Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 217 (2006) 709–721
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

A constrained optimization algorithm for total energy minimization in electronic structure calculations

Chao Yang *, Juan C. Meza, Lin-Wang Wang

Computational Research Division, Lawrence Berkeley National Laboratory, 1 Cyclotron Road, Berkeley, CA 94720, United States

Received 27 July 2005; received in revised form 13 December 2005; accepted 18 January 2006
Available online 10 March 2006

Abstract

A new direct constrained optimization algorithm for minimizing the Kohn–Sham (KS) total energy functional is presented in this paper. The key ingredients of this algorithm involve projecting the total energy functional into a sequence of subspaces of small dimensions and seeking the minimizer of total energy functional within each subspace. The minimizer of a subspace energy functional not only provides a search direction along which the KS total energy functional decreases but also gives an optimal "step-length" to move along this search direction. Numerical examples are provided to demonstrate that this new direct constrained optimization algorithm can be more efficient than the self-consistent field (SCF) iteration.

Keywords: Electronic structure calculation; Total energy minimization; Nonlinear eigenvalue problems; Constrained optimization

1. Introduction

One of the fundamental problems in electronic structure calculations is to minimize the Kohn–Sham (KS) total energy functional with respect to electron wave functions. Currently, the most widely used approach for solving this minimization problem is to apply the so called Self Consistent Field (SCF) iteration to the nonlinear equation derived from the first order necessary optimality condition. In each SCF iteration, one must compute approximations to a few smallest eigenvalues and their corresponding eigenvectors of a large matrix (Hamiltonian).

Methods for minimizing the total energy directly have been examined in the past [3,9,24,19,2,14,20,25,26]. These methods construct a search direction based on the gradient of the total energy, and perform some type of line search along that direction in order to determine an optimal step length. The difficulty with these approaches is that the line search strategy must take into account the orthonormality constraints imposed on the wave functions. Due to this difficulty, the minimization algorithms developed in the past could be slower than SCF by a factor of 1.5 to 10 [14].

* Corresponding author. Tel.: +1 510 486 6424; fax: +1 510 486 5812.
E-mail address: cyang@lbl.gov (C. Yang).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2006.01.030

In this paper, we show that direct minimization of the KS functional can be made more effective than the SCF algorithm. Instead of using a simple line search strategy, we project the total energy into a subspace from which an optimal search direction and step length are determined simultaneously by solving a smaller nonlinear eigenvalue problem. In our approach, the optimal wave functions are determined simultaneously rather than "band-by-band". The orthonormality constraint of these wave functions is automatically satisfied by the solution to the small nonlinear eigenvalue problem.

2. Mathematical background

In this section, we establish the mathematical notation required to describe a new constrained optimization algorithm. We begin with the continuous formulation of the optimization problem, and establish the finite-dimensional analog using linear algebra notation.

We denote the ith electron wave function by $\psi_i(r)$, where r represents the (three-dimensional) spatial coordinates. We denote the electron charge density by $\rho(r)$. It is defined by

$$\rho(r) = \sum_{i=1}^{k} |\psi_i(r)|^2,$$

where k is the number of occupied states.

The KS total energy functional consists of several components, i.e.,

$$E_{\text{total}}[\{\psi_i\}] = E_{\text{kinetic}} + E_{\text{ion}} + E_{\text{H}} + E_{\text{XC}}, \tag{1}$$

where E_{kinetic} is the kinetic energy, $E_{\text{ion}}, E_{\text{H}}$ and E_{XC} are potential energies induced by the electron–ion interaction (ionic potential), the electron–electron interaction (Hartree potential) and the exchange correlation potential respectively.

The kinetic energy of the atomistic system is defined by

$$E_{\text{kinetic}} = -\frac{1}{2}\sum_{i=1}^{k} \int \bar{\psi}_i(r)\nabla^2\psi_i(r)\mathrm{d}r,$$

where ∇^2 is the Laplacian operator, and $\bar{\psi}$ is the complex conjugate of ψ.

The ionic potential energy consists of a local and a non-local term. The local term can be expressed by

$$E_{\text{ion(local)}} = \int V_{\text{ion}}(r)\rho(r)\mathrm{d}r,$$

where $V_{\text{ion}}(r)$ represents some local ionic potential function. The contribution from the non-local term is defined by

$$E_{\text{ion(nonlocal)}} = \sum_{i=1}^{k}\sum_{\ell}\left|\int \bar{\psi}_i(r)w_\ell(r)\mathrm{d}r\right|^2,$$

where $w_\ell(r)$ denotes a pseudo-potential reference projection function.

The Hartree potential, defined by

$$V_{\text{H}}(r) = \int \frac{\rho(r')}{|r-r'|}\mathrm{d}r',$$

is used to model the classical electrostatic average interaction between electrons. Its contribution to the total energy is defined by

$$E_{\text{H}} = \frac{1}{2}\int V_{\text{H}}(r)\rho(r)\mathrm{d}r.$$

The exchange correlation function ϵ_{xc} is used to model the non-classical and quantum interaction between electrons. The potential energy induced by this function is defined by

$$E_{\text{XC}} = \int \rho(r)\epsilon_{\text{xc}}(\rho(r))\mathrm{d}r.$$

It is important to note that the minimization of the total energy (1) must be carried out under the orthonormality constraint

$$\int \bar{\psi}_i(r)\psi_j(r)\mathrm{d}r = \delta_{i,j}.$$

In the following discussion, we will use A^{T} to denote the transpose of a matrix A, and A^* to denote the complex conjugate of A. A submatrix of A consisting of rows i through j and columns p through q will be denoted by the notation $A(i{:}j,p{:}q)$. If the submatrix contains all rows (columns) of A, it will be denoted by $A(:,p{:}q)$ ($A(i{:}j,:)$).

With an appropriate discretization scheme, the electron wave function $\psi_i(r)$ can be approximated by a vector $x_i \in \mathbb{C}^n$, where n is the spatial degree of freedom, i.e., the number of real space grid points. If we let $X = (x_1, x_2, \ldots, x_k)$, then the charge density associated with the k occupied states can be expressed by

$$\rho(X) = \mathrm{diag}\ (XX^*),$$

where diag(A) denotes a column vector consisting of diagonal entries of the matrix A.

Under the same discretization, the Laplacian operator ∇^2 can be approximated by a Hermitian matrix $L \in \mathbb{C}^{n\times n}$. The discretized local ionic potential can be represented by a diagonal matrix D_{ion}, and the discrete form of the Hartree potential can be represented by the product of a Hermitian matrix $S \in \mathbb{C}^{n\times n}$ with $\rho(X)$.

Using the notation established above, we can express the discrete form of various components of the total energy (1) by

$$E_{\mathrm{kinetic}} = \frac{1}{2}\mathrm{trace}(X^*LX), \tag{2}$$

$$E_{\mathrm{ion(local)}} = \mathrm{trace}(X^*D_{\mathrm{ion}}X), \tag{3}$$

$$E_{\mathrm{ion(nonlocal)}} = \sum_i \sum_\ell \left|x_i^* w_\ell\right|^2, \tag{4}$$

$$E_{\mathrm{H}} = \frac{1}{2}\rho(X)^{\mathrm{T}} S\rho(X), \tag{5}$$

$$E_{\mathrm{XC}} = \rho(X)^{\mathrm{T}}(\epsilon_{\mathrm{xc}}[\rho(X)]). \tag{6}$$

Once discretized, the minimization problem becomes

$$\begin{aligned} &\min \quad E_{\mathrm{total}}(X) \\ &\text{s.t.} \quad X^*X = I_k, \end{aligned} \tag{7}$$

where I_k denotes a $k\times k$ identity matrix.

The Lagrangian associated with (7) is

$$\mathscr{L}(X) = E_{\mathrm{total}}(X) - \mathrm{trace}[\Lambda^{\mathrm{T}}(X^*X - I_k)], \tag{8}$$

where Λ is a $k\times k$ matrix containing the Lagrange multipliers associated with the constraints specified by $X^*X = I_k$.

The solution to (7) must satisfy the first order necessary condition

$$\begin{aligned} &\nabla_X \mathscr{L}(X) = 0, \\ &X^*X = I_k. \end{aligned} \tag{9}$$

Here, $\nabla_X \mathscr{L}$ represents an $n\times k$ matrix whose (i,j)th entry is the partial derivative of $\mathscr{L}$ with respect to the (i,j)th entry of X.

It is easy to verify that

$$\nabla_X E_{\mathrm{kinetic}} = \frac{1}{2}LX, \tag{10}$$

$$\nabla_X E_{\mathrm{ion(local)}} = D_{\mathrm{ion}}X, \tag{11}$$

$$\nabla_X E_{\mathrm{ion(nonlocal)}} = \sum_\ell \left(w_\ell w_\ell^*\right)X, \tag{12}$$

$$\nabla_X E_H = \mathrm{Diag}(S\rho(X))X, \tag{13}$$

$$\nabla_X E_{\mathrm{XC}} = \mathrm{Diag}(\mu_{\mathrm{xc}}(\rho))X, \tag{14}$$

where

$$\mu_{\mathrm{xc}}(\omega) \equiv \frac{\mathrm{d}[\omega\epsilon_{\mathrm{xc}}(\omega)]}{\mathrm{d}\omega}$$

is the derivative of the exchange-correlation function. Here the notation $\mathrm{Diag}(\rho)$ represents a diagonal matrix whose diagonal is determined by the vector ρ, and we scaled (10)–(14) by 1/2 to be consistent with the convention used in the electronic structure community.

Substituting (10)–(14) into (9), we obtain the Kohn–Sham equation

$$\begin{aligned} &H(X)X = X\Lambda_k, \\ &X^*X = I_k, \end{aligned} \tag{15}$$

where $H(X)$ is the Kohn–Sham Hamiltonian defined by

$$H(X) = \frac{1}{2}L + D_{\mathrm{ion}} + \sum_{\ell} w_\ell w_\ell^* + \mathrm{Diag}(S\rho(X)) + \mathrm{Diag}(\mu_{\mathrm{xc}}(\rho(X))). \tag{16}$$

Because the vector ρ in (16) depends on X, the eigenvalue problem defined by (15) is nonlinear. Note that the solution to (7) is not unique. If X is a solution, then XQ is also a solution for some $Q \in \mathbb{C}^{k\times k}$ such that $Q^*Q = I_k$. That is, the solution to the constrained minimization problem or, equivalently, the nonlinear equations (15) is a k-dimensional invariant subspace in $\mathbb{C}^n$ rather than a specific matrix. In particular, Q can be chosen such that Λ_k is diagonal. In this case, X consists of k Kohn–Sham eigenvectors associated with the k smallest eigenvalues of (15).

3. The self-consistent field iteration

The most widely used method for computing the wave functions associated with the minimum total energy is the so-called *self consistent field* (SCF) iteration, which corresponds to a fixed point iterative scheme applied to the Kohn–Sham equation (15). Given an initial guess of X, say $X^{(0)}$, one forms the discrete Hamiltonian,

$$H^{(1)} = \frac{1}{2}L + D_{\mathrm{ion}} + \sum_{\ell} w_\ell w_\ell^* + \mathrm{Diag}(S\rho(X^{(0)})) + \mathrm{Diag}(\mu_{\mathrm{xc}}(\rho(X^{(0)}))), \tag{17}$$

and computes eigenvectors $X^{(1)}$ associated with the k smallest eigenvalues of $H^{(0)}$. These eigenvectors defines a new Hamiltonian $H^{(2)}$. In the basic version of the SCF iteration, the difference between $\rho(X^{(i-1)})$ and $\rho(X^{(i)})$ is examined at each step to determine whether the iteration should be terminated. If the change in the charge density remains large, the eigenvectors associated with the k smallest eigenvalues of $H^{(i)}$ are computed, and this process continues until the $\|\rho(X^{(i-1)}) - \rho(X^{(i)})\|$ becomes negligibly small. In this case, $X^{(i)}$ consists of a set of wave functions that are self-consistent with respect to the KS equation (15).

For completeness, we outline the major steps of the basic version of a SCF calculation in Fig. 1.

Depending on the discretization scheme used, it may not be necessary or possible to form the Hamiltonian $H^{(i)}$ explicitly in the SCF calculation. This is particularly true when the continuous problem is discretized by a spectral method using a plane wave basis. In that case, $H^{(i)}$ only exists in the form of a matrix vector multiplication procedure.

For large-scale problems or problems in which $H^{(i)}$ cannot be formed explicitly, it is usually not feasible to solve the linear eigenvalue problem $H^{(i)}X^{(i)} = X^{(i)}\Lambda^{(i)}$ by using a QR [7,8] type of eigensolver. Iterative methods such as the Lanczos [16], preconditioned conjugate gradient [10] or a Jacobi–Davidson type of method [23,5,18] are more appropriate in this setting.

Note that computing the k smallest eigenvalues of $H^{(i)}$ is equivalent to solving the following constrained quadratic minimization problem:

$$\begin{aligned} &\min \quad q(X) = \frac{1}{2}\mathrm{trace}[X^*H^{(i)}X] \\ &\text{s.t.} \quad X^*X = I_k. \end{aligned} \tag{18}$$

SCF Iteration

Input: The matrices L, D_{ion}, S, the vectors w_ℓ, $\ell = 1, 2, \ldots$. The derivative of the exchange-correlation function $\mu_{xc}(x)$; an initial guess $X^{(0)}$ for the optimal wave function $X \in \mathbb{C}^{n\times m}$;

Output: $X \in \mathbb{C}^{n\times m}$ such that $X^*X = I_m$ and $E_{tot}(X)$ is minimized, where $E_{tot}(X)$ is defined by (10) - (14).

1. for $i = 1, 2, \ldots$ until convergence
2. Form $H^{(i)} = H(X^{(i-1)})$;
3. Compute $X^{(i)}$ such that $H^{(i)}X^{(i)} = X^{(i)}\Lambda^{(i)}$, and $\Lambda^{(i)}$ contains the k smallest eigenvalues of $H^{(i)}$;
4. end for

Fig. 1. The SCF iteration.

Because the the gradient of the objective in (18) is the same as the gradient of $E_{\text{total}}(X)$ at $X^{(i)}$. The minimization of the quadratic objective in (18) can lead to a reduction of E_{total}. Thus, the SCF iteration can be viewed as a procedure that minimizes the total energy functional by minimizing a sequence of quadratic surrogate functionals.

The major computational cost of the SCF scheme is in solving the linear eigenvalue problem (15) or the quadratic minimization problem (18) at Step 3 in each iteration. Because the condition number of $H^{(i)}$ is typically large ($>10^6$), and the eigenvalues of interest are typically clustered at the low end of the spectrum, solving such a problem can be extremely challenging. In the materials science community, the common practice is to apply a few preconditioned conjugate gradient iterations to solve (18) only approximately. This approach is plausible not only because it is computationally efficient, but also because the gradients of (18) and E_{total} match only at $X^{(i)}$. That is, if one moves too far along a descent direction associated with $q(X)$ by solving (15) or (18) very accurately, then $E_{\text{total}}(X)$ may actually increase.

The SCF iteration can also be viewed as a fixed-point iteration. However, because one cannot write down an explicit mapping from $X^{(i-1)}$ to $X^{(i)}$ even if the linear eigenvalue problem

$$H^{(i)}X^{(i)} = X^{(i)}\Lambda^{(i)}, \tag{19}$$

can be solved exactly, it is somewhat difficult to analyze the convergence of the SCF iteration using the standard theory for a fixed point iteration without making certain assumptions.

Although the basic SCF iteration works on some problems, it tends to fail for many large systems. In practice, a modified scheme in which a mixture of $H^{(i)}$ and $H^{(i-1)}$ is used to replace $H^{(i)}$ in Step 2 of the algorithm is much more effective. This scheme is called *charge* or *potential mixing* [15,14]. Furthermore, charge mixing is often combined with the use of the direct inversion of iterative subspace (DIIS) algorithm proposed in [21,22,11] to accelerate the convergence of SCF. However, there is no theoretical guarantee that these techniques will always work. In fact, there are cases in which these techniques fail also.

4. A constrained optimization algorithm for total energy minimization

The algorithm we present in this section aims at minimizing the KS total energy functional directly. This general approach has been discussed in a number of papers [9,24,2,19,14,26,25]. In [24,19], a conjugate gradient (CG) type of algorithm is used to minimize the total energy. The minimization is carried out "band-by-band", i.e., the total energy is minimized with respect to one wave function at a time. For the jth band (wavefunction), the search direction $p_j^{(i)}$ is generated from a linear combination of the wavefunction $x_j^{(i)} = X^{(i)}e_j$ and the residual

$$r_j = H^{(i)}x_j^{(i)} - x_j^{(i)}\lambda_j,$$

where λ_j is the jth eigenvalue of the projected Hamiltonian $X^{(i)*}H^{(i)}X^{(i)}$. Note that r_j is simply the jth column of the gradient matrix $\nabla_X \mathscr{L}(X^{(i)})$. Similar to a standard CG algorithm, the linear combination of $x_j^{(i)}$ and r_j is chosen so that $p_j^{(i)}$ is $H^{(i)}$-conjugate to the previous search direction $p_j^{(i-1)}$. The new wavefunction $x_j^{(i+1)}$ is then computed by minimizing the KS total energy in the subspace spanned by $x_j^{(i)}$ and $p_j^{(i)}$. To simplify this minimization problem, $p_j^{(i)}$ is first orthogonalized against $x_j^{(i)}$ and normalized so that $\|p_j^{(i)}\| = 1$. The new wavefunction is parameterized by

$$x_j^{(i+1)} = x_j^{(i)}\cos\theta + p_j^{(i)}\sin\theta,$$

where the optimal θ is obtained by a standard line search procedure. Instead of using the KS functional to perform the line search, Teter et al. [24] proposed using a surrogate function that is cheaper to evaluate. However, this approach was shown in [14] to be less efficient than the SCF iteration. We believe this is primarily due to the "band-by-band" nature of the algorithm.

The methods presented in [9,2,26,25] were designed to minimize the total energy with respect to all wave functions (associated with the occupied state) simultaneously. The method developed in [9] modifies the unconstrained conjugate gradient search direction so that the orthonormality constraint $X^*X = I_k$ can be satisfied. The approaches taken in [2,25] reparameterize the search direction so that standard unconstrained minimization can be used directly. The algorithm developed in [26] first computes the search direction via a limited-memory BFGS [17] scheme, the search direction is then modified through a parallel transport technique [6] to ensure that the orthonormality constraint $X^*X = I_k$ is satisfied in the line search procedure. In all of these methods, the search direction is computed first, and an optimal step length is then determined to reduce the total energy along computed search direction.

The direct minimization algorithm we present here also seeks the optimal wave functions associated with all occupied states simultaneously. However, we choose the search direction and the step length simultaneously from a subspace that consists of the existing wave functions $X^{(i)}$, the gradient of the Lagrangian (8) and the search direction produced in the previous iteration. A special strategy is developed to minimize the total energy within the search space while maintaining the orthonormality constrained required for $X^{(i+1)}$. This strategy requires us to solve a projected nonlinear eigenvalue problem as we will show below.

Let $R^{(i)}$ be the preconditioned gradient of the Lagrangian (8) with respect to X evaluated at $X^{(i)}$, and let $P^{(i-1)}$ be the search direction obtained in the $i-1$st iteration. In our algorithm, the wave function update is performed within the $3k$-dimensional subspace spanned by $X^{(i)}, R^{(i)}$ and $P^{(i-1)}$. This is in the same spirit as the locally optimal block preconditioned conjugate gradient (LOBPCG) algorithm proposed in [13] for solving large-scale linear eigenvalue problems. Note that the inclusion of $P^{(i-1)}$ is important. It prevents the search direction constructed at the ith step from being parallel to the steepest descent direction which often results in a tiny step between $X^{(i)}$ and $X^{(i+1)}$ (Hence a small reduction in E_{total} from $X^{(i)}$ to $X^{(i+1)}$).

If we let

$$Y = (X^{(i)}, R^{(i)}, P^{(i-1)}),$$

we can then express the new approximation, $X^{(i+1)}$, by

$$X^{(i+1)} = YG, \tag{20}$$

where $G \in \mathbb{C}^{3k\times k}$ is chosen to minimize $\widehat{E}(G) \equiv E_{\text{total}}(YG)$, i.e. we must solve

$$\begin{aligned} &\min_G E_{\text{total}}(YG) \\ &\text{s.t. } G^*Y^{\mathrm{T}}YG = I_k. \end{aligned} \tag{21}$$

The first order necessary condition of (21) can be derived by examining the gradient of $\widehat{E}(G)$ with respect to G.

It is easy to verify that

$$\nabla_G \widehat{E}_L(G) = \frac{1}{2}\nabla_G[\mathrm{trace}(G^*Y^*LYG)] = (Y^*LY)G, \tag{22}$$

$$\nabla_G \widehat{E}_V(G) = \nabla_G[\mathrm{trace}(G^*Y^*D_{\mathrm{ion}}YG)] = Y^*D_{\mathrm{ion}}YG, \tag{23}$$

$$\nabla_G \widehat{E}_W(G) = \nabla_G\left[\sum_i \sum_\ell \left|w_\ell^* Y g_i\right|^2\right] = \sum_\ell (Y^*w_\ell)(Y^*w_\ell)^* G, \tag{24}$$

$$\nabla_G \widehat{E}_R(G) = \frac{1}{2}\nabla_G\left[\rho(YG)^{\mathrm{T}} S\rho(YG)\right] = Y^*\mathrm{Diag}[S\rho(YG)]YG, \tag{25}$$

$$\nabla_G \widehat{E}_X(G) = \nabla_G[\rho(YG)\epsilon_{\mathrm{xc}}(\rho(YG))] = Y^*\mathrm{Diag}[\mu_{\mathrm{xc}}(\rho(YG))]YG. \tag{26}$$

Again, (22)–(26) have been scaled by 1/2 to be consistent with the convention used in the electronic structure community.

If we define

$$\widehat{H}(G) = Y^*\left[\frac{1}{2}L + D_{\mathrm{ion}} + \sum_\ell w_\ell w_\ell^* + \mathrm{Diag}(S\rho(YG)) + \mathrm{Diag}(\mu_{\mathrm{xc}}(\rho(YG)))\right]Y, \tag{27}$$

then, solving (21) is equivalent to solving

$$\widehat{H}(G)G = BG\Omega_k, \tag{28}$$

$$G^*BG = I_k, \tag{29}$$

where $B = Y^*Y$ and the $k \times k$ diagonal matrix Ω_k contains the k smallest eigenvalues of (28).

Note that the projected nonlinear eigenvalue problem defined by (28) and (29) is much smaller than the nonlinear eigenvalue solved in an SCF iteration. The reduction in size provides us with more flexibility in terms of the algorithms we can choose to solve the nonlinear eigenvalue problem. For example, if we apply an SCF iteration to compute the desired eigenpairs of (28) and (29), the linear eigenvalue problem that emerges from each SCF iteration can be solved by using the LAPACK [1] implementation of the QR algorithm [7,8]. We may even apply a standard nonlinear constrained minimization algorithm such as a sequential quadratic programming technique (SQP) [4] to (21) directly. Furthermore, it should be noted that it is not necessary to solve Eqs. (28) and (29) to full accuracy in the early stage of the direct minimization process because all we need is a G that yields sufficient decrease in the objective function within the subspace spanned by columns of Y.

Once G is computed, we can update the wave function following (20). In addition, we can compute the search direction associated with this update

$$P^{(i)} \equiv X^{(i+1)} - X^{(i)}G(1:k,:) = Y(:,k+1:3k)G(k+1:3k,:).$$

Because the solution to (28) and (29) ensures columns of $X^{(i+1)}$ are orthonormal, there is no need to explicitly orthogonalize $P^{(i)}$ against $X^{(i)}$ in our algorithm.

A complete description of the constrained minimization algorithm is shown in Fig. 2. We should point out that solving the projected optimization problem in Step 6 of the algorithm may require us to evaluate the projected Hamiltonian (27) repeatedly as we search for the best G. However, since the first three terms of $\widehat{H}$ do not depend on G, they can be computed and stored in advance. Only the last two terms of (27) need to be updated. These updates requires the charge density, the Hartree and the exchange-correlation potentials to be recomputed.

5. Computational complexity

In this section, we estimate the computational complexity of the direct constrained minimization (DCM) algorithm outlined in Fig. 2, and compare it with the complexity of the SCF iteration. Our estimation is based on the number of matrix vector multiplications (MATVEC) performed in these algorithms. Each MATVEC is of the form $y \leftarrow Hx$, where H is the KS Hamiltonian defined in (16).

Algorithm: A Constrained Minimization Algorithm for Total Energy Minimization

Input: An initial set of wave functions $X^{(0)} \in \mathbb{C}^{n\times k}$, where k is the number of occupied states; the matrices L, D_{ion}, S; the vectors w_ℓ, $\ell = 1, 2, \ldots$. The derivative of the exchange-correlation function $\mu_{xc}(x)$; a preconditioner K;

Output: $X \in \mathbb{C}^{n\times k}$ such that the KS total energy functional $E_{total}(X)$ is minimized and $X^*X = I_k$.

1. Orthonormalize $X^{(0)}$ such that $X^{(0)^*}X^{(0)} = I_k$;

2. for $i = 0, 1, 2, \ldots$ until convergence

3. Compute $\theta_j = e_j^T X^{(i)^*} H^{(i)} X^{(i)} e_j$, $j = 1, 2, \ldots, k$;

4. Compute $R = K^{-1}\left[H^{(i)}X^{(i)} - X^{(i)}D\right]$,

where $D = \mathrm{Diag}(\theta_1, \theta_2, \ldots, \theta_m)$;

5. if $(i > 1)$ then

$Y \leftarrow (X^{(i)}, R, P^{(i-1)})$

else

$Y \leftarrow (X^{(i)}, R)$;

endif

6. $B \leftarrow Y^*Y$;

7. Find $G \in \mathbb{C}^{2k\times 2k}$ or $\mathbb{C}^{3k\times 3k}$ that minimizes $E_{total}(YG)$

subject to the constraint $G^*BG = I$;

8. Set $X^{(i+1)} = YG$;

9. if $(i > 1)$ then

$P^{(i)} \leftarrow Y(:, k+1:3k)G(k+1:3k, :)$;

else

$P^{(i)} \leftarrow Y(:, k+1:2k)G(k+1:2k, :)$;

endif

10. end for

Fig. 2. A direct constrained minimization algorithm for total energy minimization.

The exact cost of each MATVEC depends on how the wavefunctions are discretized. We focus on a spectral discretization scheme in which plane waves are used as the basis of decomposition. These basis functions are eigenfunctions of the Laplacian operator (L) associated with the kinetic energy of the atomistic system. Thus, to perform each MATVEC, one only needs to store the Fourier coefficients of each wave function $x_j = Xe_j$ instead of x_j itself so that $y \leftarrow Lx_j$ can be carried out in $\mathcal{O}(n)$ floating point operations (flops) in the frequency domain. However, because the potential terms of the Hamiltonian (with the exception of the non-local ionic potential) are diagonal in the spatial domain, one must convert the Fourier space representation of x_j into the real space representation before operations involving these potential terms are performed. The complexity of this conversion is $\mathcal{O}(n \log n)$ when it is carried out by a Fast Fourier Transform (FFT). The result of the real space calculation must be converted back to the frequency domain (by an FFT) for subsequent computations. Thus the cost of each MATVEC for a plane wave discretization is dominated by two FFT calculations.

In the DCM algorithm, k MATVECs are performed in each outer iteration to obtain the gradient. If SCF is used to solve the projected problem (21), then each outer DCM iteration contains a number of inner SCF iterations in which the projected Hamiltonian (27) must be updated repeatedly. The update of the projected Hartree potential requires us to compute $S\rho(YG)$. Because S is the inverse of L, this calculation is typically carried out by a fast Poisson solver. The complexity of this computation is approximately $\mathcal{O}(n \log n)$, which is equivalent to a single MATVEC used in the SCF iteration asymptotically. Thus, if p inner SCF iterations are taken in the DCM algorithm, the total number of MATVECs used per DCM iteration is $k + p$.

When an SCF iteration is used to solve (9) directly, the quadratic minimization problem (19) (or the linear eigenvalue problem (15)) is often solved approximately by applying a number of preconditioned conjugate gradient (PCG) (inner) iterations. Each PCG iteration requires k matrix vector (MATVEC) multiplications followed by k preconditioning operations. The Laplacian operator L is often used as the preconditioner. Because it is diagonal in the frequency space, the cost of preconditioning is relatively small compared to an FFT calculation. If m PCG iterations are taken on average to compute an approximate eigenpair of (19), then the total number of MATVECs used per SCF iteration is $m \times k$.

We should point out that both the SCF and the DCM algorithms perform an additional $\mathcal{O}(n \cdot k^2)$ basic linear algebra (BLAS) operations in each inner iteration. In SCF, these operations are used to maintain the orthogonality of the Fourier representation of x_j. In DCM, these operations are used to update the charge density $\rho(YG)$ and to compute the projected potential term

$$Y^*[\mathrm{Diag}(S\rho(YG) + \mu_{\mathrm{xc}}(\rho(YG)))]Y.$$

The cost of these BLAS operations is relatively small compared to that associated with FFT when $n \gg k$, and these operations tend to scale very well on parallel machines compared to FFT calculations. However, when k becomes a significant fraction of n, the cost of these calculations cannot be ignored.

6. Numerical examples

In this section, we provide two numerical examples that demonstrate the performance the DCM algorithm, and compare it with that of the SCF iteration implemented in the plane wave based electronic structure calculation software package PEtot [27].

We measure the convergence of both algorithms by examining the relative reduction of the total energy computed in each outer iteration. The relative reduction is evaluated by

$$\Delta E_i = E_{\mathrm{total}}(X^{(i)}) - E_{\mathrm{min}},$$

where E_{min} is a lower bound of the total energy.

The computational cost of both the SCF and DCM is estimated by the number of matrix vector multiplications $y \leftarrow H^{(i)}x$ performed in these two algorithms. To illustrate that this is a reliable measure when $k \ll n$, we also provide the timing measurements. Our computation is performed on the IBM SP maintained at the National Energy Research Scientific Computing Center (NERSC). Each IBM SP node contains 16 Power3 CPUs and 16 GB memory. Each Power3 CPU runs at a 375Mhz clock speed, and has 2 MB L2 cache. Both SCF and DCM are parallelized using MPI. We used the IBM math library ESSL for dense matrix and FFT calculations.

6.1. The SiH4 test problem

In the first example, we applied both algorithms to a simple SiH4 test problem. The wavefunction is defined on a $32 \times 32 \times 32$ real space grid. The number of plane wave basis functions used in the Fourier representation is 2103. The number of occupied states for this molecule is $k = 4$.

In the SCF calculation, we set the convergence tolerance of each PCG run to $\tau = 10^{-12}$ and the maximum number of PCG iterations allowed to 10. That is, we terminate the PCG iteration when

$$\|H^{(i)}x_j^{(i)} - \lambda_j^{(i)}x_j^{(i)}\| \leqslant 10^{-12},$$

or when the number of PCG iterations taken reaches 10. In our experiment, the PCG convergence tolerance was never reached before the maximum number of iterations were taken. Thus each outer SCF iteration consumed 4×10 MATVECS. Both Pulay (DIIS) [21,22] and Kerker [12] charge mixing schemes were used in the outer SCF iteration to accelerate the convergence.

In DCM, the projected minimization problem was solved by applying a simple SCF iteration (without charge mixing) to (28). We set the number of inner SCF iterations to 3.

Fig. 3 shows that both SCF and DCM reached the same total energy level after 15 outer iterations. Clearly, DCM consumed a much smaller number of MATVECs. Furthermore, we observed that the reduction in total

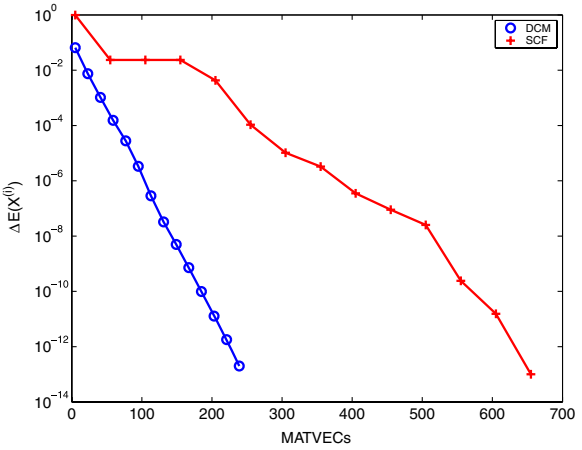

Fig. 3. Comparing the convergence of SCF and DCM in terms of the number of MATVECs performed for the SiH4 problem.

energy is monotonic in DCM. In the PEtot version of SCF, the KS Hamiltonian was not updated until the end of 4th outer iteration. This was required to carry out the charge mixing procedure for accelerating the convergence of SCF. As a result, the total energy objective does not show significant improvement between the 2nd and the 4th outer iteration.

Because the number of occupied states ($k = 4$) in SiH4 is relatively small, both SCF and DCM are dominated by the FFT computation required in each MATVEC. Hence, the MATVEC count shown in Fig. 3 gives a good measure of how the two methods compare in terms of computational speed. In Fig. 4 we plot the reduction of the total energy with respect to the wall clock time used in both SCF and DCM. Both codes were parallelized and executed on 16 Power3 CPUs. The figure shows that DCM is almost 4 times faster than SCF in terms of wall clock time.

6.2. The PtNiO test problem

In the second example, we apply both algorithms to a larger system consisting of 9 atoms and 86 valence electrons. It represents a thin PtNi slab with one O atom attached to the surface. The system is used for catalysis to dissociate O_2 molecules. The wavefunction is defined on a $96 \times 48 \times 48$ real space grid. The number of plane wave basis functions used in the Fourier representation is 15181, and the number of occupied states for this molecule is $k = 43$.

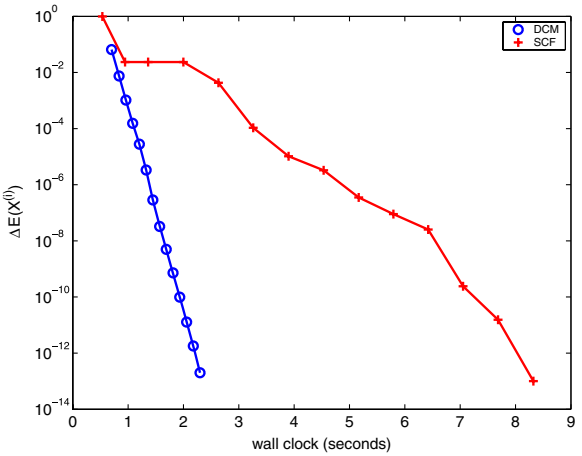

Fig. 4. Comparing the performance of DCM with SCF in terms of wall clock time for the SiH4 problem.

In the SCF calculation, we used the same PCG convergence tolerance as that used in the SiH4 example. That is, each outer SCF iteration consumed 43×10 MATVECS. Both Pulay (DIIS) and Kerker charge mixing schemes were used in the outer SCF iteration to accelerate the convergence.

In the DCM calculation, the projected minimization problem was solved by applying SCF iteration to (28) also. We set the number of inner SCF iterations to 5. Without charge mixing, the inner SCF iteration diverged quickly for this problem. After we introduced both Pulay and Kerker charge mixing in the inner SCF iteration, the total energy converged rapidly. However, the convergence is not monotonic as we can see in Fig. 5. When we used the Pulay mixing in the inner SCF iteration, we mixed charge densities produced from the previous outer iterations also. A maximum of 20 previous charge densities were allowed in the Pulay mixing.

Fig. 5 shows that both SCF and DCM had some difficulty in maintaining a monotonic reduction of the total energy during the course of convergence. After 20 outer iterations, DCM was able to reach a much lower total energy level than that reached by SCF. The DCM run also used a much smaller number of MATVECs than used by SCF. However, performing charge mixing in the inner iteration does not seem to completely stabilize the convergence of the total energy.

In Fig. 6 we plot the reduction of the total energy with respect to the wall clock time used in both SCF and DCM. Both codes were executed on 64 Power3 CPUs. The figure shows that DCM is roughly 2.5 times faster than SCF in terms of wall clock time, which is consistent with the MATVEC count shown in Fig. 5.

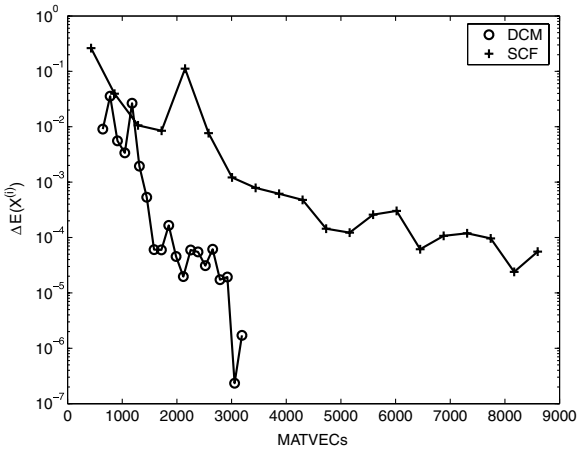

Fig. 5. Comparing the convergence of SCF and DCM in terms of the number of MATVECs performed for the PtNiO problem.

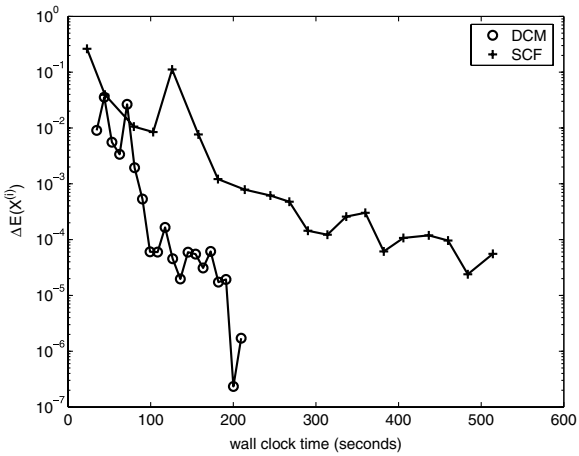

Fig. 6. Comparing the convergence of SCF and DCM in terms of wall clock time (seconds) for the PtNiO problem.

7. Concluding remarks

A direct constrained minimization (DCM) algorithm for computing the ground state total energy of a large scale atomistic system is presented in this paper. The algorithm constructs a new search direction from the subspace $\mathscr{S}$ spanned by the current approximation to the optimal wavefunction, the preconditioned gradient of the total energy and the previous search direction. The optimal search direction and step length are computed by minimizing the total energy functional within $\mathscr{S}$ subject to an orthonormality constraint. Solving this smaller optimization problem is equivalent to solving a small nonlinear eigenvalue problem. In our computational scheme, all wave functions associated with the occupied states are updated simultaneously instead of ''band-by-band''.

We compared the convergence of the DCM algorithm with that of the SCF iteration implemented in PEtot through two numerical examples. We demonstrated that DCM can be more efficient than SCF. Because DCM modifies the wave functions associated with all occupied states simultaneously (a block algorithm), we can make better use of the cache and memory in each MATVEC thereby further improving the performance of the computation.

It is observed that neither DCM nor SCF requires each inner iteration to converge to full accuracy in order to reach the optimal solution. In our numerical experiment, we set the number of inner iterations in both methods to a fixed number. However, this is clearly not the best strategy. More research is required to develop a more effective stopping criterion for the inner iteration in both methods.

In our numerical experiment, the projected optimization problem (21) is solved by applying an SCF iteration to (28). This simple scheme worked very well for the SiH4 system. For many systems such as the PtNiO example shown in Section 6.2, it is necessary to use charge mixing scheme in the inner SCF iteration in order to prevent the algorithm from diverging.

We would like to point out that the projected problem (21) typically has a much smaller dimension compared to the original total energy minimization problem (7). Hence, there is more flexibility in terms of choosing an efficient method for solving the projected problem. For example, it may also be possible to replace the inner SCF iteration with a Quasi–Newton type of algorithm such as the one used in [26]. We will experiment and compare different approaches for solving projected problem (21) in future studies. We will also perform a more comprehensive comparison of DCM with other implementations of the SCF iteration and direct minimization algorithms in the future.

Acknowledgements

This work was supported by the Director, Office of Science, Division of Mathematical, Information, and Computational Sciences of the US Department of Energy under contract number DE-AC03-76SF00098. This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the US Department of Energy under Contract No. DE-AC03-76SF00098.

References

[1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, D. Sorensen, LAPACK Users' Guide, second ed., SIAM, Philadelphia, PA, 1992.
[2] T.A. Arias, M.C. Payne, J.D. Joannopoulos, Ab initio molecular dynamics: analytically continued energy functionals and insights into iterative solutions, Phys. Rev. Lett. 69 (1992) 1077–1080.
[3] P. Bendt, A. Zunger, New approach for solving the density-functional self-consistent field problem, Phys. Rev. B 26 (1982) 3114–3137.
[4] P.T. Boggs, J.W. Tolle, Sequential quadratic programming, Acta Numerica (1995) 1–52.
[5] E.R. Davidson, The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real symmetric matrices, J. Comput. Phys. 17 (1975) 87–94.
[6] A. Edelman, T.A. Arias, S.T. Smith, The geometry of algorithms with orthogonality constraints, SIAM J. Matrix Anal. Appl. 20 (2) (1998) 303–353.
[7] J.G.F. Francis, The QR transformation a unitary analogue to the LR transformation - part 1, Comput. J. 4 (October) (1961) 265–271.
[8] J.G.F. Francis, The QR transformation – part 2, Comput. J. 4 (January) (1962) 332–345.
[9] M.J. Gillan, Calculation of the vacancy formation in aluminum, J. Phys. Condens. Matter 1 (1989) 689–711.

[10] M.R. Hestenes, W. Karush, A method of gradients for the calculation of the characteristic roots and vectors of a real symmetric matrix, J. Res. Nat. Bur. Stand. 47 (1951) 45–61.
[11] J. Hutter, H.P. Lüthi, M. Parrinello, Electronic structure optimization in plane-wave-based density functional calculations by direct inversion in the iterative subspace, Comput. Mater. Sci. 2 (1994) 244–248.
[12] G.P. Kerker, Efficient iteration scheme for self-consistent pseudopotential calculations, Phys. Rev. B 23 (1981) 3082–3084.
[13] A.V. Knyazev, Toward the optimal preconditioned eigensolver: locally optimal block preconditioned conjugate gradient method, SIAM J. Sci. Comput. 23 (2001) 517–541.
[14] G. Kresse, J. Furthmüller, Efficiency of ab initio total energy calculations for metals and semiconductors using a plane-wave basis set, Comput. Mater. Sci. 6 (1996) 15–50.
[15] G. Kresse, J. Furtthmüller, Efficient iterative schemes for *ab initio* total-energy calculations using a plane-wave basis set, Phys. Rev. B 54 (16) (1996) 11169–11185.
[16] C. Lanzcos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, J. Res. Nat. Bur. Stand. 45 (4) (1950) 255–282, Research Paper 2133.
[17] D.C. Liu, J. Nocedal, On the limited memory method for large scale optimization, Math. Program., B 45 (3) (1989) 503–528.
[18] J. Olsen, P. Jorgensen, J. Simons, Passing the one-billion limit in full configuration-interaction (fci) calculations, Chem. Phys. Lett. 169 (1990) 463–472.
[19] M.C. Payne, M.P. Teter, D.C. Allen, T.A. Arias, J.D. Joannopoulos, Iterative minimization techniques for *ab initio* total energy calculation: molecular dynamics and conjugate gradients, Rev. Modern Phys. 64 (4) (1992) 1045–1097.
[20] B.G. Pfrommer, J. Demmel, H. Simon, Unconstrained energy functionals for electronic structure calculations, J. Comp. Phys. 150 (1999) 287–298.
[21] P. Pulay, Convergence acceleration of iterative sequences: The case of SCF iteration, Chem. Phys. Lett. 73 (2) (1980) 393–398.
[22] P. Pulay, Improved SCF convergence acceleration, J. Comput. Chem. 3 (4) (1982) 556–560.
[23] G.L.G. Sleijpen, H.A. Van der Vorst, A Jacobi–Davidson iteration method for linear eigenvalue problems, SIAM J. Matrix Anal. Appl. 17 (2) (1996) 401–425.
[24] M.P. Teter, M.C. Payne, D.C. Allan, Solution of Schrödinger's equation for large systems, Phys. Rev. B 40 (18) (1989) 12255–12263.
[25] J. VandeVondele, Jürg Hutter, An efficient orbital transformation method for electronic structure calculations, J. Chem. Phys. 118 (2003) 4365–4369.
[26] T. Van Voorhis, M. Head-Gordon, A geometric approach to direct minimization, Mol. Phys. 100 (11) (2002) 1713–1721.
[27] L. Wang, Parallel Total Energy Calculation Software, http://hpcrd.lbl.gov/~linwang/PEtot/PEtot.html.