W_k = \mathcal{O}\left(M_{k_0} + \sum_{k=k_0+1}^n \left(N \frac{N_k}{\vartheta} + \varpi \left(\frac{N_k}{\vartheta} + M_H\right)\right)\right) \\ &= \mathcal{O}\left(\frac{N + \varpi}{\vartheta} \sum_{k=k_0+1}^n N_k + \varpi \log N_n M_H + M_{k_0}\right) \\ &= \mathcal{O}\left(\frac{N + \varpi}{\vartheta} N_n + \varpi \log N_n M_H + M_{k_0}\right). \end{aligned} \quad (3.56)$$

Then the proof is completed.  $\square$

**Remark 3.2.** For complicated molecular system, the number of the desired eigenpairs ( $N$ ) and the number of nonlinear iteration times ( $\varpi$ ) will be large. But fortunately, the software package FreeFEM [29] has a good scalability. Thus, when the number of processes  $\vartheta$  is large, we can still derive an asymptotically optimal computational work estimate.

## 4 Numerical experiments

In this section, we provide four numerical examples to validate the efficiency of the proposed nonnested augmented subspace method in this paper. These numerical examples are implemented on the high performance computing platform LSSC-IV in the State Key Laboratory of Scientific and Engineering Computing, Chinese Academy of Sciences. Each computing node has two 18-core Intel Xeon Gold 6140 processors at 2.3 GHz and 192 GB memory. All the parallel solving processes in this paper use 72 cores. We adopt the open source finite element software package FreeFEM [23, 29] to do the numerical simulation. The anisotropic measurement is generated by the Meshmet of Hessian matrix based on adaptive function (density function), and the adaptive mesh is generated by the software package Mmg3d [17, 20]. The involved linear boundary value problems are solved by PETSc-GAMG [5], and the linear eigenvalue problems are solved by Krylov-Schur method [25] in SLEPc [46]. In all the numerical experiments, we set the threshold in Algorithm 2 to be  $K_{asm} = 4$ , and set the positive constant  $\mu$  in Algorithm 3 to be  $\mu = 8M\left(\sum_{k=1}^M Z_K^2\right)$ .

For the nonlinear eigenvalue problems, we use the Anderson mixing scheme to accelerate the convergence rate [4]. The detailed process can be described as follows: Let  $\rho_{h,in}^{(m)}$  and  $\rho_{h,out}^{(m)}$  represent the input and output electron densities of the  $m$ -th self-consistent iteration. To input the  $(m+1)$ -th self-consistent iteration,  $\rho_{h,in}^{(m+1)}$  is computed as follows:

$$\rho_{h,in}^{(m+1)} = \beta \tilde{\rho}_{h,out} + (1 - \beta) \tilde{\rho}_{h,in}, \quad (4.1)$$

where

$$\tilde{\rho}_{h,in(out)} = \alpha_m \rho_{h,in(out)}^{(m)} + \sum_{j=m_0}^{m-1} \alpha_j \rho_{h,in(out)}^{(j)}, \quad (4.2)$$

and the sum of the coefficients equals to one, i.e.,

$$\alpha_{m_0} + \alpha_{m_0+1} + \cdots + \alpha_m = 1. \quad (4.3)$$

According to (4.3), the equation (4.2) can be written as

$$\tilde{\rho}_{h,in(out)} = \rho_{h,in(out)}^{(m)} + \sum_{j=m_0}^{m-1} \alpha_j (\rho_{h,in(out)}^{(j)} - \rho_{h,in(out)}^{(m)}), \quad (4.4)$$

Here the positive integer  $m_0 \in [1, m]$ , and the depth of the mixing scheme is defined by  $\text{depth} := m - m_0 + 1$ ,  $\beta$  is a given weight value.

Denoting  $F^{(m)} = \rho_{h,\text{out}}^{(m)} - \rho_{h,\text{in}}^{(m)}$  and  $\tilde{F} = \tilde{\rho}_{h,\text{out}} - \tilde{\rho}_{h,\text{in}}$ . Based on (4.4), there holds

$$\tilde{F} = F^{(m)} + \sum_{j=m_0}^{m-1} \alpha_j (F^{(j)} - F^{(m)}). \quad (4.5)$$

The coefficients  $\alpha_j$  are determined by minimizing  $\|\tilde{F}\|_2^2 = \|\tilde{\rho}_{h,\text{in}} - \tilde{\rho}_{h,\text{out}}\|_2^2$ , which amounts to solve the following equations:

$$\sum_{k=m_0}^{m-1} (F^{(m)} - F^{(j)}, F^{(m)} - F^{(k)}) \alpha_k = (F^{(m)} - F^{(j)}, F^{(m)}), \quad j = m_0, \dots, m-1. \quad (4.6)$$

In our numerical experiments, we choose the  $\text{depth} = 5$ , the damping coefficient  $\beta = 0.7$ .

### 4.1 Kohn-Sham equation for Helium

In the first example, we solve the following Kohn-Sham equation for Helium:

$$\begin{cases} \left( -\frac{1}{2} \Delta - \frac{2}{|x|} + \int_{\Omega} \frac{\rho(y)}{|x-y|} dy + V_{xc} \right) \psi = \lambda \psi, & \text{in } \Omega, \\ \psi = 0, & \text{on } \partial\Omega, \\ \int_{\Omega} |\psi|^2 dx = 1, \end{cases} \quad (4.7)$$

where  $\Omega = (-10, 10)^3$ . In this example, the electron density  $\rho = 2|\psi|^2$ . The exchange-correlation potential is chosen according to (3.2) and (3.3). The experimental value of non-relativistic ground state energy of Helium atom is -2.90372 hartree [50].

Here, we use three types of algorithms to solve the Kohn-Sham equation (4.7). The first one uses the self-consistent field iteration to solve (4.7) directly on the fixed structure mesh. The second one uses the self-consistent field iteration to solve (4.7) directly on the adaptive moving mesh (i.e. step 3 of Algorithm 2 is solved directly by the self-consistent field iteration). The third one uses the nonnested augmented subspace method to solve (4.7). In the moving mesh adaptive process, the metric matrix is given by the modified Hessian matrix (3.8) of  $\rho^{1/2}(x)$ . The tolerance in Algorithm 2 is set to be  $1E-3$ . The tolerance in Algorithm 3 is set to be  $2E-4$ .

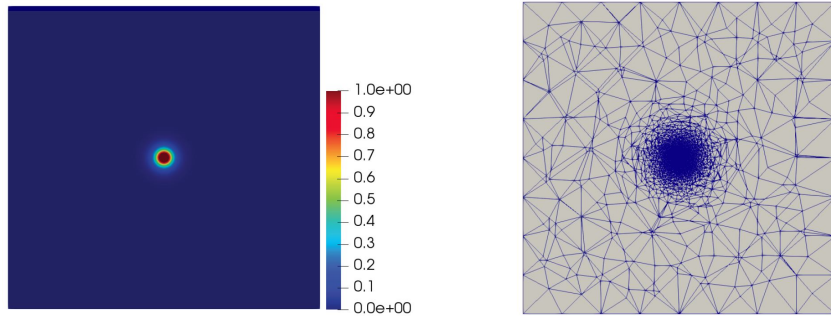


Figure 1: Contour plot of the electron density (left) and the adaptive moving mesh (right) for Example 1

The density function and the adaptive mesh of the nonnested augmented subspace method are presented in Figure 1. The ground state energy and the corresponding error estimates derived by these three methods are presented in Figure 2. It can be seen from Figure 2 that the ground state energy of Helium atom first decreases quickly and then tends to be stable with the adaptive refinement of the mesh. Besides, we also can see that the adaptive iterative methods have a better accuracy than the finite element method with the uniform mesh.



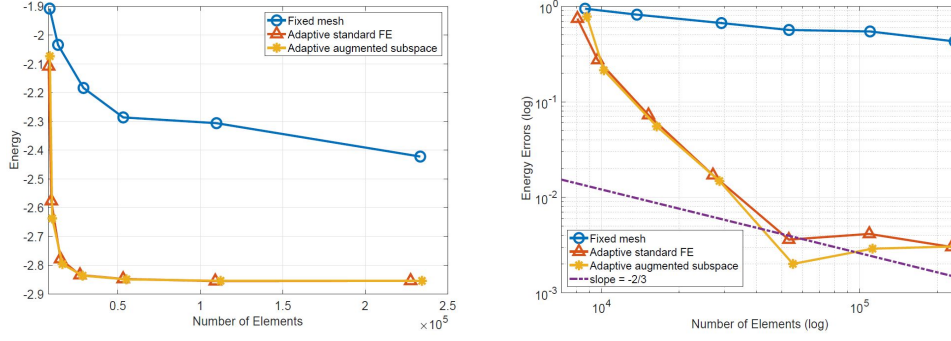


Figure 2: The ground state energy (left) and the corresponding error estimates (right) for Example 1

## 4.2 Kohn-Sham equation for Hydrogen-Lithium

In the second example, we solve the following Kohn-Sham equation for Hydrogen-Lithium:

$$\begin{cases} \left( -\frac{1}{2}\Delta - \frac{3}{|x-r_1|} - \frac{1}{|x-r_2|} + \int_{\Omega} \frac{\rho(y)}{|x-y|} dy + V_{xc} \right) \psi_i = \lambda_i \psi_i, & \text{in } \Omega, \quad i = 1, 2, \\ \psi_i = 0, & \text{on } \partial\Omega, \quad i = 1, 2, \\ \int_{\Omega} \psi_i \psi_j d\Omega = \delta_{i,j}, & i, j = 1, 2, \end{cases} \quad (4.8)$$

where  $\Omega = (-10, 10)^3$ . In this equation, the electron density  $\rho = 2(|\psi_1|^2 + |\psi_2|^2)$ ,  $r_j$  ( $j = 1, 2$ ) is the position of lithium atom and hydrogen atom and we take  $r_1 = (-1.0075, 0, 0)$ ,  $r_2 = (2.0075, 0, 0)$ . The exchange-correlation potential is chosen according to (3.2) and (3.3). The experimental value of non relativistic ground state energy of Hydrogen-Lithium molecular system is -8.070 hartree [12]. In this example, we set the tolerance in Algorithm 2 to be  $1E - 3$  and tolerance in Algorithm 3 to be  $2E - 4$ . We also use these three numerical methods as described in the first example to solve (4.8).

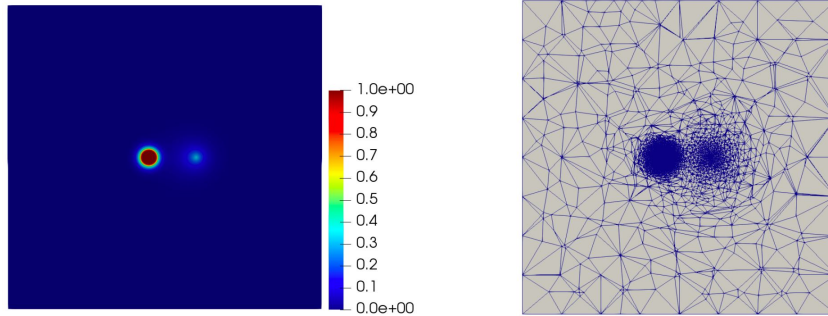


Figure 3: Contour plot of the electron density (left) and the adaptive moving mesh (right) for Example 2

The density function and the adaptive mesh for Hydrogen-Lithium are presented in Figure 3. The ground state energy and the corresponding error estimates derived by these three methods are presented in Figure 4. It can be seen from Figure 4 that the ground state energy of Hydrogen-Lithium decreases quickly to a stable constant with the adaptive refinement of the mesh. Besides, we also can see that the adaptive iterative methods have a better accuracy than the finite element method with uniform mesh. At the same time, it can be seen that the convergence rate of Algorithm 3 tends to  $2/3$  with mesh refinement.

In Table 1, we present the computational time of the nonnested augmented subspace method and the direct adaptive finite element method (i.e. solve the Kohn-Sham equation directly using the moving mesh adaptive technique).

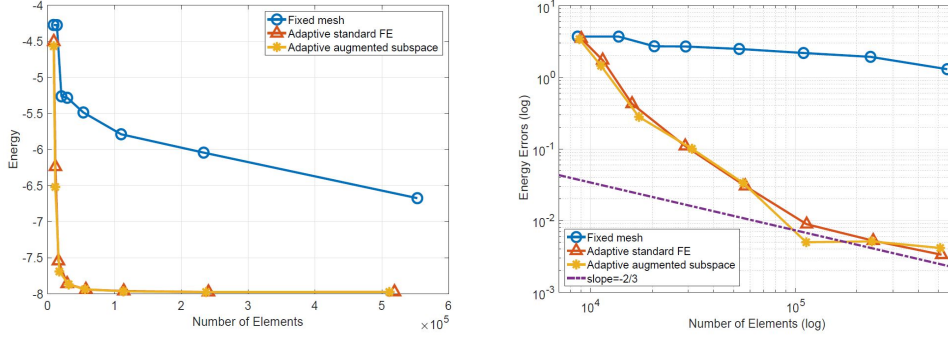


Figure 4: The ground state energy (left) and the corresponding error estimates (right) for Example 2

Elements (Direct method)	9008	11463	15943	29015	57078	113729	240435	518786
Time (Direct method)	7.024	10.231	14.188	18.587	<b>23.694</b>	<b>32.339</b>	<b>46.215</b>	<b>74.595</b>
Elements (Algorithm 3)	8819	11240	17235	31297	55690	113168	236780	510949
Time (Algorithm 3)	8.642	11.176	15.370	19.260	<b>23.340</b>	<b>30.560</b>	<b>38.086</b>	<b>49.016</b>

Table 1: CPU time (in seconds) for the nonnested augmented subspace method and the direct adaptive finite element method.

It can be seen that the nonnested augmented subspace method is more efficient when the number of mesh elements reaches 55690 or more, and the advantage is more obvious with the increase of the number of mesh elements.

### 4.3 Kohn-Sham equation for Methane

In the third example, we consider the Kohn-Sham equation for Methane. To show the efficiency of the nonnested augmented subspace method intuitively, we also use these three methods described in Example 4.1 to solve the Methane molecules. The computational domain is set to be  $\Omega = (-10, 10)^3$ , and the atomic positions of Methane are shown in Table 2.

Atom	x	y	z	Nuclear charge
C <sub>1</sub>	0.0000	0.0000	0.0000	6
H <sub>2</sub>	1.3092	1.3092	1.3092	1
H <sub>3</sub>	-1.3092	-1.3092	1.3092	1
H <sub>4</sub>	1.3092	-1.3092	-1.3092	1
H <sub>5</sub>	-1.3092	1.3092	-1.3092	1

Table 2: The position and the nuclear charge of each atom in Methane.

For a full potential calculation, there are total ten electrons. Since we don't consider spin polarisation, five eigenpairs need to be calculated. The tolerance setting in Algorithms 2 and 3 is the same as that of Example 4.1. The reference value of non relativistic ground state energy of methane molecule is set to be -40.41 hartree [28].

Figure 5 shows the electron density distribution and adaptive mesh of Methane molecule. Figure 6 shows the change curve of ground state energy and the corresponding error estimates for Methane molecule with the increase of the number of mesh elements.

Table 3 shows the computational time of the nonnested augmented subspace method and the direct adaptive finite element method. It can be seen that when the number of mesh elements reaches 499064, the nonnested augmented subspace method has an advantages in solving efficiency over the direct adaptive finite element method.

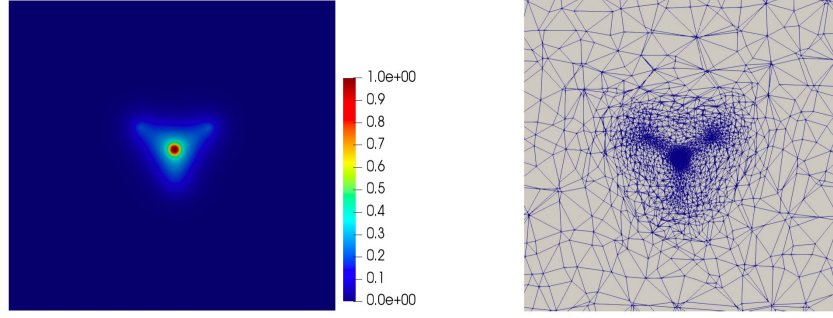


Figure 5: Contour plot of the electron density (left) and the adaptive moving mesh (right) for Example 3

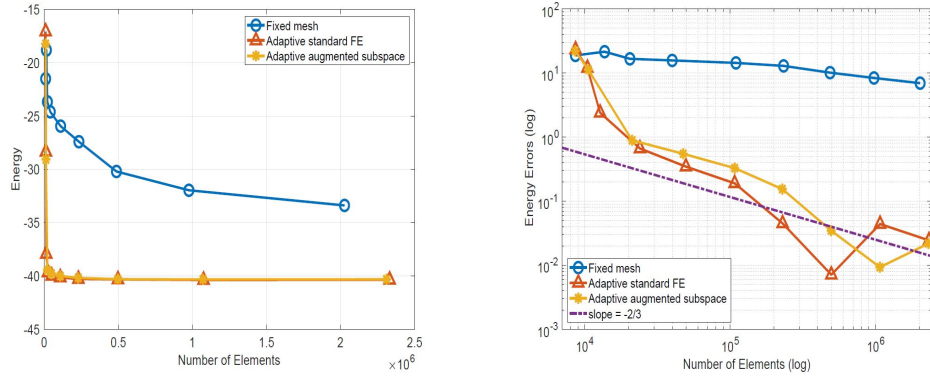


Figure 6: The ground state energy (left) and the corresponding error estimates (right) for Example 3

Elements (Direct method)	10208	11496	22267	45982	102415	225074	492919	1072524	2329210
Time (Direct method)	10.904	15.141	20.737	29.902	42.412	68.716	<b>140.839</b>	<b>312.876</b>	<b>756.633</b>
Elements (Algorithm 3)	10545	10599	22682	51646	108014	230730	499064	1083722	2342607
Time (Algorithm 3)	10.762	14.027	19.386	30.579	53.005	70.812	<b>86.976</b>	<b>149.184</b>	<b>209.735</b>

Table 3: CPU time (in seconds) for the nonnested augmented subspace method and the direct adaptive finite element method.

#### 4.4 Kohn-Sham equation for Benzene

In the last example, we consider the Kohn-Sham equation for the Benzene molecule. The computational domain is taken as  $\Omega = (-10, 10)^3$ , and the atomic positions of Methane are shown in Table 4.

Atom	x	y	z	Nuclear charge
C <sub>1</sub>	0.0000	1.3970	0.0000	6
C <sub>2</sub>	1.2098	0.6985	0.0000	6
C <sub>3</sub>	1.2098	-0.6985	0.0000	6
C <sub>4</sub>	0.0000	-1.3970	0.0000	6
C <sub>5</sub>	-1.2098	-0.6985	0.0000	6
C <sub>6</sub>	-1.2098	0.6985	0.0000	6
H <sub>7</sub>	0.0000	2.4810	0.0000	1
H <sub>8</sub>	2.1486	1.2405	0.0000	1
H <sub>9</sub>	2.1486	-1.2405	0.0000	1
H <sub>10</sub>	0.0000	-2.4810	0.0000	1
H <sub>11</sub>	-2.1486	-1.2405	0.0000	1
H <sub>12</sub>	-2.1486	1.2405	0.0000	1

Table 4: The position and the nuclear charge number of each atom in Benzene.

For a full potential calculation, there are total 42 electrons. Since we don't consider spin polarisation, 21 eigenpairs need to be calculated. The tolerance setting in Algorithms 2 and 3 is the same as that of Example 4.1. The reference value of non relativistic ground state energy of methane molecule is set to be -231.78 hartree [28].

Elements (Direct method)	8946	13095	21675	58635	198145	535501	1245081	2789529
Time (Direct method)	19.079	28.454	40.958	63.495	<b>109.606</b>	<b>223.390</b>	<b>470.353</b>	<b>1027.426</b>
Elements (Algorithm 3)	8897	12505	20535	62348	207362	550236	1262297	2796022
Time (Algorithm 3)	14.235	21.239	35.367	62.337	<b>79.789</b>	<b>133.229</b>	<b>187.119</b>	<b>294.440</b>

Table 5: CPU time (in seconds) for the nonnested augmented subspace method and the direct adaptive finite element method.

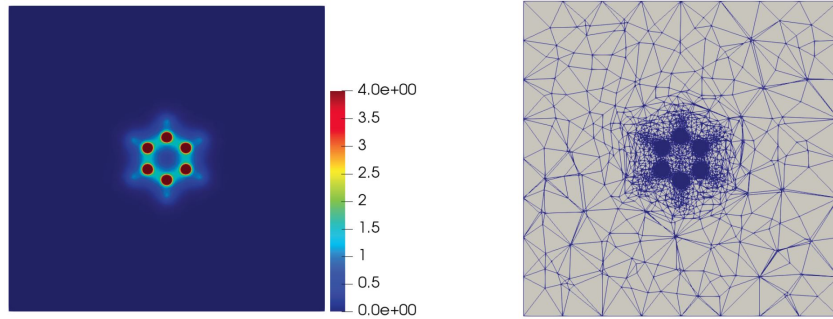


Figure 7: Contour plot of the electron density (left) and the adaptive moving mesh (right) for Example 4

Three numerical methods described in Example 4.1 are tested in this example. Figure 7 shows the electron density and adaptive mesh of the nonnested augmented subspace method. It can be seen from Figure 7 that the mesh nodes gather in the region with singular electron density (near the atomic centers), while in the region far away from the atomic centers, the mesh node distribution is sparse, which is consistent with the distribution of electron density function.

Figure 8 shows the decline curve of ground state energy and corresponding error estimates with mesh adaptive movement. From Figure 8, we can see that compared with the moving mesh adaptive method (red line and yellow

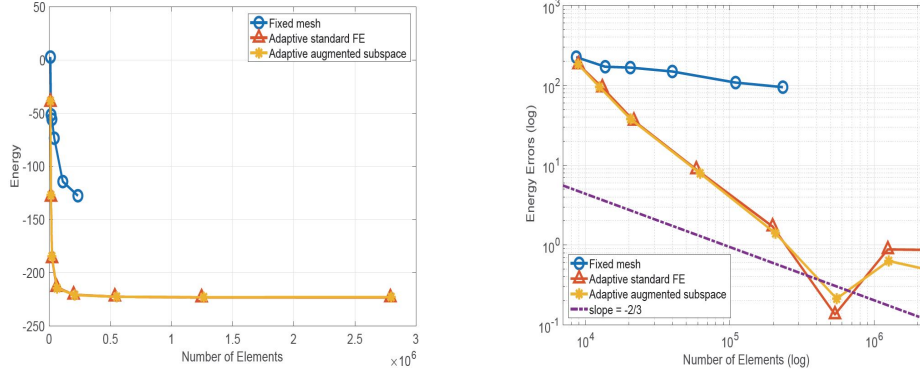


Figure 8: The ground state energy (left) and the corresponding error estimates (right) for Example 4

line), the finite element method performed on the fixed structural mesh (blue line) derives a slower convergence rate. Moreover, when the number of mesh elements reaches 233280, the nonlinear iteration on fixed structure mesh still does not converge even if the Anderson acceleration technology is adopted.

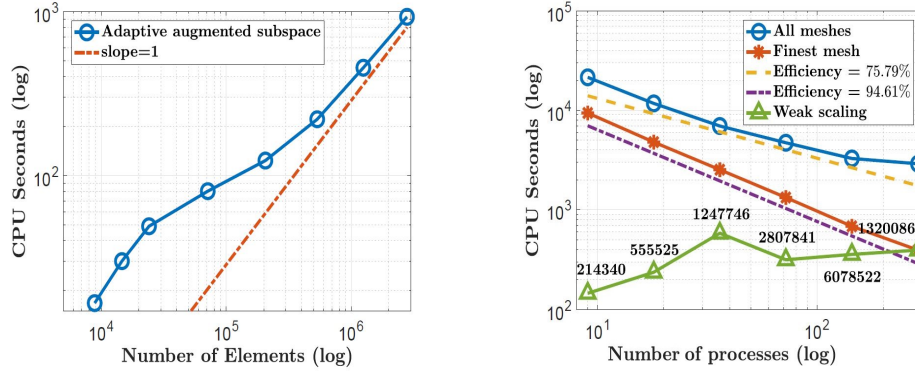


Figure 9: The computational time and parallel scalability of the nonnested augmented subspace method for Example 4

Table 5 compares the computational time of the nonnested augmented subspace method and the direct adaptive finite element method. It can be seen that when the number of mesh elements reaches 207362, the nonnested augmented subspace method has advantage in solving efficiency over the direct adaptive finite element method.

In addition, we also present the computational time of the nonnested augmented subspace method for Benzene in Figure 9 to test its linear complexity and the parallel scalability. Figure 9 shows that when the number of mesh elements reaches 500000, the nonnested augmented subspace method has a linear computational complexity with the increase of the number of mesh elements. The strong scalability and weak scalability is also presented in Figure 9. In Figure 9, the label “All meshes” (blue line) represents the total computational time from the coarsest mesh to the finest mesh. The labeled “finest mesh” (red line) represents only the computational time in the finest mesh, The green line is the result of weak scalability, and 9, 18, 36, 72, 144 and 288 processes are used, respectively. From Figure 9, we can find that the nonnested augmented subspace method has a good scalability.

## 5 Concluding remarks

In this paper, a nonnested augmented subspace method is proposed to solve the Kohn-Sham equation based on the moving mesh adaptive technique and augmented subspace method. Because the moving mesh adaptive technology is used to generate nonnested mesh sequence, the redistributed mesh is almost optimal. In the solving process, we transform the classical nonlinear eigenvalue problem on a fine mesh into a series of linear boundary value problems

of the same scale on the adaptive mesh sequence, and subsequently correct the approximate solutions by solving the small-scale nonlinear eigenvalue problems in a special low-dimensional augmented subspace. Since solving large-scale nonlinear eigenvalue problem is avoided which is time-consuming, the total solving efficiency can be greatly improved. Some numerical experiments are also presented to verify the computational efficiency and energy accuracy of the algorithm proposed in this paper.

## 6 Acknowledgement

This research is supported partly by National Key R&D Program of China 2019YFA0709600, 2019YFA0709601, National Natural Science Foundations of China (Grant Nos. 11922120, 11871489 and 11771434), FDCT of Macao SAR (0082/2020/A2), MYRG of University of Macau (MYRG2020-00265-FST), the National Center for Mathematics and Interdisciplinary Science, CAS, Beijing Municipal Natural Science Foundation (Grant No. 1232001), and the General projects of science and technology plan of Beijing Municipal Education Commission (Grant No. KM202110005011).

## References

- [1] R. A. Adams, *Sobolev spaces*, Academic Press, New York, 1975.
- [2] F. Alauzet and P.J. Frey, *Estimateur d'erreur géométrique et métriques anisotropes pour l'adaptation de maillage. partie i : aspects théoriques*, Rapport de recherche RR-4759, INRIA, 2003.
- [3] R. Alexander, P. Manselli and K. Miller, *Moving finite elements for the Stefan problem in two dimensions*, Ren. Accad., 1979, Naz. Lincei (Rome) LXVII:57–61.
- [4] D. Anderson, *Iterative procedures for nonlinear integral equations*, J. Assoc. Comput. Mach., 12 (1965), 547–560.
- [5] S. Balay, S. Abhyankar, M. Adams, et al., *PETSc users manual*, Technical Report ANL-95/11- Revision 3.13, 2020.
- [6] G. Bao, G. Hu and D. Liu, *Numerical solution of the Kohn-Sham equation by finite element methods with an adaptive mesh redistribution technique*, J. Sci. Comput., 55(2) (2012), 372–391.
- [7] T. L. Beck, *Real-space mesh techniques in density functional theory*, Rev. Modern Phys., 72 (2000), 1041–1080.
- [8] E. J. Bylaska, M. Host and J. H. Weare, *Adaptive finite element method for solving the exact Kohn-Sham equation of density functional theory*, J. Chem. Theory Comput., 5 (2009), 937–948.
- [9] E. Cancès, R. Chakir and Y. Maday, *Numerical analysis of nonlinear eigenvalue problems*, J. Sci. Comput., 45(1–3) (2010), 90–117.
- [10] E. Cancès, R. Chakir and Y. Maday, *Numerical analysis of the planewave discretization of some orbital-free and Kohn-Sham models*, ESAIM Math. Model. Numer. Anal., 46 (2012), 341–388.
- [11] A. Castro, H. Appel, M. Oliveira, C. A. Rozzi, X. Andrade, F. Lorenzen, M. A. L. Marques, E. K. U. Gross, A. Rubio, *Octopus: A tool for the application of time-dependent density functional theory*, Phys. Status Solidi B, 243 (2006), 2465–2488.
- [12] W. Cencek and J. Rychlewski, *Benchmark calculations for  $He^{2+}$  and LiH molecules using explicitly correlated Gaussian functions*, Chem. Phys. Lett., 320 (2000), 549–552.
- [13] J. R. Chelikowsky, N. Troullier and Y. Saad, *Finite-difference pseudopotential method: Electronic structure calculations without a basis*, Phys. Rev. Lett., 72 (1994), 1240–1243.
- [14] H. Chen, X. Dai, X. Gong, L. He, Z. Yang and A. Zhou, *Adaptive finite element approximations for Kohn-Sham models*, Multi. Model. Simul., 12(4) (2014), 1828–1869.
- [15] H. Chen, L. He and A. Zhou, *Finite element approximations of nonlinear eigenvalue problems in quantum physics*, Comput. Methods Appl. Mech. Engrg., 200(21) (2011), 1846–1865.

- [16] H. Chen, X. Gong, L. He, Z. Yang and A. Zhou, *Numerical analysis of finite dimensional approximations of Kohn-Sham models*, Adv. Comput. Math., 38 (2013), 225–256.
- [17] C. Dapogny, C. Dobrzynski and P.J. Frey, *Three-dimensional adaptive domain remeshing, implicit domain meshing, and applications to free and moving boundary problems*, J. Comput. Phys., 262 (2014), 358–378.
- [18] Y. Di, R. Li, T. Tang, et al., *Moving mesh finite element method for the incompressible Navier-Stokes equations*, SIAM J. Sci. Comput., 26(3) (2005), 1036–1056.
- [19] Y. Di, H. Xie and X. Yin, *Anisotropic meshes and stabilization parameter design of linear SUPG method for 2D convection-dominated convection-diffusion equations*, J. Sci. Comput., 76(1) (2018), 48–68.
- [20] P.J. Frey and F. Alauzet, *Anisotropic mesh adaptation for CFD computations*, Comput. Methods Appl. Mech. Engrg, 194 (2005), 5068–5682.
- [21] W. J. Hehre, R. F. Stewart and J. A. Pople, *Self-consistent molecular-orbital methods. I. Use of Gaussian expansions of Slater-type atomic orbitals*, J. Chem. Phys., 51 (1969), 2657–2664.
- [22] F. Jensen, *Introduction to Computational Chemistry*, Wiley Publishers, 1999.
- [23] F. Hecht, *FreeFEM documentation release 4.6*, 2020, <https://doc.freefem.org/pdf/FreeFEM-documentation.pdf>.
- [24] F. Hecht, *Bidimensional anisotropic mesh generator. Technical report, INRIA, Roc-quencourt*, 1997. Sourcecode: <http://www-rocq1.inria.fr/gamma/cdrom/www/bamg/eng.htm>.
- [25] V. Hernandez, J.E. Roman, A. Tomas, et al., *Krylov-Schur methods in SLEPc*, SLEPc Technical Report STR-7, 2007. <http://slepc.upv.es>.
- [26] P. Hohenberg and W. Kohn, *Inhomogeneous electron gas*, Phys. Rev., 136 (1964), B864–B871.
- [27] S. Jia, H. Xie, M. Xie and F. Xu, *A full multigrid method for nonlinear eigenvalue problems*, Science China: Mathematics, 59 (2016), 2037–2048.
- [28] R. Johnson, *Nist computational chemistry comparison and benchmark database*, 2011, <http://cccbdb.nist.gov>.
- [29] P. Jolivet, F. Hecht, F. Nataf, et al., *Scalable domain decomposition preconditioners for heterogeneous elliptic problems*, Proceedings of the 2013 ACM/IEEE conference on Supercomputing. SC13. Best paper finalist. ACM, 80 (2013), 1–11.
- [30] W. Kohn and L. Sham, *Self-consistent equations including exchange and correlation effects*, Phys. Rev. A, 140 (1965), 1133–1138.
- [31] R. Li, T. Tang and P.W. Zhang, *Moving mesh methods in multiple dimensions based on harmonic maps*, J. Comput. Phys., 170 (2001), 562–588.
- [32] R. Li, T. Tang and P.W. Zhang, *A moving mesh finite element algorithm for singular problems in two and three space dimensions*, J. Comput. Phys., 177 (2002), 365–393.
- [33] L. Lin, J. Lu, L. Ying and W. E, *Adaptive local basis set for Kohn-Sham density functional theory in a discontinuous Galerkin framework I: Total energy calculation*, J. Comput. Phys., 231 (2012), 2140–2154.
- [34] Q. Lin and H. Xie, *A multi-level correction scheme for eigenvalue problems*, Math. Comp., 84(291) (2015), 71–88.
- [35] B. Liu, H. Chen, G. Dusson, J. Fang and X. Gao, *An adaptive planewave method for electronic structure calculations*, 2021, arXiv:2012.14806v2.
- [36] A. Masud and R. Kannan, *B-splines and NURBS based finite element methods for Kohn-Sham equations*, Comput. Methods Appl. Mech. Engrg., 241-244 (2012), 112–127.
- [37] K. Miller and R.N. Miller, *Moving finite element methods I*, SIAM J. Numer. Anal., 18 (1981) 1019–1032.
- [38] K. Miller, *Moving finite element methods II*, SIAM J. Numer. Anal., 18 (1981), 1033–1057.



- [39] N. A. Modine, G. Zumbach and E. Kaxiras, *Adaptive-coordinate real-space electronic-structure calculations for atoms, molecules and solids*, Phys. Rev. B, 55 (1997), 289–301.
- [40] P. Motamarri, M. Nowak, K. Leiter, J. Knap and V. Gavini, *Higher-order adaptive finite-element methods for Kohn-Sham density functional theory*, J. Comput. Phys., 253 (2013), 308–343.
- [41] J. Moulin, P. Jolivet and O. Marquet, *Augmented Lagrangian preconditioner for large-scale hydrodynamic stability analysis*, Comput. Methods Appl. Mech. Engrg., 351 (2019), 718–743.
- [42] J. E. Pask, B. M. Klein, P. A. Sterne and C. Y. Fong, *Finite element methods in electronic-structure theory*, Comput. Phys. Comm., 135 (2001), 1–34.
- [43] J. E. Pask and P. A. Sterne, *Finite element methods in ab initio electronic structure calculations*, Model. Simul. Mater. Sci. Eng., 13 (2005), R71–R96.
- [44] J. Perdew, A. Zunger, *Self-interaction correction to density-functional approximation for many-electron systems*, Phys. Rev., 23 (1981), 5048–5079.
- [45] M. Reed and B. Simon, *Methods of Modern Mathematical Physics*, Academic Press, New York, 1975.
- [46] J.E. Roman, C. Campos, E. Romero, et al., *Krylov-Schur methods in SLEPc*, Technical Report DSIC-II/24/02, 2020. <http://slepc.upv.es>.
- [47] V. Schauer and C. Linder, *All-electron Kohn-Sham density functional theory on hierarchic finite element spaces*, J. Comput. Phys., 250 (2013), 644–664.
- [48] C. K. Skylaris, P. D. Haynes, A. A. Mostofi and M. C. Payne, *Introducing ONETEP: Linear-scaling density functional simulations on parallel computers*, J. Chem. Phys., 122 (2005), 084119.
- [49] P. Suryanarayana, V. Gavini, T. Blesgen, K. Bhattacharya and M. Ortiz, *Non-periodic finite-element formulation of Kohn-Sham density functional theory*, J. Mech. Phys. Solids, 58 (2010), 256–280.
- [50] A. Veillard and E. Clementi, *Correlation energy in atomic systems. V. Degeneracy effects for the second row atoms*, J. Chem. Phys., 49 (1968), 2415–2421.
- [51] H. Xie, *A multigrid method for eigenvalue problem*, J. Comput. Phys., 274 (2014), 550–561.
- [52] H. Xie, *A multigrid method for nonlinear eigenvalue problems*, Science China: Mathematics (Chinese), 45(8) (2015), 1193–1204.
- [53] H. Xie and X. Yin, *Metric tensors for the interpolation error and its gradient in  $L^p$  norm*, J. Comput. Phys., 256 (2014), 543–562.
- [54] W.Z. Huang and R.D. Russell, *Adaptive moving mesh methods*, Springer, New York, 2011.