
Density Mixing Notes

Nick Woods

February 2017

Contents

Contents	i
1 Introduction	1
2 Theory	2
2.1 The Problem	2
2.2 Characterisation of the Solutions	3
2.3 Improving the SCF Process	9
2.3.1 Advanced Density Mixing	9
2.3.1.1 Newton's Method	10
2.3.1.2 Broyden's Methods	11
2.3.2 Preconditioning and Dielectric Models	14
3 Application	16
3.1 CASTEP	16
References	17

Introduction

Theory

2.1 The Problem

Solving a set of non-linear equations is the final step in the solution of many practical problems in Physics. This can be formulated as finding the ‘fixed point’ of a multidimensional function

$$\mathbf{g}(\mathbf{x}^*) = \mathbf{x}^*, \quad (2.1)$$

or recast as a root finding problem

$$\mathbf{f}(\mathbf{x}^*) = \mathbf{g}(\mathbf{x}^*) - \mathbf{x}^* = \mathbf{0} \quad (2.2)$$

i.e. find $\mathbf{x}^* \in \mathbb{R}^n$ such that the function $\mathbf{f}: \mathbb{R}^n \rightarrow \mathbb{R}^n$, defined above, is zero. Fixed point iteration theory states that for a sufficiently well behaved (continuous, etc.) function and input argument, the iterative process $\mathbf{g}(\mathbf{x}_n) = \mathbf{x}_{n+1}$ will converge with some order of convergence. The convergence (both speed and success) of a less well behaved operation than the \mathbf{g} above can be improved beyond fixed point iterations if \mathbf{x}_{n+1} is chosen ‘well’. That is, if one chooses $\mathbf{x}_{n+1} = \mathbf{h}(\mathbf{g}(\mathbf{x}_n^{\text{in}}) = \mathbf{x}_n^{\text{out}}, \mathbf{x}_n^{\text{in}})$ with \mathbf{h} a function defined to assist convergence toward Eq. (2.2). All methods below seek to find the form of \mathbf{h} such that Eq. (2.2) can be achieved, and be achieved as fast as possible in the specific case of Kohn-Sham DFT.

The above can now be rearranged in terms of Kohn-Sham (KS) DFT. The KS Hamiltonian, H^{ks} , solves an eigenproblem,

$$H^{\text{ks}}\psi_i = \epsilon_i\psi_i. \quad (2.3)$$

However, one finds that the solutions, ψ_i , depend on the input, H^{ks} , through the electron density – $H^{\text{ks}} = H^{\text{ks}}[\rho]$ where,

$$\rho(\mathbf{r}) = \sum_{i \in \text{occupied}} |\psi_i(\mathbf{r})|^2. \quad (2.4)$$

Hence, the above process (Eq. (2.3) and Eq. (2.4)) can be compacted into a *self-consistent field operator* F , satisfying

$$F[\rho^{\text{in}}] = \rho^{\text{out}}. \quad (2.5)$$

The ρ^* that satisfies both Eq. (2.3) and Eq. (2.4), i.e. $\rho^{\text{in}} = \rho^{\text{out}}$, is the electron density that solves KS DFT for a given external potential, system size, etc. One now seeks the function $\mathbf{h}(\rho^{\text{in}}, \rho^{\text{out}})$ such that $\rho^{\text{in}} = \rho^{\text{out}}$ is satisfied (within some tolerance) as quickly and robustly as possible – ‘*density mixing*’. The corresponding function to \mathbf{f} in Eq. (2.2) is now defined as the *residual* – $R[\rho(\mathbf{r})] = \rho^{\text{in}} - \rho^{\text{out}}$, where \mathbf{h} drives R toward zero, thus finding the ground state electron density ρ^* ($R[\rho^*] = 0$). A few ‘problem dependent’ properties can now be noted: firstly, the evaluation of the operation F is extremely costly, as it involves (iterative) diagonalisation of an $N_{\text{pw}} \times N_{\text{pw}}$ matrix. This means one would favour a density mixing scheme which converges in a low iteration count, at the cost of many computations per iteration (to an extent). Furthermore, the analytic form of the derivative $\partial_\rho F$ is not readily available, meaning some finite-differenced numerical approximation will be often need to be computed (atleast for the first step in an iterative process). This derivative – the Jacobian – is conventionally stored in a matrix which is prohibitively large (size dependent on the discretisation of the space), both in terms of computation cost and in storage requirements (when applying some mixing scheme). One therefore seeks a density mixing scheme which can be executed with limited memory requirements (i.e. not storing such large matrices), and converges in a low number of SCF cycles. With this in mind, the difficulties in applying some of the ‘naive’ approaches to the SCF process can be outlined.

2.2 Characterisation of the Solutions

The simplest update to the density would be to treat the above system as a fixed point iteration, feeding the output density back in as the input density by computing $\rho_{n+1} = F[\rho_n]$ – this turns out to be unsuitable for a variety of reasons. To elaborate, first note that the iterates will take the form $\rho_{n+1}^{\text{in}} = \rho_n^{\text{out}} = F[\rho_n^{\text{in}}]$. In practice, the initial guess of the density is not far from the converged density by simply modelling atoms at \mathbf{R}_μ as pseudoatomic orbitals of charge at \mathbf{R}_μ . This can be seen in Fig. 2.1, where a bulk FCC Al structure took 5 SCF iterations to converge within tolerance. By-eye inspection reveals that the initial guess does not change much from the converged density, and in fact after only one iteration the densities become visually indistinguishable as plotted.

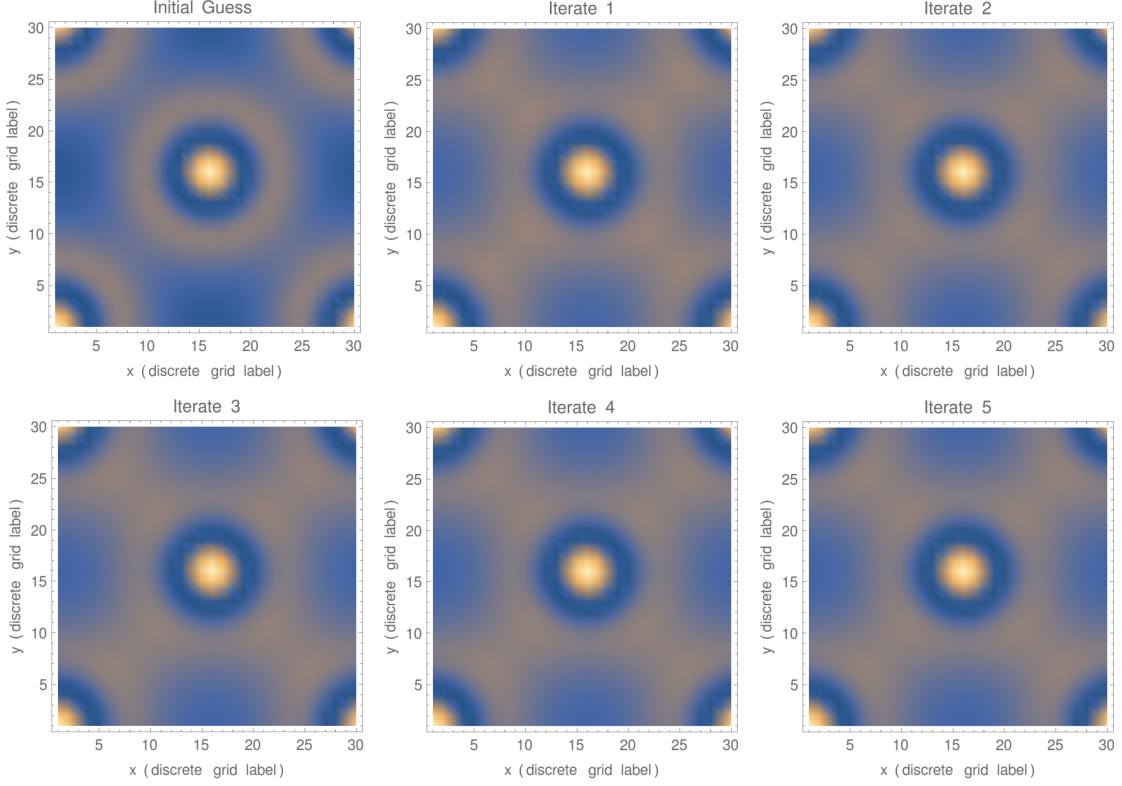


FIGURE 2.1: SCF convergence of FCC aluminium in CASTEP using Pulay mixing.

If it is the case that the $\rho^{\text{initial guess}} \sim \rho^*$ the operation F can be linearised about ρ^* ,

$$\rho^* + \delta\rho_n^{\text{out}} = F[\rho^* + \delta\rho_n^{\text{in}}] \approx F[\rho^*] + \left. \frac{\delta F}{\delta \rho} \right|_{\rho^*} \delta\rho_n^{\text{in}} = \rho^* + \left. \frac{\delta F}{\delta \rho} \right|_{\rho^*} \delta\rho_n^{\text{in}}, \quad (2.6)$$

$$\implies \delta\rho_n^{\text{out}} = \left. \frac{\delta F}{\delta \rho} \right|_{\rho^*} \delta\rho_n^{\text{in}}, \quad (2.7)$$

$$\implies \delta\rho_{n+1}^{\text{in}} = \left. \frac{\delta F}{\delta \rho} \right|_{\rho^*} \delta\rho_n^{\text{in}}. \quad (2.8)$$

Any use of Eq. (2.7) denotes a density mixing scheme in the *linear response regime* – i.e. it is assumed that small perturbations to the input charge density of KS DFT depend linearly on the output charge density. In general this clearly doesn't hold, but as mentioned, for charge densities sufficiently close to convergence, the relation is a good approximation. $F : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ and $\rho \in \mathbb{R}^3$, so the derivative term at a point \mathbf{r}, \mathbf{r}' in real space (ignoring any discretisation of the domain) will be a matrix $\frac{\delta F}{\delta \rho} \sim M \in \mathbb{R}^{3 \times 3}$. It will later be shown that M is closely related to the DFT dielectric response of the input system. From a mathematical perspective, one can ask what the conditions on M are to guarantee convergence. Convergence is defined as $\delta\rho_{n+1}^{\text{in}} \rightarrow 0$ as $n \rightarrow \infty$, or

$$M\delta\rho_n^{\text{in}} = (M)^n \delta\rho_1^{\text{in}} \rightarrow 0. \quad (2.9)$$

It will be assumed here (although it can be shown [1]) that M is of full rank, and can be diagonalised using $\{\mathbf{e}_i\}$ as an orthonormal basis – $M_{ij}^d = \delta_{ij}\lambda_i|\mathbf{e}_i\rangle\langle\mathbf{e}_j|$ with $\{\lambda_i\}$ the spectrum of M ¹. Since $\langle\mathbf{e}_i|\mathbf{e}_j\rangle = \delta_{ij}$,

$$((M^d)^n)_{ii} = (\lambda_i)^n|\mathbf{e}_i\rangle\langle\mathbf{e}_i|, \quad (2.10)$$

$$\implies M\delta\rho_n^{\text{in}} \rightarrow 0 \text{ iff } |\lambda_i| < 1 \ \forall i. \quad (2.11)$$

The question now becomes, for typical DFT input systems, is the condition $|\lambda_i| < 1$ always satisfied? The answer is no, and for some illustrative models studied in Ref. [1] $|\lambda_i| \sim 100$, causing a strong divergence from ρ^* . Clearly a simple fixed point iteration-esque update to ρ_{n+1} is unsuitable in general DFT calculations. It seems natural to now ask how one could incorporate a parameter to suppress $|\lambda_i|$ such that it satisfies the convergence criteria for any system of interest. This leads to the most simple density mixing scheme able to converge a (not-so-wide) variety of input systems – *linear mixing*. Instead of using $\rho_{n+1}^{\text{in}} = \rho_n^{\text{out}}$, one incorporates a damping parameter α as such,

$$\rho_{n+1}^{\text{in}} = \rho_n^{\text{in}} + \alpha(\rho_n^{\text{out}} - \rho_n^{\text{in}}) = \rho_n^{\text{in}} + \alpha R[\rho_n^{\text{in}}]. \quad (2.12)$$

It can be seen that, defining convergence of $\delta\rho$ similarly to above, the convergence criteria now becomes $|1 + \alpha(\lambda_i - 1)| < 1$. As α can be arbitrarily tuned such that this is true, the problem has been (superficially) solved. Supposing M is positive definite (i.e. not worrying about the case of $\lambda_{\min} < 0$), α must be chosen such that $|1 + \alpha(\lambda_{\max} - 1)| < 1$, meaning ρ is guaranteed to converge for all λ_i . However, the speed of convergence is related to how close $|1 + \alpha(\lambda_{\max} - 1)|$ is to unity – if it is only slightly less, $|1 + \alpha(\lambda_{\max} - 1)|^n$ will take large n to reach zero within tolerance. Moreover, if α is too low, the change in the ρ_{n+1} becomes minuscule per iteration for eigenvalues at or close to λ_{\min} , leading to slow convergence. Therefore, an optimal value of α must be deduced, but how efficacious this choice is clearly depends on the ratio $\frac{\lambda_{\max}}{\lambda_{\min}}$ – an important quantity in numerical analysis, the *condition number* of M . In general M is not well conditioned, e.g. for metallic materials the condition number of M turns out to be divergent proportional to the size of the unit cell [source?], and hence certain systems can take large amounts of time to converge (to the point where a user would say the calculation doesn't converge in a practical sense).

The above discussion on the convergence of fixed point iterations and linear mixing was mathematical in nature, but the lack of convergence of both methods can also be interpreted physically in a process dubbed ‘charge sloshing’. One starts by interpreting $M = \frac{\delta F}{\delta \rho}$ in Eq. (2.7). The functional derivative of the operator F has been investigated by multiple sources [1, 2], and it can be shown that M relates to the dielectric constant

¹Here, M^d is a transformed matrix sharing some of the same properties of M , namely, $M^n \rightarrow 0$ iff $(M^d)^n \rightarrow 0$.

matrix ϵ in the following way,

$$M = \mathbf{I} - \epsilon^{\text{DFT}}. \quad (2.13)$$

The dielectric describes how the electrons in a given system will respond to an (applied) perturbation in the electron density $-\frac{\delta\rho^{\text{out}}}{\delta\rho^{\text{in}}}$. Applying a perturbation to the electron density amounts to mimicking the application an external electric field, and measuring the response of the electrons (which is the classical electromagnetism definition of a dielectric). It will be shown below that knowing the dielectric of the input system can greatly improve convergence to the ground state, firstly however a subtle but important distinction can be made. When one says ‘the dielectric of the input system’, what is meant by this is the dielectric of the input system *as it would be in* KS DFT. The idea of density mixing is to converge KS DFT to its ground state as quickly as possible, and the dielectric that is most efficacious in achieving this is not the dielectric of the real material, but the dielectric as it would be in KS DFT². This fact becomes not so subtle when the dielectric (or some other mathematical object designed to assist convergence) requires the *band gap* as a parameter. One might initially (correctly) identify that DFT systematically underestimates the band gap, and then (incorrectly) deduce that convergence processes relying on the band gap are unsuitable (or at the very least non-optimal). However, it is precisely the DFT band gap that is required for optimal performance, as density mixing is a numerical technique designed to extract the correct output answer from the theory.

Analysis in [3] confirms that *at the fixed point* ρ^* , ϵ (and therefore M) is positive definite as presumed (which removes some divergence possibilities, allowing α to remain positive). Breaking down ϵ component by component reveals that perturbations in the charge density and *Hartree* potential³ affect one-another as so,

$$\delta V_h(\mathbf{r}) = \int \frac{\delta\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}', \quad (2.14)$$

$$\delta\rho(\mathbf{r}) = \int \chi^{\text{DFT}}(\mathbf{r}, \mathbf{r}') \delta V_h(\mathbf{r}') d\mathbf{r}'. \quad (2.15)$$

In this context, χ^{DFT} is the independent (DFT) electron *susceptibility* – a real, symmetric matrix characterising how ‘susceptible’ the electrons in an input system are to changes in the potential, $\chi^{\text{DFT}} \sim \frac{\delta\rho}{\delta V_h}$. One can see that M will pick up amplified contributions from changes in the charge density at long range from Eq. (2.14). In Fourier space, Eq. (2.14) takes the form,

$$\delta\tilde{V}_h(\mathbf{G}) \sim \frac{\delta\tilde{\rho}(\mathbf{G})}{|\mathbf{G}|^2}. \quad (2.16)$$

²Obviously these are ideally the same, but it is unfortunately not always the case.

³Ignoring the exchange-correlation component for now, as it does not affect the ensuing analysis.

Hence, finite changes in the density at short wavelength Fourier modes ($\delta\rho$ at long range in real space) will be amplified by the factor of $|G|^{-2}$. This then has a back-reaction on the iteratively updated output density, Eq. (2.15). Depending on the susceptibility, a large change will be incurred in $\delta\rho_{n+1}^{\text{out}}$, and the system will fail to converge. The dependence on χ (and therefore ϵ) is the link between the mathematical analysis above, and the discussion here. If the maximum eigenvalue of $U\chi - U$ being the kernel of the Hartree potential – is too large (or the spectrum is not dense in the linear mixing case), the electrons are ‘too’ susceptible to changes in the potential causing divergence (or extremely slow convergence). The physical picture being that, when the material is ‘too’ susceptible (relating to the eigenvalues of χ) a small change in the Hartree potential (for example the one induced in the mixing schemes) causes a large change in the charge density from low $|G|$ wavevectors. This in turn back-reacts onto V_h trying to accommodate for the change in ρ , which sets up a ‘back-and-forth’ of continuous overcorrection of ρ and V_h – sloshing. Unfortunately, charge sloshing is present in all mixing algorithms to an extent, where the effect is more pronounced in high χ materials, like metals, and particularly when the unit cell (and charge inside the cell) is large⁴. In reference to the earlier discussion, the reason why Ref. [1] finds $\lambda_i \sim 100$ is precisely because they are modelling a metallic system whose electrons respond much more freely to the ‘applied field’ in contrast to an insulating system.

Charge sloshing can be seen in action in Fig. 2.2. A graphene nanoribbon consisting of 12 atoms was attempted to be converged using linear mixing (and for reference, actually converged using Pulay mixing in Fig. 2.3). This system is (semi-)metallic, and has a large unit cell in the plotted direction (across the ribbon) as it includes 10\AA of vacuum space. It can be seen that the initial guess, as expected, is close to the converged answer (when comparing to the Pulay case). However, after just two iterations, large amounts of charge starts getting displaced per iteration. The mixing algorithm attempts to move charge to the edge of the unit cell, and then overcorrects in the next iteration by moving all charge to the centre of the unit cell. What is left by iteration five does not resemble the input system at all, and hence the instability has taken over and linear mixing has failed to converge.

Another source of what one could interpret as a sloshing instability is related to how KS orbital occupations are determined for systems where the Fermi energy crosses the bands (metallic systems again, but for differing reasons to above). In a DFT calculation the software will converge the total energy,

$$E_{\text{tot}} = \sum_{i \in \text{occupied}} \epsilon_i - \text{double counting}, \quad (2.17)$$

⁴It seems intuitive that for more charge present over longer ranges, the sloshing will be more pronounced.

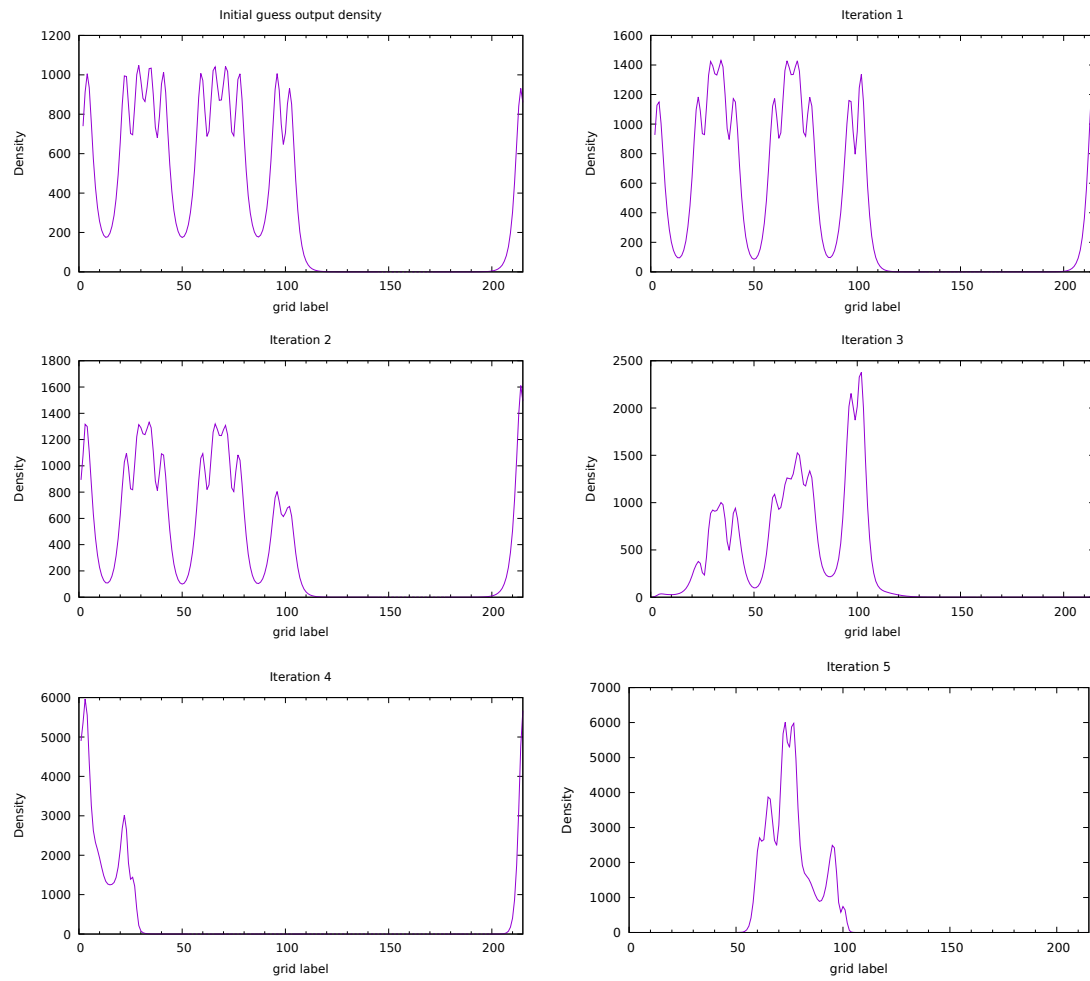


FIGURE 2.2: Electron densities at subsequent iterations of the SCF process for a 12 atom ZGNR using linear mixing.

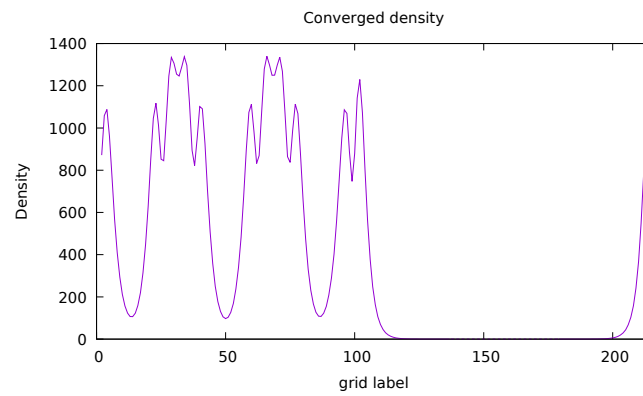


FIGURE 2.3: Converged electron density for a 12 atom ZGNR using Pulay mixing.

where occupation refers to the electrons in the system occupying specific single particle KS orbitals ϕ_i with energy ϵ_i such that E_{tot} is minimised. In semiconductors and insulators, determination of the occupation is simple: it is discontinuous and the lowest energy N orbitals are filled with electrons, and any bands above E_f (which are completely separate to bands below E_f) have their corresponding orbitals unoccupied. In metals however, the bands cross at E_f and the act of choosing (discontinuously) one of the orbitals to be fully occupied, and the other to be not occupied, can raise the energy of the occupied band over the previously unoccupied one in the next SCF iteration. The DFT program will then choose the previously unoccupied orbital to now be the occupied orbital, and thus the process repeats and convergence of E_{tot} may never be reached. This form of sloshing is less serious than the sloshing as a product of V_h and χ , as one can introduce a fix in the form of *partial occupancy*. In implementing this, the occupancy of orbitals with energies around E_f are no longer discontinuous, but smeared according to some ‘finite temperature’ distribution f_i (e.g. Fermi-Dirac), allowing E_{tot} to be converged much more systematically and reliably by the code.

This first form of sloshing is atleast partially solved by the use of a model dielectric. In practice, the model dielectric matrix $\sim M$ is applied to the electron density in Fourier space, and is specifically designed to weight changes of ρ at low $|G|$ less, heavily damping the sloshing effect. In mathematical terms, the dielectric matrix acts as a *preconditioner*, which takes the problem at hand (i.e. driving $\rho^{\text{guess}} \rightarrow \rho^*$) and changes the problem (by application of the preconditioner) to make it more suitably solved by the mixing algorithm.

The above analysis has highlighted two key considerations when seeking to improve the SCF cycles in KS DFT – the choice of mixing algorithm, and the choice of preconditioner. It is common in practice to apply the preconditioner at each iteration of the mixing algorithm (specified by **h**). Now the base theory has been laid out, one can begin to seek so-called *advanced mixing* algorithms to assist convergence, and improved preconditioners/dielectric models.

2.3 Improving the SCF Process

2.3.1 Advanced Density Mixing

Firstly a range of mixing methods will be introduced and detailed in (almost) chronological order, with the goal of arriving at the methods currently implemented in CASTEP. The methods below can be categorised into two broad classes: *direct inversion of the iterative subspace* (DIIS) – a method using a history of iterates to perform a least squares weighting of the subsequent iterate in order to achieve zero ‘error’ – and quasi-Newton

methods. The foundation of quasi-Newton techniques is the Newton-Raphson method for approximating roots, which will first be outlined.

2.3.1.1 Newton's Method

As discussed, the SCF solution of KS DFT is a root finding problem, requiring $R[\rho^*] = 0$. Reintroducing the mathematical notation from the start of Section. 2.1, Newton's method for finding \mathbf{x}^* such that $\mathbf{f}(\mathbf{x}^*) = 0$ starting from an initial guess \mathbf{x}_0 will be derived. Firstly, the assumption that \mathbf{x}_0 is close to the root will be made, such that the addition of a small pertubation vector $\mathbf{e} \in \mathbb{R}^n$ yields the root, $\mathbf{f}(\mathbf{x} + \mathbf{e}) = 0$. The goal is to therefore find the \mathbf{e} that satisfies this equation, or in practice, an \mathbf{e} that will drive subsequent values of \mathbf{x} closer to satisfying it. To find this \mathbf{e} , the Taylor expansion of $f_i(\mathbf{x} + \mathbf{e})$ about \mathbf{x} is computed,

$$f_i(\mathbf{x} + \mathbf{e}) = f_i(\mathbf{x}) + \sum_j \frac{\partial f_i(\mathbf{x})}{\partial x_j} h_j + \mathcal{O}(\|\mathbf{e}\|^2), \quad (2.18)$$

$$\mathbf{f}(\mathbf{x} + \mathbf{e}) = \mathbf{f}(\mathbf{x}) + J_f(\mathbf{x})\mathbf{e} + \mathcal{O}(\|\mathbf{e}\|^2). \quad (2.19)$$

where $J_f(\mathbf{x})$ is the Jacobian matrix of f at \mathbf{x} - $J_f(\mathbf{x})_{ij} = \frac{\partial f_i}{\partial x_j}$. The larger $\|\mathbf{e}\|$ is, the further away the Taylor expansion will deviate from the exact evaluation of the function. Ideally, the initial \mathbf{e} that is calculated will bring \mathbf{x} much closer to the root (provided it was 'close' to the root to begin with), but will over or undershoot due to the first order nature of the Taylor expansion. Taylor expansions are then repeatedly done as \mathbf{e} (the absolute error in $\mathbf{x} - \mathbf{x}^*$) tends to zero. This defines the self-consistent process to find \mathbf{x}^* from a suitable initial value,

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \mathbf{e}. \quad (2.20)$$

Where \mathbf{e} is obtained from Eq. (2.19) via

$$\mathbf{f}(\mathbf{x} + \mathbf{e}) = \mathbf{0} \implies \mathbf{e} = -J_f^{-1}(\mathbf{x})\mathbf{f}(\mathbf{x}), \quad (2.21)$$

producing

$$\mathbf{x}_{n+1} = \mathbf{x}_n - J_f^{-1}(\mathbf{x})\mathbf{f}(\mathbf{x}). \quad (2.22)$$

This the famous Newton-Raphson update formula, which translated into the language of KS DFT looks like,

$$\rho_{n+1}^{\text{in}}(\mathbf{r}) = \rho_n^{\text{in}}(\mathbf{r}) - J_R^{-1}[\rho_n^{\text{in}}(\mathbf{r})](\rho_n^{\text{out}}(\mathbf{r}) - \rho_n^{\text{in}}(\mathbf{r})). \quad (2.23)$$

The appeal of (quasi-)Newton methods lies mostly in the order of convergence. It can be shown [proof?] that if there exists an interval I where \mathbf{x}_0 is ‘sufficiently close’ to \mathbf{x}^* , \mathbf{f}' is bounded away from zero and \mathbf{f}'' exists (and is continuous), the rate of convergence of the Newton method is *quadratic*⁵. Unfortunately, Newton’s method as it appears here is unsuitable for KS DFT. The Jacobian is costly to compute⁶, and is stored in an $N_{\text{gridpoints}} \times N_{\text{gridpoints}}$ matrix. For even just a few iterations of KS DFT, the computational cost and storage requirements of J make this method not feasible for practical implementation in planewave codes. The computational cost problem can be greatly alleviated by considering approximate updates of J at each iteration, instead of full reconstruction of J (ideally without much loss of convergence properties) – Broyden’s methods.

2.3.1.2 Broyden’s Methods

Broyden’s methods for non-linear root finding were first published in 1965 [4], yet the concepts and machinery he proposed remain foundational to many methods in practical use today. Broyden supposes that instead of recomputing the exact Jacobian matrix at every iteration, one can construct the Jacobian at only the initial iteration, and update it at every subsequent iteration in a way that preserves its convergence properties and saves the computational expense of reconstructing it in full. Since density mixing only induces relatively small changes in ρ , the changes in the Jacobian will be (in some sense) small per iteration. This fact lends itself naturally to asking the question; what is this ‘small’ update such that one can keep the near quadratic convergence of Newton by utilising data from previous iterations, but also avoid the computational expense of constructing Jacobian’s at every iteration?

Broyden proposed two methods for updating the Jacobian: method one (producing the so-called ‘good Broyden formula’) updates the Jacobian according to some assumptions and *then* inverts it as is required in Newton’s formula, whereas the second (‘bad’) method updates the inverse of the Jacobian explicitly under similar assumptions. The first method will be detailed below, whereas the second will simply be stated as it follows a similar methodology to the first. Supposing one knows the Jacobian $J_{R,n-1}[\rho]$, an update is required to find $J_{R,n}[\rho]$ such that

$$J_{R,n}[\rho] = J_{R,n-1}[\rho] + C_n \quad (2.24)$$

$$\rho_{n+1}^{\text{in}}(\mathbf{r}) = \rho_n^{\text{in}}(\mathbf{r}) - J_{R,n}^{-1}[\rho_n^{\text{in}}(\mathbf{r})](\rho_n^{\text{out}}(\mathbf{r}) - \rho_n^{\text{in}}(\mathbf{r})) \quad (2.25)$$

can be computed. This update C_n can be cleverly obtained by first enforcing a ‘secant’ condition – i.e. force $J_{R,n}[\rho]$ to obey a finite-difference equation that is approximately

⁵As far as root finding methods go, quadratic convergence is better than most.

⁶Some finite differencing technique is required to build up $J_R(\rho)$ at each SCF iteration.

true, which manifestly uses known data. For a simple one dimensional system, the definition of the derivative of f at x_n is obtained by taking the following limit,

$$f'(x_n) = \lim_{x_{n-1} \rightarrow x_n} \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}. \quad (2.26)$$

Meaning if x_n and x_{n-1} are sufficiently close, the secant condition is the finite-difference approximation,

$$f'(x_n) \approx \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}. \quad (2.27)$$

Importantly, in this one dimensional case, $f'(x_n)$ is *uniquely* defined by Eq. (2.27) – this is not the case in $k > 1$ dimensions, which will be detailed below. Generalising now to the N dimensional system of KS DFT, the secant condition becomes,

$$J_{R,n}[\rho_n^{\text{in}}](\rho_n^{\text{in}} - \rho_{n-1}^{\text{in}}) = R[\rho_n^{\text{in}}] - R[\rho_{n-1}^{\text{in}}]. \quad (2.28)$$

If J is a $k \times k$ matrix of full rank, it has k linearly independent basis vectors forming a k -dimensional vector space, $\{e_i\} \in V$. Enforcing the secant condition in Eq. (2.28) specifies how J_n should act on all vectors parallel to $\rho_n^{\text{in}} - \rho_{n-1}^{\text{in}} \in \mathbb{R}^k$ – corresponding to one basis vector in an appropriately rotated basis. Linear algebra now tells us that there is a $(k-1)$ -dimensional sub-vector space $V' \subset V$ consisting of all elements of the vector space V orthogonal to $\rho_n^{\text{in}} - \rho_{n-1}^{\text{in}}$: $V' = \{z \mid z \in \mathbb{R}^k, (\rho_n^{\text{in}} - \rho_{n-1}^{\text{in}}) \cdot z = 0\}$. Importantly, Eq. (2.28) does *not* specify how J_n acts on elements of V' , of which there are $k-1$ linearly independent members. This means there exists k simultaneous equations to determine the k^2 degrees of freedom in J_n . The update therefore remains underdetermined with information provided only by secant, and by enforcing the secant condition we have effectively gained *one rank* of information, leaving $k-1$ ranks of information unknown. Up to this point, what has been defined is a class of Broyden updates – i.e. all rank one updates to the Jacobian that satisfy the most recent iterative secant condition (of which there are infinitely many). Now a particular method Broyden proposed within this class can be outlined; known as ‘Broyden 1 (B1)’ or ‘Broyden’s good method’.

To recover the missing information, Broyden supposes J_n acts on elements of V' the same way as J_{n-1} does

$$J_{R,n}[\rho_n^{\text{in}}]z = J_{R,n-1}[\rho_{n-1}^{\text{in}}]z \quad \forall z \in V'. \quad (2.29)$$

This is a sort of ‘minimal change’ condition on J_n , motivated by the fact that changes in ρ are small per iteration, and so changes in J_n will follow suit. These two conditions (minimal change and secant) provide a unique solution to the update matrix C_n which is crucially of rank one (from the secant information).

To condense, B1 seeks the update matrix C_n under two conditions,

$$1. J_{R,n}[\rho_n^{\text{in}}](\rho_n^{\text{in}} - \rho_{n-1}^{\text{in}}) = R[\rho_n^{\text{in}}] - R[\rho_{n-1}^{\text{in}}] \rightarrow J_{R,n}\Delta\rho_n^{\text{in}} = \Delta R_n, \quad (2.30)$$

$$2. J_{R,n}z = J_{R,n-1}z. \quad (2.31)$$

One can show the following ansatz satisfies these conditions,

$$J_{R,n} = J_{R,n-1} + \frac{\Delta R_n - J_{R,n-1}\Delta\rho_n^{\text{in}}}{|\Delta\rho_n^{\text{in}}|^2}(\Delta\rho_n^{\text{in}})^T, \quad (2.32)$$

and thus provides a unique solution for C_n . The notation uv^T defines the *outer product* of $u, v \in \mathbb{R}^k$. Vectors are rank one objects, and it is a mathematical fact that the outer product of two vectors is a rank one matrix. Thus $J = J' + uv^T$ is called a *rank one update*. In order to show Eq. (2.32) satisfies conditions 1. and 2., first note that the outer product has the property that $uv^T x = u(v \cdot x)$. Operating with Eq. (2.32) on the vector z shows immediately, via this property, that 2. is satisfied as $\Delta\rho_n^{\text{in}}$ is defined orthogonal to z (zeroing the right most term). Proving 1. is simply a matter of operating now on $\Delta\rho_n^{\text{in}}$. $(\Delta\rho_n^{\text{in}})^T \cdot \Delta\rho_n^{\text{in}} = |\Delta\rho_n^{\text{in}}|^2$ which cancels the denominator, then $J_{R,n-1}\Delta\rho_n^{\text{in}}$ immediately cancels leaving the desired result.

It is worth noting that the Newton update formula requires the inverse Jacobian, not the Jacobian. An explicit formula for the update of the inverse of a matrix like $J_{R,n}$ can be given in terms of the components of $J_{R,n}$ in Eq. (2.32) via the Sherman-Morrison formula,

$$J_{R,n}^{-1} = J_{R,n-1}^{-1} + \frac{\Delta\rho_n^{\text{in}} - J_{R,n-1}^{-1}\Delta R_n}{\Delta\rho_n^{\text{in}T} J_{R,n-1}^{-1} \Delta R_n} \Delta\rho_n^{\text{in}T} J_{R,n-1}^{-1}. \quad (2.33)$$

The simplicity of this expression owes to the fact that the update of $J_{R,n-1}$ is of rank one – higher rank updates add algebraic (and hence computational) complexity to Eq. (2.33).

The above derivation of the B1 Jacobian update is equivalent to solving the following constrained optimisation problem,

$$\underset{J_{R,n-1} \in S}{\text{minimise}} \quad \|J_{R,n} - J_{R,n-1}\|_f^2 \quad (2.34)$$

where $\|\cdot\|_f$ is the *Frobenius norm* of a matrix,

$$\|A\|_f = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2}, \quad (2.35)$$

allowing one to formalise the notion of the ‘closeness’ of two matrices. The set $S = \{A \mid A \in \mathbb{R}^{k \times k}, A\Delta\rho_n^{\text{in}} = \Delta R_n\}$ imposes the secant condition. In words, one seeks to

find the update that minimises the Frobenius norm (difference) of $J_{R,n} - J_{R,n-1}$ subject to the secant condition, which turns out to be Eq. (2.32). Broyden's 'bad' method (B2) – dubbed 'bad' as he himself dismissed it as inferior to B1 in his work⁷ – is simply the same process involved in deriving B1, but instead minimising the Frobenius norm of $J_{R,n}^{-1} - J_{R,n-1}^{-1}$, yielding

$$J_{R,n}^{-1} = J_{R,n-1}^{-1} + \frac{\Delta\rho_n^{\text{in}} - J_{R,n-1}^{-1}\Delta R_n}{|\Delta R_n|^2} \Delta R_n^T. \quad (2.36)$$

do bad broyden, compare, marks and luke? but first a note on the convergence properties.

2.3.2 Preconditioning and Dielectric Models

some more background in DFT

why is it hard? expand on the DFT dependent difficulties to the solution of the fixed point problem.

the key foundational methods: Pulay, broyden 1 and 2, anderson, kerker.

note on skipping of spin so far.

where are dielectrics?

step by step improvements of each method until castep

Marks and luke...

Plan:

- Implement Levine Louie preconditioner in CASTEP ($\omega = 0$?)

find out what the form of the equation is – same as mathematica?

what is the other parameter in the model? E_g/E_f , and r_s ? compared to A, B (G0) in Kerker.

r_s related to q_f ? Fermi wave vector of non interacting electron gas of same density.

test systems?

- try to find out about multiseccant (history dependent) broyden methods

⁷As it turns out, B2 has some important advantages over B1 which will be elaborated on in further sections.

- find out the form of the equation from the marks and luke paper

what are all the vectors?

then implement

Application

3.1 CASTEP

References

- [1] P. H. Dederichs and R. Zeller. Self-consistency iterations in electronic-structure calculations. *Phys. Rev. B* 28, 5462, (1983).
- [2] X. Gonze. Towards a potential-based conjugate gradient algorithm for order-n self-consistent total energy calculations. *Phys. Rev. B* 54, 4383, (1996).
- [3] P.-M. Anglade and X. Gonze. Preconditioning of self-consistent-field cycles in density-functional theory: The extrapolar method. *Phys. Rev. B* 78, 045126, (2008).
- [4] C. G. Broyden. A class of methods for solving nonlinear simultaneous equations. *Math. Comp.* 19, 577-593, (1965).