Comparing two iteration algorithms of Broyden electron density mixing through an atomic electronic structure computation*

Man-Hong Zhang(张满红)†

School of Electricial and Electronics Engineering, North China Electric Power University, Beijing 102206, *China*

(Received 23 November 2015; revised manuscript received 19 January 2016; published online 5 April 2016)

By performing the electronic structure computation of a Si atom, we compare two iteration algorithms of Broyden electron density mixing in the literature. One was proposed by Johnson and implemented in the well-known VASP code. The other was given by Eyert. We solve the Kohn-Sham equation by using a conventional outward/inward integration of the differential equation and then connect two parts of solutions at the classical turning points, which is different from the method of the matrix eigenvalue solution as used in the VASP code. Compared to Johnson's algorithm, the one proposed by Eyert needs fewer total iteration numbers.

Keywords: self-consistent field, electron density mixing, Broyden algorithm, density functional

PACS: 31.15.A–, 31.15.xr **DOI:** 10.1088/1674-1056/25/5/053102

1. Introduction

In modern density functional computation of electronic structures, the exchange correlation potential is a functional of the electron density.[1] If the latter is known, we can further determine a lot of the ground state properties, such as total energy and its derivatives with respect to other variables such as atom positions, strain or electric field.[2] However, we can only set up an approximate electron density from the start, then calculate the exchange-correlation potential to solve the Kohn–Sham (KS) equation[1] to get single-particle states and build a new electron density through the Fermi–Dirac distribution.[3,4] Mixing the previous and current output electron density to generate a new input electron density and the exchange-correlation potential for the next cycle of solving the Kohn–Sham equation. The iteration continues till a converged electron density and the single-particle energy eigenvalues are obtained. Mixing the electron density properly and solving the Kohn–Sham equation either in matrix eigenvalue form or in differential equation form consist of a self-consistent field (SCF) computation process.

In crystal solids, we may use the space periodicity to expand the electron density and the potential over the discrete reciprocal lattice vectors in the reciprocal space, and use the plane wave basis or other proper basis functions to transform the KS equation into a matrix-eigenvalue form. In this way we can fully adopt the linear algebra methods to get energy bands and wave functions.[3,4] The VASP code has been implemented in the following density mixing schemes:[3,4] linear mixing, Kerker mixing,[5] Broyden mixing,[6] and Pulay mixing.[7] Different reciprocal lattice vector components of density may have different mixing weights.[3] The Broyden mixing algorithm used in the electronic structure computation is adapted from Broyden's original iteration method[8] and was first proposed by Vanderbilt and Louie[9] and refined by Johnson,[6] Kresse *et al.*,[3] and Eyert.[10] Among all of electron density mixing algorithms, Broyden electron density mixing (BEDM) has been most often used in the SCF computation in VASP code.[3] This algorithm uses the multiple-step recursive equations to calculate the inverse Jacobian matrix. In contrast, Eyert proposed a very simple BEDM algorithm without recursion.[10]

In this paper we compare two BEDM algorithms through the electronic structure computation of a Si atom under the local density approximation.[3,4] The KS equation is solved by using the conventional outward/inward integration of the differential equation and then connecting two parts of solutions at the classical turning points.[11] It is different from the method of the matrix eigenvalue solution as used in the VASP code. As the BEDM is nonlinear, the resulting electron density needs a rescaling in order to keep the charge conservation. In addition, we find that when the BEDM is used, we must use the shooting method to search the accurate energy eigenvalues for shallow bound states instead of using the energy eigenvalue correction formula, which is expressed through the first-derivative discontinuity of radial wave functions at the connection point and can be found in the textbook.[11] In the shooting method, a lot of guessed energy eigenvalues are chosen and then the outward/inward integrations of the differential equation are performed to see whether two parts of solutions can be connected with the smooth first derivative at the classical turning points. It is a very time-consuming process. In contrast, it is found that in the linear electron density mixing (LEDM) we can use this perturbation correction to calculate the estimation of the energy eigenvalues in the next cycle and obtain the converged results. Our results show that the BEDMs need less iteration

*Project supported by the National Natural Science Foundation of China (Grant No. 61176080).

†Corresponding author. E-mail: zhangmanhong@ncepu.edu.cn

© 2016 Chinese Physical Society and IOP Publishing Ltd

numbers but longer computation time compared to the LDM. Of the two BEDM algorithms, to reach the same converged results, the one proposed by Eyert takes less total iteration numbers.

2. Theoretical basis

2.1. Relation among the Newton–Raphson method and several Broyden methods

The BEDMs used in this paper are different from Broyden's original methods and belong to the so-called modified Broyden methods.[3,6,10] Eyert has given a very good account of relations among the Newton–Raphson method, Broyden iteration methods, and the modified Broyden methods.[10] For readers to better understand the Broyden methods we give a simple explanation here.

A multi-dimensional iteration may be defined as a nonlinear operator which when applied to an input vector $|x^{(m)}\rangle$ of length N produces an output vector $|y^{(m)}\rangle$ of the same length. Here we have used the Dirac state symbol. The superscript m ($m \geq 1$) denotes the iteration number. Although the input and output vectors often take the electron density in the computation of electronic structures, they can be any variables. We define a residual vector as follows:

$$\begin{aligned} |R^{(m)}\rangle &= |y^{(m)}\rangle - |x^{(m)}\rangle \\ &= |R^{(m)}[|x^{(m)}\rangle]\rangle. \end{aligned} \tag{1}$$

In the second step we have written $|R^{(m)}\rangle$ as multi-dimensional functions of the input vector $|x^{(m)}\rangle$. Self-consistency is achieved when the residual vector vanishes, so $|y^{(m)}\rangle = |x^{(m)}\rangle$. In actual iterations, it is the length of the residual vector that is used to measure the convergence. Hence, an inner product is usually defined in the space spanned by input and output vectors. The inner product also induces a norm in the same space. The norm of the residual vector is formally written as $\langle R^{(m)}|R^{(m)}\rangle$.

In the $(m+1)$-th iteration, the residual vector can be linearly approximated as

$$|R^{(m+1)}\rangle = |R^{(m)}\rangle - J^{(m)}(|x^{(m+1)}\rangle - |x^{(m)}\rangle). \tag{2}$$

Here $J^{(m)}$ is the Jacobian matrix, defined by

$$J_{ij}^{(m)} = -\frac{\partial R_i^{(m)}}{\partial x_j^{(m)}}. \tag{3}$$

If the residual vector on the left side of Eq. (2) is required to vanish, we obtain the following proposal for the new input vector:

$$|x^{(m+1)}\rangle = |x^{(m)}\rangle + (J^{(m)})^{-1}|R^{(m)}\rangle. \tag{4}$$

If we know the analytical dependence of $|y^{(m)}\rangle$ on $|x^{(m)}\rangle$ and the dimension N is small, we can use Eqs. (1) and (4) to obtain a new input vector $|x^{(m+1)}\rangle$. At each iteration step, we calculate the exact Jacobian matrix $J^{(m)}$ at $|x^{(m)}\rangle$ from Eq. (1), invert it, and then calculate $|x^{(m+1)}\rangle$ by using Eq. (4); it is the so-called Newton–Raphson method.

In most cases, the exact Jacobian matrix is not available. In the m-th iteration we use an approximated one in Eq. (4) to calculate a new input vector $|x^{(m+1)}\rangle$ and the corresponding residual vector $|R^{(m+1)}\rangle$. To continue the iteration, the following difference vectors are defined:

$$|\Delta x^{(m)}\rangle = |x^{(m+1)}\rangle - |x^{(m)}\rangle, \tag{5a}$$

$$|\Delta R^{(m)}\rangle = |R^{(m+1)}\rangle - |R^{(m)}\rangle. \tag{5b}$$

To get an estimate of $J^{(m+1)}$, we require it to be the following equation:

$$|\Delta R^{(m)}\rangle = -J^{(m+1)}|\Delta x^{(m)}\rangle. \tag{6}$$

Far away from the final solution, the residual vector $|R^{(m+1)}\rangle$ does not vanish. Equation (14) fixes only the projection of the new Jacobian on the vector $|\Delta x^{(m)}\rangle$. The update $J^{(m+1)} - J^{(m)}$ fulfills a similar equation,

$$(J^{(m+1)} - J^{(m)})|\Delta x^{(m)}\rangle = -|\Delta R^{(m)}\rangle - J^{(m)})|\Delta x^{(m)}\rangle. \tag{7}$$

To determine $J^{(m+1)}$ completely, $J^{(m+1)} - J^{(m)}$ can be formally written as

$$\begin{aligned} & J^{(m+1)} - J^{(m)} \\ = & (J^{(m+1)} - J^{(m)}) \frac{|\Delta x^{(m)}\rangle\langle\Delta x^{(m)}|}{\langle\Delta x^{(m)}|\Delta x^{(m)}\rangle} \\ & + (J^{(m+1)} - J^{(m)}) \left(I - \frac{|\Delta x^{(m)}\rangle\langle\Delta x^{(m)}|}{\langle\Delta x^{(m)}|\Delta x^{(m)}\rangle} \right), \end{aligned} \tag{8}$$

where I is the unit operator and the second term presents the projection of the update onto the space orthogonal to $|\Delta x^{(m)}\rangle$. Broyden argued to neglect the second term and obtained the following rank-1 update formula:[8]

$$\begin{aligned} & \boldsymbol{J}^{(m+1)} - \boldsymbol{J}^{(m)} \\ = & \frac{1}{\langle\Delta x^{(m)}|\Delta x^{(m)}\rangle} \\ & \times (|\Delta R^{(m)}\rangle\langle\Delta x^{(m)}| + \boldsymbol{J}^{(m)}|\Delta x^{(m)}\rangle\langle\Delta x^{(m)}|). \end{aligned} \tag{9}$$

The initial Jacobian matrix $J^{(1)}$ can be taken as a constant diagonal matrix,

$$\boldsymbol{J}^{(1)} = \frac{1}{\beta}\boldsymbol{I}_{N\times N}. \tag{10}$$

Here $\boldsymbol{I}_{N\times N}$ is the $N \times N$ unit matrix and the parameter β is in the range of $0 < \beta < 1$. Equations (1), (4), (9), and (10) form an iteration process. In the above method we update Jacobian $\boldsymbol{J}^{(m)}$ directly. It is called Broyden's first method. The drawback of this method is that when N is very large, we have to store very large $N \times N$ matrices and calculate the matrix inverse used in Eq. (4). Broyden's second method uses the inverse Jacobian matrix $\boldsymbol{G}^{(m)} = (\boldsymbol{J}^{(m)})^{-1}$ directly and

thus avoids calculating the matrix inverse. Corresponding to Eqs. (4), (5), (9), and (10), we have

$$|x^{(m+1)}\rangle = |x^{(m)}\rangle + G^{(m)}|R^{(m)}\rangle, \tag{11}$$

$$|\Delta x^{(m)}\rangle = -G^{(m+1)}|\Delta R^{(m)}\rangle, \tag{12}$$

$$G^{(m+1)} - G^{(m)} = -\frac{1}{\langle \Delta R^{(m)}|\Delta R^{(m)}\rangle}(|\Delta x^{(m)}\rangle\langle \Delta R^{(m)}| + G^{(m)}|\Delta R^{(m)}\rangle\langle \Delta R^{(m)}|); \tag{13}$$

$$G^{(1)} = \beta I_{N\times N}. \tag{14}$$

Both Broyden's first and second methods need to store large $N \times N$ matrices and only use the iteration information in the last step. Vanderbilt and Louie proposed to use iteration information in the previous several steps of iteration.[9] For example, if the inverse Jacobian matrix is used to perform iteration, instead of Eq. (12), we require $G^{(m+1)}$ to fulfill the following M equations:

$$|\Delta x^{(l)}\rangle = -G^{(m+1)}|\Delta R^{(l)}\rangle, \ \forall\, l = m+1-M, \ldots, m. \tag{15}$$

In this way, we can use all the iteration information in the previous M steps of iteration. Eyert has shown how to define a projection operator and then followed Broyden's argument to derive an equation of $G^{(m+1)}$ based on $G^{(m+1-M)}$, $|\Delta x^{(l)}\rangle$, and $|\Delta R^{(l)}\rangle$ for $l = m+1-M, \ldots, m$. For the initial guess for inverse Jacobian matrix, we use

$$G^{(m+1-M)} = \beta I_{N\times N}. \tag{16}$$

Such iteration methods are called modified Broyden methods. Johnson and Eyert applied the method to the self-consistent electronic structure computation using the electron density as iteration vectors.[6,10] They reformulated it by using the variational schemes to determine $G^{(m+1)}$ through the definition of cost functions. In the following we give some details.

2.2. Implementation of BEDMs during SCF computation

During the SCF computation process, given an input electron density $\rho_{\rm in}$, we generate an output electron density $\rho_{\rm out}$ through solving the KS equation. The self-consistency respect to $\rho_{\rm in}$ requires that the charge density residual vector $R[\rho_{\rm in}]$[3]

$$R[\rho_{\rm in}] = \rho_{\rm out} - \rho_{\rm in}, \tag{17}$$

is zero. $\rho_{\rm out}$ and $R[\rho_{\rm in}]$ are both complicated functional of $\rho_{\rm in}$. Let us denote $\rho_{\rm in}^{(m)}$ as the input electron density in the m-th step of iteration. In the BEDMs, it is assumed to be near the zero value of $R[\rho]$,

$$R[\rho] = R[\rho_{\rm in}] - J^{(m)}(\rho - \rho_{\rm in}). \tag{18}$$

As no explicit analytical dependence on $\rho_{\rm in}$ is available for $\rho_{\rm out}$, it is impossible to compute the exact Jacobian matrix, so we have to take an approximation for $J^{(m)}$. To require $R[\rho] = 0$, we gain an optimal charge density ρ^* in the form of

$$\rho^* = \rho_{\rm in}^{(m)} + (J^{(m)})^{-1}R[\rho_{\rm in}^{(m)}]. \tag{19}$$

In successive steps, an improved approximation of inverse Jacobian matrix $G^{(m)} \equiv (J^{(m)})^{-1}$ is built up based on Broyden's second method and a new charge density is obtained from the current approximation of inverse Jacobian matrix $G^{(m)}$, the current charge density $\rho_{\rm in}^{(m)}$, and the current residual vector $R[\rho_{\rm in}^{(m)}]$ through the following equation:

$$\begin{aligned}\rho_{\rm in}^{(m+1)} &= \rho_{\rm in}^{(m)} + G^{(m)}R[\rho_{\rm in}^{(m)}] \\ &= (I - G^{(m)})\rho_{\rm in}^{(m)} + G^{(m)}\rho_{\rm out}^{(m)}.\end{aligned} \tag{20}$$

Here equation (20) clearly shows the forming of $\rho_{\rm in}^{(m+1)}$ through the density mixing.

In Johnson's BEDM algorithm, information of all previous steps of iterations is used to calculate $G^{(m)}$ for the current iteration. For the m-th iteration, this is done via a least square minimization of the following error function:

$$\begin{aligned}E_1 &= w_0||G^{(m+1)} - G^{(m)}||_{\rm F}^2 \\ &\quad + \sum_{i=1}^{m} w_i||(|\Delta\rho^{(i)}\rangle + G^{(m+1)}|\Delta R^{(i)}\rangle)||^2.\end{aligned} \tag{21}$$

Here we have defined

$$|\Delta\rho^{(m)}\rangle \equiv |\rho_{\rm in}^{(m+1)}\rangle - |\rho_{\rm in}^{(m)}\rangle, \tag{22a}$$

$$|\Delta R^{(m)}\rangle \equiv |R[\rho_{\rm in}^{(m+1)}]\rangle - |R[\rho_{\rm in}^{(m)}]\rangle. \tag{22b}$$

In the first and second terms of Eq. (21), the Frobenius matrix norm $||A||_{\rm F} = \sqrt{\sum_{i=1}^{n}\sum_{i=1}^{n}(A_{ij})^2}$ and the vector norm $||A||^2 = \langle A|A\rangle = \sum_{i=1}^{n}(A_i)^2$are used, respectively.[6] For weight factor w_0 and w_i, the relation $w_0 << w_i$ fulfills.[3,6] Minimizing the error function in Eq. (21) with respect to $G^{(m+1)}$, we can derive a set of recursive equations for $G^{(m+1)}$ in terms of $G^{(1)}$, $|\Delta\rho^i\rangle$and $|\Delta R^i\rangle$ with $i = 1, 2, \ldots, M, M+1$. After completing $M+1$ steps of recursive iterations, we reset $G = G^{(1)} = \beta I_{N\times N}$ and restart the next cycle of recursive iterations. Our implementation of this BEDM uses the corrected equations given by Kresse *et al.*[3] The original equations given by Johnson have several mistakes.[3,6] Eyert proposed to minimize the following error function to get $G^{(m+1)}$:[10]

$$\begin{aligned}E_2 &= (w_0)^2||G^{(m+1)} - G^{(m+1-M)}||_{\rm F}^2 \\ &\quad + \sum_{i=m+1-M}^{m}\frac{(w_i)^2}{a_{ii}}||\Delta\rho^{(i)} + G^{(m+1)}\Delta R^{(i)}||^2,\end{aligned} \tag{23}$$

where $a_{ij} = \langle \Delta R^i|\Delta R^j\rangle$ and $G^{(m+1-M)} = \beta I_{N\times N}$. M is the iteration depth and is the same as in Johnson's algorithm. Weight factors in Eqs. (21) and (23) may be different, but the relation $w_0 << w_i$ fulfills in both BEDM algorithms.[3,6] Minimizing the error function in Eq. (23) with respect to $G^{(m)}$, we can derive a very easy equation for $G^{(m)}$ without using recursion. Further we can derive an iteration equation for $\rho_{\rm in}^{(m)}$

$$\rho_{\rm in}^{(m+1)} = \rho_{\rm in}^{(m)} + G^{(m+1-M)}R^{(m)}\rangle$$

$$- \sum_{i=m-M}^{m-1} \gamma_i^{(m)} [\Delta\rho^{(i)} + \boldsymbol{G}^{(m+1-M)} \Delta R^{(i)}]. \quad (24)$$

The quantities of $\gamma_i^{(m)}$ are defined in Eyert's paper and can be easily calculated through solving a small set of linear algebra equations.[10] However, the definition of the inverse Jacobian matrix here has one minus symbol difference and is the same as that used by Kresse *et al.*[3] Compared to those complex equations for $\boldsymbol{G}^{(m)}$ in Johnson's algorithm,[3,6] Eyert's algorithm does not even need to calculate and store $\boldsymbol{G}^{(m)}$.[10]

To better distinguish a BEDM from the other, we further denote Eyert's BEDM as BEDM1 and Johnson's BEDM as BEDM2. In the following we explain how to use these BEDMs to compute the atomic electronic structure.

2.3. Solving KS equation

As usual, we use the following transformation to discrete the radial coordinate:[10]

$$r = r_0 \exp(x). \quad (25)$$

Here, r_0 is a constant. The uniform discretion in x with a step of h gives a discretion which is dense in small r and dilute in large r. With this variable transformation, differentiation and integration with respect to r can be changed into those with respect to the variable x: $\mathrm{d}r = r\mathrm{d}x$. We denote the uniform x-discretion points as $x_1, x_2, \ldots, x_N$ with an increasing order. Correspondingly we have a set of non-uniform r-points: $r_1, r_2, \ldots, r_N$. Here, N is the number of total discretion points.

The spherical averaged radial electron density $\rho(r)$ and the bulk electron density $n(r)$ fulfill the following equations:

$$\rho(r) = \frac{n(r)}{4\pi r^2}, \quad (26a)$$

$$\int_0^\infty \mathrm{d}r \rho(r) = Z. \quad (26b)$$

Here, Z is the atomic number of an atom and for Si, $Z = 14$.

The KS equation for the ground state in a spherically symmetrical potential is

$$-\frac{1}{2}\frac{\mathrm{d}^2\chi_{nl}(r)}{\mathrm{d}r^2} + \left[(V_{\mathrm{HXC}}(r) - E_{nl}) + \frac{l(l+1)}{2r^2}\right]\chi_{nl}(r) = 0, \quad (27)$$

where $V_{\mathrm{HXC}}(r)$ is the Hartree potential plus the exchange–correlation potential. The Hartree atomic unit is adopted: $m_{\mathrm{e}} = 1$, $\hbar = 1$, and $e^2 = 1$. The wave function can be written as

$$\begin{aligned}\psi(r,\theta,\varphi) &= R_{nl}(r)\mathrm{Y}_{lm}(\theta,\varphi) \\ &= \frac{\chi_{nl}(r)}{r}\mathrm{Y}_{lm}(\theta,\varphi) \\ &= \frac{\phi_{nl}(r(x))}{\sqrt{r}}\mathrm{Y}_{lm}(\theta,\varphi). \end{aligned} \quad (28)$$

The normalization is implemented through $\int_0^\infty \mathrm{d}r|\chi_{nl}(r)|^2 = 1$. With the introduction of $\phi_{nl}(r(x))$, the following form is obtained:

$$-\frac{\mathrm{d}^2\phi_{nl}(x)}{\mathrm{d}x^2} + [2r^2(V_{\mathrm{HXC}}(r) - E_{nl}) + (l+1/2)^2]\phi_{nl}(x) = 0. \quad (29)$$

Given a guess of the eigenvalue E_{nl}, this equation is integrated both outward and inward using Numerov's method.[11] At the exact eigenvalue E_{nl}, two pieces of wave function are connected with smooth 0-th and 1-st derivative of $\chi_{nl}(r_{\mathrm{ctp}})\chi'_{nl}(r_{\mathrm{ctp}})$ at the classical turning point r_{ctp}. If $\chi'_{nl}(r_{\mathrm{ctp}})$ is not continuous in two sides of point r_{ctp}, the following improvement to the eigenvalue E_{nl} in the $(m+1)$-th iteration can be estimated from the m-th iteration:[11]

$$\begin{aligned} & E_{nl}^{(m+1)} \\ &= E_{nl}^{(m)} + \frac{(\chi'_{nl}(r_{\mathrm{ctp}}-\delta) - \chi'_{nl}(r_{\mathrm{ctp}}+\delta))\chi_{nl}(r_{\mathrm{ctp}})}{2\int_0^\infty \mathrm{d}r|\chi_{nl}(r)|^2}, \end{aligned} \quad (30)$$

Here, δ is a very small positive number used to define the left and right derivative of χ_{nl} with respect to r. In addition to using this correction formula (denoted as E-pert), we also try the shooting method (denoted as shooting) to search for the exact energy eigenvalue based on the discontinuity of $\chi'_{nl}(r_{\mathrm{ctp}})$ in a certain energy window. Compared to the E-pert method, the shooting method is slower, but has a higher accuracy.

In the LEDM, the $(m+1)$-th input density $|\rho_{\mathrm{in}}^{(m+1)}]\rangle$ is calculated by

$$|\rho_{\mathrm{in}}^{(m+1)}\rangle = \beta|\rho_{\mathrm{out}}^{(m)}\rangle + (1-\beta)|\rho_{\mathrm{in}}^{(m)}\rangle. \quad (31)$$

In this scheme, if both $|\rho_{\mathrm{in}}^{(m)}\rangle$ and $|\rho_{\mathrm{out}}^{(m)}\rangle$ are normalized according to Eq. (26), $|\rho_{\mathrm{in}}^{(m+1)}\rangle$ is also normalized.

2.4. Defining the discrete radial electron density

We now define $|\rho_{\mathrm{in}}^{(m)}\rangle$ vector in a BEDM as

$$|\rho_{\mathrm{in}}^{(m)}\rangle = S_{\mathrm{wr}}[\rho_{\mathrm{in}}^{(m)}{}_{,1}, \rho_{\mathrm{in}}^{(m)}{}_{,2}, \ldots, \rho_{\mathrm{in}}^{(m)}{}_{,N}]'. \quad (32)$$

Here, $\rho_{\mathrm{in}}^{(m)}{}_{,i} = \rho_{\mathrm{in}}^{(m)}(r_i)$ and the transpose operation '$\prime$' has been taken to define $|\rho_{\mathrm{in}}^{(m)}\rangle$ as a column vector. $\boldsymbol{S}_{\mathrm{wr}} = \mathrm{diag}(\sqrt{wr_1}, \sqrt{wr_2}, \sqrt{wr_3}, \ldots, \sqrt{wr_N})$ is a weight matrix. A similar definition applies to $|\rho_{\mathrm{out}}^{(m)}\rangle$ with the same weight vector. One choice is to choose $\boldsymbol{S}_{\mathrm{wr}}$ as an $N \times N$ matrix identity matrix,

$$\boldsymbol{S}_{\mathrm{wr}} = \boldsymbol{I}_{N\times N}. \quad (33)$$

The other choice of the weight matrix is

$$\begin{aligned} \boldsymbol{S}_{\mathrm{wr}} = \mathrm{diag}\Bigg(&\sqrt{\frac{h}{3}r_1}, \sqrt{\frac{4h}{3}r_2}, \sqrt{\frac{4h}{3}r_3}, \ldots, \sqrt{\frac{2h}{3}r_{N-2}}, \\ &\sqrt{\frac{4h}{3}r_{N-1}}, \sqrt{\frac{h}{3}r_N}\Bigg). \end{aligned} \quad (34)$$

Here, N is further assumed to be an odd integer. Equation (34) originates from making the variable transformation from r to x

and then performing a numerical integration with the Simpson rule in the left of Eq. (26b). Other choices of weight matrix are also possible. With such definitions of column vector $|\rho_{\rm in}^{(m)}\rangle$ and $|\rho_{\rm out}^{(m)}\rangle$, we can compute $|R[\rho_{\rm in}^{(m)}]\rangle$, $|\Delta\rho^{(m)}\rangle$, $|\Delta R^{(m)}\rangle$ and the corresponding inner products very easily. For example, if equation (34) is chosen for the weight vector, the following inner product is calculated:

$$\begin{aligned} a_{ii} &= \langle \Delta R^i | \Delta R^i \rangle \\ &= \sum_{k=1}^{N} |\Delta R^i(r_k)|^2 \times ((\boldsymbol{S}_{\rm wr})_{k,k})^2 \\ &= \int_0^{\infty} {\rm d}r |\Delta R^i(r)|^2. \end{aligned} \tag{35}$$

To summarize, in the LEDM we use the density column vector defined by Eq. (31). In BEDMs we compare two definitions (Eqs. (33) and (34)) of the density column vector; the reason to do this is that in solids, especially in metals, proper weight factors are needed to avoid charge sloshing. To analyze convergent properties in different schemes (different density mixing and different weight vector to define the different column vector $|\rho_{\rm in}^{(m)}\rangle$ for iteratively computing the Jacobian matrix $\boldsymbol{G}^{(m)}$), the following quantity is used:

$$\begin{aligned} & ||(|\rho_{\rm out}^{(m)}\rangle - |\rho_{\rm in}^{(m)}\rangle)||^2 \\ &= \langle R^{(m)} | R^{(m)} \rangle \\ &= \sum_{k=1}^{N} |R^{(m)}(r_k)|^2 \times ((\boldsymbol{S}_{\rm wr})_{k,k})^2 \\ &= \int_0^{\infty} {\rm d}r |R^{(m)}(r)|^2. \end{aligned} \tag{36}$$

Here the same set of weight factors defined in Eq. (34) are used in all schemes to fairly compare the convergent error defined by $\lg ||(|\rho_{\rm out}^{(m)}\rangle - |\rho_{\rm in}^{(m)}\rangle)||^2$.

3. Results and discussion

We take $\beta = 0.3$ in the initial inverse Jacobian matrix and in Eq. (31). We set $w_0 = 0.01$ and $w_i = 1$ in Eqs. (21) and (23). The iteration depth in two BEDM algorithms is taken as $M = 5$. The hydrogen wave functions are used as the input. For the exchange–correlation potential, we have tested both the local density approximation (LDA)[12] and the generalized gradient approximation using the PBE functional.[13] It is found that the convergence behaviors in both cases are very similar, so we only present the results based on LDA. As in the initial cycles, there appear the drastic variations of $V_{\rm HXC}(r)$, wave functions and the corresponding E_{nl}, the stable algorithms of the shooting method plus the LEDM are used in the first 5 times of iterations. With $M = 5$, each cycle of BEDM corresponds to 6 times of iterations in the LEDM. So we compare two BEDMs in 100 cycles and the LEDM in 600 times of iterations. It should be emphasized that two kinds of weight matrices (Eqs. (33) and (34)) are used to define the different column vector $|\rho_{\rm in}^{(m)}\rangle$ for iteratively computing the inverse Jacobian matrix $\boldsymbol{G}^{(m)}$ to see the effect of different choices of weight matrices. In the convergent error $\log_{10} ||(|\rho_{\rm out}^{(m)}\rangle - |\rho_{\rm in}^{(m)}\rangle)||^2$, $|\rho_{\rm out}^{(m)}\rangle$ and $|\rho_{\rm in}^{(m)}\rangle$ are converted into forms defined using the same weight vector in Eq. (34).

Figure 1 shows the calculated radial electron density $\rho(r)$ after 600 times of iterations under three conditions: (i) LEDM, E-pert; (ii) BEDM1, E-pert; and (iii) BEDM1, shooting. It can be seen that $\rho(r)$ in cases (i) and (iii) are the same, while in case (ii) the highest peak of $\rho(r)$ is missing. This peak corresponds to 3s and 3p states with small absolute value of E_{nl}. The wave functions of these states are very sensitive to the error of E_{nl}. Figure 2 shows the corresponding convergent errors in three cases. In cases of (i) and (iii), it decreases rapidly with the iteration number, while in case (ii), it retains very large values even after 200 times of iterations. Its final decrease after 250 times of iterations gives completely wrong wave functions for 3s and 3p states in a large r region. The shooting method is more accurate, but it is too slow. To accelerate the iteration speed, we have adopted a compound method: for core states the E-pert method is used to get new values of E_{nl}, for valence states such as 3s and 3p states, the shooting method is used with a much smaller energy search window. It greatly enhances the computation speed. The results are calculated using this method. The trends in Figs. 1 and 2 do not depend on the BEDM and weight matrix.

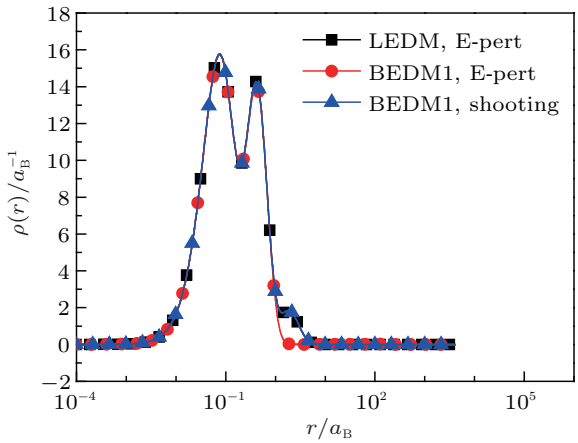


Fig. 1. (color online) Calculated radial electron density. $\rho(r)$ after 600 times of iterations in conditions of LEDM, E-pert, BEDM1, E-pert, and BEDM1, shooting.

Figure 3 shows the convergent error in LEDM, BEDM1, and BEDM2. It is found that two kinds of weight matrices used to calculate $|\rho_{\rm in}^{(m)}\rangle$ and $G^{(m)}$ hardly affect the convergent error. With respect to the iteration time, BEDM1 is the fastest, LEDM is the slowest and BEDM2 is in between. The conservation of charges in the BEDM iteration process is also examined. With the atomic number $Z = 14$ and the input total electron number of $N_{\rm e} = \int_0^{\infty} {\rm d}r \rho_{\rm in}(r)$ before renormalizing $\rho_{\rm in}(r)$ to Z, in all four iteration processes with the BEDM shown in

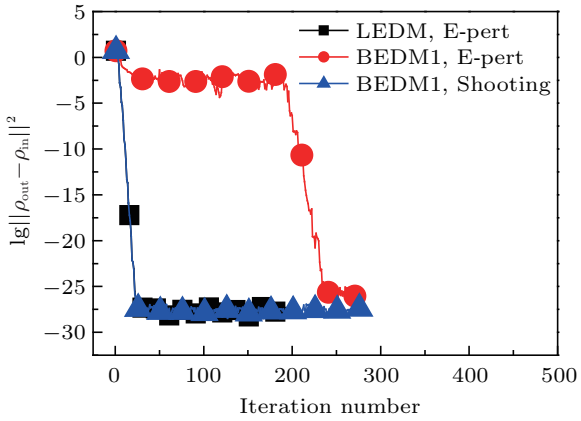


Fig. 2. (color online) Convergent errors versus the iteration number in three cases shown in Fig. 1.

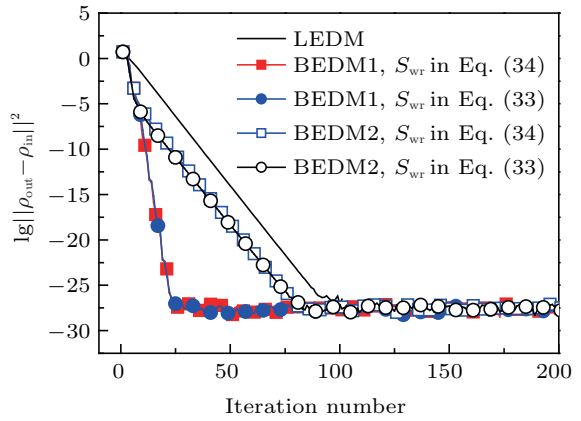


Fig. 3. (color online) Convergent errors versus the iteration number for different electron density mixing schemes and weight matrices.

Fig. 3, the quantity $(N_e - Z)/Z$ decreases from -1.9×10^{-11} to -2.3×10^{-11} after 1 cycle of the BEDM iteration and stays almost constant during later iterations. This means although the BEDMs are nonlinear, the charge conservation almost keeps with present choices of weight factors.

Figure 4 compares the converged radial electron density $\rho(r)$ and those after 1 cycle (6 times of iterations) of BEDM1 and BEDM2 iteration process. The profile from BEDM1 is in much better agreement with the converged result. As both start from the same initial electron density and the same initial Jacobian matrix, this result indicates that BEDM1 is more accurate than BEDM2.

Finally using Eq. (24) we easily show how the weight matrix affects the iteration vectors. We can easily prove the following equation:

$$|\rho^{(m)}_{\text{in,wr2}}\rangle = \boldsymbol{S}_{\text{wr2}}\boldsymbol{S}^{-1}_{\text{wr1}}|\rho^{(m)}_{\text{in,wr1}}\rangle. \tag{37}$$

The same transformation matrix $\boldsymbol{S}_{\text{wr2}}\boldsymbol{S}^{-1}_{\text{wr1}}$ applies to the transformation of $|\Delta\rho^{(m)}\rangle$, $|R^{(m)}\rangle$, and $|\Delta R^{(m)}\rangle$. In this way we can transform the iteration vectors from weight matrix $\boldsymbol{S}_{\text{wr1}}$ to $\boldsymbol{S}_{\text{wr2}}$ very easily. Meanwhile neither the initial inverse Jacobian matrix nor the form of Eq. (24) will change.

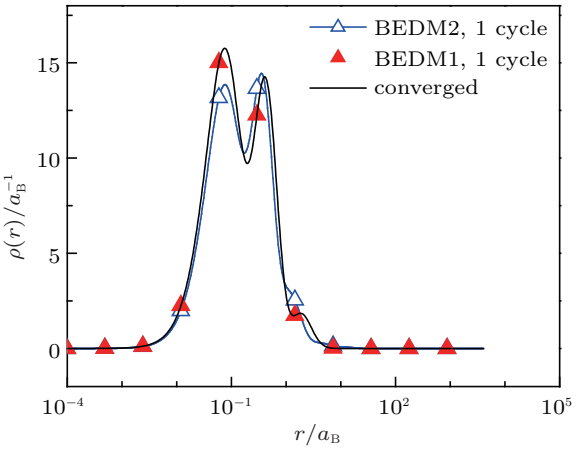


Fig. 4. (color online) Comparison of the radial electron density $\rho(r)$: the converged one and those after 1 cycle (6 times of iterations) of BEDM1 and BEDM2 iteration processes.

4. Conclusions

In summary, we have compared two BEDM algorithms proposed by Johnson and Eyert during SCF atomic electronic structure computation. We found when integrating the differential equation to solve the KS equation, both algorithms need to use the slower but more accurate shooting method to search the energy eigenvalues. The usual perturbation estimation of the correction to the energy eigenvalues is stable with LEDM, but unstable with two BEDMs. A new compound method is proposed to accelerate the search of the energy eigenvalues: using the perturbation estimation for core states and the shooting method for the shallow valence states. Of two BEDMs, the one proposed by Eyert is simpler, more accurate and converges in a much smaller number of iterations in the computation of atomic electronic structures.

References

[1] Kohn W and Sham L 1965 *Phys. Rev. A* **140** 1133
[2] Gonze X 1995 *Phys. Rev. A* **52** 1096
[3] Kresse G and Furthmueler J 1996 *Comput. Mater. Sci.* **6** 15
[4] Kresse G and Joubert D 1999 *Phys. Rev. B* **59** 1758
[5] Kerker G P 1981 *Phys. Rev. B* **23** 3082
[6] Johnson D D 1988 *Phys. Rev. B* **38** 12087
[7] Pulay P 1980 *Chem. Phys. Lett.* **73** 393
[8] Broyden C G 1965 *Math. Comput.* **19** 577
[9] Vanderbilt D and Louie S G 1984 *Phys. Rev. B* **30** 6118
[10] Eyert V 1996 *J. Comput. Phys.* **124** 271
[11] Johnson W R 2007 *Atomic Structure Theory* (Berlin: Springer-Verlag) pp. 44–45
[12] Perdew J P and Zunger A 1981 *Phys. Rev. B* **23** 5048
[13] Perdew J P, Burke K and Ernzerhof M 1996 *Phys. Rev. Lett.* **77** 3865