Publication I

K. Baarman, T. Eirola, and V. Havu. Robust acceleration of self-consistent field calculations for density functional theory. *The Journal of Chemical Physics*, 134, 134109; doi:10.1063/1.3574836 , April 2011.

Robust acceleration of self consistent field calculations for density functional theory

K. Baarman,[1,a] T. Eirola,[1] and V. Havu[2]
[1]*Department of Mathematics and Systems Analysis, Aalto University School of Science, Espoo, Finland*
[2]*Department of Applied Physics, Aalto University School of Science, Espoo, Finland*

(Received 9 September 2010; accepted 16 March 2011; published online 6 April 2011)

We show that the type 2 Broyden secant method is a robust general purpose mixer for self consistent field problems in density functional theory. The Broyden method gives reliable convergence for a large class of problems and parameter choices. We directly mix the approximation of the electronic density to provide a basis independent mixing scheme. In particular, we show that a single set of parameters can be chosen that give good results for a large range of problems. We also introduce a spin transformation to simplify treatment of spin polarized problems. The spin transformation allows us to treat these systems with the same formalism as regular fixed point iterations. © *2011 American Institute of Physics*. [doi:10.1063/1.3574836]

I. INTRODUCTION

A nonrelativistic system is completely described by the Schrödinger equation. It is, however, intractable for all but the simplest cases. In 1965 Kohn and Sham introduced the equation[1]

$$\left(\frac{-\hbar^2\nabla^2}{2m} + V_{\mathrm{Eff}}(\mathbf{r}, \rho(\mathbf{r}))\right)\phi_i(\mathbf{r}) = E_i\phi_i(\mathbf{r}). \tag{1}$$

Here

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

where the index α goes over all occupied orbitals and f_α is the occupation number. This density functional theory (DFT) formulation is in principle equivalent to the full quantum mechanical explanation of matter.[2] DFT however becomes significantly more tractable as the electron density ρ is only dependent on three variables, whereas the wave function of the Schrödinger equation is in $\mathbb{R}^{3N}$ for an $N-$particle system. In practice an approximation is introduced into the DFT formulation because we do not know the exact V_{Eff}.

The price to pay for the considerable reduction in complexity is that Eq. (1) is nonlinear. The nonlinear equation can be solved either as a minimization problem or as a nonlinear eigenvalue problem. Solving Eq. (1) as a nonlinear eigenvalue problem is done as a self consistent field (SCF) problem. The SCF iteration is required as Eqs. (1) and (2) are interdependent. A significant cost of the SCF iteration is the need to solve the lowest eigenvectors of Eq. (1) for each step of the iteration.[3,4]

Instead of solving the Kohn–Sham equation [Eq. (1)] through the SCF cycle a direct minimization of the ground state energy can be done. This is a mathematically appealing method that does not require the lowest eigenvector to be solved.[5,6] Direct minimization has, therefore, attracted a lot of attention, and several methods for minimization of the ground state energy are available.[7–10] These methods are typically based on constructing a search direction and subsequently minimizing the energy functional on the search direction.[6–8,10] One challenge present in direct minimization is the requirement that the Kohn–Sham orbitals remain orthogonal, and the minimizers are therefore constrained to a nonlinear surface.[11–13]

SCF problems are also encountered in several other computational science problems, and several strategies are available for solving these problems. The SCF problem is a fixed point problem: find $\mathbf{x}$ such that $\mathbf{g}(\mathbf{x}) = \mathbf{x}$. Here $\mathbf{g}(\mathbf{x})$ is the output of a function that is typically both complex and expensive to evaluate. For the DFT calculations we take $\mathbf{x}$ to represent the approximate electron density of the system, and $\mathbf{g}(\cdot)$ is composed of both solving the necessary eigenstates from Eq. (1) and constructing the next density approximation from Eq. (2). Directly mixing the approximate electron density instead of a representation in some basis provides a basis independent mixing scheme, provided that the approximation of the density is sufficiently close to the true density. In the mathematics community mixing is known as acceleration by linear combinations with previous iterates.

A simple approach is to solve the SCF problem as a fixed point iteration, where the result from the previous step is used as the trial solution for the next step. If a good initial guess is available, an under-relaxed fixed point iteration can be an efficient solution method. However, for Eqs. (1) and (2) the cost of each iteration is significant, and it is of interest to reduce the number of iterations required for convergence as much as possible.

The Pulay method is widely used to accelerate the fixed point iteration in DFT calculations of electron structure.[5,14,15] While typically quite efficient, the Pulay method suffers from a lack of robustness. The convergence path can be highly non-smooth, and optimal parameter choice depends very much on the problem. If the parameters are not well chosen, convergence rate can suffer dramatically or the method does not

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

converge at all. Unfortunately, a good choice of parameters is in general not known *a priori*, and an extensive search for efficient mixer parameters can itself become computationally expensive. This lack of robustness of the Pulay method is also present in direct minimization techniques.[7,8]

To solve the problem in practice we take $\mathbf{f}(\mathbf{x}) = \mathbf{x} - \mathbf{g}(\mathbf{x})$ and require[16]

$$\mathbf{f}(\mathbf{x}) = \mathbf{0}. \tag{3}$$

This type of problem is ideally solved with Newton's method when the derivative of $\mathbf{f}$ is available and the starting point is not too far from the solution. However, for DFT problems the derivative is typically unavailable in analytic form and does not necessarily exist everywhere.[17] Local finite difference approximations of $D\mathbf{f}$ are impractical because n is typically very large, and evaluation of $\mathbf{f}$ is expensive.[4] It is therefore necessary to use a mixing scheme that does not rely on the use of $D\mathbf{f}$.

We can consider $\mathbf{x}_k$ and $\mathbf{f}_k := \mathbf{f}(\mathbf{x}_k)$ for $k = 1, 2, \ldots$ as already discretized vectors in $\mathbb{R}^n$ instead of continuous densities and attempt to find a sequence of vectors $\mathbf{x}_0, \mathbf{x}_1, \ldots$ such that $\mathbf{f}_k \to \mathbf{0}$ as $k \to \infty$. To do this we construct an approximate inverse derivative, $\mathbf{G}_k$, of the true inverse derivative, $D\mathbf{f}_k^{-1}$, with the information gained from previous evaluations of $\mathbf{f}$. If we then substitute $\mathbf{G}_k$ for $D\mathbf{f}_k^{-1}$ in Newton's method we obtain a quasi-Newton method.

The construction of $\mathbf{G}_k$ based on secant conditions provides robustness and ensures that the approximation can be constructed even when $D\mathbf{f}$ does not exist. This permits us to construct $\mathbf{G}_k$ as a low rank update of an initial approximation, $\mathbf{G}_0 = \sigma\mathbf{I}$. Here σ corresponds to the under-relaxation of the fixed point part of the iteration. Using a low rank update to construct $\mathbf{G}_k$ makes it possible to store a representation of $\mathbf{G}_k$ and makes operating with it inexpensive. Strong differentiability of $\mathbf{f}$ is not necessary for superlinear convergence of quasi-Newton methods.[18]

The Broyden family of methods is a well known class of secant methods.[5,19–21] We have considered the type 1, or Broyden *Good* method, the type 2, or Broyden *Bad* method, and a generalization of the Broyden method that simultaneously takes into account several secant conditions.[22,23] The type 1 methods differ from the type 2 methods in that the type 1 method constructs an approximation of $D\mathbf{f}$ that is then inverted, while the type 2 method directly constructs an approximation of $(D\mathbf{f})^{-1}$. For both the type 1 and 2 methods, the update is chosen such that the change of the approximation is minimal in the Frobenius norm, i.e., the least squares sense, while the new secant condition is satisfied. This generally causes updates to partly override previous secant conditions. The generalized methods in contrast requires a set of secant conditions to be satisfied simultaneously.

We have compared the performance of secant methods with the Pulay method. The comparisons are done with the GPAW code, a real space DFT implementation of the projector augmented wave method.[24–26] GPAW makes use of the ASE atomistic simulation environment.[27] We utilize the default implementations of the Pulay method available in GPAW and evaluate performance of the Broyden method by comparing it with the results obtained by the Pulay method for a number of test cases.

In Sec. II we recall the Pulay method and the type 2 Broyden method and present variations that we have implemented and tested. We then report the results obtained in Sec. III and present our conclusions in Sec. IV. A more detailed presentation of secant methods is available in Appendix A, while our implementation of the type 2 Broyden method is reported in Appendix B.

II. METHODS

The most straightforward way to solve Eq. (3) is to start with a trial solution $\mathbf{x}_0$ and update the vectors by

$$\tilde{\mathbf{x}}_{k+1} = \mathbf{x}_k - \mathbf{f}(\mathbf{x}_k). \tag{4}$$

To achieve convergence, the update of the trial solution is under-relaxed by $\beta \in (0, 1]$ such that

$$\mathbf{x}_{k+1} = (1 - \beta)\mathbf{x}_k + \beta\tilde{\mathbf{x}}_{k+1}. \tag{5}$$

This approach unfortunately requires $\beta \ll 1$ for many cases resulting in extremely slow convergence. Under-relaxation is closely related to level shifting, which is another technique used to force convergence.[28] To improve the rate of convergence, Eq. (4) is typically replaced by a more advanced mixer.

While we do not explicitly guard against an unphysical negative trial density, construction of the next density will give a globally positive density. In this case $||\mathbf{f}_{k+1}||$ becomes large, and the mixer will be repulsed from the unphysical density.

A. Residual minimization

The Pulay method is a mixing strategy based on minimization of the norm of the linear combination of the evaluations of $\mathbf{f}(\mathbf{x}_k)$. The next trial vector is obtained by constructing it from the minimizing coefficients and the previous trial vectors. That is, solve[14]

$$\min_{c_i} ||\sum_{i=1}^{m} c_i \mathbf{f}(\mathbf{x}_i)||, \tag{6}$$

subject to

$$\sum_{i=1}^{m} c_i = 1, \tag{7}$$

and construct the next trial vector as

$$\tilde{\mathbf{x}}_{m+1} = \sum_{i=1}^{m} c_i \mathbf{x}_i. \tag{8}$$

In other words, the Pulay mixer assumes that if a suitably weighted residual average is close to zero, then the corresponding linear combination of trial solutions would be a good candidate for the next trial solution.

B. Secant methods

Another class of accelerators for Eq. (4) is secant methods. These are based on the attempt to improve the approximation for the Jacobian matrix, $D\mathbf{f}$, of the SCF problem. As the Jacobian for a problem of n degrees of freedom requires storage of n^2 values, storage of a full representation of $D\mathbf{f}$ is unfeasible for large problems. Instead we use low rank updates of an initial guess for the Jacobian, and treat the estimated Jacobian as an operator in contrast to a matrix. We are in general interested in the inverse of the Jacobian and can write the update rules directly for the inverse operator.

One class of secant operators is the Broyden family of methods. We have implemented and tested the type 1 and type 2 Broyden methods as well as the type 2 Andersson's method.[16] Initial trials indicate that the type 2 Broyden methods work better than the type 1 Broyden methods. Marks and Luke attribute this to a better handling of ill posed problems by the type 2 Broyden method.[29] The type 2 Broyden also seemed more robust than the type 2 Andersson's method in initial trials. We have also tested the nonlinear Eirola–Nevanlinna method,[30,31] but it has not showed a benefit over the type 2 Broyden method in our tests. Methods other than the type 2 Broyden method is presented in more detail in Appendix A.

The vector update for the type 2 Broyden method is[4,19]

$$\tilde{\mathbf{x}}_{k+1} := \mathbf{x}_k - \mathbf{G}_k \mathbf{f}_k, \tag{9}$$

where $\mathbf{G}_k$ is an approximation of $(D\mathbf{f}_k)^{-1}$ by $\sigma\mathbf{I}$ plus a low rank matrix. Multiplying the initial guess by σ permits separate under-relaxation of the simple fixed point iteration and the directions in which we have gained information by the secant conditions. We set $\mathbf{G}_0 = \sigma\mathbf{I}$ and update the approximation of $D\mathbf{f}_k^{-1}$ by the recursion

$$\mathbf{G}_{k+1} = \mathbf{G}_k + (\Delta\mathbf{x}_k - \mathbf{G}_k \Delta\mathbf{f}_k)\frac{\Delta\mathbf{f}_k^T}{\Delta\mathbf{f}_k^T \Delta\mathbf{f}_k}, \tag{10}$$

where $\Delta\mathbf{f}_k = \mathbf{f}_{k+1} - \mathbf{f}_k$ and $\Delta\mathbf{x}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$. This update rule satisfies both the secant condition $\mathbf{G}_{k+1}\Delta\mathbf{f}_k = \Delta\mathbf{x}_k$ and the no change condition

$$\mathbf{G}_k\mathbf{q} = \mathbf{G}_{k+1}\mathbf{q}, \ \forall\mathbf{q} \quad \text{such that} \quad \mathbf{q}^T\Delta\mathbf{f}_k = 0. \tag{11}$$

The history of the mixer is limited to m steps. In practice, once the limit is reached, we discard the oldest trial solutions and reindex the remaining trial solutions.

Instead of directly using Eq. (9) to calculate $\tilde{\mathbf{x}}_{k+1}$ we calculate the coefficients needed to construct it in the reduced space, $\text{span}(\mathbf{f}_0, \ldots, \mathbf{f}_{m-1}, \mathbf{x}_0, \ldots, \mathbf{x}_{m-1})$. We present our implementation in more detail in Appendix B.

C. Look ahead Broyden

To prevent the Broyden method from proceeding in a direction that degrades the solution look ahead strategies can be used. We have tried a look ahead strategy that does not update the current trial solution when $||\mathbf{f}||$ grows and instead bases the update on the trial solution from the previous iteration. To take the next step we update the approximation of $D\mathbf{f}^{-1}$ with the information gained from the regressive step and re-evaluate the trial solution. Equation (9) then becomes

$$\tilde{\mathbf{x}}_{k+1} := \mathbf{x}_{k-1} - \mathbf{G}_k \mathbf{f}_{k-1}, \tag{12}$$

and Eq. (5) becomes

$$\mathbf{x}_{k+1} = (1 - \beta_{\text{tmp}})\mathbf{x}_{k-1} + \beta_{\text{tmp}}\tilde{\mathbf{x}}_{k+1}. \tag{13}$$

We have combined this look ahead strategy with a temporary reduction of β for the problematic step. This strategy does not produce a clear benefit compared to the basic Broyden method.

D. Spin transformation

It is often necessary to couple spin channels when mixing electron densities for spin polarized calculations. One option is to mix the sum and the difference of the up and down spin channels separately and to recombine the result to obtain the next trial solution. This approach is also used by the spin enabled Pulay mixer available in GPAW.

For the Broyden methods we have implemented spin coupling slightly differently, by applying a spin transformation before mixing, and applying the inverse transform after mixing. In practice, we use the matrix

$$\mathbf{S} = \mathbf{S}^{-1} = \frac{1}{\sqrt{2}}\begin{bmatrix} \mathbf{I} & \mathbf{I} \\ \mathbf{I} & -\mathbf{I} \end{bmatrix}, \tag{14}$$

and the full mixing procedure becomes

$$\tilde{\mathbf{x}}_{n+1} = \mathbf{S}\,\text{mix}(\mathbf{S}\,\mathbf{x}_n), \tag{15}$$

when we consider $\mathbf{x} = \begin{bmatrix} \mathbf{x}_\uparrow \\ \mathbf{x}_\downarrow \end{bmatrix}$, where $\mathbf{x}_\uparrow$ and $\mathbf{x}_\downarrow$ are the up and down spin channels, respectively. This choice effectively forms the sum and the difference of the spin channels, but retains them in the same vector. This permits connections between the density sum and difference to be formed in the approximation of $D\mathbf{f}$. Mixing only one transformed vector instead of two vectors separately has the advantage that it allows us to handle spin polarized calculations with the same code that we use for unpolarized calculations.

The spin transformation is consistent in the sense that Eq. (15) reduces to $\tilde{\mathbf{x}}_{n+1} = \mathbf{x}_n$ if $\text{mix}(\cdot)$ is the identity function. Using spin transformations also permits us to precondition the mixer. In principle it could be advantageous to apply different weighing to the density sums than to the density differences. We are not aware of other works where spin coupling has been implemented this way.

III. RESULTS

We have compared the Broyden type 2 and the Pulay mixer on several models, including both easy and more demanding ones. Below we provide details of the tests.

Model 1. CH_4 molecule calculation with grid size $h = 0.09$.
Model 2. C_{60} molecule calculation with grid size $h = 0.18$.

TABLE I. Number of iterations required for convergence. Comparison between best case Pulay accelerator ($\mathbf{P}_B$), Pulay iteration with default parameter choice ($m = 3$, $\beta = 0.25$, $\mathbf{P}_D$), best case Broyden type 2 ($\mathbf{B}_B$) and Broyden type 2 method with default parameter choice ($m = 10$, $\beta = 0.8$, $\sigma = 0.25$, $\mathbf{B}_D$). Failure to convergence in 500 iterations is indicated by a dash.

Molecule	Spin	$\mathbf{P}_B$	$\mathbf{P}_D$	$\mathbf{B}_B$	$\mathbf{B}_D$
CH_4	No	20	39	19	27
C_{60}	No	17	19	16	20
Si_{dia}	No	20	23	21	26
C_6H_6	No	19	25	19	36
Al_{fcc}	Yes	153	–	30	44
Co_3	Yes	–	–	134	205

Model 3. Periodic Si_{dia} bulk in diamond lattice calculation with grid size $h = 0.09$.
Model 4. C_6H_6 molecule with calculation grid size $h = 0.18$.
Model 5. Spin polarized computation of Al_{fcc} surface with hydrogen adsorbate with grid size $h = 0.25$.
Model 6. Spin polarized computation of linear Co_3 with interatomic distance $d = 2.03453544$ and grid size $h = 0.15$.

The initial guess is provided by GPAW, and it is based on a sum of the independent atomistic densities. Three fixed point iteration steps are also taken by GPAW before engaging the mixer. The iteration is stopped when all convergence indicators satisfy the convergence criteria, 10^{-4} electron change in integral of absolute density, total energy change 10^{-3} eV per atom, and 10^{-9} integral of absolute eigenstate change. Exchange correlation is computed using local density approximation for systems without spin polarization and local spin density approximation for spin polarized system.[32]

Spin polarized computations for the Broyden mixer uses the spin transformation described in Sec. II D, while the reference method is calculated using the MixerSum method found in GPAW. The MixerSum method separately mixes the sum and the difference of the up and the down spin channels and recombines the results to produce the next trial density.

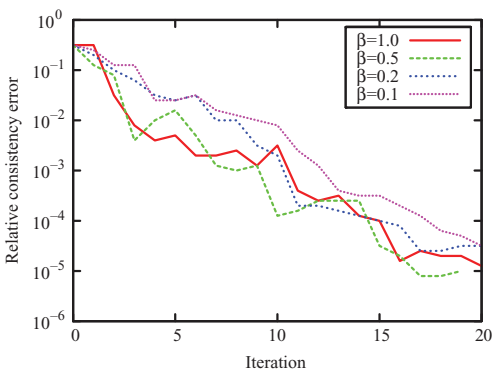

FIG. 1. Effect of varying beta for Pulay method calculation of CH_4 with $m = 5$.

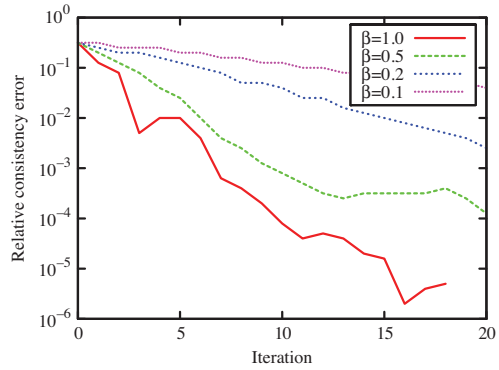

FIG. 2. Effect of varying beta for type 2 Broyden method calculation of CH_4 with $m = 5$, $\sigma = 0.9$.

For the Pulay method we have calculated the test cases for an array of mixer parameter values consisting of $(m, \beta) \in \{3, 5, 10\} \times \{0.05, 0.1, 0.25, 0.5, 0.8, 1.0\}$, where m is the history, and β is the fraction of the new estimate we use to update the previous trial vector. The default parameters in the GPAW code is $\beta = 0.25$, $m = 3$. We have calculated the test cases for the type 2 Broyden method over a the parameter array $(m, \beta, \sigma) \in \{3, 5, 10\} \times \{0.25, 0.5, 0.8, 1.0\} \times \{0.16, 0.25, 0.5, 0.8, 1.0\}$. Here σ is the multiplier of the initial guess for $D\mathbf{f}^{-1}$, and m and β are as above. The default parameters were fixed to $m = 10$, $\beta = 0.8$, and $\sigma = 0.25$, and compared to the results obtained by the Pulay method. These values were chosen to smooth convergence and improve performance in more demanding cases, at the cost of decreased performance in the easiest cases. We present the best case for both mixers, as well as the result for the default mixer parameters.

The results presented in Table I show that while the Pulay method is efficient for easy cases, convergence is difficult to achieve for the more demanding models. This behavior seems to stem from two sources. On the one hand the convergence curve of the Pulay method is very nonsmooth, even

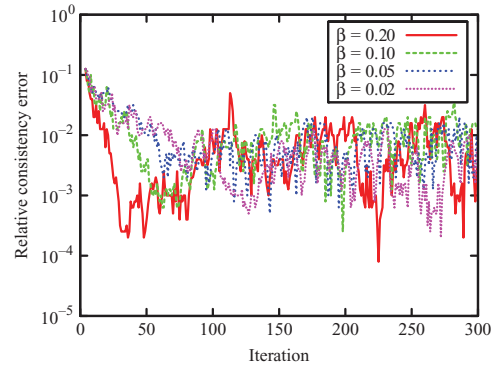

FIG. 3. Consistency error for Co_3 calculated with Pulay's method ($m = 5$). Although the correct spin polarizaton has been fixed consistency error is highly nonuniform.

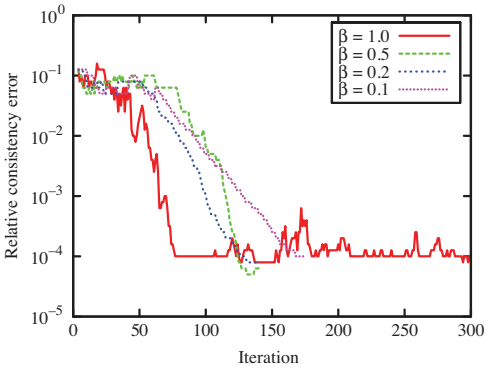

FIG. 4. Consistency error for Co_3 calculated with Broyden's second method ($m = 10, \sigma = \beta/5$). Convergence is significantly smoother than for Pulay's method even when significantly more aggressive under-relaxation is used (see Fig. 3). With $\beta = 1.0$ convergence is reached in 330 iterations.

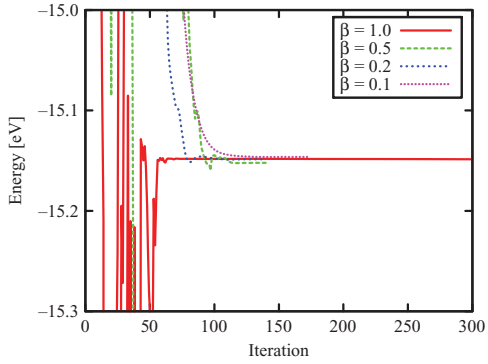

FIG. 6. Energy of Co_3 calculated with Broyden's second method ($m = 5$, $\sigma = \beta/5$). Energy evolution is significantly smoother than for Pulay's method even when significantly higher β is used (see Fig. 5). With $\beta = 1.0$ convergence is reached in 330 iterations.

for conservative mixer parameter choices. This is illustrated in Fig. 1, where the convergence curve of the Pulay method is shown. In particular, the curve does not become significantly smoother with more conservative β. On the other hand, the Pulay method typically requires more conservative under-relaxation, even for easier models. For convergence of the more demanding models extremely conservative mixer parameters is required and convergence rate suffers. In contrast the Broyden method generally has a smoother convergence curve and a more intuitive response to more conservative underrelaxation. This is illustrated in Fig. 2.

One of the demanding models presented in Table I is Al_{fcc}. The Pulay spin mixer requires 153 iterations to converge for the best case parameter choice, but does not converge with default parameter choice in 500 iterations. Convergence is only achieved when $n = 3$ and $\beta = 0.05$. This parameter choice for the Pulay method is not efficient for more general calculations. While it is possible that the Pulay method would converge for some part of the parameter space, that part of the space is often difficult to find.

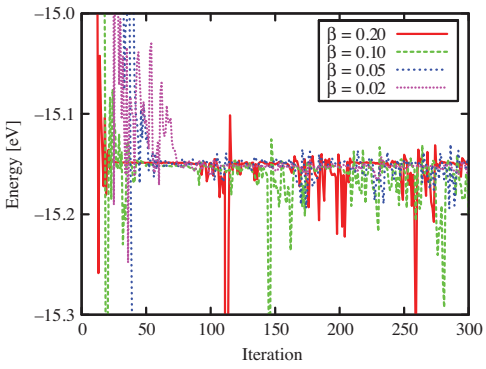

FIG. 5. Energy of Co_3 calculated with Pulay's method $m = 5$. The fluctuations are not smooth, and the system does not converge in 500 iterations.

Another even more challenging computation presented in Table I is Co_3. Convergence behavior for Co_3 using Pulay's method and Broyden's second method are shown in Figs. 3–5, and 6. In Fig. 3 the relative consistency error satisfies the convergence criteria at one step, but the eigenstate change does not satisfy the convergence criteria. Initial convergence is relatively fast, but the solution fails to stabilize. In Fig. 5, the energy of the system is presented. Both consistency error and energy fluctuations are very far from smooth.

In contrast, the convergence behavior of Broyden's second method, presented in Figs. 4 and 6, is significantly smoother. For Co_3 we obtained a HOMO–LUMO gap of 0.17 eV, and it has been shown that systems become more challenging as the gap size decreases.[33–35] Trilinear cobalt is an open shell transition metal, and charge sloshing is typically strong for these systems.[3] Transition metal system can also have several critical points close to the minimum.[7] Together, these properties serve to make the system a challenging benchmarking problem for SCF methods.

IV. CONCLUSIONS

The behavior of the type 2 Broyden method is not very sensitive to parameter choice. It converges reliably for a default set of parameters over a large class of problems, and the response on parameter changes is intuitive. The convergence curve of the type 2 Broyden method is relatively smooth, which increases confidence in convergence indicators. The smooth convergence behavior can also be used to estimate the number of iterations still required for convergence.

Characteristic of the behavior of the type 2 Broyden method is that the method first has to come close enough to the solution so that that the linear approximation is valid. For the more demanding cases the final number of iterations required is heavily dependent on the number of iterations required to reach the neighborhood of the solution. For this reason it is important to not use too conservative underrelaxation as that inhibits the initial search. Strong underrelaxation also slows convergence in the later stage. A slight

under-relaxation seems preferable to stabilize the initial convergence and smoothen the convergence curve. The smoother convergence also makes it easier to evaluate convergence rates for different parameters and increases confidence in obtained solution.

Pulay's method is typically very fast when it converges, and Kresse and Furthmüller found that Broyden's second method is outperformed by Pulay's method.[5] Our experience suggests that using σ around 0.25 together with a more aggressive β can improve performance of Broyden's method, and we are not aware if this was implemented by them. This indicates that the secant conditions generally provide accurate information of the SCF iteration, while a simple fixed point iteration will not perform well.

Convergence speed of Broyden's second method is not significantly worse than for Pulay's method for less challenging systems. However, Pulay's method sometimes fails to converge or performs very poorly for more challenging systems. For these systems Broyden's method can offer a more robust alternative. It remains possible that the Pulay method would converge well once the proper parameter set has been found. Searching for such a set can be prohibitively expensive.

In conclusion, we consider the type 2 Broyden method a competitive general purpose density mixer for DFT calculations. A fixed parameter set for the type 2 Broyden method gives good convergence speeds for the more demanding models and acceptable convergence for the easy models. In addition, similar treatment of both spin polarized and unpolarized systems can be achieved by the use of spin transformation. While problem optimized mixer parameters can produce improved convergence, the Broyden method has the advantage that for novel cases where the optimal parameter choice is not known a solution can still be obtained in one run.

ACKNOWLEDGMENTS

This research was supported by the Magnus Ehrnrooth Foundation, Grant No. MA2010n3, and by the Academy of Finland, Grant No. 128474.

APPENDIX A: SECANT METHODS

Secant condition based methods can be divided into groups based on two criteria. We base our division on whether the method attempts to approximate $D\mathbf{f}$ or its inverse. These are known as type 1 and type 2 methods, respectively. Another classification criterion is the group size of the simultaneous secant conditions. At one extreme are the type 1 and type 2 Broyden methods, where one secant condition is satisfied. At the other extreme we have the Andersson's methods, where all secant conditions are simultaneously satisfied. It is also possible to construct methods with intermediate or variable secant group sizes.

This presentation follows closely that given by Fang and Saad.[16] The type 1 Broyden method can be derived from the requirement that the update of approximate derivative $\mathbf{J}$ satisfy the secant condition

$$\mathbf{J}_{k+1}\Delta\mathbf{x}_k = \Delta\mathbf{f}_k, \tag{A1}$$

and the no change condition

$$\mathbf{J}_k\mathbf{q} = \mathbf{J}_{k+1}\mathbf{q},\ \forall\mathbf{q} \quad \text{such that} \quad \mathbf{q}^T\Delta\mathbf{f}_k = 0. \tag{A2}$$

The no change condition states that the update cannot increase information in directions orthogonal to the change. The no change condition is equivalent to minimization of the change in $\mathbf{J}$ (Ref. 36)

$$E(\mathbf{J}_{k+1}) = ||\mathbf{J}_{k+1} - \mathbf{J}_k||_F^2, \tag{A3}$$

subject to the secant condition. The update formula for $\mathbf{J}$ is then

$$\mathbf{J}_{k+1} = \mathbf{J}_k + (\Delta\mathbf{f}_k - \mathbf{J}_k\Delta\mathbf{x}_k)\frac{\Delta\mathbf{x}_k^T}{\Delta\mathbf{x}_k^T\Delta\mathbf{x}_k}. \tag{A4}$$

To obtain the update formula for $\mathbf{G}_{k+1}$, the Sherman–Morrison formula is applied to previous equation, giving

$$\mathbf{G}_{k+1} = \mathbf{G}_k + (\Delta\mathbf{x}_k - \mathbf{G}_k\Delta\mathbf{f}_k)\frac{\Delta\mathbf{x}_k^T\mathbf{G}_k}{\Delta\mathbf{x}_k^T\mathbf{G}_k\Delta\mathbf{f}_k}. \tag{A5}$$

Following a similar path the generalized type 2 Broyden method for secant groups of size l is[23]

$$\mathbf{G}_k = \mathbf{G}_{k-1} + (\mathbf{X}_k - \mathbf{G}_{k-1}\mathbf{F}_k)(\mathbf{F}_k^T\mathbf{F}_k)^{-1}\mathbf{F}_k^T, \tag{A6}$$

where $\mathbf{X} = [\mathbf{x}_{k-l}, \ldots, \mathbf{x}_{k-1}]$ and $\mathbf{F} = [\mathbf{f}_{k-l}, \ldots, \mathbf{f}_{k-1}]$.

In the special case $\mathbf{G}_{k-1} = -\sigma\mathbf{I}$ we get Anderson's method[22,23]

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \sigma\mathbf{f}_k - (\mathbf{X}_k + \sigma\mathbf{F}_k)\left(\mathbf{F}_k^T\mathbf{F}_k\right)^{-1}\mathbf{F}_k^T\mathbf{f}_k. \tag{A7}$$

APPENDIX B: COMPACT IMPLEMENTATION OF TYPE 2 BROYDEN METHOD

We have based our implementation of the type 2 Broyden method on the observation that $\mathbf{G}_k$ is only used to operate on vectors in the set $\{\mathbf{f}_0, \ldots, \mathbf{f}_{m-1}\}$ and that the range of $\mathbf{G}_k - \sigma\mathbf{I}$ is span$(\mathbf{f}_0, \ldots, \mathbf{f}_{m-1}, \mathbf{x}_0, \ldots, \mathbf{x}_{m-1})$. We can then represent $\mathbf{G}_k$ as a $2m \times m$ matrix for all $k < m$. A similar approach has been used by Kawata *et al.*[37]

To construct a compact representation of $\mathbf{G}_k$ we introduce the matrices $\mathbf{X} = [\mathbf{x}_0, \ldots, \mathbf{x}_{m-1}]$, $\mathbf{F} = [\mathbf{f}_0, \ldots, \mathbf{f}_{m-1}]$, and $\mathbf{Y} = [\mathbf{F}, \mathbf{X}]$. Here the columns of $\mathbf{Y} \in \mathbb{R}^{n\times 2m}$ spans the range of $\mathbf{G}_k - \sigma\mathbf{I}$ and the columns of $\mathbf{F} \in \mathbb{R}^{n\times m}$ are the vectors $\mathbf{G}_k$ will operate on during one step of the type 2 Broyden method calculation.

We will use $\hat{\mathbf{G}}_k \in \mathbb{R}^{2m\times m}$ to denote the compact approximate inverse derivative, and $\hat{\mathbf{f}}_k = \mathbf{e}_k \in \mathbb{R}^m$ and $\hat{\mathbf{x}}_k = \mathbf{e}_{m+k} \in \mathbb{R}^{2m}$. It then holds that $\mathbf{f}_k = \mathbf{F}\hat{\mathbf{f}}_k$ 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 $\Delta\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k+1} - \hat{\mathbf{x}}_k$.

If we require that $\mathbf{G}_0 = \mathbf{Y}\hat{\mathbf{G}}_0\hat{\mathbf{f}}$, we can transform Eq. (10) into the compact form

$$\hat{\mathbf{G}}_{k+1} = \hat{\mathbf{G}}_k + (\Delta\hat{\mathbf{x}}_k - \hat{\mathbf{G}}_k\Delta\hat{\mathbf{f}}_k)\frac{\Delta\hat{\mathbf{f}}_k^T\mathbf{S}}{\Delta\hat{\mathbf{f}}_k^T\mathbf{S}\Delta\hat{\mathbf{f}}_k}, \tag{B1}$$

where $\mathbf{S} = \mathbf{F}^T\mathbf{F}$. This update formula holds in particular for the initial guess $\mathbf{G}_0 = \sigma\mathbf{I}$ when we define $\hat{\mathbf{G}}_0 = [\frac{\sigma\mathbf{I}}{\mathbf{0}}]$.

While $\hat{\mathbf{G}}_{m-1}$ can be explicitly calculated, we have used a recursive implementation. In either case $\mathbf{S}$ should be stored between iterations and only updated when new $\mathbf{f}_k$ are available. In this case the pre- and postprocessing of one iteration step is $\mathcal{O}(nm)$, where n is the degrees of freedom, m the history length, and $n \gg m$ generally. The cost of one step of the compact Broyden's method is then $\mathcal{O}(m^3)$, and cost is dominated by the pre- and postprocessing steps. The cost of evaluation of $\mathbf{f}_k$ depends on the DFT code used.

The reported calculations have been made with a recursive implementation of the type 2 Broyden method. This procedure calculates each required $\hat{\mathbf{G}}_i\hat{\mathbf{f}}_j$ once provided that we use at most one step look ahead. That is, when initially called, we only attempt to calculate $\hat{\mathbf{G}}_k\mathbf{f}_k$ or $\hat{\mathbf{G}}_k\mathbf{f}_{k-1}$. We assume that the procedure has been initialized by updating $\mathbf{S}$ and will be finalized by clearing storage of all $\Delta\hat{\mathbf{x}}_{i-1} - \hat{\mathbf{G}}_{i-1}\Delta\hat{\mathbf{f}}_{i-1}$. To calculate $\hat{\mathbf{G}}_i\hat{\mathbf{f}}_j$ the recursion then proceed as follows:

1. if i is 0:
2. return $\hat{\mathbf{G}}_0\hat{\mathbf{f}}_j$
3. if $\Delta\hat{\mathbf{x}}_{i-1} - \hat{\mathbf{G}}_{i-1}\Delta\hat{\mathbf{f}}_{i-1}$ is not stored:
4. calculate $\hat{\mathbf{G}}_{i-1}\hat{\mathbf{f}}_{i-1}$
5. if i is $j+1$:
6. $\hat{\mathbf{G}}_{i-1}\hat{\mathbf{f}}_j := \hat{\mathbf{G}}_{i-1}\hat{\mathbf{f}}_{i-1}$
7. calculate $\hat{\mathbf{G}}_{i-1}\hat{\mathbf{f}}_i$
8. if i is j:
9. $\hat{\mathbf{G}}_{i-1}\hat{\mathbf{f}}_j := \hat{\mathbf{G}}_{i-1}\hat{\mathbf{f}}_i$
10. store $\Delta\hat{\mathbf{x}}_{i-1} - \hat{\mathbf{G}}_{i-1}\Delta\hat{\mathbf{f}}_{i-1}$
11. if $\hat{\mathbf{G}}_{i-1}\hat{\mathbf{f}}_j$ is not known from line 6 or 9:
12. calculate $\hat{\mathbf{G}}_{i-1}\hat{\mathbf{f}}_j$
13. return $\hat{\mathbf{G}}_{i-1}\hat{\mathbf{f}}_j + (\Delta\hat{\mathbf{x}}_{i-1} - \hat{\mathbf{G}}_{i-1}\Delta\hat{\mathbf{f}}_{i-1})\frac{\Delta\hat{\mathbf{f}}_{i-1}^T\mathbf{S}\hat{\mathbf{f}}_j}{\Delta\hat{\mathbf{f}}_{i-1}^T\mathbf{S}\Delta\hat{\mathbf{f}}_{i-1}}$

[1] W. Kohn and L. J. Sham, Phys. Rev. **140**, A1133 (1965).
[2] P. Hohenberg and W. Kohn, Phys. Rev. **136**, B864 (1964).
[3] G. Kresse and J. Furthmüller, Phys. Rev. B **54**, 11169 (1996).
[4] Y. Saad, J. R. Chelikowsky, and S. M. Shontz, SIAM Rev. **52**, 3 (2010).
[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] T. van Voorhis and M. Head-Gordon, Mol. Phys. **100**, 1713 (2000).
[8] E. Cancès, J. Chem. Phys. **114**, 10616 (2001).
[9] A. A. Mostofi, P. D. Haynes, C.-K. Skylaris, and M. C. Payne, J. Chem. Phys. **119**, 8842 (2003).
[10] C. Freysoldt, S. Boeck, and J. Neugebauer, Phys. Rev. B **79**, 241103 (2009).
[11] A. Edelman, T. A. Arias, and S. T. Smith, SIAM J. Matrix Anal. Appl. **20**, 303 (1998).
[12] J. VandeVondele and Jürg Hutter, J. Chem. Phys. **118**, 4365 (2003).
[13] V. Weber, J. VandeVondele, J. Hutter, and A. M. N. Niklasson, J. Chem. Phys. **128**, 084113 (2008).
[14] P. Pulay, Chem. Phys. Lett. **73**, 393 (1980).
[15] D. R. Bowler and M. J. Gillian, Chem. Phys. Lett. **325**, 473 (2000).
[16] H.-r. Fang and Y. Saad, Numer. Linear Algebra Appl. **16**, 197 (2009).
[17] E. Cancès and K. Pernal, J. Chem. Phys. **128**, 134108 (2008).
[18] L. Qi, Oper. Res. Lett. **20**, 223 (1997).
[19] C. G. Broyden, Math. Comput. **19**, 577 (1965).
[20] J. M. Martínez, J. Comp. Appl. Math. **124**, 97 (2000).
[21] G. P. Srivastava, J. Phys. A: Math. Gen. **17**, L317 (1984).
[22] D. G. Anderson, J. ACM **12**, 547 (1965).
[23] V. Eyert, J. Comput. Phys. **124**, 271 (1996).
[24] J. J. Mortensen, L. B. Hansen, and K. W. Jacobsen, Phys. Rev. B **71**, 035109 (2005).
[25] J. Enkovaara, C. Rostgaard, J. J. Mortensen, J. Chen, M. Dulak, L. Ferrighi, J. Gavnholt, C. Glinsvad, V. Haikola, H. A. Hansen, H. H. Kristoffersen, M. Kuisma, A. H. Larsen, L. Lehtovaara, M. Ljungberg, O. Lopez-Acevedo, P. G. Moses, J. Ojanen, T. Olsen, V. Petzold, N. A. Romero, J. Stausholm, M. Strange, G. A. Tritsaris, M. Vanin, M. Walter, B. Hammer, H. Häkkinen, G. K. H. Madsen, R. M. Nieminen, J. K. Nørskov, M. Puska, T. T. Rantala, J. Schiøtz, K. S. Thygesen, and K. W. Jacobsen, J. Phys.: Condens. Matter **22**, 253202 (2010).
[26] The latest version of GPAW can be publicly accessed and anonymously downloaded from http://wiki.fysik.dtu.dk/gpaw
[27] S. R. Bahn and K. W. Jacobsen, Comput. Sci. Eng. **4**, 56 (2002).
[28] V. R. Saunders and I. H. Hillier, Int. J. Quantum Chem. **7**, 699 (1973).
[29] L. D. Marks and D. R. Luke, Phys. Rev. B **78**, 075114 (2008).
[30] U. M. Yang, "A family of preconditioned iterative solvers for sparse linear systems," Ph.D. thesis (University of Illinois at Urbana-Champaign, 1995).
[31] T. Eirola and O. Nevanlinna, Numer. Linear algebra Appl. **121**, 511 (1989).
[32] J. P. Perdew and Y. Wang, Phys. Rev. B **45**, 13244 (1992).
[33] F. Tassone, F. Mauri, and R. Car, Phys. Rev. B **50**, 10561 (1994).
[34] J. F. Annett, Comput. Mater. Sci. **4**, 23 (1995).
[35] E. H. Rubensson, E. Rudberg, and P. Sałek, J. Math. Phys. **49**, 032103 (2008).
[36] J. E. Dennis, Jr. and J. J. Moré, SIAM Rev. **19**, 46 (1977).
[37] M. Kawata, C. M. Cortis, and R. A. Friesner, J. Chem. Phys. **108**, 4426 (1998).