JCTC Journal of Chemical Theory and Computation

Article

pubs.acs.org/JCTC

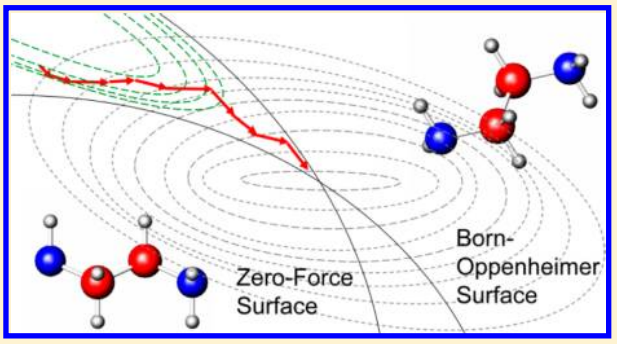
Fixed-Point Optimization of Atoms and Density in DFT

L. D. Marks*

Department of Materials Science and Engineering, Northwestern University, Evanston, Illinois 60201, United States

Supporting Information

ABSTRACT: I describe an algorithm for simultaneous fixed-point optimization (mixing) of the density and atomic positions in Density Functional Theory calculations which is approximately twice as fast as conventional methods, is robust, and requires minimal to no user intervention or input. The underlying numerical algorithm differs from ones previously proposed in a number of aspects and is an autoadaptive hybrid of standard Broyden methods. To understand how the algorithm works in terms of the underlying quantum mechanics, the concept of algorithmic greed for different Broyden methods is introduced, leading to the conclusion that if a linear model holds that the first Broyden method is optimal, the second if a linear model is a poor approximation. How this relates to the algorithm is discussed in terms of electronic phase transitions during a self-consistent run which results in discontinuous changes in the Jacobian. This leads to the need for a nongreedy algorithm when the charge density crosses phase boundaries, as well as a greedy algorithm within a given phase. An ansatz for selecting the algorithm structure is introduced based upon requiring the extrapolated component of the curvature condition to have projected positive eigenvalues. The general convergence of the fixed-point methods is briefly discussed in terms of the dielectric response and elastic waves using known results for quasi-Newton methods. The analysis indicates that both should show sublinear dependence with system size, depending more upon the number of different chemical environments than upon the number of atoms, consistent with the performance of the algorithm and prior literature. This is followed by details of algorithm ranging from preconditioning to trust region control. A number of results are shown, finishing up with a discussion of some of the many open questions.



1. INTRODUCTION

At the present time, a relatively large fraction of the world's supercomputer time is used for density-functional (DFT) calculations. While these are often static calculations with fixed atomic positions, the more demanding ones involve relaxing the atomic positions. This is typically done with a double-loop approach: in the inner loop, the density is iterated toward self-consistency; in the outer, the atoms are moved. Numerous algorithms exist for both loops. One interesting feature of the inner-loop is that it is relatively fast; problems with 10^3 to 10^6 or more variables for the density converge in 20−100 iterations. (If they did not then DFT would not be a useful technique.) Somewhat different is the Car−Parrinello method[1] where the wave functions and atoms can be simultaneously varied.

In principle, it is inefficient to have two loops; much more efficient would be to simultaneously vary the density and atomic positions. The double loops are avoided in the Car−Parrinello method (e.g.,[1−5]), but this is a kinetic model with fictitious masses and a solution of Newton's laws of motion involving wave functions. Somewhat similar methods exist but also with wave functions where either the total energy as a function of wave function occupancy or form is treated variationaly (e.g., refs 3,6−16) or by a least-squares (L2) method (e.g., ref 17). However, there are technical disadvantages in terms of speed and storage with wave functions compared to a density only approach. An alternative approach involving simultaneous changes of the density and atoms was suggested by Bendt and Zunger in 1983,[18] which various groups tried but failed to convert into a successful numerical algorithm—it only converged for very simple systems.[19]

The purpose of this article is to describe a viable fixed-point optimization approach fusing the loops that is based upon a derivative free quasi-Newton (QN) approach. The method works not just for simple demonstration problems but for complicated ones, is robust and rather faster than a double loop, shows linear or sublinear scaling in general with the number of electrons and atoms, and has already been used in a number of papers.[20−31] The algorithm does not directly involve the energy of the system, the true forces, or fictitious masses, so while it has some similarities to Car−Parrinello and molecular methods it uses a vector that does not necessarily have any mathematically exact connection to the forces. The algorithm is an optimizer because from the Kohn−Sham equations the fixed-point solution is the variational minimum.

The structure of this paper is to start with a brief outline of the QN algorithm, and then a brief analysis in terms of algorithmic greed. While this is algorithm mathematics, by itself it does not explain what is important, and for this a deeper analysis is needed. It is pointed out that during the fixed-point

Received: August 22, 2012

   dx.doi.org/10.1021/ct4001685 | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

optimization and even for a standard fixed-point solution for the electron densities, the system traverses different electronic phases which have different dielectric responses and hence Jacobians. Therefore the Jacobian can change discontinuously, which influences how greedy the algorithm can be. This leads to a need to mitigate the greed of Broyden's first method and explains the need for a less greedy adaptive algorithm as well as regularization and trust-region control. These then lead to a method for choosing a fixed-point Broyden family member based upon the ansatz that the extrapolated eigenvalues are real and position since this is a necessary condition at the variational minimum.

I next provide a brief analysis of the eigenspectrum of the Jacobian, indicating how this connects to the convergence, primarily via the elastic wave spectrum and the eigenvalues of the dielectric matrix. This explains why the algorithm scales linearly or sublinearly with system size, and as a rule the number of self-consistent cycles to convergence only scales weakly with the number of atoms. Some details are then provided about the specifics of preconditioning, scaling, regularization, and trust-region control, which follow from numerical, algorithmic, and physical considerations. After a brief presentation of some typical results, the paper concludes with a discussion of the numerous open issues, as well as some suggestions for future developments.

For brevity, this article does not attempt to compare the new algorithm to all other QN methods beyond a standard benchmark optimizer, analyze the more formal mathematics of algorithm convergence and related issues, analyze the convergence of the new approach with different flavors of DFT, or compare it to different strategies such as the Car−Parrinello method, which can solve the same problem by a different route. All these are important topics but are left to future work, and some are mentioned at the end.

2. ALGORITHM DESCRIPTION

Consider some density $\rho(r,R)$ as a function of position r and a vector of N_{av} variable atomic positions $R = (R_1, R_2, ...R_M)$ for N_a inequivalent atoms. For this density, the solutions of the equations of Kohn−Sham density-functional theory[32,33] can be written as

$$(\hat{H}_0 + V_p)\phi_i = \varepsilon_i\phi_i \tag{1}$$

$$F(\rho(r, R)) = \sum_i (1 + \exp((\varepsilon_i - \mu)/kT))^{-1}|\phi_i(r)|^2 \tag{2}$$

with eigenvectors (orbitals) ϕ_i and eigensolutions (energies) ε_i where $\hat{H}_0$ is the noninteracting single-particle Hamiltonian, V_p the effective local potential for $\rho(r,R)$, μ the chemical potential, k Boltzmann's constant, and T the temperature. The term $F(\rho(r,R))$ will be referred to throughout as the SCF mapping. In conventional methods, we seek a fixed-point of $F(\rho(r,R))$ as the solution for a given density and atomic positions, i.e.

$$F(\rho(r, R)) - \rho(r, R) = D(\rho(r, R)) = 0 \tag{3}$$

where $D(\rho(r,R))$ is the density residual. This form is specific to pure DFT calculations where the only active variables are the density; the forms for orbital-based density methods are different and would involve either a density matrix or the wave functions as the active variables. In addition to the density condition, for a complete solution we seek the minimum of the total energy of the system, i.e., for a total energy $E(\rho(r,R))$ (including electronic entropy[33]), the derivative as a function of the atomic positions:

$$\partial E(\rho(r, R))/\partial R_i = g_i(\rho(r, R)) = 0 \tag{4}$$

The standard method is to solve eq 3 for some fixed R, change the atomic positions using a minimization algorithm to move toward the solution of eq 4, reconverge the density fixed-point subproblem, and iterate. Rather than solving these serially, they can be merged; i.e., we seek the fixed-point solution of

$$(D(\rho(r, R)), -g(\rho(r, R)) = -G(\rho(r, R)) = 0 \tag{5}$$

with $G(\rho(r,R))$ the generalized residual vector. For a DFT code where the basis set is not atom position dependent, the gradients are the negative of the Hellmann−Feynman forces. For a LAPW-based method, additional corrections for the basis set are needed, commonly called Pulay corrections.[34−36] These are calculated with the Kohn−Sham density of $F(\rho(r,R))$, whereas the Hellmann−Feynman calculations use $\rho(r,R)$. Hence the gradients $g(\rho(r,R))$ in eqs 4 and 5 are not the Born−Oppenheimer surface gradients; they are instead vectors which share a common fixed-point. (The Born−Oppenheimer surface is defined as the energy as a function of R for $D(\rho(r,R)) = 0$.) In some respects; it would be better to refer to these as "pseudo-gradients" to emphasize this.

The residual $G(\rho(r,R))$ is not a true gradient, although it will appear in some of the equations where one would appear in an optimization problem. There is no reason a priori why $-G(\rho(r,R))$ should be a decent direction for the energy or the density residual. The most that can be said is that when it is zero the total energy as a function of both the density and atomic positions is a variational local minimum.

Dropping the (r,R) notation for brevity, and using ρ_n and R_n to describe the density and positions, respectively, for iteration 'n' as well as G_n, this suggests a standard Newton algorithm:

$$(\rho_{n+1}, R_{n+1}) = (\rho_n, R_n) - H_nG_n \tag{6}$$

where H_n is the inverse of the Jacobian for the change in density/pseudogradients with density/atomic positions. (The use of "H" here is conventional, unfortunately confusable with a Hamiltonian.) This is a "linear model," as higher-order terms in the Taylor series are neglected. The computational cost of calculating H_n is prohibitive, so instead it is approximated using a QN method. Introducing the new variables:

$$y_{j,n} = G_n - G_j \tag{7}$$

$$s_{j,n} = (\rho_n, R_n) - (\rho_j, R_j) \tag{8}$$

and the matrices $S_n = [s_{n-k,n}, s_{n-k+1,n}, ...s_{n-1,n}]$ and $Y_n = [y_{n-k,n}y_{n-k+1,n}...y_{n-1,n}]$, we require that H_n or its inverse, the Jacobian B_n, satisfy the multisecant equations:

$$H_nY_n = S_n \text{ or } B_nS_n = Y_n \tag{9}$$

for which there is a general rank-one solution

$$H_n = \sigma_nI + (S_n - \sigma_nY_n)(Y_n^TW)^{-1}W^T \tag{10}$$

with W being any vector of size $N_e + N_{av}$. Here, σ_n is what will be called the *algorithm greed*. As discussed in earlier work,[37,38] this determines how large a step the algorithm will take in directions about which there is no earlier information. While in principle both S_n and Y_n can include all previous densities, in practice only a small number, between 6 and 10, is needed, with 8 being the current default; more discussion is given later.

dx.doi.org/10.1021/ct4001685 | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

Different forms for W and slightly different methods of constructing the mathematics lead to variations on the above; all the common methods used for DFT can be described in this fashion as discussed previously.[37,38] The two most common are to take $W = S_n$, a multisecant form of Broyden's first method (MSB1), often called "good Broyden," whereas taking $W = Y_n$ is a multisecant form of his second method (MSB2), sometimes called "bad Broyden." In the original paper by Broyden,[39] where nonmultisecant methods were introduced, his first method (B1) worked; his second (B2) did not. Since a number of papers soon afterward reached the same conclusion, B2 was largely dismissed, although more recent work has questioned whether it really fails in all cases. Since the most recent information overwrites earlier information, a sequential method is a greedy algorithm (see later for a discussion of algorithm greed); a multisecant approach treats all previous steps on an equal basis. All DFT methods in common use involve some variant of the MSB2 method for reasons which will become clearer later.

For the problem of interest here, choosing $W = Y_n$ does not completely fail but often stagnates without fully converging the atomic positions. This appears to be similar to what was found with previous attempts to implement the Bendt and Zunger approach.[18,19] While there is a large amount of mathematical literature on the equivalent rank-two updates for QN optimization and how they behave for different types of problems, less has been published for nonlinear equation solvers, the main focus being on choosing a method that gives the best conditioning.[40,41] An interesting hypothesis by Martinez[42] is that which method is optimal depends upon the error in the secant equation for B1 or B2.

The discussion above suggests a linear combination of MSB1 and MSB2, i.e.

$$W = Y_n + \alpha S_n \tag{11}$$

This algorithm, which will be referred to as "Multisecant Rank One" (MSR1), tends to MSB1 if the absolute values of the determinants $\alpha|\det(S_n^T Y_n)| > |\det(Y_n^T Y_n)|$ and to MSB2 if the inequality is reversed. This algorithm is one member of the "fixed-point Broyden family" and should share the convergence properties of similar algorithms (e.g., refs 40, 41, 43−54).

To complete the algorithm, the critical question is what should α be? For this, some analysis is needed, and as a prelude I will first show (section 2.1) that $0 < \alpha < \infty$, and for a linear model with exact arithmetic, $\alpha \to \infty$ is optimal. Then I will show that due to electronic phase transitions this is too greedy (section 2.2), followed by an ansatz for choosing α based upon positive projected eigenvectors for the extrapolated component of the inverse Jacobian (section 2.3). This exploits the fact that at the solution the Jacobian has only positive eigenvectors since it originates from a variational minimum. The next section finishes with a brief analysis of the convergence before moving on to some specific algorithm details in section 3.

2.1. Analysis I: Algorithm Greed and Accuracy of the Linear Model. All the variants discussed here and elsewhere in the DFT literature are linear approximations, solutions to a quadratic local optimization problem.[37,38] MSB1 is the minimum-norm solution for

$$B_n S_n = Y_n \tag{12}$$

while MSB2 is the minimum-norm for the inverse equation

$$H_n Y_n = S_n \tag{13}$$

MSB1 yields the smallest B and the best new density based upon the previous history; MSB2 yields the smallest H and the smallest residual. In terms of algorithms, MSB1 is a greedy algorithm; MSB2 is not. A definition of "greedy algorithms" is appropriate:[55]

> *A greedy algorithm always makes the choice that looks best at the moment. That is, it makes a locally optimal choice in the hope that this choice will lead to a globally optimal solution.*

MSB2 only uses the information stored in Y_n and makes no assumptions beyond this. In contrast, MSB1 projects the next direction onto the prior history of steps in S_n. Hence it includes components of the residual normal to the prior history of Y_n.

Greedy algorithms work well if the problem has what is called "optimal substructure" (e.g., ref 55); i.e., the problem can be broken down into a combination of optimal solutions to a number of subproblems. A completely linear problem has this property; it can be decomposed into subproblems corresponding to solving along the set of vectors specified by the eigenvectors of B (or H) or alternatively a Krylov subspace using the matrix B (or H).

From this, it follows that algorithm greed is good or bad depending upon the linearity of the problem. Note that this connects to the hypothesis of Martinez[42] mentioned earlier, albeit in a different way.

Next, consider the effects of nonlinearities. For MSB1, the largest relative errors will be in the smallest absolute value eigenvalues of B. Hence the worse determined eigenvalues of B^{-1}, which is used to determine the next step, are the largest ones. From standard error analysis if the determinant $(\det(S_n{}^T Y_n))$ is small, the errors will be large. (For completeness, while rank-deficient behavior of $S_n{}^T Y_n$ is formally an issue, it can be avoided by using a regularization technique, as described later.) In contrast, with MSB2 the largest relative errors will be in the smallest absolute value eigenvalues of H.

Hence if the problem is well described by the linear model, we want to use MSB1 ($\alpha \to \infty$), whereas if it is not, we want to tend toward MSB2 ($\alpha = 0$). What yields nonlinearities in a DFT problem?

2.2. Analysis II: Electronic Phase Transitions. Analysis of the performance of the two main variants of Broyden's methods requires care. The literature is very large when one includes areas beyond DFT. There is general consensus on the mathematical details, although there are some disagreements and rediscoveries. Both approaches have been shown to be superlinearly convergent for a linear approximation and locally superlinearly convergent (see refs 37, 38, 42, 56, 57, and references therein).

As already mentioned, all other methods found in the DFT literature can be expressed as slightly different minimum-norm solutions,[37,38] as can the Direct Inversion of the Iterative Subspace (DIIS) approach.[58] Assuming that there is no intentional or unintentional regularization, with exact arithmetic they are all essentially the same and for a linear problem should be equivalent. However, any moderately conscientious reading of the literature reveals that they are not in practice. When the two were specifically compared[37,38] with the Wien2k APW+lo code,[59] MSB2 was more stable than MSB1.

This leads to the conclusion that the mathematics of a linear model by itself is not enough, and one has to consider the properties (quantum mechanics) of the underlying problem.

A key result is that the Hamiltonian of the underlying problem can change discontinuously due to electronic phase

dx.doi.org/10.1021/ct4001685 | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

transitions. One way to monitor this is to analyze how the residual changes in the next step as a function of a simple Pratt step, i.e. $(\rho_1,R_0) = (1-\sigma)(\rho_0R_0) + \sigma F(\rho_0,R_0)$ as a function of σ. Results are shown for a number of representative small problems in Figure 1, all using the conventional PBE

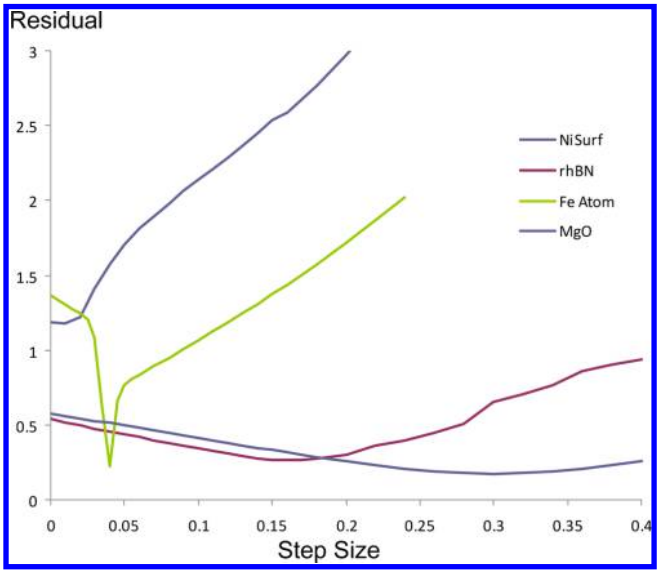


Figure 1. Plot of the residual in the second step as a function of the size of a simple Pratt step in the first step, for a number of different problems as described in the text. The jumps in the derivatives are real and are attributed to changes in the electronic phase.

functional[60] with room-temperature for the electronic temperature and starting from relativistic atomic densities.[61] The cases are (1) an isolated Fe atom in a cubic cell of size 28 atomic units (au) run without spin-polarization starting a configuration of Ar $3d^{6.5}4s^{1.5}$, (2) bulk MgO with the published X-ray lattice parameters, (3) a 7-layer (total) Ni (111) 1 × 1 surface, run spin polarized in a 4.757008 × 4.757008 × 34.956725 au hexagonal cell, and a 11 × 11 × 1 k-mesh, and (4) a simplified version of a BN monolayer on Rh,[63,64] run without spin polarization, started from atomic densities, a 7-layer (111) Rh bulk with a monolayer of BN on either side in a 5.11238 × 5.11238 × 45.04544 au hexagonal cell, and a (deliberately small) 4 × 4 × 1 k-mesh.

The clearest example is the Fe atom where there are three different electronic phases as a function of the step size as shown in Figure 2: (a) for small steps, an electronic configuration Ar $3d^84s^0$, (b) for larger steps, an electronic configuration of Ar $3d^64s^2$, and (c) for intermediate steps, a mixed phase (coexistence of a and b) with an electronic configuration Ar $3d^{8-x}4s^x$.

In the other cases where there are discontinuities in the gradient, there are also electronic phase transitions of different types. The difference in the gradient on either side of the phase transitions is because the dielectric properties are different; as will be discussed later, the exact Jacobian can be represented in terms of the dielectric matrix. These phases have energies much higher than the ground state, so they are different from what one observes in real physical systems but are "real" as far as the DFT algorithm is concerned. Increasing the electronic temperature (see ref 65 and references therein) broadens the region over which the mixed phase, such as b in Figure 2,

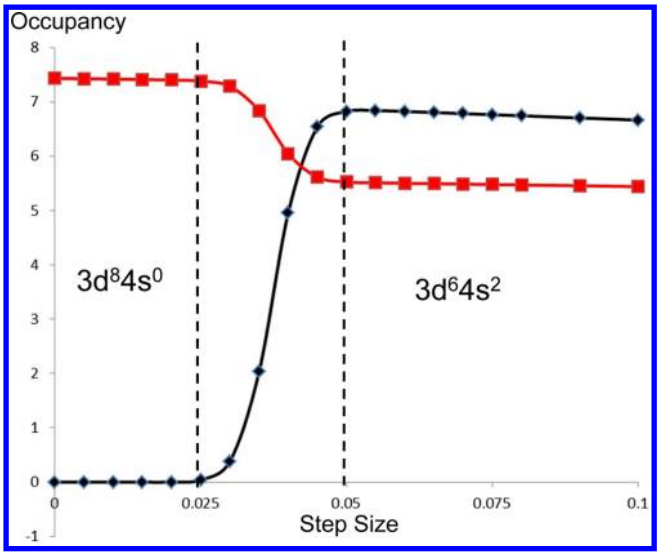


Figure 2. Electronic configuration of $F(\rho)$ in the second step as a function of the size of the first Pratt step for an Fe atom, with the 4s occupancy within the muffin-tins in black (×10) and the 3d in red. There are two pure phases as marked, as well as a mixed phase regime. Not all the density is within the muffin tins, so the occupancies are not the integer values of $3d^84s^0$ or $3d^64s^2$.

occurs, explaining the common heuristic of increasing the electronic temperature to improve convergence.

At the other extreme is bulk MgO, a very well-behaved problem without significant phase transitions.

The other two examples show features intermediate between these extremes. Interesting is the Rh/BN case where there are a number of small jumps and changes in the gradient. These are not artifacts; they are small phase transitions which do not change the behavior as dramatically as for the Fe atom. The nickel surface also has these.

One might hypothesize that one should therefore only take small steps along the unpredicted direction, but this is not correct. For the isolated Fe atom case, the gradient for small steps is small, so this will lead to large predicted QN steps—both B and H are simplex gradients (e.g., ref 66). As shown in Figure 1, it is better to use an "optimum" step, and going slightly larger than the optimum is safer than going too small. This rationalizes the need to control the algorithm greed σ using a trust-region strategy. Note that the concept of reducing the algorithm greed if the convergence is poor as often found in the unwritten DFT literature can be inappropriate.

2.3. Ansatz for α. From the previous two sections, $\alpha \to \infty$ is optimal for a linear problem with exact arithmetic; $\alpha = 0$ is the safe choice when electronic phase transitions exist. The simplest choice (used in an earlier version of MSR1) is $\alpha = 1$. While this can work, often it oscillates about the solution or spirals around it. To avoid this, the algorithm uses an ansatz for the Hessian borrowed from the optimization literature, as well as concepts from fixed-point stability theory. Forcing the Hessian to be positive definite in an optimization ensures that the steps are descent directions. From stability theory, oscillatory or spiral behavior is associated with negative or imaginary values, respectively, for the eigenvalues of the Jacobian.

We know that the fixed-point solution is a variational minimum so at the solution has real, positive eigenvalues. The component of the inverse Jacobian with respect to Y_n is fixed by

dx.doi.org/10.1021/ct4001685 | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

the curvature condition (eq 9), but there is an extrapolated component orthogonal to Y_n corresponding to the matrix

$$T_n^T = S_n^T - (S_n^T Y_n)(Y_n^T Y_n)^{-1} Y_n^T \tag{14}$$

This is the extrapolated component of the eigenvector of the part of the inverse Jacobian $S_n(Y_n^T W)^{-1} W^T$ that originates from the prior history. We choose the eigenvalues of this projected component to be purely real and positive, i.e. find the largest α such that all ξ scalars that satisfy

$$S_n(Y_n^T W)^{-1} W^T T_n = \xi S_n \tag{15}$$

are real and positive. Since the matrix $W^T T_n$ has real, positive eigenvalues, this reduces to choosing α such that $Y_n^T W$ has real, positive eigenvalues. This can occur for a range of values, the optimum choice is the largest possible value, i.e., the most greedy one. This is implemented using a bracketing search with an upper limit of $\alpha = 3$.

The extent to which this ansatz is appropriate for general fixed-point problems which are not variational minima is not obvious, a topic for further work, as is the question of whether this ansatz is the best choice in all possible cases. For the problems of interest here, it removes the oscillatory and spiral behavior and also improves the stability.

2.4. Analysis III: Convergence. As mentioned above, superlinear convergence (i.e., the exponential decay rate increases near the solution) of B1 and B2 has already been proved for a linear model (see refs 37, 38, 42, 56, and 57 and references therein), so we can extrapolate to superlinear convergence for MSB1, MSB2, and MSR1. However, the convergence proofs in the literature do not fully define the properties. One useful connection recently investigated in some detail[67] is that between MSB2 and DIIS within a linear model and the older Generalized Minimal Residual Method GMRES (e.g., ref 56) technique for solving linear equations. It is also worth noting that one can also connect to the conjugant-gradient (CG) method, as well as the concept of Krylov spaces. In the related area of optimization, it has been shown that with an exact line-search all Broyden-type methods become equivalent to CG (e.g., ref 57).

It follows that in a well-behaved case the convergence of the algorithms will approach those of GMRES and CG. I will state here the general convergence results without proof; the interested reader is referred to refs 56, 57, 68, and 69 and references therein for specific analyses: (a) In the linear case, the algorithm will converge in at most k-iterations, where k is the number of eigenvalues of the Jacobian. (b) The convergence of the algorithm depends upon the number of clusters of eigenvalues as well as the width of the clusters. In the limit of an infinitely small width of the clusters, it will converge in k iterations where k is the number of clusters. In the more general case, it will converge more slowly as the width of the clusters increases, but in most cases the number of clusters is more important than the width (with standard generalization caveats). From this, we may deduce that the eigenspectrum of the Jacobian controls the convergence rate.

Consider first the atomic component of the Jacobian. The eigenspectrum is that of elastic waves (phonons/vibrations with the same mass for all atoms), and the clusters can be interpreted as the bands of the elastic eigenvalues. For a single unit cell, only the Γ point of the first Brillouin zone is relevant. For a larger structure considered as a superstructure of a single unit cell, rather than the complexity of the eigenspectrum scaling with the number of unit cells forming the superstructure, the clusters (bands) will be folded back. The number of clusters of eigenvalues will depend upon the number of chemically distinct environments, while the width will depend upon the number of atoms. Hence the convergence rate depends at most linearly upon the size of a problem if the number of chemically different environments does not change and can have a sublinear dependence. That the convergence is not simply a function of the number of atoms and does not scale to some high power of this number can be seen by inspecting, for instance, the prior results of Baker[11] or Moss and Li[17] and plotting the number of iterations to convergence versus the number of atoms/variables as well as the results shown later.

Next, consider the electronic component. Using conventional dielectric theory for the effect of an external charge ρ_{ext} and the charge it induces ρ_{ind}, we can write

$$s_{j,n} = (\rho_n, R_n) - (\rho_j, R_j) \equiv \rho_{ext} \tag{16}$$

$$y_{j,n} = G_n - G_j \equiv \rho_{ind} - \rho_{ext} \tag{17}$$

So the exact Jacobians be written as

$$B_{q,q'} = I - 1/\varepsilon_{q,q'}; \quad H_{q,q'} = \varepsilon_{q,q'}/(I - \varepsilon_{q,q'}) \tag{18}$$

Where $\varepsilon(q,q')$ is the dielectric response in reciprocal space with q, q' reciprocal lattice vectors. Note that this is equivalent to the use of a form

$$\varepsilon_{q,q'} = \frac{\int \varepsilon_{q,q'}(k)\rho(k, r, R)\, \mathrm{d}k}{\int \rho(k, r, R)\, \mathrm{d}k} \tag{19}$$

where the integral over "k" is for the first Brillouin zone, and $\rho(k,r,R)$ is the SCF mapping for each k-point in the zone, taken here for simplicity at the fixed-point solution. Assuming a linear mapping from reciprocal space to the specific basis-set of the DFT calculation using just the density, which is true for the Wien2k code and many others (but perhaps not all), the spectrum of the eigenvalues of B (or its inverse H) directly maps to that of the basis-set used, and those of B are those of the negative of the inverse dielectric response shifted by +1.

The eigenvalues are related to the dielectric band structure.[70−76] For a few insulators, this has been calculated, and similar to the electronic band structure, there are only a limited number of bands, as would be expected since simple density-functional perturbation theory can be used (see ref 77 and references therein). For metals, it is more complicated since there are degenerate states, so the number of eigenvalue clusters will be larger. However, in neither case will the number of clusters be close to the number of variables in the basis set, and for the most deeply bound states the dielectric band structure will be very flat; i.e., the cluster will be narrow.

A number of inferences therefore follow which are consistent with existing empirical data from DFT calculations discussed in existing code documentation: (1) Metals will in general converge slower than insulators, because there are more clusters and/or the width of the clusters is larger. (2) Using what is called "Temperature Smearing," i.e., a large temperature in the Fermi−Dirac occupancy or Gaussian smearing (see ref 65 and references therein), will improve convergence since it will mix orbitals which would otherwise be part of different clusters, thereby reducing the number of clusters. This is related to the broadening of the mixed electronic phase described

dx.doi.org/10.1021/ct4001685 | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

Table 1. Summary Data for the Models Shown with the Total Numbers of Atoms, Inequivalent Atoms N_a, Atomic Position Variables N_{av}, the Matrix Size of the Hamiltonian, the Number of Density Variables N_e, the Number of Semi-Core and Valence States below the Fermi Energy, the Largest Change in the Position of Any of the Atoms As Well As the Approximate Number of SCF Cycles to Convergence with MSR1 as well as BFGS, both SCF and Outer-Loop Iterations

Figure	name	atoms	ineq atoms N_a	atomic variables N_{av}	matrix size	density size N_e	states	max change (au)	SCF MSR1	SCF/Iter BFGS
3	$Ni(OH)_2$	5	3	2	601	5644	15	0.20	27	108/8
4	MgO (111)+H_2O	52	21	42	6344	28842	176	2.12	241	414/23
6	$Al_{106}Fe_2$	108	42	92	5945	82643	491	0.48	76	148/8
7	TiO_2 (001) 2 × 2	40	16	35	6518	25361	163	0.53	156	349/13
8	$SrTiO_3$ (001) 2 × 1	108	44	96	16106	65752	400	2.87	480	645/40
9	MgO (111) octapole	44	12	20	10029	17046	176	1.09	92	170/13
		68	18	32	10185	21732	272	1.19	109	158/10
		92	24	44	10341	26418	358	1.27	138	152/10
		116	30	56	10497	31104	464	1.35	151	189/14
10	$C_2N_2H_8$	24	6	18	5764	16085	26	3.20	575	575/45

earlier. (3) When the problem is close to a minimum for the atomic positions, the mixing will be faster since the number of different chemical environments and hence local dielectric responses is reduced compared to a case far from equilibrium with strange atomic positions.

The convergence of the density subproblem will scale as the number of clusters of the dielectric response (C^E), which will scale more as the number of active states below or near the Fermi energy, which is a much smaller number than N_e (see for instance Table 1), that of the atomic positions as the number of clusters of the force matrix or energy Hessian (C^F).

What about the method described here? This is going to depend upon the eigenvalues of a general dielectric response which includes the effect of varying the atomic positions as well as the electronic response, and there does not appear to be any analysis of this to date. The total number of clusters can be bounded as

$$\max(C^E, C^F) < C < C^E C^F \tag{20}$$

This compares to a serial method which will converge with a rate that will scale as the product $C^E C^F$. A reasonable hypothesis is that the fused problem will scale as the number of different environments. This suggests that the algorithm will scale sublinearly with problem size which is consistent with experience to date and the results shown later, albeit hard to prove beyond reasonable doubt. Much more important than size is the quantum mechanical problem; not all hard problems are large and not all large problems are hard, as demonstrated by the results shown later.

3. ALGORITHM DETAILS

From the previous sections it is apparent that the unpredicted component needs to be controlled; one needs to safeguard for changes in the Jacobian, and different parts of the basis-set need to be appropriately scaled so the eigenvalues are compressed to the smallest number of narrow clusters. With a gradient-based optimization problem, how to do this is well documented, but for the fixed-point problem there has been little analysis. Some parts are relatively straightforward, for instance preconditioning to compress the eigenspectrum. Others, such as how to scale different parts of the basis set, are not obvious.

It is important to recognize that with straight density functional methods there is no reliable metric of exactly how far a given density is from the solution without incurring excessive additional computational cost; the algorithm does not know the correct energy when the density is not self-consistent, which elements of the basis set matter the most, or how large are the non-self-consistent force terms.

For step sizes, it is always tempting to take as large a step as possible. However, since this information is used for subsequent iterations, this approach can lead to problems. Very small steps are also bad, as discussed in section 2.2—the algorithm can starve to death. What is needed is to mitigate the greed, not too much and not too little. The strategy here is that the trust region limits should only be occasionally active (<5% of iterations, ideally none) and are present for safety in case something unexpected occurs.

The specific details for the unpredicted step size and trust region given below have been approximately tuned for the Wien2k code using a range of problems and are not accessible to routine users; with other codes, similar but different tuning parameters will need to be developed. The Wien2k code uses (e.g., ref 78) an all electron APW-lo method with atomic spheres (muffin-tins) within which the density is represented as a product of a logarithmic radial density (CLM) multiplied by spherical harmonics, plus an interstitial region where a Fourier series is used (PW). To solve the SCF mapping, either a full diagonalization of the Hamiltonian or an iterative one[79] (e.g., Figure 10) is used. The parameters below have been adjusted to the extent possible to work independently of whether the user starts from densities for isolated ions, that for a converged density with fixed atomic positions or anywhere in between.

There are some metrics which can be borrowed from optimization theory and will play an important role later so will be introduced first, as well as a running average form:

Bounded Running Average. This is defined as

$$\begin{aligned} &Ave(a_n, m, C) \\ &\quad = \{Ave(a_{n-1}, m-1, C) \\ &\qquad + \max(1/C, \min(C, a_n)\}/2 \\ &Ave(a_{n-m}, 0, C) = \max[1/C, \min\{C, a_{n-m}\}] \end{aligned} \tag{21}$$

where m is the number of prior history steps used. This is used to minimize oscillations of the parameters.

Multisecant Shanno-Phua Scaling. Borrowing from methods for choosing the initial Hessian scaling,[80] I will define the multisecant form as the value of σ_n^{SP} which minimizes the Froebius norm, $norm_F$ (the square root sum of the squares of the elements, also the square root of the sum of the singular values squared):

dx.doi.org/10.1021/ct4001685 | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

$$\sigma_n^{SP} = \max(0.05, \min_x norm_F(xS_n^T Y_n - Y_n^T Y_n)) \quad (22)$$

Multisecant Step Size. This is an estimate of the relative total step size σ_n^{Step} as the ratio of the largest singular values (spectral norm) of S_n divided by that for Y_n

The next section details the components of the algorithm in the order they are used.

3.1. Initial Steps. For the very first Pratt step, a heuristic based upon tests is used:

$$\sigma_0 = \max(\sigma_{\max}(0.25 + \exp(-dQ_0)), \sigma_{\min}) \quad (23)$$

with dQ_n being the multiplicity weighted RMS change of the density within the muffin tins for iteration n; a typical value is $\sigma_0 = 0.035$. Rather than just accepting this, in the next iteration it is tested using σ_0^{SP}. If this implies that a larger step should be used, a second expanded Pratt step was taken up to an additional twice σ_0. Alternatively, if $|G|$ increased, a second Pratt step of half the distance is used.

3.2. Rescaling of the Total Charge Density. For just densities, since both ρ_n and $F(\rho_n)$ have the correct integrated density, no rescaling is needed. However, when the atomic positions change, the total density is not preserved in a LAPW method. This is not a serious issue, but it is slightly better to correct it. Since the core electrons are almost completely confined within the muffin-tins, the most appropriate approach is to apply a correction to just the valence electrons, approximating the core electrons as frozen. This correction, a simple rescaling, is done after the next QN density has been calculated, although in principle it could be added as a constraint.

3.3. Choice of the Number of Steps. The default number of history steps is 8, which is approximately optimal. If the algorithm is not behaving well, for instance due to numerical noise near the minimum or when transitioning an electronic phase transition, increasing this number adds stability. The simple heuristic of increasing the number of iterations by 1 up to a maximum of 16 is used if $dQ_n > 2dQ_{n-1}$ and decreasing it by 1 to a minimum of 8 if $dQ_n < dQ_{n-1}$ while retaining a record for the next iteration of the number of memory steps to use.

3.4. Preconditioning. An issue for a LAPW-based method is scaling of the plane-wave components versus the logarithmic sampling within the muffin-tins, plus the pseudogradients and atomic positions, as well as a density-matrix term for Hubbard U methods or similar. In addition, Wien2k makes maximum use of symmetry so the multiplicity of different components of the plane wave representation and of the atoms should be taken into account.

First, unlike other parts of the problem, the atomic positions and gradients are in different units. Natural units for them are atomic units and mRy/atomic units and have been used. Small adjustments might be better for particular problems, but QN methods are weakly sensitive to small scaling changes.

Second, we want to precondition the variables such that the effects of multiplicity are removed. This leads to a preconditioning matrix acting on the variables of

$$\Theta_n = \begin{pmatrix} 1/\sqrt{m_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1/\sqrt{m_N} \end{pmatrix} \quad (24)$$

where m_i is the multiplicity of the given variable, for instance the number of symmetry equivalent atoms or reciprocal lattice vectors.

Next, writing $D_n^i(\rho_n)$ for the component of the density residual projected onto the 'i' distinctly different component of the basis set, then similar to previous work the algorithm solves for the weights ω in

$$\begin{aligned} &\omega^i \sum_n \| D^i(\rho_n)\|^2 / L_n \\ &= \omega^g \sum_n \| g_n \|^2 / L_n,\ L_n \\ &= \left(\sum_i \omega^i \| D^i(\rho_n)\|^2 + \omega^g \| g_n \|^2\right) \end{aligned} \quad (25)$$

This leads to a second preconditioning matrix acting on the full basis set of form

$$\Omega_n = \begin{pmatrix} \sqrt{\omega^{CLM}/\omega^{PW}}\, I^{PW} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sqrt{\omega^{CLM}/\omega^{g}}\, I^{g} \end{pmatrix} \quad (26)$$

where I^{PW} is an identity matrix for the interstitial components, and I_g is that for the atomic positions and gradients and (if appropriate) the density matrix or on-site exact-exchange terms.[81,82] For algorithmic reasons, rather than allowing the scaling parameters to be completely free, the maximum change for any cycle relative to the previous one was bounded between 1/1.5 and 1.5 to avoid oscillatory behavior.

Following this, the different steps of the history were scaled as in earlier work:

$$\Psi_n = \begin{pmatrix} 1/\| y_1 \| & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1/\| y_n \| \end{pmatrix} \quad (27)$$

3.5. Regularization. Equations 10−12 contain the inverse

$$(Y_n^T Y_n + \alpha Y_n^T S_n)^{-1} = A_n^{-1} \quad (28)$$

Some singular values of A could be small, which could lead to instabilities. Following a similar concept to earlier work, we replace this by a conventional Tikonov regularized inverse,[83−86] i.e., use

$$(A_n^T A_n + \lambda^2 I)^{-1} A_n \quad (29)$$

and use a single-valued decomposition to avoid calculating the inverse. There exist methods such as a generalized cross-validation approach,[87] which can estimate the value of λ, but in tests this too strongly damps small eigenvalues. The best method currently is to use a value of $\sim 10^{-6}$ times the spectral norm of $Y_n^T Y_n$.

3.6. Control of the Unpredicted Step. The total step can be split into two parts:

$$\begin{aligned} p_n^P &= -S_n (Y_n^T W)^{-1} W^T G \text{ and } p_n^U \\ &= -(I - Y_n)(Y_n^T W)^{-1} W^T G \end{aligned} \quad (30)$$

where p_n^P is the predicted part of the step based upon the prior information, and $\sigma_n p_n^U$ is the unpredicted part about which nothing is known. Similar to previous work, an implicit trust-

dx.doi.org/10.1021/ct4001685 | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

region was used to control the unpredicted step via the generalized residual by a combination of three terms. The first control limits the total change for any given step as

$$\sigma_{n+1} = \sigma_n \max(1.0/1.5, \min(1.5, C_{n+1})) \quad (31)$$

where C_{n+1} is the bounded running average of previous improvements

$$C_{n+1} = Ave(|R_n|/|R_{n+1}|, m+1, 2) \quad (32)$$

Next, σ_n^{SP} is used to prevent the greed from becoming either too small or too large based upon the prior history by

$$\sigma_{n+1} = \max(1.5\sigma_{n+1}, \sigma_n^{SP}/3) \text{ if } \sigma_{n+1} < \sigma_n^{SP}/3$$
$$= \max(\sigma_{n+1}/1.5, 3\sigma_n^{SP}) \text{ if } \sigma_{n+1} > 3\sigma_n^{SP} \quad (33)$$

This forces larger greed when the total step size is large, which is needed for soft modes, while also decreasing it for small step sizes. As a final constraint, the range was limited as

$$\sigma_{\min} \le \sigma_{n+1} \le \min(1, \sigma_{\max} * \sigma_n^{Step}) \quad (34)$$

where σ_{max} is a user input, by default 0.2, and

$$\sigma_{\min} = \min(\sigma_{base}, \max(0.005, 0.75\sigma_{base}/dQ)) \quad (35)$$

with σ_{base}=0.01 by default. This allows the algorithm greed to be reduced down to 0.005 for very large density residuals but not otherwise go too small; it is not that critical and could be omitted.

The combination of eqs 30−35 allows the algorithm greed to be automatically decreased when it is not making good progress, and increased when it is. The incorporation of the two metrics is important for adequate treatment of soft modes and without this the algorithm can stagnate. As illustrated earlier in Figure 1, in general it is better to slightly overestimate than underestimate the greed.

3.7. Trust-Region Control. The linear QN model is only reasonably valid for steps smaller than some absolute magnitude, what is called a trust-region, and the magnitude of the largest reasonable step, the trust-region radius. Existing trust-region methods are based upon limiting the total step, i.e., the combination of the predicted and unpredicted steps in eq 30. While this is appropriate when G is a true gradient, there is no guarantee that it is even close to one here. The approach taken is to first limit the predicted step, and only limit the unpredicted if this fails. The algorithm uses a conventional trust-region model of

$$T = 1/2\,|G_n - (S_n(Y_n^T W)^{-1} W^T)^{-1} p^P|^2 - \eta\{|p^P + p^U|^2 - d^2\} \quad (36)$$

where d is a maximum allowed change applied both to the atomic positions and the total change with a Lagrange multiplier η. We minimize T, for which the solution is (for constant p^U)

$$\{(S_n(Y_n^T W)^{-1} W^T)^{-2} + \eta I\} p^P = (S_n(Y_n^T W)^{-1} W^T)^{-1} G_n \quad (37)$$

While this form is correct, it is numerically involved to include the regularization described in the previous section. Multiplying by $(S_n(Y_n^T W)^{-1} W^T)^2$ and requiring that p^P is a linear combination of the prior steps gives the equivalent form:

$$\{I + \eta A_n^{-2}\}(S_n^T S_n + Y_n^T S_n)c = A_n^{-1}(S_n^T + Y_n^T) G_n \quad (38)$$

where the inverse of A_n is the regularized form defined above. This was solved by a back-tracking bracketing strategy.

If only one history was present or the trust-region condition was never satisfied because it was violated by the unpredicted step, the algorithm automatically switched to, in order, (a) a conventional double-dogleg strategy[88] using the trust-region model above, again only for the unpredicted step, and (b) a Cauchy step[57] with steepest-descent for the predicted step added to the unpredicted step, and the combination scaled such that the constraints were obeyed.

A standard trust-region method adjusts the radius, increasing it if the algorithm is making good progress, decreasing it if not. Here "progress" is not well-defined, so the approach has been taken of limiting the total step as well as that of the atoms so neither are too large.

First, for the atoms the largest atomic movement was limited so that the muffin-tins do not overlap, and by

$$d^{Atoms} = \min(d_{\max}^{Atoms}, \sqrt{Ave(|g_n|^2 + 4, m, \infty)}) \quad (39)$$

where $d_{\max}^{Atoms}$ is a user parameter, by default 0.1 au. This limits the largest atomic displacement to the range 0.02−0.1 au.

Next, the total step as a fraction of a full Pratt step (i.e., $|R_{n+1}|$) was limited by

$$d^{tot} < 4Ave(\mu_{n+1} * \max(0.5, \min(4, \alpha\sigma_n^{Step})), m, \infty)$$
$$\mu_{n+1} = \max(0.1, \min(1.0, \sigma_{n+1}/0.15)) \quad (40)$$

This (1) reduces or increases the total step depending upon the unpredicted step greed, an implicit trust-region control, (2) reduces or increases the total step depending upon the previous steps—this is needed for soft modes, and (3) reduces or increases the total step depending upon to what extent the algorithm is using MSB1 versus MSB2, appropriate since when α is small the linear model will only be valid for a limited total step.

The total step is limited to the range 0.2−16 times a simple Pratt step. Typical values when α is small, i.e. the linear model has limited validity, are d^{tot} of 1−2, whereas when it is optimizing soft modes, d^{tot} is closer to 10−12.

For a well-posed problem, e.g. a MgO surface starting from the densities of isolated atoms, the trust-region control might be activated in the second step, and for ∼2% of the steps, it is not important. In a few cases it can be activated more than this, for 5−10% of the steps or very close to the final solution if there is numerical noise.

3.8. Algorithm Summary. The same algorithm is used for self-consistency of the density alone, and for a full fixed-point optimization; the only difference is that the atom positions and gradients are not part of the problem when only the density is used, and there is no need to rescale the valence density.

(1) Choose an initial (ρ_0, R_0) (user input) and then σ_0 from section 3.1 and generate $(\rho_1, R_1) = (\rho_0, R_0) - \sigma_0 G_0$ so long as the step is not too large (if needed, use a linear trust region to reduce the step). Rescale the valence density using section 3.2. Set $n = 1$.

(2) Check if the new density represents an adequate reduction (section 3.1). If it does, move to 3; otherwise, take another Pratt step with either a reduced or larger algorithm greed (section 3.6) and rescale the valence density using section (3.2).

(3) If the convergence criteria are met, stop; otherwise determine the number of history steps to use from section 3.3.

dx.doi.org/10.1021/ct4001685 | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

(4) Generate the scaled and preconditioned matrices S_n and Y_n as well as scaled and preconditioned G_n using section 3.4.

(5) Solve for Broyden-family term α as discussed in section 2.3, and calculate the regularization term from section 3.5.

(6) Calculate the unpredicted greed from section 3.6.

(7) Calculate the next full Newton step (eqs 6 and 10). If this violates the trust-region limits, solve the trust-region subproblem for the step (section 3.7). Calculate the new density.

(8) Rescale the valence density section 3.2.

(9) Calculate the new potential, and solve the Kohn−Sham equations. Set $n = n + 1$ and repeat step 3.

4. RESULTS

I will show a few representative examples here. In all cases, comparisons are made to a traditional double-loop method where MSR1 was used for the density-only part, and for the atomic positions, the Broyden−Fletcher−Goldfarb−Shanno (BFGS) algorithm drmng code,[43,88−91] which is available from Netlib and is a standard against which numerous other codes have been benchmarked. It is a reverse-communications code written in standard Fortran 77, so it can be readily incorporated into any DFT code. (The only modification to the code when it was added in 2004 was a minor trap to avoid overflows with the secant update for very small steps and a printout of when the curvature condition failed, as this often indicates a user-error in the model.) In all cases, an approximate initial Hessian was used based upon a simple spring model,[92] which yields a ∼30% reduction in the minimization cost compared to a unitary matrix.

Starting from a user specified Hessian estimate and initial trust-region radius, this BFGS algorithm updates the Cholesky factorization of the Hessian via a safeguarded method to ensure that it remains positive-definite. If the energy increases, the Hessian is not updated, and a backtracking strategy is used with a reduced trust-region radius. A similar back-tracking method is used if the next step is inappropriate, for instance due to overlapping muffin tin spheres. A conventional double-dogleg procedure[88] is used for the step ranging from a Cauchy[57] (steepest descent) step, a double-dogleg (step from the Cauchy point toward the unrestricted step), or full Newton steps depending upon the trust-region radius. This radius is increased if the total energy reduction is "good enough," where standard heuristics are used to compare the expected and obtained energy reductions.

For completeness, the additional cost of calculating the forces at every iteration in the Wien2k code is relatively small, about 5−10% of the cost of a single iteration based upon benchmarks, mainly due to the Pulay corrections.[34−36] In principle, the forces only need to be calculated with a converged density with a conventional double loop. The additional cost is small, so in all cases below, the number of SCF iterations can be taken as an absolute measure of the computational cost; the computational cost of including the atomic variable is very small (<0.01%).

In all cases, default parameters were used, and no attempt has been made to optimize any settings in Wien2k or the algorithm itself for the different cases. The results shown are representative of what a novice user would obtain within numerical accuracy and reproducibility. Table 1 summarizes many of the key parameters for the different examples shown; see the Supporting Information for the initial positions in all cases.

Note that in the MSR1 results the forces and energy shown are not the true ones (they are for the BFGS calculations), and as mentioned earlier the term pseudogradient and similarly pseudoenergy would be more appropriate. Both converge to the true values at the fixed-point but are otherwise only representative, and the energy as shown is not variational. However, both are useful metrics and in practice tend to converge in a representative fashion.

The first example, shown in Figure 3, is $Ni(OH)_2$ run spin-polarized using an on-site exact-exchange hybrid[81,82] for the Ni

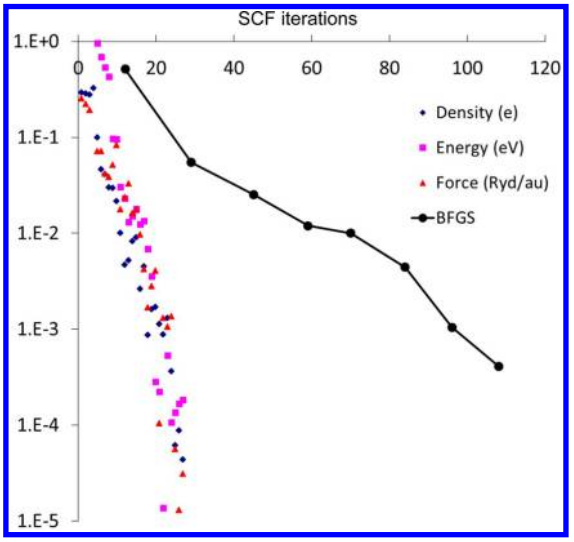


Figure 3. Plot of the convergence of the RMS density residual within the muffin tins, the energy above the minimum in eV, and the RMS force in Ry/au (so they have similar scales) for bulk $Ni(OH)_2$ with an on-site exact-exchange hybrid as well as for the double-loop BFGS method. Earlier iteration values of the energy are not shown, as the numbers are meaningless when the density residual is very large.

d states, starting from the X-ray structure and lattice parameters and neutral atoms. This is a relatively simple problem with only two atomic free variables and, since it is a large band gap insulator, a simple dielectric matrix. The calculation converges the density and forces in 27 iterations, the largest displacement being movement of the H atom by about 0.2 au; it takes 8 atomic iterations with BFGS, a total of 108 SCF mappings. This is about the same as a calculation for bulk $Mg(OH)_2$. The various metrics shown in the figure all scale exponentially with the number of iterations with a form of approximately $\exp(-0.337n)$ where n is the number of iterations, i.e. linear convergence with a rate of convergence of ∼0.7. Note that this is the conventional definition of convergence of an optimization algorithm.

The second in Figure 4 is a more complicated representation of a hydroxylated surface of MgO (111) (see ref 93 for details). The calculations were done using the PBE functional and an initial 10 iterations with just the density, then using iterative diagonalization.[79] This is a much longer calculation and converges relatively slowly (convergence rate 0.9792) due to soft modes for the surface hydroxyl groups. The largest displacement of an atom from the original position was 2.12 au. The MSR1 algorithm was about 2 times faster than BFGS, which had a convergence rate of 0.9842; per iteration the fractional reduction in the energy is twice as large. It should be mentioned that the forces apparently converge quicker than the energy, which is correct remembering that these are not the

dx.doi.org/10.1021/ct4001685 | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

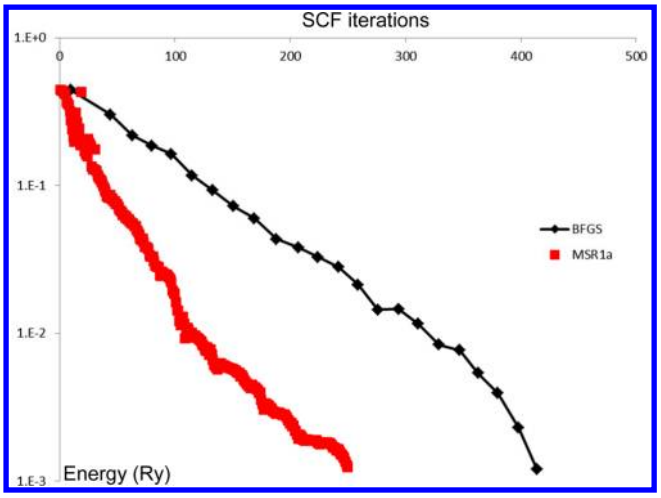


Figure 4. Comparison of the energy in Ry with respect to the minimum for the MSR1 algorithm in red and a conventional double loop in black, for a 52 atom MgO (111) surface with chemisorbed water.

true forces for self-consistent densities. The algorithm is effectively minimizing the L2 of the generalized residual *G*, which has many more density terms than the pseudoforces. It is therefore natural for the algorithm to take a trajectory intermediate between the Born−Oppenheimer surface and the surface of zero-force (and inconsistent densities) as illustrated in Figure 5.

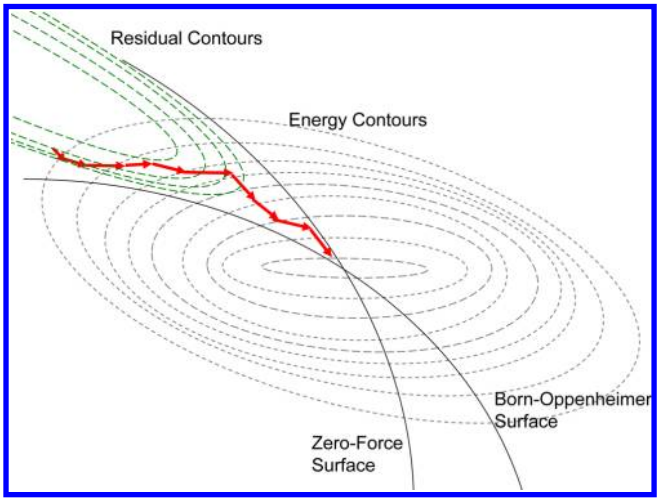


Figure 5. Illustration of the Born−Oppenheimer and zero-force surfaces.

Another example in Figure 6 is a 22.81 × 32.26 × 32.26 au structure with *Amm*2 symmetry containing 42 unique atoms, 108 total, of which two are Fe and the rest Al. This is a supercell where two of the nearest neighbor Al atoms in the initial perfect fcc lattice have been replaced by Fe. It was run spin-polarized with a k-mesh of 3 × 3 × 3, starting from atomic densities with the Al nonmagnetic and the Fe in a configuration of Ar $3d^7 4s^1$ with an initial unbalanced spin of 2. The two Fe atoms change positions somewhat, the distance between them increasing from 9.79 to 11.21 au, while nine of the 11 Al atoms around each move closer from 9.79 to 9.37 (one atom) or 5.04 (10 atoms) au, with the largest displacement of any atom being about 0.48 au.

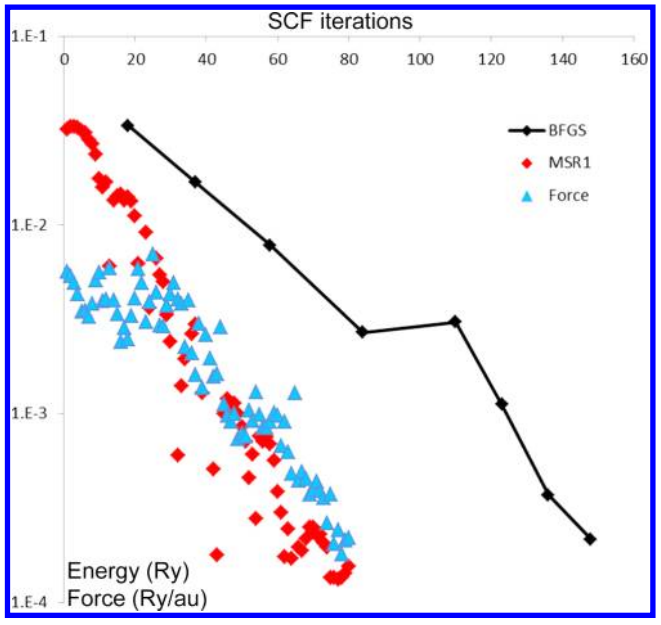


Figure 6. Plot showing the convergence of the RMS force in Ry/au and energy in Ry both for the $Al_{106}Fe_2$ model as well as for the energy with a conventional double loop.

Additional examples are shown in Figures 7−10 all run without spin-polarization using the PBE functional with additional details in Table 1:

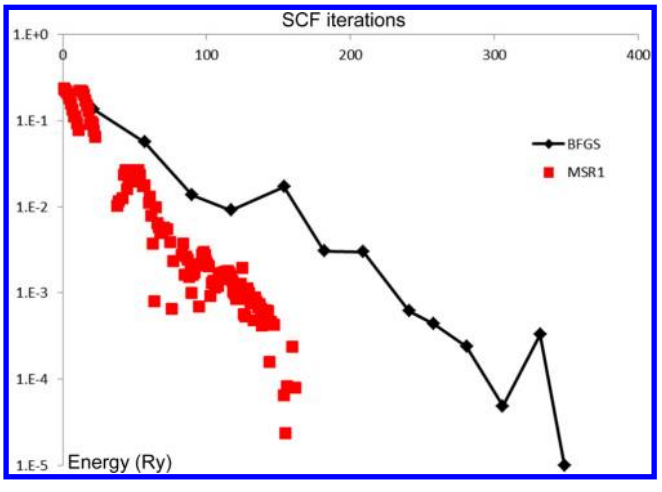


Figure 7. Comparison of the energy in Ry with respect to the minimum for the MSR1 algorithm in red and a conventional double loop in black, for a reconstructed TiO_2 (001) surface with an oxygen vacancy.

•Figure 7 is a reduced (i.e., metallic) reconstructed TiO_2 (001) surface taken from ref 94 in a 17.5 × 52.6 × 12.3 au unit cell with *B*2/*m* symmetry, a 4 × 1 × 4 k-point mesh and room-temperature Fermi−Dirac occupancies.

•Figure 8 is a 2 × 1 $SrTiO_3$ (001) 2 × 1 surface with chemisorbed water taken from ref 95 with a 14.9 × 56.7 × 14.9 au unit cell and *P*2/*m* symmetry, run using iterative diagonalization with a 2 × 1 × 2 k-point mesh.

•Figure 9 is a 2 × 2 MgO (111) octapolar terminated surface (see ref 93) with different numbers of inequivalent atoms (i.e., layers) as marked on the figure and started from bulk atomic

dx.doi.org/10.1021/ct4001685 | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

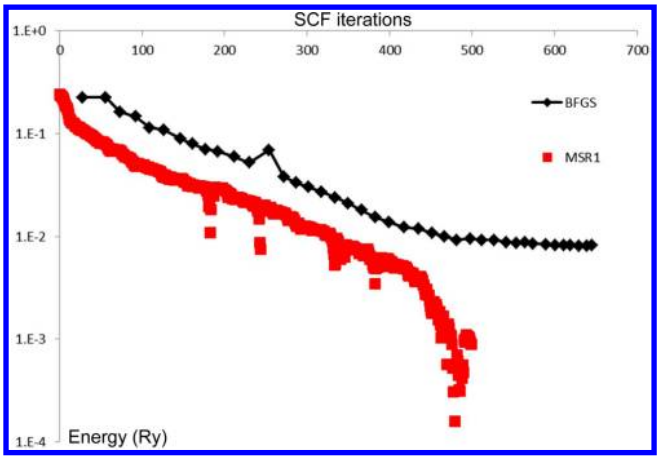


Figure 8. Comparison of the energy in Ry with respect to the minimum for the MSR1 algorithm in red and a conventional double loop in black, for a $SrTiO_3$ (001) 2 × 1 surface with chemisorbed water.

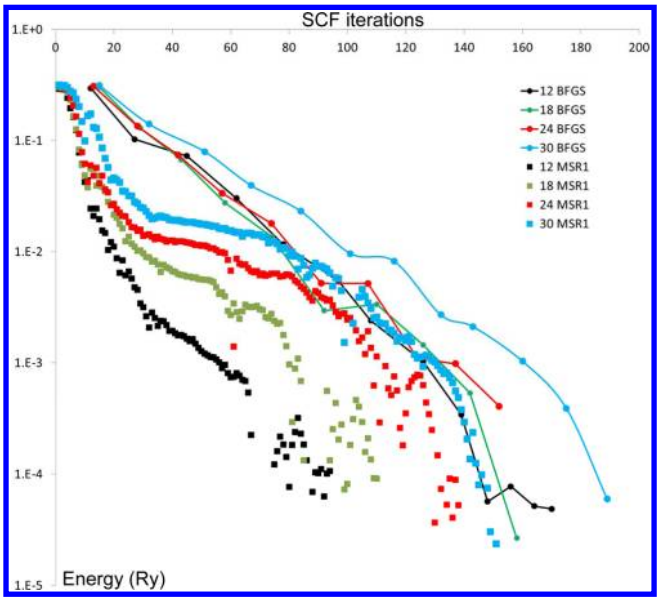


Figure 9. Comparison of the energy in Ry with respect to the minimum for the MSR1 algorithm and a conventional BFGS double loop, for a range of oxygen-terminated octapolar 2 × 2 MgO (111) surfaces with different numbers of inequivalent atoms as marked.

positions, all with a 11.34 × 11.34 × 117.16 au unit cell and $P\bar{3}m1$ symmetry with a 4 × 4 × 1 k-point mesh in all cases.

•Figure 10 is the exception, an apparently simple $C_2N_2H_8$ molecule in a P21/c 13.46 × 17.63 × 9.17 au unit cell obtained by taking bulk polyethylene, cutting out four carbons, replacing the outer two with nitrogen, and removing two hydrogens. Results are shown with both full and iterative diagonalization using a 4 × 3 × 6 k-point mesh.

As general comments on Figures 7−10: (1) With the exception of Figure 10, MSR1 is approximately twice as fast as BFGS. (2) They are all noisier than the simpler examples shown earlier, and in some cases the BFGS code takes too large a step and the (variationaly correct) energy increases, which should not be a surprise and is not a problem. (3) Superlinear convergence for MSR1 is clearly observed in Figures 7, 8, for some cases Figure 9, and Figure 10. (4) Superlinear convergence for BFGS is observable in Figures 9 and 10. (5)

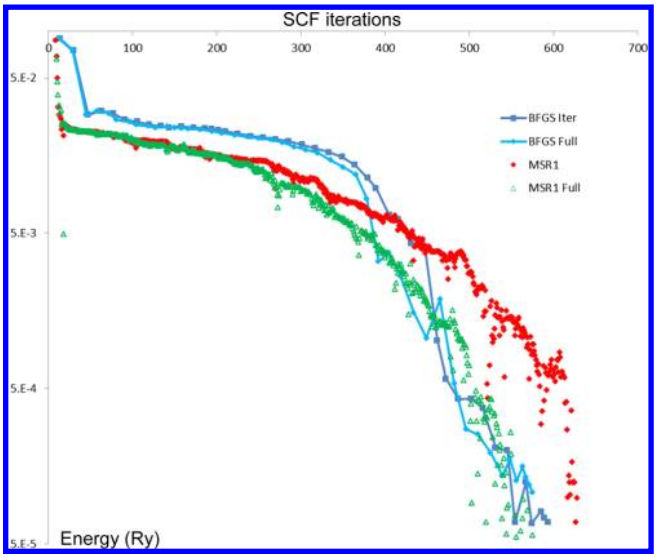


Figure 10. Comparison of the energy in Ry with respect to the minimum for the MSR1 algorithm and a conventional BFGS double loop for a $C_2N_2H_8$ molecule started relatively far from equilibrium using both full diagonalization as well as iterative diagonalization.

In Figure 8, BFGS finds a higher minimum, whereas MSR1 finds a lower one. Since these methods are all local (not global) minimizers and there is a stochastic component, this can always occur. (6) As shown for the comparison in Figure 10, the iterative diagonalization is less accurate, which slows down the convergence more noticeably for MSR1. The overall computational cost of the iterative diagonalization is sufficiently advantageous that this does not matter. (7) The number of iterations to convergence does not have a simple dependence upon the number of atoms or the number of occupied states which can also be seen from Table 1 and the specific comparisons in Figure 9, and discussed in section 2.3.

5. DISCUSSION

The algorithm was first released in a simpler form (without the trust-region control and with $\alpha = 1$) in April 2011. That form worked very well for insulators but had some problems with d-electron metals. The addition of a trust-region control for the total step size solved many of the instabilities, and they have been reduced further by forcing the projected component of the eigenvectors to have positive, real eigenvalues (section 2.3). The version with $\alpha = 1$ and trust-region control was released as part of the main Wien2k package in July 2012. It has already been used in a number of papers.[20−31] The more advanced version with the ansatz for control of α is scheduled for release in 2013 and has been in use at a number of locations for some time. It handles soft modes much better than the earlier versions which could need restarting.

No algorithm is perfect. In general use, MSR1 is robust and offers speed improvements of roughly a factor of 2 compared to the double-loop BFGS, sometimes more and occasionally less. Comparable speed improvements can be expected with a plane-wave code, perhaps better since there are fewer complications with the scaling and conditioning of the basis-set.

It has been used for many more cases than those shown, and some generalizations may be useful with the standard caveat that there are exceptions to all generalizations. One class of problems where the algorithm can be better is where there is a large change in the electronic configuration when the atomic

dx.doi.org/10.1021/ct4001685 | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

positions change, for instance transition from a metal to an insulator. When this occurs, the Hessian with respect to the atom positions will change substantially, so information from when it was a metal will be less relevant. Conventional BFGS (or a limited-memory version) will only slowly accommodate to this over a number of full self-consistency calculations, whereas the algorithm herein will after a few iterations.

How the physical problem is posed matters. A calculation started from unusual densities or atomic positions will have to traverse odd electronic phases to reach a minimum, or even to achieve self-consistency of the density. Increasing the effective temperature and adjusting the k-point sampling are standard ways of improving the convergence. While it is not uncommon to use rather large effective temperatures (e.g., 0.2 eV equivalent to a temperature of ∼2400 K, see for instance ref 65 and references therein), the author's experience with MSR1 and Wien2k is that using room temperature or at most 500−600 K is more than adequate, and higher temperatures than this can be worse. Most cases where self-consistency is hard to achieve, albeit not necessarily all, can be traced to user errors in how the model for the DFT experiment has been set up. It is worth stressing that the convexity of the quantum mechanical and numerical problem is under the users' control in most cases, and improving it is often more important than anything else. A simple example of this is the slower convergence of the iterative method in Figure 10, although it is faster in terms of net CPU time.

There is certainly room for improvement. The current bounds used for the trust-region controls in section 3.6−3.7 may be too complicated although they have qualitatively the right dependencies; they do avoid a need for user input. No preconditioning of the gradients or transformation to internal coordinates is currently used, and previous work has indicated that something as simple as a spring-model can be very effective[92] or possibly some approximation to the dielectric matrix, for instance just the diagonal term.[72,96] As a caveat, the use of a spring model preconditioner has been tested by the author within MSR1 and did not appear to help. Going in the opposite direction, it would be very useful to be able to estimate the elastic mode eigenvalues and vectors from the prior history. While in principle this information is available, to date reasonable attempts by the author to extract this information from the MSR1 step history have failed.

There is also some more investigation which could be useful. The dielectric band structure has been calculated for a few simple insulators, but it is hard to find useful information for metals. It would also be useful if analyses existed not just for the electronic component but for the complete dielectric response including atomic positions.

A few important comments for readers interested in implementing MSR1 in other DFT variants follow. Experience in the four years since MSB2 was first introduced is that the mixer is an excellent debugger of code (and operating systems) and quite sensitive to noise. In some cases, changes in the underlying Wien2k code have led to parts being faster, but introduced noise. Sometimes the net effect for the self-consistent calculation was to make it slower. The current version of Wien2k is much more stable than the version when MSB2 was first introduced, and as a consequence the regularization used is smaller than it was. The regularization of MSR1 is different, and much smaller than that for MSB2. The parameters given herein are not those which will give the faster results in all cases, rather ones which are robust for general use. With other codes, the exact values will need to be determined from tests; something as simple as bulk $Mg(OH)_2$ should be easy; more demanding are d-electron metals. The experience of the author is that it is best to avoid heuristic controls as much as possible, and let the implicit trust-region control for the unpredicted step and the explicit trust-region control for the total step magnitudes take control, and be as noninvasive as possible.

Numerical accuracy is an issue; QN methods are sensitive to this. Wien2k can be more accurate than pseudopotential codes but also contains many density contributions which change little. There are also many numerical algorithm issues such as aliasing of the spherical harmonic components of the exchange-correlation potential within the muffin tins as well as how numerical differentiation is done, all of which can have unexpected consequences. Even with double-precision arithmetic, it can be difficult to achieve convergence to better than 0.1 mRy/au in the forces and 1.0 mau in the positions particularly for heavy elements and large cells.[97]

A second set of comments more for the new user; this algorithm is different. Sometimes the atoms move away from the initial positions as the forces and density residual simultaneously reduce and then move back toward where they started from. This is quite reasonable since the elastic/dielectric response of the system is changing and there may be phase transitions in the electronic subsystem. This does raise interesting questions as to whether it is better to start the optimization from a somewhat self-consistent electronic configuration and to what extent one should stay close to the Born−Oppenheimer surface. For mathematical reasons, hugging the Born−Oppenheimer surface is inefficient, and it is better to run as unconstrained as possible relying upon the variational convergence of the density and atoms. In many cases, this is an issue of what the user is most comfortable with. In addition, the final convergence is also different and is sometimes only to the relative accuracy of the approximations used in the calculation (e.g., reciprocal space sampling) rather than the apparently precise convergence of standard double-loop methods which can appear to give energies accurate to meV or even μeV.

In summary, this note outlined an algorithm for simultaneous fixed-point optimization of the electron density and atomic positions. The algorithm is stable, roughly twice as fast as conventional double-loop methods. The behavior of the algorithm can be understood in terms of electronic phase transitions as the density and atomic positions change, and how these and other nonlinearities couple into the accuracy of a linear fit of the underlying quantum mechanics as well as choices for algorithmic greed. The algorithm used is an adaptive variant which can exploit the greed of MSB1 under favorable circumstances, but will switch to MSB2 when this is not favorable. Key additional elements of the algorithm are the use of an extrapolated component of the Jacobian with real, positive eigenvalues, control of the unpredicted step, regularization, and trust-region control of the overall step.

■ ASSOCIATED CONTENT

Ⓢ Supporting Information

Crystallographic Information Format (CIF) files for the initial structures used in the manuscript named by the figures where they appear and (if appropriate) the structure name. This information is available free of charge via the Internet at http://pubs.acs.org/.

dx.doi.org/10.1021/ct4001685 | J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

AUTHOR INFORMATION

Corresponding Author

*E-mail: l-marks@northwestern.edu.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

This work was supported by the NSF on grant number DMR-0906306. The author is indebted to Russell Luke for numerous discussions on fixed-point problems, to Sanjay Mehrotra, Mihai Anitescu, and Dimitri Bertsekas for useful information about the convergence of QN methods, and to Lyudmila V. Dobysheva for the Al/Fe structure. This work would have been impossible without numerous discussion with Peter Blaha.

NOMENCLATURE

α, parameter for Broyden-family control between "good" and "bad" multisecant Broyden forms; $Ave(a_n,m,C)$, bounded running average, used when controlling the step size to avoid fluctuations; B, Jacobian, a $(N_e + N_{av})x(N_e + N_{av})$ matrix; B1, sequential version of Broyden's first method; B2, sequential version of Broyden's second method; $\varepsilon_{q,q'}$, dielectric matrix for reciprocal lattice vectors $\mathbf{q}$ and $\mathbf{q}'$; $D(\rho(r,R))$, residual of the density only, a vector with N_e components; $\mathrm{Det}(A)$, abbreviation for the determinant of the matrix A; $F(\rho(r,R))$, SCF mapping, i.e. density after solving Kohn−Sham equations—a vector with N_e elements; Ψ, scaling matrix of size $(N_e + N_{av})x(N_e + N_{av})$; g, pseudogradient of energy with respect to atomic positions—converges to the true gradient for a self-consistent density—a vector of N_{av} components; G, generalized residual—a vector with $N_e + N_{av}$ components; H, inverse Jacobian $(N_e + N_{av})x(N_e + N_{av})$ matrix; I, identity matrix; λ, Tikonov regularization parameter; m, multiplicity of a given density value or atomic position; M, number of history steps used in the multisecant method; MSB1, multisecant version of B1; MSB2, multisecant version of B2; MSR1, adaptive multisecant algorithm, multisecant rank one; N_e, number of elements of the density; N_a, number of atoms—set to zero for algorithms which only mix the density; N_{av}, number of free atomic variables, $N_{av} \leq 3N_a$; r, 3D position vector; R, 3D vector of atomic positions, size N_{av}; $\rho(r,R)$, density as a function of position r and atomic positions R—a vector with N_e elements; s, step for both density and atomic positions—a vector with $N_e + N_{av}$ components; S, matrix of prior s, of size $(N_e + N_{av})xM$; Θ, Ω, preconditioning matrices of size $(N_e + N_{av})x(N_e + N_{av})$; σ, algorithm greed—used to be called mixing factor—controls the unpredicted step; σ^{SP}, multisecant Shanno-Phua step size, used in the step control; σ^{Step}, total step size estimate for the previous history; W, vector of size $N_e + N_{av}$ for satisfying the secant equations: its form changes with algorithm; ω, weights for different parts of the $N_e + N_{av}$ vector; y, change of G—avector with $N_e + Nv_a$ components; Y, matrix of prior y, of size $(N_e + N_{av})xM$

REFERENCES

(1) Car, R.; Parrinello, M. Unified Approach For Molecular-Dynamics and Density-Functional Theory. *Phys. Rev. Lett.* **1985**, *55*, 2471−2474.

(2) Gross, A. Reactions at surfaces studied by ab initio dynamics calculations. *Surf. Sci. Rep.* **1998**, *32*, 291−340.

(3) Headgordon, M.; Pople, J. A. Optimization of Wave-Function and Geometry in the Finite Basis Hartree-Fock Method. *J. Phys. Chem.* **1988**, *92*, 3063−3069.

(4) Iannuzzi, M.; Laio, A.; Parrinello, M. Efficient Exploration of Reactive Potential Energy Surfaces Using Car-Parrinello Molecular Dynamics. *Phys. Rev. Lett.* **2003**, *90*, 238302.

(5) Kuhne, T. D.; Krack, M.; Mohamed, F. R.; Parrinello, M. Efficient and Accurate Car-Parrinello-Like Approach to Born-Oppenheimer Molecular Dynamics. *Phys. Rev. Lett.* **2007**, *98*, 066401.

(6) Schlegel, H. B. Optimization of Equilibrium Geometries and Transition Structures. *J. Comput. Chem.* **1982**, *3*, 214−218.

(7) Csaszar, P.; Pulay, P. Geometry Optimization by Direct Inversion in the Iterative Subspace. *J. Mol. Struct.* **1984**, *114*, 31−34.

(8) Headgordon, M.; Pople, J. A.; Frisch, M. J. Quadratically Convergent Simultaneous-Optimization of Wavefunction and Geometry. *Int. J. Quantum Chem.* **1989**, 291−303.

(9) Pulay, P.; Fogarasi, G. Geometry Optimization in Redundant Internal Coordinates. *J. Chem. Phys.* **1992**, *96*, 2856−2860.

(10) Payne, M. C.; Teter, M. P.; Allan, D. C.; Arias, T. A.; Joannopoulos, J. D. Iterative Minimization Techniques for Abinitio Total-Energy Calculations - Molecular-Dynamics and Conjugate Gradients. *Rev. Mod. Phys.* **1992**, *64*, 1045−1097.

(11) Baker, J. Techniques for Geometry Optimization - a Comparison of Cartesian and Natural Internal Coordinates. *J. Comput. Chem.* **1993**, *14*, 1085−1100.

(12) Peng, C. Y.; Ayala, P. Y.; Schlegel, H. B.; Frisch, M. J. Using Redundant Internal Coordinates to Optimize Equilibrium Geometries and Transition States. *J. Comput. Chem.* **1996**, *17*, 49−56.

(13) Farkas, O.; Schlegel, H. B. Methods for Geometry Optimization of Large Molecules. I. An O(N-2) Algorithm for Solving Systems of Linear Equations for the Transformation of Coordinates and Forces. *J. Chem. Phys.* **1998**, *109*, 7100−7104.

(14) Baysal, C.; Meirovitch, H.; Navon, I. M. Performance of Efficient Minimization Algorithms As Applied to Models of Peptides and Proteins. *J. Comput. Chem.* **1999**, *20*, 354−364.

(15) Farkas, O.; Schlegel, H. B. Methods for Optimizing Large Molecules - Part III. An Improved Algorithm for Geometry Optimization Using Direct Inversion in the Iterative Subspace (GDIIS). *Phys. Chem. Chem. Phys.* **2002**, *4*, 11−15.

(16) Saad, Y.; Chelikowsky, J. R.; Shontz, S. M. Numerical Methods for Electronic Structure Calculations of Materials. *SIAM Rev.* **2010**, *52*, 3−54.

(17) Moss, C. L.; Li, X. S. First order simultaneous optimization of molecular geometry and electronic wave function. *J. Chem. Phys.* **2008**, *129*, 114102.

(18) Bendt, P.; Zunger, A. Simultaneous Relaxation Of Nuclear Geometries And Electric Charge-Densities In Electronic-Structure Theories. *Phys. Rev. Lett.* **1983**, *50*, 1684−1688.

(19) Kresse, G. *Private communication*, 2011.

(20) Spiel, C.; Blaha, P.; Suchorski, Y.; Schwarz, K.; Rupprechter, G. CeO_2/Pt(111) interface studied using first-principles density functional theory calculations. *Phys. Rev. B* **2011**, *4*, 045412.

(21) Koch, H.; Laskowski, R.; Blaha, P.; Schwarz, K. Adsorption of gold atoms on the h-BN/Rh(111) nanomesh. *Phys. Rev. B* **2011**, *84*, 245410.

(22) Kienzle, D. M.; Becerra-Toledo, A. E.; Marks, L. D. Vacant-Site Octahedral Tilings on SrTiO3 (001), the (root 13 x root 13)R33.7 degrees Surface, and Related Structures. *Phys. Rev. Lett.* **2011**, *106*, 176102.

(23) Lin, Y. Y.; Becerra-Toledo, A. E.; Silly, F.; Poeppelmeier, K. R.; Castell, M. R.; Marks, L. D. The (2 × 2) reconstructions on the SrTiO(3) (001) surface: A combined scanning tunneling microscopy and density functional theory study. *Surf. Sci.* **2011**, *605*, L51−L55.

(24) Marshall, M. S. J.; Becerra-Toledo, A. E.; Marks, L. D.; Castell, M. R. Surface and Defect Structure of Oxide Nanowires on SrTiO(3). *Phys. Rev. Lett.* **2011**, *107*, 086102.

(25) Becerra-Toledo, A.; Castell, M.; Marks, L. D. Water on SrTiO3 (001) surfaces I. *Surf. Sci.* **2012**, *7−8*, 762−765.

(26) Becerra-Toledo, A.; Marshall, M. S. J.; Castell, M.; Marks, L. D. c(4 × 2) and Related Structural Units on the SrTiO3(001) Surface: STM, DFT and Structure. *J. Chem. Phys.* **2012**, *136*, 214701.

(27) Becerra-Toledo, A. E.; Kienzle, D. M.; Enterkin, J.; Marks, L. D. Water on $SrTiO_3$ (001) surfaces II. *Surf. Sci.* **2012**, *606*, 791−802.
(28) Enterkin, J.; Becerra-Toledo, A.; Poeppelmeier, K. R.; Marks, L. D. A Chemical Approach to Understanding Oxide Surfaces. *Surf. Sci.* **2012**, *606*, 344−355.
(29) Hsu, H.; Blaha, P.; Wentzcovitch, R. Ferromagnetic insulating state in tensile-strained $LaCoO_3$ thin films from LDA+U calculations. *Phys. Rev. B* **2012**, *85*, 140404.
(30) McDermott, E. J.; Kurmaev, E. Z.; Boyko, T. D.; Finkelstein, L. D.; Green, R. J.; Maeda, K.; Domen, K.; Moewes, A. Structural and Band Gap Investigation of GaN:ZnO Heterojunction Solid Solution Photocatalyst Probed by Soft X-ray Spectroscopy. *J. Phys. Chem. C* **2012**, *116*, 7694−7700.
(31) Ghimire, N. J.; McGuire, M. A.; Parker, D. S.; Sales, B. C.; Yan, J.-Q.; Keppens, V.; Koehler, M.; Latture, R. M.; Mandrus, D. Unusual Itinerant Ferromagnetism in Non-centrosymmetric $Cr_{11}Ge_{19}$. *Phys. Rev. B* **2012**, *85*, 224405.
(32) Kohn, W.; Sham, L. J. Self-Consistent Equations Including Exchange and Correlation Effects. *Phys. Rev.* **1965**, *140*, 1133−1138.
(33) Mermin, N. D. Thermal Properties of the Inhomogeneous Electron Gas. *Phys. Rev.* **1965**, *137*, A1441−A1443.
(34) Yu, R. C.; Singh, D.; Krakauer, H. All-Electron And Pseudopotential Force Calculations Using The Linearized-Augmented-Plane-Wave Method. *Phys. Rev. B* **1991**, *43*, 6411−6422.
(35) Kohler, B.; Wilke, S.; Scheffler, M.; Kouba, R.; AmbroschDraxl, C. Force calculation and atomic-structure optimization for the full-potential linearized augmented plane-wave code WIEN. *Comput. Phys. Commun.* **1996**, *94*, 31−48.
(36) Madsen, G. K. H.; Blaha, P.; Schwarz, K.; Sjostedt, E.; Nordstrom, L. Efficient linearization of the augmented plane-wave method. *Phys. Rev. B* **2001**, *64*, 195134.
(37) Marks, L. D.; Luke, D. R. Robust mixing for ab initio quantum mechanical calculations. *Phys. Rev. B* **2008**, *78*, 075114.
(38) Marks, L. D.; Luke, D. R., Robust mixing for ab initio quantum mechanical calculations. *arXiv:0801.3098v1* 2008, Longer Preprint.
(39) Broyden, C. G. A Class of Methods for Solving Nonlinear Simultaneous Equations. *Math. Comput.—Math. Comput.* **1965**, *19*, 577−593.
(40) Oren, S. S.; Spedicato, E. Optimal Conditioning of Self-Scaling Variable Metric Algorithms. *Math. Program.* **1976**, *10*, 70−90.
(41) Spedicato, E.; Greenstadt, J. Some Classes Of Variationally Derived Quasi-Newton Methods For Systems Of Non-Linear Algebraic Equations. *Numer. Math.* **1978**, *29*, 363−380.
(42) Martinez, J. M. Practical quasi-Newton methods for solving nonlinear systems. *J. Comput. Appl. Math.* **2000**, *124*, 97−121.
(43) Goldfarb, D. A Family Of Variable-Metric Methods Derived By Variational Means. *Math. Comput.—Math. Comput.* **1970**, *24*, 23−26.
(44) Rheinbol, W. C.; Vandergr, J. S. Local Convergence of Update Methods. *SIAM J. Numer. Anal.* **1974**, *11*, 1069−1085.
(45) Gragg, W. B.; Stewart, G. W. Stable Variant of Secant Method for Solving Nonlinear Equations. *SIAM J. Numer. Anal.* **1976**, *13*, 889−903.
(46) Dennis, J. E.; Schnabel, R. B. Least Change Secant Updates for Quasi-Newton Methods. *SIAM Rev.* **1979**, *21*, 443−459.
(47) Dennis, J. E.; Walker, H. F. Convergence Theorems for Least-Change Secant Update Methods. *SIAM J. Numer. Anal.* **1981**, *18*, 949−987.
(48) Burdakov, O. P. On Superlinear Convergence of Some Stable Variants of the Secant Method. *Z. Angew. Math. Mech.* **1986**, *66*, 615−622.
(49) Martinez, J. M. Local Convergence Theory of Inexact Newton Methods Based on Structured Least Change Updates. *Math. Comput.—Math. Comput.* **1990**, *55*, 143−167.
(50) Gomesruggiero, M. A.; Martinez, J. M.; Moretti, A. C. Comparing Algorithms for Solving Sparse Nonlinear-Systems of Equations. *SIAM J. Sci. Stat. Comput.* **1992**, *13*, 459−483.
(51) Lopes, V. L. R.; Martinez, J. M. Convergence properties of the inverse column-updating method. *Optim. Method. Soft.* **1995**, *6*, 127−144.
(52) Spedicato, E.; Huang, Z. Numerical experience with Newton-like methods for nonlinear algebraic systems. *Computing* **1997**, *58*, 69−89.
(53) Luksan, L.; Spedicato, E. Variable metric methods for unconstrained optimization and nonlinear least squares. *J. Comput. Appl. Math.* **2000**, *124*, 61−95.
(54) *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*; Bauschke, H. H., Burachik, R. S., Combettes, P. L., Elser, V., Luke, D. R., Wolkowicz, H., Eds.; Springer: New York, 2011; Vol. *49*.
(55) Cormen, T. H.; Leiserson, C. E.; Rivest, R. L.; Stein, C. *Introduction to Algorithms*; MIT Press: Boston, MA, 2009.
(56) Kelly, C. T. *Iterative Methods for Linear and Nonlinear Equations*; SIAM: Philadelphia, PA, 1995.
(57) Nocedal, J.; Wright, S. *Numerical Optimization*; Springer: New York, 2006.
(58) Pulay, P. Convergence Acceleration Of Iterative Sequences - The Case Of Scf Iteration. *Chem. Phys. Lett.* **1980**, *73*, 393−398.
(59) Blaha, P.; Schwarz, K.; Madsen, G. K. H.; Kvasnicka, D.; Luitz, J. *WIEN2k, An Augmented Plane Wave + Local Orbitals Program for Calculating Crystal Properties*; Universität Wien: Vienna, Austria, 2001.
(60) Perdew, J. P.; Burke, K.; Ernzerhof, M. Generalized gradient approximation made simple. *Phys. Rev. Lett.* **1996**, *77*, 3865−3868.
(61) Desclaux, J. P. Multiconfigurational Relativisitic Dirac-Fock Program. *Comput. Phys. Commun.* **1975**, *9*, 31−45.
(62) The structures used in Figures 1, 3−4, 7−10 are provided as CIF files in the Supporting Information.
(63) Laskowski, R.; Blaha, P. Unraveling the structure of the h-BN/Rh(111) nanomesh with ab initio calculations. *J. Phys.: Condens. Matter* **2008**, *20*, 064207.
(64) Laskowski, R.; Blaha, P. Ab initio study of h-BN nanomeshes on Ru(001), Rh(111), and Pt(111). *Phys. Rev. B* **2010**, *81*, 075418.
(65) Bjorkman, T.; Granas, O. Adaptive Smearing for Brillouin Zone Integration. *Int. J. Quantum Chem.* **2011**, *111*, 1025−1030.
(66) Custodio, A. L.; Dennis, J. E.; Vicente, L. N. Using simplex gradients of nonsmooth functions in direct search methods. *IMA J. Numer. Anal.* **2008**, *28*, 770−784.
(67) Rohwedder, T.; Schneider, R. An analysis for the DIIS acceleration method used in quantum chemistry calculations. *J. Math. Chem.* **2011**, *49*, 1889−1914.
(68) Bertseka, D. P. *Constrained Optimization and Lagrange Multiplier Methods*; Athena Scientific: Noshua, NH, 1996.
(69) Bertsekas, D. P. *Nonlinear Programming*, 2nd ed.; Athena Scientific: Nashua, NH, 2004; p 780.
(70) Baldereschi, A.; Tosatti, E. Dielectric Band-Structure Of Solids. *Solid State Commun.* **1979**, *29*, 131−135.
(71) Car, R.; Tosatti, E.; Baroni, S.; Leelaprute, S. Dielectric Band-Structure Of Crystals - General-Properties And Calculations For Silicon. *Phys. Rev. B* **1981**, *24*, 985−999.
(72) Ho, K. M.; Ihm, J.; Joannopoulos, J. D. Dielectric Matrix Scheme for Fast Convergence in Self-Consistent Electronic-Structure Calculations. *Phys. Rev. B* **1982**, *25*, 4260−4262.
(73) Hybertsen, M. S.; Louie, S. G. Abinitio Static Dielectric Matrices From The Density-Functional Approach 0.1. Formulation And Application To Semiconductors And Insulators. *Phys. Rev. B* **1987**, *35*, 5585−5601.
(74) Castro, J. D. E.; Muniz, R. B.; Oliveira, L. E. A Model Calculation Of The Dielectric Band-Structure Of Diamond And Silicon. *Phys. Status Solidi B* **1990**, *158*, 743−752.
(75) Galamic-Mulaomerovic, S.; Hogan, C. D.; Patterson, C. H. Eigenfunctions of the inverse dielectric functions and response functions of silicon and argon. *Phys. Status Solidi A* **2001**, *188*, 1291−1296.
(76) Wilson, H. F.; Lu, D. Y.; Gygi, F.; Galli, G. Iterative calculations of dielectric eigenvalue spectra. *Phys. Rev. B* **2009**, *79*, 245106.
(77) Baroni, S.; de Gironcoli, S.; Dal Corso, A.; Giannozzi, P. Phonons and related crystal properties from density-functional perturbation theory. *Rev. Mod. Phys.* **2001**, *73*, 515−562.

(78) Schwarz, K.; Blaha, P.; Trickey, S. B. Electronic structure of solids with WIEN2k. *Mol. Phys.* **2010**, *108*, 3147−3166.
(79) Blaha, P.; Hofstatter, H.; Koch, O.; Laskowski, R.; Schwarz, K. Iterative diagonalization in augmented plane wave based methods in electronic structure calculations. *J. Comput. Phys.* **2010**, *229*, 453−460.
(80) Shanno, D. F.; Phua, K. H. Matrix Conditioning and Non-Linear Optimization. *Math. Program.* **1978**, *14*, 149−160.
(81) Novak, P.; Kunes, J.; Chaput, L.; Pickett, W. E. Exact exchange for correlated electrons. *Phys. Status Solidi B* **2006**, *243*, 563−572.
(82) Tran, F.; Blaha, P.; Schwarz, K.; Novak, P. Hybrid exchange-correlation energy functionals for strongly correlated electrons: Applications to transition-metal monoxides. *Phys. Rev. B* **2006**, *74*, 155108.
(83) Moreau, J. Fonctions convexes duales et points proximaux dans un espace hilbertien. *Acad. Sci. Paris* **1962**, *C.R. 255*, 2897−2899.
(84) Tikonov, A. N. Solution of incorrectly formulated problems and the regularization method. *Dokl. Akad. Nauk SSSR* **1963**, *151*, 501−504.
(85) Hoerl, A. E.; Kennard, R. W. Ridge Regression - Biased Estimation for Nonorthogonal Problems. *Technometrics* **1970**, *12*, 55−67.
(86) Hansen, P. C. *Rank-Deficient and Discrete Ill-Posed Problems*; SIAM: Philadelphia, PA, 1998.
(87) Golub, G. H.; Heath, M.; Wahba, G. Generalized Cross-Validation as a Method for Choosing a Good Ridge Parameter. *Technometrics* **1979**, *21*, 215−223.
(88) Dennis, J. E.; Mei, H. H. W. 2 New Unconstrained Optimization Algorithms Which Use Function And Gradient Values. *J. Optimiz. Theor. Appl.* **1979**, *28*, 453−482.
(89) Dennis, J. E.; More, J. J. Quasi-Newton Methods, Motivation And Theory. *SIAM Rev.* **1977**, *19*, 46−89.
(90) Dennis, J. E.; Gay, D. M.; Welsch, R. E. An Adaptive Non-Linear Least-Squares Algorithm. *ACM Trans. Math. Softw.* **1981**, *7*, 348−368.
(91) Dennis, J. E.; Gay, D. M.; Welsch, R. E. Algorithm 573 - Nl2sol - An Adaptive Non-Linear Least-Squares Algorithm E4. *ACM Trans. Math. Softw.* **1981**, *7*, 369−383.
(92) Rondinelli, J.; Bin, D.; Marks, L. D. Enhancing structure relaxations for first-principles codes: an approximate Hessian approach. *Comput. Mater. Sci.* **2007**, *40*, 345−353.
(93) Ciston, J.; Subramanian, A.; Marks, L. D. Water-driven structural evolution of the polar MgO (111) surface: An integrated experimental and theoretical approach. *Phys. Rev. B* **2009**, *79*, 085421.
(94) Wang, Y. M.; Warschkow, O.; Marks, L. D. Surface evolution of rutile TiO_2 (100) in an oxidizing environment. *Surf. Sci.* **2007**, *601*, 63−67.
(95) Becerra-Toledo, A. E.; Enterkin, J. A.; Kienzle, D. M.; Marks, L. D. Water adsorption on SrTiO3(001): II. Water, water, everywhere. *Surf. Sci.* **2012**, *606*, 791−802.
(96) Raczkowski, D.; Canning, A.; Wang, L. W. Thomas-Fermi charge mixing for obtaining self-consistency in density functional calculations. *Phys. Rev. B* **2001**, *6412*, 121101.
(97) This is an approximate number based upon tests by the author and discussions with P. Blaha.