Publication II

K. Baarman and J. VandeVondele. A comparison of accelerators for direct energy minimization in electronic structure calculations. *The Journal of Chemical Physics*, 134, 244104; doi:10.1063/1.3603445, June 2011.

A comparison of accelerators for direct energy minimization in electronic structure calculations

Kurt Baarman[1,a)] and Joost VandeVondele[2]
[1]*Department of Mathematics and Systems Analysis, Aalto University School of Science, Finland*
[2]*Institute of Physical Chemistry, University of Zurich, Switzerland*

(Received 9 December 2010; accepted 5 June 2011; published online 23 June 2011)

We compare three different methods for direct energy minimization in electronic structure calculations where the gradient of the energy functional with respect to the molecular orbitals is available. These methods make use of the preconditioned gradient to increase robustness. An orbital transformation is used to ensure that the orthogonality constraint on the orbitals remains satisfied when using standard minimization methods. In addition, we propose an adaptive scheme for estimating the curvature of the energy functional to increase the performance of a line search free quasi-Newton method. We show that the performance of all methods is similar when robustness of the methods is ensured. © *2011 American Institute of Physics*. [doi:10.1063/1.3603445]

I. INTRODUCTION

The electronic structure of molecules and atoms is of fundamental interest in physics and chemistry. A complete answer for nonrelativistic problems is provided by the Schrödinger equation. It is, however, intractable for all but the simplest cases. In 1965, Kohn and Sham introduced the Kohn-Sham energy,[1,2]

$$E^{KS}[\{\phi_\alpha\}] = T_s[\{\phi_\alpha\}] + \int V_{\text{ext}}(\mathbf{r})\rho(\mathbf{r})d\mathbf{r} + \frac{1}{2}\int V_{\text{H}}(\mathbf{r})\rho(\mathbf{r})d\mathbf{r} + E_{\text{xc}}[\rho], \tag{1}$$

and the corresponding Kohn-Sham equations,

$$\left(\frac{-\hbar^2\nabla^2}{2m} + V_{\text{eff}}(\mathbf{r},\rho(\mathbf{r}))\right)\phi_\alpha(\mathbf{r}) = E_\alpha\phi_\alpha(\mathbf{r}). \tag{2}$$

Here

$$\rho(\mathbf{r}) = \sum_\alpha f_\alpha\,|\phi_\alpha(\mathbf{r})|^2\,, \tag{3}$$

is the electronic density and the index α goes over all orthogonal Kohn-Sham orbitals, ϕ_α, and f_α is the corresponding occupation number. $T_s[\{\phi_\alpha\}]$ is the kinetic energy of the system, $V_{\text{ext}}(\mathbf{r})$ is the external potential, $V_{\text{H}}(\mathbf{r})$ is the Hartree term describing the electron-electron Coulomb interactions, and $E_{\text{xc}}[\rho]$ is the exchange correlation energy. In the Kohn-Sham equations $V_{\text{eff}}(\mathbf{r},\rho(\mathbf{r}))$ is the effective potential.

The ground state of a system can be found either through self-consistent solution of the Kohn-Sham equations or through minimization of the Kohn-Sham energy. This density functional theory (DFT) formulation is in principle equivalent to the Schrödinger equation.[3] DFT is, however, significantly more tractable as the electron density $\rho(\mathbf{r})$ is only dependent on three variables, whereas the wave function of the Schrödinger equation depends on $3N$ variables for a N-particle system. In practice, an approximation has to be introduced into DFT because the exact E_{xc} and thus V_{eff} are unknown.

One further price to pay for this dramatic reduction in complexity is that the Kohn-Sham equations are nonlinear. Repeatedly solving the lowest eigenvectors typically represent a significant part of the computational cost of finding a self-consistent solution to the Kohn-Sham equations. However, one way to avoid the explicit computation of the lowest eigenvectors at every self-consistent step is a direct optimization of Kohn-Sham energy functional.[4–7] Consequently, there exists several methods for minimization of the ground state energy.[8–12] These methods typically rely on constructing a search direction and subsequently minimizing the energy functional along this direction.[6,9,10,12,13] The optimization approach is, however, complicated by the requirement that individual orbitals remain orthogonal through the optimization process. In the CP2K (Ref. 14) code this is achieved by using an orbital transformation (OT) method.[13,15] The OT is a nonlinear change of variables that simplifies the minimization process by transforming the orthogonality constraint into a linearly constrained problem. The current comparison focuses on various minimization methods within the OT scheme. The OT scheme is aimed at systems of intermediate size, up to a few thousand atoms in the condensed phase, that are described with extended basis sets. For these systems the density matrix is dense, but the overlap matrix is sparse, and the OT method outperforms traditional diagonalization approaches, especially on parallel computers.[13,16] For larger systems, linear scaling methods will become an alternative.[17] The OT method can also be used with hybrid functionals, which include Hartree-Fock exchange.[18]

Popular methods for the minimization of the energy functional are direct inversion of the iterative subspace (DIIS), conjugate gradient (CG) minimization methods for non-quadratic problems, and methods based on gradient descent.[8,19,20] DIIS converges quickly when the method is

[a)]Electronic mail: kurt.baarman@tkk.fi.

working properly but has a reputation of breaking down easily.[9] CG is a remarkably robust method, while a disadvantage of the method is the requirement to do a line search for every evaluation of the gradient. Gradient descent methods range from the naive method of steepest descent to Newton's method. We consider a line search free quasi-Newton (QN) method using a slightly modified Broyden's type 2 or *bad* update of the approximate inverse Hessian. The QN method with Broyden's type 2 update offers a robust accelerator for self-consistent solution of the Kohn-Sham equations.[21,22] For a description of the QN updates, see, e.g., Ref. 20, Sec. II A. We also propose an adaptive scheme to decide the step length for descent in the nonsampled directions.

In Sec. II we present the methods used. We then report on the results obtained in Sec. III, and compare the results for the DIIS, CG, and QN methods. Finally, we present our conclusions in Sec. IV.

II. METHODS

Our approach to the minimization problem depends on the orbital transformation method. The N occupied orthonormal orbitals are discretized and expressed as a linear combination of M basis functions using a matrix of orbital coefficients, $\mathbf{C} \in \mathbb{R}^{M\times N}$. When expressed like this, every column of $\mathbf{C}$ corresponds to a discretized orbital. A nonlinear parameterization,[13]

$$\mathbf{C}(\mathbf{X}) = \mathbf{C}_0 \cos(\mathbf{U}) + \mathbf{X}\mathbf{U}^{-1}\sin(\mathbf{U}), \tag{4}$$

is used that transforms the nonlinear constraint on $\mathbf{C}$ into a linear constraint on $\mathbf{X} \in \mathbb{R}^{M\times N}$. Here $\mathbf{U}$ is defined as $\mathbf{U} = (\mathbf{X}^T\mathbf{S}\mathbf{X})^{1/2}$, where $\mathbf{S}$ is the overlap matrix. This transformation allows us to use a standard minimization method on the energy functional instead of being forced to follow a curved geodesic during the minimization process.[13,23]

A preconditioner obtained from the inversion of the positive definite matrix $\mathbf{P} = \mathbf{T} + \epsilon\mathbf{S}$, where $\mathbf{T}$ is the matrix representation of the kinetic energy, $\mathbf{S}$ is the overlap matrix and a positive constant, $\epsilon = 0.2$ is employed to improve the rate of convergence.[13] Nevertheless, such a simple preconditioner cannot avoid that the number of iterations increases as the HOMO-LUMO gap decreases. For conjugate gradient minimization methods, it has been shown that the number of iterations is expected to grow with the square root of the inverse gap.[24,25] The systems studied here are therefore non-metallic systems, which have a finite gap.

This allows us to consider the differentiable objective function $f(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}$ bounded from below, with the known gradient, $\mathbf{f}_k := \mathbf{P}_{\mathbf{x}}^{-1}\nabla f(\mathbf{x}_k)$. Here $\mathbf{x}$ is a vector corresponding to the parameterization, $\mathbf{X}$, of $\mathbf{C}$, $f(\mathbf{x})$ corresponds to the Kohn-Sham energy, and $\mathbf{P}_{\mathbf{x}}$ is the corresponding preconditioner. We represent the degrees of freedom, $\mathbf{x}$, as a vector to simplify presentation of the minimization methods. In practice, $\mathbf{x}$ is often stored in some other format which must be taken into account when implementing the methods.

Construction of an approximate Hessian from secant conditions permits the use of quasi-Newton methods even when f is not twice differentiable everywhere.[26] Nevertheless, we begin by considering the case where the Hessian, $\mathbf{H} := D^2 f(\mathbf{x})$, is symmetric and positive definite $\forall\,\mathbf{x}$.

A. Direct inversion of the iterative subspace

The DIIS method by Pulay is based on minimizing the norm of a linear combination of the gradient vectors.[8] The problem is then

$$\min_{c_i} \left\| \sum_{i=1}^{m} c_i \mathbf{f}_i \right\| \quad \text{subject to} \quad \sum_{i=1}^{m} c_i = 1. \tag{5}$$

The next trial solution is obtained by using the minimizing coefficients to construct the next evaluation point,

$$\mathbf{x}_{m+1} = \sum_{i=1}^{m} c_i(\mathbf{x}_i - \tau\mathbf{f}_i), \tag{6}$$

where $\tau = 0.15$ is a step size chosen for the DIIS method. The arbitrary step size τ is included to adapt DIIS for use with minimization methods. When $m = 1$, the method reduces to the method of steepest descent.

While the DIIS method is very fast in cases where it converges, it is not particularly robust. In the QUICKSTEP code, the DIIS method is modified to change to steepest descent when a DIIS step would bring the solution towards an ascent direction. This safeguard makes DIIS more robust and is possible because the gradient of the energy functional is available.

B. Conjugate gradient methods for non-quadratic problems

The CG method used as a minimization method is based on successively minimizing the function along conjugate directions. The set of vectors $\mathbf{d}_1, \ldots, \mathbf{d}_k$ are $\mathbf{A}$-conjugate if they are linearly independent and $\mathbf{d}_i^T\mathbf{A}\mathbf{d}_j = 0$ when $i \neq j$. Minimization along conjugate directions has the advantage that the solution to the quadratic problem

$$\min g(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} - \mathbf{b}^T\mathbf{x}, \tag{7}$$

is found in n steps, for $\mathbf{x} \in \mathbb{R}^n$, when an exact line search is used.[19]

In DFT calculations the objective function is generally not quadratic, and the update of the search direction must take this into account to preserve conjugacy of the search directions. The direction of steepest descent, $\mathbf{d}_1 = -\mathbf{f}_1$, is used as the first search direction and the update formula for the search direction is

$$\mathbf{d}_{j+1} = -\mathbf{f}_{j+1} + \alpha_j\mathbf{d}_j. \tag{8}$$

Several choices for α_j are available and we make use of the Polak-Ribière variant,[19]

$$\alpha_j = \frac{\mathbf{f}_j^T(\mathbf{f}_{j+1} - \mathbf{f}_j)}{||\mathbf{f}_j||^2}. \tag{9}$$

The step length is then decided by solving the problem,

$$\min_{\gamma} f(\mathbf{x}_j + \gamma\mathbf{d}_j), \tag{10}$$

using an approximate line search. The updated position becomes $\mathbf{x}_{j+1} = \mathbf{x}_j + \gamma \mathbf{d}_j$.

In CP2K an extremely robust golden section line search is available but the default choice is minimization of a quadratic interpolation along the search direction. The quadratic interpolation is in practice significantly faster than the golden section line search and is therefore used in all our calculations.

C. Gradient descent

The method of steepest descent is the most straightforward minimization scheme when the gradient is available. The iteration is

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \beta \mathbf{f}_k, \tag{11}$$

where β is a step length parameter. Several strategies are available for determining the used step length, ranging from fixed β to line search. A disadvantage of the steepest descent method is that when the optimum is in a narrow valley the steepest descent method has a tendency to zigzag at the floor of the valley. This behavior significantly decreases the performance of the method.

An improvement over the steepest descent method, when the inverse Hessian is available, is to use Newton's method. The gradient is premultiplied with the inverse Hessian to give both direction and step length. As a result, Newton's method is scale invariant, and the zigzag behavior that plagues the method of steepest descent is not present. The iteration for Newton's method is

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \beta\, \mathbf{H}(\mathbf{x}_k)^{-1} \mathbf{f}_k. \tag{12}$$

Newton's method exhibits super-linear convergence when the initial guess is close to the solution if $\beta = 1$. On the other hand, when the initial guess is further away from the solution Newton's method may diverge. This divergent behavior can be suppressed by the introduction of line search or backtracking.

For problems that are well approximated by a quadratic function, we can avoid line search by using under relaxation. We choose a fixed $\beta \in (0, 1]$ to force the method to proceed more cautiously. For quadratic problems $\beta = 1$ gives the optimum step length, while $\beta < 1$ increases robustness but slows convergence for higher order problems.

For problems where the Hessian of the objective function is available, Newton's method with line search or backtracking provides a remarkably good method. However, for DFT problems, the Hessian is in general not available. We can instead substitute the true Hessian in Newton's method for an approximation of the Hessian to produce a quasi-Newton method.[20] A well chosen approximation scheme makes it possible to retain an efficient method, while still keeping the problem size feasible. The next step of the iteration is then determined by the formula,

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \beta\, \mathbf{G}_k \mathbf{f}_k, \tag{13}$$

where $\mathbf{G}_k$ is the approximate inverse Hessian at step k.

A shortcoming of the standard quasi-Newton method is that it converges towards any point where the gradient is zero. In particular, problems can be caused by saddle points. Since the Hessian is not positive definite near saddle points, Eq. (12) leads to ascent steps towards the saddle point. Broyden's update formula ensures that a positive definite initial approximation of the inverse Hessian stays positive definite if the Hessian of the underlying optimization problem is positive definite everywhere. The latter property does not hold for DFT calculations, and it is therefore possible that the quasi-Newton method converges towards a saddle point or maxima unless precautions are taken.

1. Compact update formula

We have used Broyden's type 2 update to construct the approximate inverse Hessian. This is one of several methods by which $\mathbf{G}_k$ can be constructed. We use a compact update formula,[19,22,27]

$$\hat{\mathbf{G}}_{k+1} = \hat{\mathbf{G}}_k + (\Delta\hat{\mathbf{x}}_k - \hat{\mathbf{G}}_k \Delta\hat{\mathbf{f}}_k)\frac{\Delta\hat{\mathbf{f}}_k^T \mathbf{S}_\mathbf{Y}}{\Delta\hat{\mathbf{f}}_k^T \mathbf{S} \Delta\hat{\mathbf{f}}_k}, \tag{14}$$

where $\mathbf{Y} = [\mathbf{x}_0, \ldots, \mathbf{x}_{m-1}, \mathbf{f}_0, \ldots, \mathbf{f}_{m-1}, \mathbf{y}]$, $\mathbf{S}_\mathbf{Y} = \mathbf{Y}^T\mathbf{Y}$, and $\mathbf{y} \in \mathbb{R}^n$ is a random vector that we keep constant during the entire minimization. We define $\hat{\mathbf{G}}_0 = \sigma \mathbf{I} \in \mathbb{R}^{(2m+1)\times(2m+1)}$, $\hat{\mathbf{x}}_k = \mathbf{e}_k \in \mathbb{R}^{2m+1}$, and $\hat{\mathbf{f}}_k = \mathbf{e}_{m+k} \in \mathbb{R}^{2m+1}$. It then holds that $\mathbf{f}_k = \mathbf{Y}\hat{\mathbf{f}}_k$ and $\mathbf{x}_k = \mathbf{Y}\hat{\mathbf{x}}_k$. As expected, $\Delta\hat{\mathbf{f}}_k = \hat{\mathbf{f}}_{k+1} - \hat{\mathbf{f}}_k$ and $\Delta\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k+1} - \hat{\mathbf{x}}_k$. We have slightly modified $\hat{\mathbf{G}}$ to make $\hat{\mathbf{G}} : \mathbb{R}^{2m+1} \to \mathbb{R}^{2m+1}$, and to include $\mathbf{y}$. These changes have been made to permit the use of $\hat{\mathbf{G}}$ with iterative methods, as presented in Sec. II C 2. We have included $\mathbf{y}$ to be able to represent a starting vector for the iterative method that is not in span $(\mathbf{x}_0, \ldots, \mathbf{x}_{m-1}, \mathbf{f}_0, \ldots, \mathbf{f}_{m-1})$. Including a random vector component in the starting vector for the Arnoldi iteration ensures that we do not pick e.g. an eigenvector of $\hat{\mathbf{G}}$ as the starting vector.

This update formula has the property that the secant condition, $\mathbf{G}_{k+1}\Delta\mathbf{f}_k = \Delta\mathbf{x}_k$, is satisfied in the most recently sampled direction, and that $\mathbf{G}$ remains unchanged in nonsampled directions. This means that steepest descent is used in nonsampled directions, and the method suffers the disadvantages present in the steepest descent method for those directions.

For an approximate inverse Hessian in compact form, Eq. (13) becomes

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \beta\, \mathbf{Y}\hat{\mathbf{G}}_k\hat{\mathbf{f}}_k. \tag{15}$$

To prevent convergence towards saddle points and maxima we change the sign of $\Delta\hat{\mathbf{x}}\, \Delta\hat{\mathbf{f}}^T \mathbf{S}_\mathbf{Y}$ when $\Delta\hat{\mathbf{f}}^T \Delta\hat{\mathbf{x}} < 0$. This does not ensure a positive definite $\hat{\mathbf{G}}_{k+1}$ but seems to work well in practice.

2. Adaptive curvature information

Using a line search at each step of the QN iteration increases the cost of each step. When $\mathbf{G}_k$ is a good approximation of $\mathbf{H}^{-1}$, and we are close to the quadratic region of f, the QN method produces acceptable step lengths without the need for a line search.

We propose a way to adaptively refine the initial approximation, $\mathbf{G}_0 = \sigma\mathbf{I}$, by incorporating information gained from the eigenvalues of previously calculated $\mathbf{G}_k$. σ is a measure of the second derivative of f in all directions, and by choosing σ to be representative of the curvature of f we attempt to improve the convergence rate.

The eigenpairs of $\mathbf{G}_k$ carry information on the curvature of f. Even disregarding the eigenvectors, the eigenvalues of $\mathbf{G}_k$ represent a sampling of the curvature of f in different directions. If we can assume that this sampling is representative of the curvature of f in nonsampled directions we can obtain a crude approximation for the curvature of f.

Due to the storage requirements for a full representation of the inverse Hessian we cannot use a direct method to calculate the singular values of $\mathbf{G}_k$. Instead, we use the Arnoldi iteration to construct a Hessenberg matrix with eigenvalues approximating the eigenvalues of the full low rank approximation of the inverse Hessian.[28] The eigenvalues of the Hessenberg matrix are then solved with a standard eigenvalue solver. Construction of the Hessenberg matrix can be done entirely in the compact formulation. The required changes to the Arnoldi iteration are to take $\mathbf{S}_\mathbf{Y}$ into account when calculating the inner product. We use a $\mathbb{R}^{2m+1}$ vector with uniform values and norm 1 as the starting vector. This choice of starting vector has significant components in the sampled directions as well as a component in a random nonsampled direction.

After solving the eigenvalues, λ_α of $\mathbf{G}_k$ we construct a new estimate for the step length, $\tilde{\sigma}$. The estimator we use is

$$\tilde{\sigma} = \frac{\sum_\alpha |\lambda_\alpha|^2}{\sum_\alpha |\lambda_\alpha|}, \tag{16}$$

where the index α goes over all eigenvalues obtained by the Arnoldi iteration. This estimator puts an emphasis on larger eigenvalues of $\mathbf{G}$, and gives the best results among the estimators we have considered. In a typical DFT calculation, the amount of sampled directions is very small compared to the dimension of the problem. For this reason we do not use $\tilde{\sigma}$ directly. Instead we increase the current σ if $\tilde{\sigma}$ leads us to believe that there is room for an increase of step length. We also reduce σ by multiplying it with 0.7 whenever we are forced to backtrack. Backtracking is done to ensure that the energy does not increase during the iteration. We use the update formula,

$$\sigma_{k+1} = \max(\sigma_{\min}, \min(\eta\tilde{\sigma}_k, \omega\sigma_k)), \tag{17}$$

where $\eta = 0.7$ dampens the estimator $\tilde{\sigma}$, $\omega = 1.1$ limits the growth rate of σ, and $\sigma_{\min}$ is included to avoid breakdown by reducing the step length to far. The estimated σ is also used to determine step length for the minimization problem at later molecular dynamics (MD) time steps.

III. RESULTS

Here we report the results obtained by comparing the minimization methods. We have compared DIIS, CG, and two variants of Newton's method with Broyden's type 2 update. The comparisons are done with the QUICKSTEP (Ref. 16) code, available in the CP2K suite of programs.[14] CP2K is free software, available under the general public license (GPL).[29] We present results for two subsets of the GMTKN24 data sets, BHPERI and MB08-165.[30] In addition, an electronic ground state calculation of trilinear cobalt and a 10 time step MD calculation of the BHPERI data set has been performed as well as four 200 time step MD calculation of 64 H_2O molecules in periodic boundary conditions with different perturbations. These are the removal of an electron, the addition of an electron, and the removal of an hydrogen. These strong perturbations make the system more difficult to converge. Systems with an odd number of electrons have been described using an unrestricted formalism with the lowest multiplicity. The trilinear cobalt consists of three cobalt atoms positioned on an axis with an interatomic distance of 2.034535 Å. The benchmark cases include both finite and periodic insulators.

All calculations have been performed using pseudopotentials[31] and employed a basis of DZVP quality.[32] The exchange correlation functional for the 64 H_2O calculations is PBE,[33,34] while the remaining calculations have been done with the Padé interpolation to the Perdew-Wang functional[33] proposed in Ref. 31. The initial wave function for static calculations and the first step of MD calculations are obtained from atomic calculations. During MD, the initial wave function for subsequent steps is obtained using an extrapolation of the product of the overlap and density matrices.[16]

The convergence criterion has been $||\mathbf{f}|| \leq 3 \times 10^{-6}$ for the MB08-165, BHPERI, and BHPERI MD data sets, consistent with the setting used in our earlier work,[18] and appropriate for small gas phase molecules. The cobalt and bulk liquid systems have been studied using the tighter setting $||\mathbf{f}|| \leq 10^{-6}$ as is commonly employed for MD simulations.

The DIIS and CG minimization methods are the default implementations in the CP2K program, while the QN_{ad} method is the default adaptive implementation of the QN method when $\sigma = 1.0$ is used as the starting value for the inverse Hessian approximation, $\hat{\mathbf{G}} = \sigma\mathbf{I}$. For the nonadaptive QN method we fixed $\beta = 0.7$ and determined σ by calculating the BHPERI data set with $\sigma = 1.0, 0.5, 0.25$. The best result was obtained with $\sigma = 0.5$ and we present this as QN_{non}.

Since one iteration of the CG method includes both a gradient evaluation and a line search, the number of iterations required for convergence is not an accurate measure to compare performance to line search free methods. We therefore multiply the number of CG iterations by 1.6, a factor which has been obtained as the average cost ratio between CG and the DIIS/QN iterations for the MB08-165, BHPERI, and the MD BHPERI calculations. In practice, this ratio depends on the relative cost of the energy evaluation and the gradient evaluation, which in turns depends on system composition, system size, and computational details but remains in the range of 1.3-2.0 if quadratic interpolation for the line search algorithm is employed.

The number of iterations required for convergence of the electronic ground state calculations are presented in Table I. Average number of iterations needed per time step for MD calculations can be found in Table II. The geometry for which the maximum number of iterations was realized is identified in Table III.

All methods offer similar performance for the electronic ground state calculations presented in Table I, except for the trilinear cobalt calculation. The QN_{non} method naturally

TABLE I. Number of iterations required for convergence of electron ground state calculations. For data sets with several geometries, both mean number of iterations and iterations for the most challenging case for the minimization method in question is given. For the CG method an estimated equivalent number of iterations are given. The geometry for which the maximum number of iterations was realized is identified in Table III. Detailed results, including the electronic gap for these systems, are available as supplementary materials (Ref. 35).

Data set	DIIS	CG	QN_{ad}	QN_{non}
MB08-165 mean	42	51	48	55
MB08-165 max	127	155	139	194
BHPERI mean	37	42	40	36
BHPERI max	91	58	83	59
Trilinear Co	3376	1442	689	789

offers very good performance for the BHPERI data set for which it was optimized. Performance of the QN_{non} minimization method is worse for MB08-165 with $\sigma = 0.5$ but improves when $\sigma = 0.25$ is used. While QN_{ad} cannot match the performance of QN_{non} on the data set for which it was optimized, QN_{ad} improves performance over QN_{non} for the MB08-165 data set.

The trilinear cobalt calculation presented in Table I is a very demanding calculation for all methods. For this system the DIIS approximation fails and the minimization method needs to employ steepest descent in 69% of the steps. Progress towards the energy minimum is almost stagnant during the steepest descent steps. The lowest energy is obtained with the QN minimization method, and the final energy obtained is within 2×10^{-4} a.u. for all methods.

The DIIS method performs well on the MD calculations presented in Table II. Using the QN_{ad} method requires roughly 20% more iterations to converge, and the CG requires 40% more estimated equivalent iterations for convergence than the DIIS method.

We have also calculated the MB08-165 and BHPERI data sets with DIIS without ensuring that the method proceeds in a descent direction. For calculations on the MB08-165 data set the standard DIIS method either failed to converge in 420 iterations or converged to a state with a higher energy than the one obtained by the more robust DIIS method in 40 out of 180 cases. With the same failure criteria as above, the standard DIIS method did not converge for 19 geometries out of 61 for the BHPERI data set. Convergence of the remaining cases of the MB08-165 and BHPERI data sets required essentially the same number of iterations for the standard DIIS and more robust DIIS versions. To determine if the calculations converged to states with the same energy we have used an energy cutoff of 10^{-3} a.u.

TABLE II. Average number of iterations required per time step of MD calculations. For data sets with several geometries, both mean number of iterations and iterations for the most challenging case for the minimization method in question is given. For the CG method an estimated equivalent number of iterations are given. The geometry for which the maximum number of iterations was realized is identified in Table III.

Data set	DIIS	CG	QN_{ad}	QN_{non}
BHPERI MD mean	8.9	12	10	9.2
BHPERI MD max	14	15	14	13
64 H_2O pure	6.5	9.9	8.1	7.3
64 H_2O hole	32	49	40	24
64 H_2O electron	7.7	9.9	8.6	8.8
64 H_2O hydrogen	18	23	21	19

TABLE III. The name of the geometry, as used in the GMTKN24 database, (Ref. 30) for which the maximum number of iterations was needed to reach convergence. The number of iterations required for convergence is presented in Tables I and II.

Data set	DIIS	CG	QN_{ad}	QN_{non}
MB08-165	080	040	040	040
BHPERI	13r_9	13r_7	13r_9	13r_9
BHPERI MD	13r_9	13r_2	13r_9	13r_2

IV. CONCLUSIONS

Performance for all minimization methods are similar for the data sets we have considered. Significant differences are present only for trilinear cobalt, for which both the DIIS method and the CG method performs poorly.

DIIS gives excellent performance for the majority of problems but robustness is a concern. Without safeguards the DIIS method fails in well over 20% of our calculations. If the method is safeguarded by using steepest descent when the step proposed by DIIS is in an ascent direction the method becomes significantly more robust. However, steepest descent suffers from slow convergence in ill-conditioned problems. Therefore, frequent reliance on steepest descent indicates poor performance of the safeguarded DIIS method, and when this occurs a different minimization method should be considered.

Performance of the QN method depends on the choice of σ. When a good value for the curvature is used, performance of the QN method is excellent. The best value for σ is, however, problem dependent, and not known *a priori*. We have proposed an adaptive scheme determining σ that increases performance of the line search free QN method in cases for which an *a priori* chosen fixed value for σ is not well suited. This eliminates the need for the user to select the parameter correctly. Overall, the QN method is a robust and competitive method.

ACKNOWLEDGMENTS

One of the authors (K.B.) is grateful for the support of the Magnus Ehrnrooth foundation, Grant No. MA2010n3.

[1] W. Kohn and L. J. Sham, Phys. Rev. **140**, A1133 (1965).
[2] D. Marx and J. Hutter, *Ab Initio Molecular Dynamics* (Cambridge University Press, Cambridge, England, 2009).
[3] P. Hohenberg and W. Kohn, Phys. Rev. **136**, B864 (1964).
[4] G. Kresse and J. Furthmüller, Phys. Rev. B **54**, 11169 (1996).
[5] G. Kresse and J. Furthmüller, Comput. Mater Sci. **6**, 15 (1996).
[6] N. Marzari, D. Vanderbilt, and M. C. Payne, Phys. Rev. Lett. **79**, 1337 (1997).
[7] Y. Saad, J. R. Chelikowsky, and S. M. Shontz, SIAM Rev. **52**, 3 (2010).
[8] P. Pulay, Chem. Phys. Lett. **73**, 393 (1980).

[9] T. van Voorhis and M. Head-Gordon, Mol. Phys. **100**, 1713 (2002).
[10] Eric Cancès, J. Chem. Phys. **114**, 10616 (2001).
[11] A. A. Mostofi, P. D. Haynes, C.-K. Skylaris, and M. C. Payne, J. Chem. Phys. **119**, 8842 (2003).
[12] C. Freysoldt, S. Boeck, and J. Neugebauer, Phys. Rev. B **79**, 241103 (2009).
[13] J. VandeVondele and J. Hutter, J. Chem. Phys. **118**, 4365 (2003).
[14] See http://cp2k.berlios.de/ for the latest version of CP2K.
[15] V. Weber, J. VandeVondele, J. Hutter, and A. M. N. Niklasson, J. Chem. Phys. **128**, 084113 (2008).
[16] J. VandeVondele, M. Krack, F. Mohamed, M. Parrinello, T. Chassaing, and J. Hutter, Comput. Phys. Commun. **167**, 103 (2005).
[17] S. Goedecker and G. Scuseria, Comput. Sc. Eng. **5**, 14 (2003).
[18] M. Guidon, J. Hutter, and J. VandeVondele, J. Chem. Theory Comput. **6**, 2348 (2010).
[19] J. Nocedal and S. J. Wright, *Numerical Optimization* (Springer, New York, 2006).
[20] H.-r. Fang and Y. Saad, Numer. Linear Algebra Appl. **16**, 197 (2009).
[21] L. D. Marks and D. R. Luke, Phys. Rev. B **78**, 075114 (2008).
[22] K. Baarman, T. Eirola, and V. Havu, J. Chem. Phys. **134**, 134109 (2011).
[23] A. Edelman, T. A. Arias, and S. T. Smith, SIAM J. Matrix Anal. Appl. **20**, 303 (1998).
[24] F. Tassone, F. Mauri, and R. Car, Phys. Rev. B **50**, 10561 (1994).
[25] J. F. Annett, Comput. Mater. Sci. **4**, 23 (1995).
[26] Liqun Qi, Oper. Res. Lett. **20**, 223 (1997).
[27] M. Kawata, C. M. Cortis, and R. A. Friesner, J. Chem. Phys. **108**, 4426 (1998).
[28] L. N. Trefethen and D. Bau III, *Numerical Linear Algebra* (SIAM, Philadelphia, 1997).
[29] See http://www.gnu.org/licenses/gpl.html for a description of the GPL license.
[30] L. Goerigk and S. Grimme, J. Chem. Theory Comput. **6**, 107 (2010).
[31] S. Goedecker, M. Teter, and J. Hutter, Phys. Rev. B **54**, 1703 (1996).
[32] J. VandeVondele and J. Hutter, J. Chem. Phys. **127**, 114105 (2007).
[33] J. P. Perdew and Y. Wang, Phys. Rev. B **45**, 13244 (1992).
[34] J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. **78**, 1396 (1997).
[35] See supplementary material at http://dx.doi.org/10.1063/1.3603445 for iterations required for convergence and the HOMO-LUMO gap of the systems.