

Computational Materials Science II

PROGRESS REPORT # I

Vourvachakis S. Georgios
mse354

March 9, 2025



This work is licensed under a Creative Commons “Attribution-NonCommercial-NoDerivatives 4.0 International” license.



CONTENTS

1	Introduction	3
2	Hohenberg-Kohn Theorems and Kohn-Sham equations	4
3	Density-based Mixing	9
3.1	Fixed Point Iteration scheme	10
3.1.1	Fixed Point Iteration experiments	11
3.2	Broyden, Kerker, Pulay mixers	13
4	Reciprocal Space and Periodicity	16
4.1	Numerical Integration experiments	16
5	Grid Spacing and Cutoffs	19
5.1	Symmetry consideration on Bulk Cu experiments	20
5.1.1	Space Group Symmetry Utilization	20
5.1.2	Symmetry-agnostic Execution	22
5.2	Comparative Analysis of Grid Spacing and Cutoff on Bulk Cu	23
6	Direct Minimization of KS Hamiltonian	25
6.1	Bisection Method: root-finding Binary Search	26
6.2	Newton-Raphson's Method: Curvature information	26
6.3	Optimization experiments	27
6.3.1	Bisection method Implementation	27
6.3.2	BFGS (Quasi-Newton) Implementation	29
7	Environment Setup	31
7.1	How to Modify the Path Settings	31
	References	32

INTRODUCTION

This review presents a comprehensive exploration of Density Functional Theory (DFT) as applied in computational materials science. We begin by laying the groundwork with the Hohenberg-Kohn Theorems through the Born-Oppenheimer approximation to Kohn-Sham equations and then transition to the computational aspects that allow us to simulate material properties using DFT.

The first major section of the report introduces the fundamental concepts and simulation parameters that form the basis of Density Functional Theory. This part begins by reviewing essential principles of quantum mechanics and aspects on variational optimization, establishing the theoretical backdrop for understanding material behavior at the quantum level. It then explains how computational materials science integrates these quantum principles into simulation techniques, with a particular focus on the transition from traditional wavefunction-based methods (rooted in the Schrödinger equation) to the electron density approach of Kohn-Sham equations. Additionally, the discussion covers the key simulation parameters such as k-points, plane-wave cut-off energies, and structural optimization protocols. These elements are critical for ensuring that DFT simulations accurately reflect the electronic structure and physical properties of simple elemental solids, thus bridging the gap between theory and practical computation.

The second major section delves into the computational techniques and hands-on exercises that illustrate the application of DFT. This part discusses iterative methods for computing electron density, including the role of density mixing and the fixed point theorem where the density is treated as a vector of independent variables. It further examines various mixing algorithms such as Broyden, Pulay, and Kerker mixers, which they are implemented extensively in GPAW and VASP softwares. A series of practical exercises—ranging from GPAW simulations of bulk aluminum and aluminum clusters to studies on bulk copper with and without symmetry—demonstrate how these techniques are applied in practice. Complementary topics include numerical integration for function continuity, the impact of grid spacing and plane-wave cut-offs on wavefunction accuracy, and the application of Fourier's theorem in expanding periodic functions. Finally, the section covers optimization techniques such as the bisection and Quasi-Newton methods, which are used to fine-tune structural parameters and achieve minimum energy configurations, thereby highlighting both the theoretical and computational intricacies of DFT simulations.

By beginning with the fundamental and computational aspects of DFT, this report not only provides theoretical context but also bridges the gap to practical simulation techniques, as demonstrated by the reproduced lecture exercises. The ensuing sections aim to equip the reader with both conceptual understanding and hands-on insights into the ab-initio simulation of materials.

HOHENBERG-KOHN THEOREMS AND KOHN-SHAM EQUATIONS

The *first Hohenberg-Kohn (HK) theorem* [HK64] asserts that the electron density, $\rho(\mathbf{r})$, uniquely determines the external potential, $v(r)$, which arises from the nuclei. Since $\rho(\mathbf{r})$ is normalized to yield the total number of electrons,

$$\int \rho(\mathbf{r}) d\mathbf{r} = N ,$$

it follows that N and $v(r)$ together uniquely specify the molecular Hamiltonian. In the *Born-Oppenheimer approximation*¹ and neglecting relativistic effects (with atomic units used throughout), this Hamiltonian is written as

$$H_{BO} \equiv - \sum_i^N \frac{1}{2} \nabla_i^2 - \sum_A^n \sum_i^N \frac{Z_A}{r_{iA}} - \sum_{i < j}^N \sum_j^N \frac{1}{r_{ij}} + \sum_{B < A}^n \sum_A^n \frac{Z_A Z_B}{R_{AB}} .$$

where the summations over i and j run over the electrons, and those over A and B run over the nuclei. Here, r_{ij} , r_{iA} , and R_{AB} represent the distances between electrons, between an electron and a nucleus, and between nuclei, respectively. Since H_{BO} governs the energy of the system via Schrödinger's equation,

$$H_{BO} \Psi = E \Psi ,$$

with Ψ being the electronic wave function, the electron density $\rho(\mathbf{r})$ ultimately determines the system's energy and all its ground-state electronic properties.

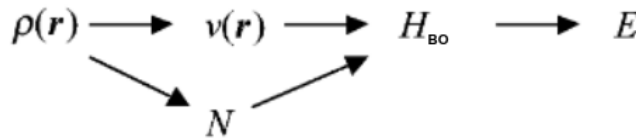


Figure 1: Interdependence of Basic Variables in the Hohenberg-Kohn Theorem [HK64].

In fact, the above scheme illustrates that the energy E is a functional of ρ , expressed as $E = E_v[\rho]$, where the subscript v highlights the explicit dependence on the external potential.

The proof of this renowned theorem is elegantly simple (for an intuitive approach, see Wilson as cited in Lowdin [Löw85]) and its impact has been profound.

¹Decoupled Electronic and nuclear terms/subsystems.

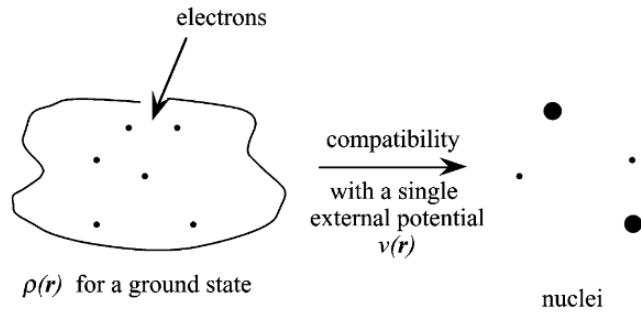


Figure 2: Schematic of the First Hohenberg-Kohn Theorem.

A schematic representation (fig.2) can be quite enlightening: if one were to present an observer with a plot of $\rho(\mathbf{r})$ corresponding to the ground-state electron density of an atom or molecule, the first HK theorem guarantees that this density uniquely determines the number of electrons N (via eq.2) and the specific arrangement of nuclei (their number, charges, and positions).

The *second HK theorem* provides a variational principle for determining $\rho(\mathbf{r})$. It states that the ground-state energy can be found by searching for the electron density that minimizes the energy functional. At the optimal $\rho(\mathbf{r})$, the energy E remains stationary with respect to variations in $\rho(\mathbf{r})$, subject to the constraint that the electron density always integrates to N (eq.2):

$$\delta \left(E - \mu \int \rho(\mathbf{r}) d\mathbf{r} \right) = 0,$$

where μ is the corresponding Lagrange multiplier.

This condition leads to

$$v(\mathbf{r}) + \frac{\partial F_{HK}}{\partial \rho(\mathbf{r})} = \mu,$$

where F_{HK} is the *Hohenberg-Kohn functional* that includes both the electronic kinetic energy functional, $T[\rho]$, and the electron-electron interaction functional, $V_{ee}[\rho]$.

Then, one can define the energy functional as

$$E_v[\rho] := \int v(\mathbf{r}) \rho(\mathbf{r}) d\mathbf{r} + F_{HK}[\rho]$$

with

$$F_{HK}[\rho] := T[\rho] + V_{ee}[\rho].$$

The Euler-Lagrange equation (eq.2) thus serves as the DFT analogue of Schrödinger's time-independent equation (eq.2). Since the Lagrange multiplier μ in eq.2 is independent of \mathbf{r} , the density $\rho(\mathbf{r})$ must be such that the left-hand side of eq.2 does not depend on \mathbf{r} . Notably, the functionals $T[\rho]$ and $V_{ee}[\rho]$ remain only partially known, representing ongoing challenges in the field.

The practical resolution of eq.2 was achieved by Kohn and Sham [KS65], who cleverly recast it into a form bearing a strong resemblance to the Hartree equations [HKS90]. This resemblance later enabled its integration into pre-existing wave-function-based software packages such as Gaussian [Fri+10] (cf. section I). Their strategy involved introducing orbitals in a manner that allowed the kinetic energy to be computed with both simplicity and good accuracy. They considered an N -electron non-interacting reference system described by the Hamiltonian:

$$H_{\text{ref}} \equiv - \sum_{i=1}^N \frac{1}{2} \nabla_i^2 + \sum_{i=1}^N v_i(\mathbf{r}) = \sum_i h_{\text{ref},i}$$

with

$$h_{\text{ref},i} \equiv -\frac{1}{2} \nabla_i^2 + v_i(\mathbf{r}),$$

which, by excluding electron–electron interactions, reproduces the exact electron density, $\rho(r)$, of the interacting system. By introducing the orbitals Ψ_i , which are eigenfunctions of the one-electron operator (eq.2), any physically acceptable density of the non-interacting system can be expressed as

$$\rho_s := \sum_i^N |\Psi_i|^2,$$

where the sum extends over the N lowest eigenstates of h_{ref} . Indeed, Harriman demonstrated by explicit construction that every non-negative, normalized density (i.e., all physically acceptable densities) can be represented as a sum of the squares of an arbitrary number of orthonormal orbitals [Har81].

The Hohenberg–Kohn functional, $F_{HK}[\rho]$, is then expressed as

$$F_{HK}[\rho] := T_s[\rho] + J[\rho] + E_{xc}[\rho].$$

Here, $T_s[\rho]$ denotes the kinetic energy functional of the reference system:

$$T_s[\rho] := \sum_i^N \left\langle -\frac{1}{2} \nabla_i^2 \right\rangle_{\Psi_i},$$

$J[\rho]$ is the classical Coulomb interaction energy:

$$J[\rho] := \frac{1}{2} \int \int \frac{\rho(\mathbf{r})\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r} d\mathbf{r}',$$

and the remaining energy contributions are gathered into the $E_{xc}[\rho]$ functional, which represents the *exchange–correlation energy*. This exchange–correlation term accounts for the difference between the exact kinetic energy and $T_s[\rho]$, includes the non-classical component of $V_{ee}[\rho]$, and corrects the self-interaction present in $J[\rho]$.

By combining eqs. 2, 2, 2, and 2, one can rewrite the Euler–Lagrange equation (eq. 2) as follows:

$$\mu = v_{\text{eff}}(\mathbf{r}) + \frac{\partial T_s}{\partial \rho},$$

noting that all derivatives with respect to $\rho(\mathbf{r})$ are taken while keeping the total number of electrons N fixed (this constraint is not explicitly indicated in the subsequent derivatives for simplicity). An effective potential, $v_{\text{eff}}(\mathbf{r})$, is introduced here,

$$\begin{aligned} v_{\text{ref}}(\mathbf{r}) &= v(\mathbf{r}) + \frac{\partial J}{\partial \rho} + \frac{\partial E_{\text{xc}}}{\partial \rho} \\ &= v(\mathbf{r}) + \int \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}' + v_{\text{xc}}(\mathbf{r}). \end{aligned}$$

It contains the exchange–correlation potential, $v_{\text{xc}}(\mathbf{r})$, which is defined as

$$v_{\text{xc}} := \frac{\partial E_{\text{xc}}}{\partial \rho(\mathbf{r})}.$$

The lagrange multiplier, μ , defined in eq. 2, along with the normalization condition (eq. 2), precisely matches the equation obtained for a non-interacting N -body electron system subjected to an external potential, $v_{\text{eff}}(\mathbf{r})$. Thus, for a given v_{eff} , one can determine $\rho(\mathbf{r})$ such that the right-hand side of eq. 2 becomes independent of \mathbf{r} , as expressed by

$$\rho(\mathbf{r}) = \int \sum_i^N |\Psi_i(x^\nu)|^2 d\sigma,$$

where x^ν represents the four-vector containing both spatial and spin coordinates, and the integration is performed over the spin variable σ .

We can also express the electron density of an N -electron system invoking the proper occupational statistics that they obey:

$$\rho(\mathbf{r}) = \sum_{i=1}^N f_i |\Psi_i(\mathbf{r})|^2,$$

with $\{f_i\}_{i=1}^N$ the occupation numbers and for *Fermi-Dirac* occupations: $f_i = (1 + \exp\{\beta(E_i - \mu(T))\})^{-1}$, with $\mu(T)$ denoting the chemical potential and $\beta \equiv 1/k_B T$.

Furthermore, the molecular orbitals Ψ_i must satisfy the one-electron equations:

$$\left(-\frac{1}{2}\nabla^2 + v_{\text{eff}}(\mathbf{r})\right) \Psi_i = \epsilon_i \Psi_i$$

We can formulate the above equations in a more complete and procedural manner, splitting the effective potential as (compare with the form in eq. 2)

$$v_{\text{eff}}(\mathbf{r}; \rho_{\text{in}}) := v(\mathbf{r}) + v_{\text{HF}}(\mathbf{r}; \rho_{\text{in}}) + v_{\text{xc}}(\mathbf{r}; \rho_{\text{in}})$$

where we highlighted the conditioning on an input electron density, ρ_{in} , and recognized the *Hartree-Fock functional*,

$$v_{\text{HF}}(\mathbf{r}; \rho) \equiv \frac{\delta J[\rho]}{\delta \rho} = \int \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}'.$$

Therefore, we obtain the following updating scheme:

$$\begin{aligned} \left(-\frac{1}{2} \nabla^2 + v(\mathbf{r}) + v_{\text{HF}}(\mathbf{r}; \rho_{\text{in}}) + v_{\text{ex}}(\mathbf{r}; \rho_{\text{in}}) \right) \Psi_i(\mathbf{r}) &= \epsilon_i \Psi_i(\mathbf{r}) \\ \rho_{\text{out}} &= \sum_{i \in \text{occupied}} |\Psi_i(\mathbf{r})|^2, \end{aligned}$$

assuming a local exchange-correlation potential.

This result is recovered within a *variational optimization* framework when seeking the orbitals that minimize the energy functional (eq.2) under the orthonormality constraints $\langle \Psi_i | \Psi_j \rangle = \delta_{ij}, \forall i, j \in [N]$. In this way, the *Kohn-Sham equations* (eq.2) emerge as one-electron equations, similar in nature to the Hartree or Hartree-Fock equations, and they are solved iteratively.

A Self-consistent cycle can be developed as a convergent iterative procedure (contraction mapping on electron density updates, $T[\rho_{\text{in}}] = \rho_{\text{out}}$) as shown is shown below:

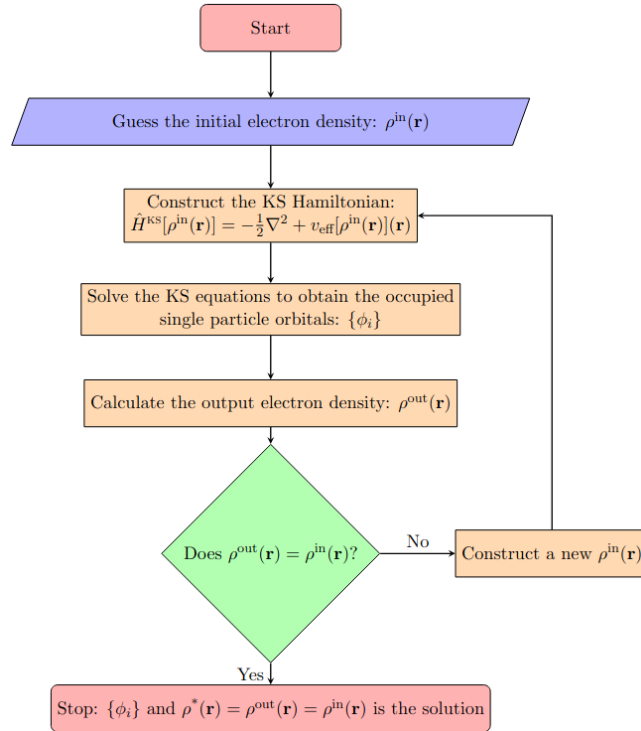


Figure 3: Flow chart depicting a self-consistent solution to the KS equations[Woor18].

The above *Self Consistent Field* can be briefly summarized by the following queue structure:

$$\rho_{\text{in}}(\mathbf{r}) \rightarrow \{v_{\text{HF}}, v_{\text{xc}}\} \rightarrow \{E_i, \Psi_i\} \rightarrow \rho_{\text{out}}(\mathbf{r}).$$

The cost of incorporating electron correlation is the introduction of the exchange–correlation potential, v_{xc} , whose exact form remains unknown and for which no systematic improvement strategy exists.

It is important to note that, in its Kohn–Sham formulation, density functional theory now offers a computational tool characterized by an exceptional quality-to-cost ratio, as extensively demonstrated in the work by Koch and Holthausen [KH01]. This aspect is underscored in this report, given that many—if not most—of the applications reported in the literature are based on DFT computational methods summarized in the below scheme.

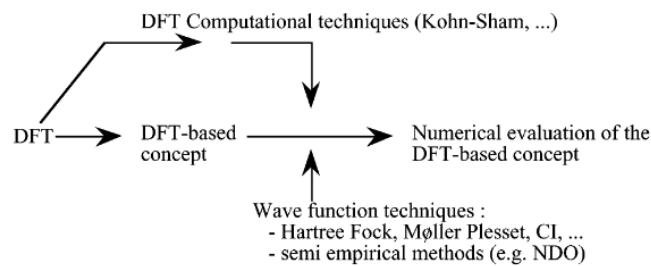


Figure 4: Conceptual DFT in practice.

Over the years, there has been substantial progress in the development of functionals, with perhaps the most widely used being the B3LYP functional[Bec93],[LYP88]². Its performance, in conjunction with various basis sets, has been thoroughly evaluated for molecular geometries[KW13], vibrational frequencies[SR96], ionization energies and electron affinities[VDG97], dipole and quadrupole moments[DTGoo], atomic charges[DMG96], infrared intensities[DMG96], and magnetic properties (e.g., chemical shifts[RPF99]).

3

DENSITY-BASED MIXING

Density-based mixing is defined as the process of finding a fixed point of the Kohn–Sham (KS) map,

$$T[\rho^*] = \rho^*, \text{ or, equivalently identifying a zero of the residual, } R[\rho^*] = T[\rho^*] - \rho^* = 0.$$

In practice, this means devising an *iterative method* in which the input and output densities from the current iteration are combined to produce an updated density that is closer to the converged solution. In fact, one can utilize the full history of densities generated during the iterations to construct the next input density, leading to a density mixing scheme defined by the function

$$\rho_{in}^{n+1} = f(\{\rho_{in}^j, \rho_{out}^j\}), \forall j \in [n].$$

The objective of this section essentially is to determine the optimal form of the linear functional $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ specifically for the KS operator $T : \mathbb{R} \rightarrow \mathbb{R}$.

²The B3LYP approach belongs to the hybrid (i.e., mixed) approximations for the exchange-correlation functional.

A vast body of literature exists on numerical methods for solving non-linear systems[Faur6], yet there is no universally superior method that outperforms all others in every application. Consequently, the function f must be adapted to incorporate characteristics specific to the KS system. For instance, in KS DFT one often benefits from a reasonably accurate initial guess for the electron density (commonly constructed as a sum of well-placed atomic orbitals), naturally favoring algorithms that perform well when starting from an accurate density. Additionally, the electron density can be expressed in a plane wave basis expansion,

$$\rho(\mathbf{r}) = \sum_{\mathbf{G}} c_{\mathbf{G}} e^{i\mathbf{G} \cdot \mathbf{r}} \in \mathbb{C}^{\mathbb{N}_{\mathbf{G}}},$$

where the allowed plane waves are restricted by the condition $\mathbf{G} \cdot \mathbf{R} = 2\pi m, \forall m \in \mathbb{N}$; that is, the plane waves must share the periodicity of the unit cell. This formulation differs from the more intuitive real-space picture (with \mathbf{r} representing a specific configuration). Since the number of \mathbf{G} -vectors required to represent ρ is typically very large ($10^4 - 10^6$), it is crucial that the algorithm is designed to avoid excessive memory consumption. Thus, the function f should define a robust and efficient scheme that can be executed within limited memory resources.

With these points in mind, one can distinguish two aspects of density mixing: the numerical analysis used to select or define an algorithm that will lead to convergence, and the preconditioning of this algorithm. In the first aspect, KS DFT is treated as a black box, while in the second, as much prior knowledge about the KS system as possible is incorporated to aid convergence.

3.1 FIXED POINT ITERATION SCHEME

The simplest conceivable update is to treat the self-consistent field (SCF) process as a *fixed-point iteration* (FPI), where the output density is directly fed back as the new input density by setting

$$\rho_{in}^{n+1} = T[\rho_{in}^n] = \rho_{out}^n.$$

However, this straightforward approach proves to be unsuitable for several reasons. Notably, when the initial density guess is already close to convergence, one can linearize the operation T about the ideal fixed point ρ^* . By writing

$$\rho^n = \rho^* + \delta\rho^n,$$

and performing a Taylor expansion of T around ρ^* :

$$\begin{aligned} \rho^* + \delta\rho_{out}^n &= T[\rho^* + \delta\rho_{in}^n] \\ &\approx T[\rho^*] + \frac{\delta T[\rho_{in}^n]}{\delta\rho_{in}^n} \Big|_{\rho^*} \delta\rho_{in}^n \\ &= \rho^* + \frac{\delta T[\rho_{in}^n]}{\delta\rho_{in}^n} \Big|_{\rho^*} \delta\rho_{in}^n, \end{aligned}$$

it follows that the change in the output density—and consequently the change in the subsequent input density—is linearly related to the current input density via

$$\delta \rho_{in}^{n+1} \approx M \delta \rho_{in}^n,$$

where the matrix operator M is defined as

$$M := \frac{\delta T[\rho_{in}^n]}{\delta \rho_{in}^n} \Big|_{\rho^*} = \frac{\delta \rho_{out}^n}{\delta \rho_{in}^n} \in \mathbb{C}^{N_G \times N_G}.$$

This matrix is of particular significance beyond fixed-point iterations as it is closely related to the *Kohn-Sham dielectric function* and can be employed to *precondition* density mixing schemes. It is therefore necessary to ensure that the spectral radius of M , given by $\rho(M) \equiv \max_i |\lambda_i|$ is less than 1.

However, further analysis by Dederichs and Zeller [DZ83] indicates that in practice, $\rho(M) \sim 100$, which leads to a pronounced divergence from the fixed point ρ^* .

3.1.1 Fixed Point Iteration experiments

In this part we present the experiments done to study the (oscillatory) convergence of the Fixed Point Iteration routine.

Below are the results for the fixed point iteration scheme following the theorem:

Theorem 3.1 (special case of Banach fixed-point theorem). *Given a function $f : \mathbb{R} \rightarrow \mathbb{R}$ and a point $x_0 \in \text{dom}(f)$, the fixed-point iteration is*

$$x^{n+1} = f(x^n), \forall n = 0, 1, 2, \dots,$$

which gives rise to the sequence $\{x_i\}$ of iterated function applications $x_0, f(x_0), f(f(x_0)), \dots$ which ideally converges to a point x^ . If f is continuous, then one can prove that the obtained x^* is a fixed point of f , i.e., $x^* = f(x^*)$. More generally, the function f can be defined on any metric space with values in that same space.*

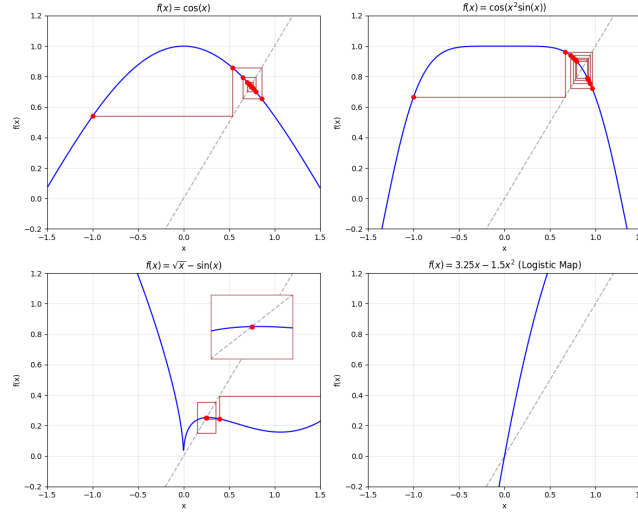


Figure 5: Lémeray (staircase) diagrams[Lém97] for the trial functions: $\cos(x)$, $\sqrt{x} - \sin(x)$, $\cos(x^2 \sin(x))$, and $3.25x - 1.5x^2$ respectively. Initial condition: $x_0 = 1$. Maximum iterations: 10.

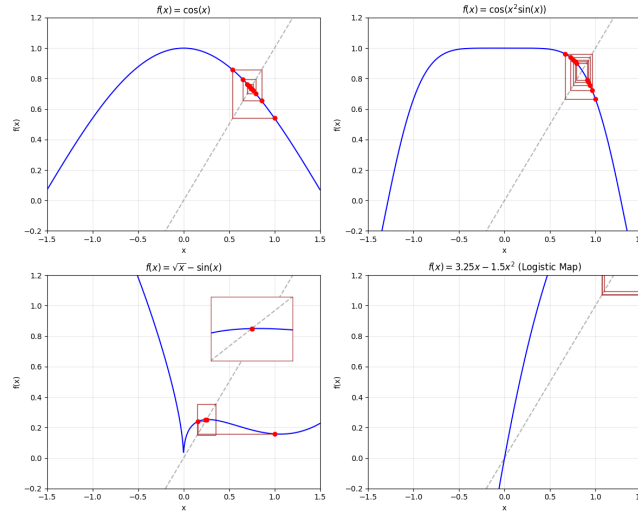


Figure 6: Lémeray (staircase) diagrams for the trial functions: $\cos(x)$, $\sqrt{x} - \sin(x)$, $\cos(x^2 \sin(x))$, and $3.25x - 1.5x^2$ respectively. Initial condition: $x_0 = -1$. Maximum iterations: 10.

Above we experimented with the following functions:

$$\begin{aligned} f_1(x) &= \cos(x), \\ f_2(x) &= \sqrt{|x|} - \sin(x), \\ f_3(x) &= \cos(x^2 \sin(x)), \\ f_4(x) &= 3.25x - 1.5x^2. \end{aligned}$$

where we use $|x|$ on the square root in f_2 , such that $\text{dom}(f_2) = \mathbb{R}$. We run for 10 iterations with initial conditions $x_0 = 1$ and $x_0 = -1$, respectively, and a stopping criterion: $|x| > 10^6$.

The iterates of f_1 converge to the Dottie number (≈ 0.739) for both initializations, due to the stable derivative $|f_1'(x)| = |\sin(x)| < 1$ near the fixed point.

The generated sequence of f_2 oscillates for both initializations, reflecting the lack of stable fixed points for $\sqrt{|x|} = \sin(x)$ and physically meaningful solutions beyond $x^* = 0$, where f_2' is unbounded.

Despite its complexity, f_3 's iteration converges to a stable fixed point near $x \approx 0.8$. The nested trigonometric operations dampen oscillations, and the derivative $|f_3'(x)|$ near the fixed point likely remains below 1, ensuring stability.

Contrary to typical logistic maps (e.g., $f(x; r > 0) = rx(1 - x)$), this parameterization leads to divergence. The fixed points $x = 0$ and $x = 1.5$ are unstable due to $|f_4'(x)| = |3.25 - 3x|$. At $x = 1.5$, $|f_4'(x = 1.5)| = 1.25 > 1$, causing oscillations to grow. For $x_0 = 1$, the iteration overshoots the unstable point $x_u = 1.5$, accelerating divergence.³

For divergent cases like the logistic map here, hybrid methods (e.g., Newton-Raphson) or relaxation techniques (e.g., $x^{n+1} = \alpha g(x^n) + (1 - \alpha)x^n$) could enforce stability[Faur6].

3.2 BROYDEN, KERKER, PULAY MIXERS

Broyden, Kerker, and Pulay mixing schemes all aim to mitigate the potential divergence or slow convergence in self-consistent calculations by leveraging different mathematical strategies (quasi-Newton updates, reciprocal-space damping, or Direct Inversion in the Iterative Subspace (DIIS)). Kresse mixing, the default in widely used codes like VASP and GPAW, combines the benefits of these approaches with carefully chosen default parameters, delivering a reliable balance between convergence stability and speed for a broad range of material systems.

Broyden's method is a quasi-Newton algorithm used to solve nonlinear systems iteratively[Bro65]. In DFT, it approximates the inverse Jacobian of the residual (difference between input and output densities) using historical data from previous iterations. This avoids explicit Jacobian calculations, reducing computational cost. The update formula combines a rank-1 update to the inverse Jacobian with a weighted history of past densities and residuals. Broyden accelerates convergence compared to simple linear mixing but requires storing several previous iterations.

The basic idea is to treat the residual $R_i = \rho_{out}^i - \rho$ (the difference between output and input densities at iteration i) as a *root-finding search*. By building up an approximate Jacobian (or derivative) of this function

³While the standard logistic map can exhibit convergence for specific parameters (e.g., $r < 3$), the given parameterization ($r = 3.25$) exceeds this threshold, leading to instability. Adjusting the parameter to $r < 3$ would stabilize convergence to a fixed point.

“on the fly”, Broyden mixing adapts the mixing parameter β at each step:

$$\rho^{i+1} = \beta \rho^i + (1 - \beta) \rho_{out}^i, \quad 0 \leq \beta \leq 1.$$

Instead of keeping β fixed, it is refined iteratively based on accumulated information from previous steps (sometimes referred to as a posteriori derivatives).

This method was among the most common until around 2000, often simply setting $\beta = 0.4$ if no more sophisticated scheme was available.

Kerker mixing is specifically tailored for reciprocal-lattice (plane-wave) DFT codes[Ker81]. Unlike a uniform β in real-space, Kerker mixing makes the mixing factor G -dependent, where \mathbf{G} is a reciprocal-lattice vector:

$$\rho_{min}(\mathbf{G}) = \rho_{in}(\mathbf{G}) + A \frac{G^2}{G^2 + B^2} [\rho_{out}(\mathbf{G}) - \rho_{in}(\mathbf{G})].$$

Similarly, the mixing parameter for wave-vector \mathbf{k} is scaled as

$$\Delta \rho_{mixed}(\mathbf{k}) = \beta \cdot \frac{k^2}{k^2 + k_o^2} \Delta \rho_{residual}(\mathbf{k}),$$

where $\Delta \rho_{residual} := \rho_{SCF} - \rho^n$ is the residual which should converge to zero, and k_o (screening parameter) damps long-wavelength ($k \rightarrow 0$) components. This suppresses low-frequency oscillations, improving convergence for metals.

This mixing strategy addresses charge sloshing instabilities (long-wavelength density oscillations) in metallic systems or systems with long-range Coulomb interactions, by preconditioning the residual in reciprocal space. Specifically, low- G components of the density can lead to slow SCF convergence. Kerker mixing damps these problematic long-wavelength modes by making the mixing weaker (i.e., effectively smaller β) for small $|G|$. Parameters A and B must be tuned (empirically or heuristically) for different systems to optimize convergence.

Pulay mixing, or *Direct Inversion in the Iterative Subspace* (DIIS), extrapolates the next density using a linear combination of previous densities and residuals[Pul80]:

$$\rho^{i+1} = \sum_{j=1}^{N_{max}} a_j \rho_j, \quad \sum_{j=1}^{N_{max}} a_j = 1.$$

where N_{max} is the number of past densities stored (the “subspace”). The coefficients a_j are chosen to minimize the norm of a composite residual vector by solving a small linear system. Pulay is highly effective but memory-intensive due to storing multiple densities/residuals.

Summarizing, Pulay mixer has the following attributes:

- **Linear Combination:** Rather than mixing only the current and previous densities, Pulay uses a weighted combination of multiple past densities.

- **DIIS Procedure:** The coefficients a_j are obtained by minimizing $\langle M|M \rangle$, where $M = \sum_k a_k R_k$ is a linear combination of residuals $\{R_k\}$.
- **Memory-accuracy Trade-off:** In practice, larger N_{\max} often yields better convergence but at the cost of storing more data and performing a larger minimization problem.

In modern plane-wave codes such as VASP and GPAW, a scheme often referred to as *Kresse mixing* is used by default [KF96]. It combines Kerker preconditioning and Pulay DIIS to balance stability and convergence. Essentially, *preconditioning* handles charge sloshing by damping long-wavelength residuals and *DIIS extrapolation* uses historical data to accelerate convergence.

Default Parameters (GPAW):

- $\beta=0.1$: Mixing strength for the Kerker preconditioner. Controls the step size for residual incorporation.
- $\text{weight}=50.0$: Screening parameter k_0^2 in the Kerker denominator. Larger values reduce damping of long-wavelength components.
- $\text{nmaxold}=3$: Number of previous iterations stored for DIIS. Governs the subspace size for extrapolation.

These default values are robust for many systems, but for challenging cases (e.g., large vacuum regions, metallic surfaces, or magnetic systems), tuning them can significantly improve SCF convergence rates.

The residual for wavevector \mathbf{k} is preconditioned as

$$\Delta\rho_{\text{mixed}}(\mathbf{k}) = \beta \cdot \frac{k^2}{k^2 + \text{weight}} \Delta\rho_{\text{residual}}(\mathbf{k}),$$

DIIS then optimizes the density using the last nmaxold preconditioned residuals and densities, with preconditioning kernel: $\Delta\rho_{\text{mixed}}(\mathbf{k})/\beta\Delta\rho_{\text{residual}}(\mathbf{k}) := \frac{k^2}{k^2 + \text{weight}}$.

This hybrid approach ensures stable and efficient convergence in the self-consistent procedure for both insulating and metallic systems.

We can expand the above equation obtaining the density update in reciprocal space:

$$\rho^{n+1}(\mathbf{k}) = \rho^n(\mathbf{k}) + \beta \cdot \frac{k^2}{k^2 + \text{weight}} [\rho_{\text{SCF}}(\mathbf{k}) - \rho^n(\mathbf{k})],$$

where $\rho^n \in \mathbb{C}^{N_G}$ is the density at iteration n in reciprocal space (see eq.3), ρ_{SCF} is the output density from the self-consistent field calculation, β is the mixing parameter controlling the overall step size, and weight is chosen to control the damping of long-wavelength (small $|\mathbf{k}|$) components. After applying this preconditioner in reciprocal space, an inverse Fourier transform is taken to update the density in real space.

RECIPROCAL SPACE AND PERIODICITY

In a crystalline solid, atoms are arranged in a periodic lattice. Real-space lattice vectors, often denoted \mathbf{a}_1 , \mathbf{a}_2 , and \mathbf{a}_3 , define the fundamental repeating unit (the unit cell). In reciprocal space, one defines a complementary set of lattice vectors, \mathbf{b}_1 , \mathbf{b}_2 , and \mathbf{b}_3 , which are mathematically constructed so that $\mathbf{b}_i \cdot \mathbf{a}_j = 2\pi \delta_{ij}$. The periodicity in real space implies a corresponding periodicity in reciprocal space, meaning that the same electronic structure repeats every reciprocal lattice vector \mathbf{G} .

Because $\mathbf{R} \cdot \mathbf{G} = 2\pi m$, where $m \in \mathbb{N}$, for lattice vectors $\mathbf{R}_{n_1, n_2, n_3} = n_1 \mathbf{a}_1 + n_2 \mathbf{a}_2 + n_3 \mathbf{a}_3$ and reciprocal lattice vectors $\mathbf{G}_{m_1, m_2, m_3} = m_1 \mathbf{b}_1 + m_2 \mathbf{b}_2 + m_3 \mathbf{b}_3$, the exponential factor $e^{i\mathbf{R} \cdot \mathbf{G}} = 1$. This relationship underlies many of the simplifications in solid-state calculations, including Bloch's theorem and the way we treat integrals over reciprocal space.

A cornerstone of periodic systems is *Bloch's theorem*, which states that the electronic wavefunction $\psi_{\mathbf{k}}(\mathbf{r})$ for a crystal can be written as

$$\psi_{\mathbf{k}}(\mathbf{r}) = e^{i\mathbf{k} \cdot \mathbf{r}} u_{\mathbf{k}}(\mathbf{r}),$$

where \mathbf{k} is a wave-vector in reciprocal space, and $u_{\mathbf{k}}(\mathbf{r}) = u_{\mathbf{k}}(\mathbf{r} + \mathbf{R})$. Conceptually, this means each wavefunction is labeled by a specific \mathbf{k} within the first Brillouin zone (the fundamental region in reciprocal space). Periodicity in real space (\mathbf{r}) maps onto periodicity in \mathbf{k} , reducing what would otherwise be an unmanageably large problem to one that only needs to be solved for \mathbf{k} values within that zone.

In practice, many of the physical properties of a crystal—such as the total energy, electron density, or density of states—require an integral over reciprocal space of the form

$$\int_{\text{BZ}} f(\mathbf{k}) d\mathbf{k}.$$

Because this integral cannot be performed analytically for arbitrary materials, computational approaches discretize it by selecting a finite set of k -points within the Brillouin zone (or, more commonly, the irreducible wedge of that zone, which exploits symmetry to further reduce computational effort). This k -point sampling effectively turns the integral into a sum over a set of k -points, each weighted according to the chosen sampling scheme. Methods like the Monkhorst-Pack scheme[ZZ23] or Gamma-centered grids[Tog+23] help ensure that these k -points capture the essential physics with as few points as possible.

4.1 NUMERICAL INTEGRATION EXPERIMENTS

Here we present the experiments conducted on trapezoidal method for $f(x) = x^2$ integration and optimal grid point selection for a specific accuracy bound of the Fermi-Dirac function, $f(x; \sigma) = \left(1 + e^{\frac{x-\mu}{\sigma}}\right)^{-1}$.

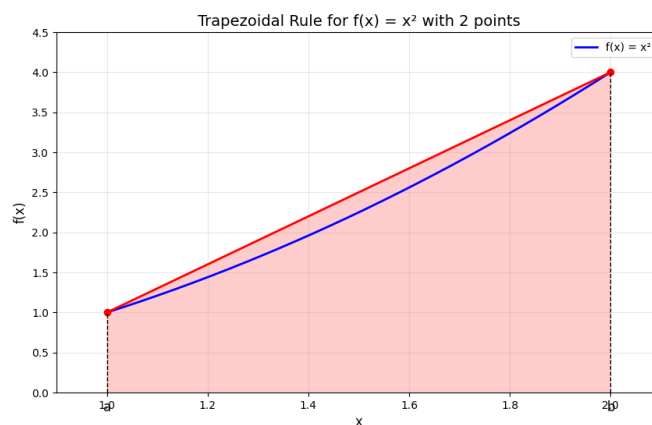


Figure 7: Plot of the exact function curve (blue) and the linear approximation segment in (red).

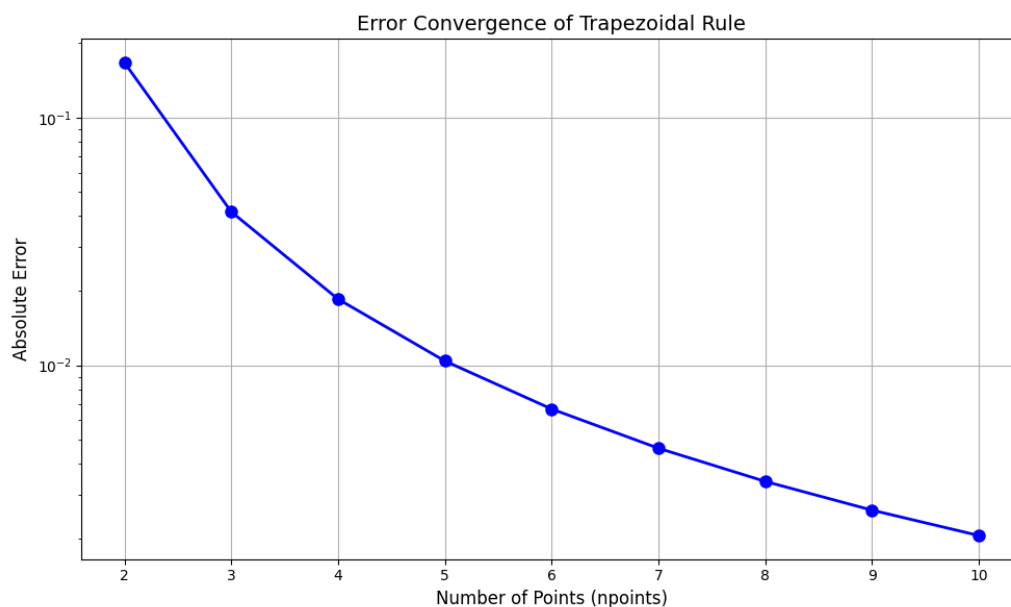


Figure 8: Error dependence curve for different npoints (subintervals: npoints - 1).

npoints	subintervals	Approximation	Absolute Error	Relative Error (%)
2	1	2.5000000000	0.1666666667	7.1428571429
3	2	2.3750000000	0.0416666667	1.7857142857
4	3	2.3518518519	0.0185185185	0.7936507937
5	4	2.3437500000	0.0104166667	0.4464285714
6	5	2.3400000000	0.0066666667	0.2857142857
7	6	2.3379629630	0.0046296296	0.1984126984
8	7	2.3367346939	0.0034013605	0.1457725948
9	8	2.3359375000	0.0026041667	0.1116071429
10	9	2.3353909465	0.0020576132	0.0881834215

Figure 9: Table showing approximation errors for different numbers of points (1 to 10) (tabulate library).

The exact baseline was set by using adaptive Gaussian quadrature integration using the method `scipy.integrate.quad` which in turn uses the Fortran library QUADPACK. This calculation returned (double precision): $f_{\text{exact}}(x) = 2.333333333333335 \approx \int_1^2 x^2 dx$.

Theorem 4.1 (composite trapezoidal rule). *Let $\{x_k\}$ be a partition of $[a, b]$ such that $a = x_0 < x_1 < \dots < x_{N-1} < x_N = b$ and Δx_k be the length of the k -th subinterval (that is, $\Delta x_k = x_k - x_{k-1}$), then*

$$\text{Trapez} \left[\int_a^b f(x) dx \right] = \sum_{k=1}^N \frac{f(x_{k-1}) + f(x_k)}{2} \Delta x_k.$$

In the regular spacing regime ($\Delta x_k \equiv k \Delta x$):

$$\text{Trapez} \left[\int_a^b f(x) dx \right] \equiv \frac{\Delta x}{2} (f(x_0) + 2f(x_1) + \dots + 2f(x_{N-1}) + f(x_N)).$$

As depicted by the above results, the trapezoidal (1st order) approximation becomes more accurate as the resolution of the partition increases (that is, for larger N , all Δx_k decrease).

Below we investigate the convergence properties for different values of temperature parameter $\sigma > 0$ of the Fermi-Dirac distribution:

$$f(x; \sigma) = \left(1 + e^{\frac{x-1}{\sigma}} \right)^{-1}.$$

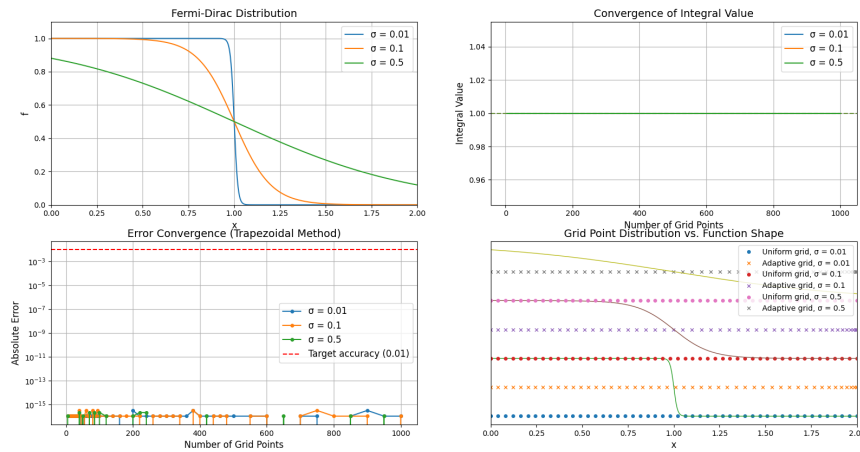


Figure 10: Fermi-Dirac distribution function for various $\sigma \in \{0.01, 0.1, 0.5\}$. Determination of the resolution (number of grid points) needed to achieve 0.01 accuracy for each σ value. Results for different grid point mechanisms.

Interestingly, all three cases appear to converge to exactly 1.0 (top-right subplot), regardless of "sharpness" (σ value). The fact that the lines appear flat suggests that even with relatively few grid points, the integral approximation is quite accurate.

The logarithmic plot on the bottom left panel shows the absolute error in the numerical integration compared to the exact value⁴. The red dashed line represents the target accuracy of 0.01. The most striking feature is that the errors for all σ values appear to be extremely small (below 10^{-13}) even with very few grid points. This suggests that the trapezoidal rule is highly effective for this particular function. The exact analytical integral of this function over $[0, 2]$ appears to be exactly 1.0.

For smaller σ values (0.01, 0.1), the function exhibits a sharper transition at $x = 1$, requiring more grid points to accurately capture this steep change. The function becomes smoother and easier to integrate numerically as σ increases to 0.5. From the integral analysis, $\sigma = 0.01$ requires many more grid points due to the very steep transition, $\sigma = 0.1$ demands an intermediate number of grid points, and $\sigma = 0.5$ converges with fewer grid points due to the smoother function profile.

Remark 4.2. If the density varies rapidly, when compared with the grid spacing, then a small change in the position of the grid origin can produce a large change in the output function (and its integral). More generally, functions with sharp features (like the step-like behavior with small σ) require more grid points for accurate integration unless you use adaptive methods that concentrate points where they're most needed.

5

GRID SPACING AND CUTOFFS

In DFT, *plane-wave basis sets* are widely used to represent electronic wavefunctions due to their mathematical simplicity and compatibility with periodic boundary conditions. Starting from Bloch's theorem, we can expand the crystal-periodic function $u_{\mathbf{k}}(\mathbf{r})$ in plane waves yielding

$$\psi_{\mathbf{k}}(\mathbf{r}) = e^{i\mathbf{k}\cdot\mathbf{r}} \sum_{n=1}^{N_{\text{PW}}} c_n e^{i\mathbf{G}_n\cdot\mathbf{r}},$$

where \mathbf{G}_n are reciprocal lattice vectors. The accuracy of this representation depends on the number of plane waves (N_{PW}) included in the basis set, with larger basis sets capturing finer details of the wavefunction and reducing truncation errors. The concepts of grid spacing and energy cutoffs are critical for balancing computational efficiency and accuracy, as discussed in Sholl and Steckel's "Density Functional Theory: A Practical Introduction" (subsection 3.1)[SSII].

To standardize this process, DFT codes use the plane-wave cutoff energy (E_{PW}), defined as

$$E_{\text{PW}} := \frac{\hbar^2 |\mathbf{G}_{\text{max}}|^2}{2m},$$

where \mathbf{G}_{max} is the largest reciprocal lattice vector retained in the basis. This cutoff ensures all plane waves with kinetic energy below E_{PW} are included. Higher cutoffs improve accuracy but increase computational

⁴The plot shows some oscillating behavior at extremely small error values, which is likely due to floating-point precision limitations rather than actual integration errors.

cost, as the workload scales cubically with the number of plane waves. For example, a typical cutoff of 500eV corresponds to a real-space grid spacing of a $h \sim 0.1\text{\AA}$, balancing resolution and efficiency.

The relationship between plane-wave cutoffs and real-space grid spacing is governed by the Nyquist-Shannon sampling theorem⁵. Finer grids (smaller grid spacing h) allow higher $|\mathbf{G}_{\text{max}}|$ values, enabling larger E_{PW} . Conversely, coarser grids risk aliasing errors⁶, where high-frequency wavefunction components are misrepresented as low-frequency artifacts. Codes like VASP or GPAW automate grid generation from E_{PW} to mitigate such errors.

Practical considerations include convergence testing, where users incrementally increase E_{PW} until total energies or forces stabilize, and the choice of pseudo-potentials. Hard pseudo-potentials (e.g., for transition metals) require higher cutoffs to resolve localized interactions. Insufficient cutoffs or coarse grids can also cause the egg-box effect⁷, introducing spurious energy oscillations as atoms move across grid points.

While E_{PW} governs basis set completeness, k -point sampling determines Brillouin zone resolution. These parameters are independent: dense k -point meshes integrate electronic states over reciprocal space, while plane-wave cutoffs resolve wavefunction details at each k -point. Neither compensates for deficiencies in the other; both must be converged for reliable results. For instance, a well-converged E_{PW} ensures accurate wavefunction representation, while sufficient k -points capture electronic band structure.

In summary, achieving reliable DFT results requires balancing accuracy and computational cost. Higher E_{PW} (smaller h) enhances precision but demands greater resources. Codes often automate grid generation from E_{PW} , streamlining workflow. Users should systematically test convergence for both E_{PW} and k -points, tailoring parameters to system-specific requirements, such as pseudo-potential hardness and material geometry. This approach ensures robust simulations while avoiding artifacts like aliasing or the egg-box effect.

5.1 SYMMETRY CONSIDERATION ON BULK CU EXPERIMENTS

5.1.1 Space Group Symmetry Utilization

On the first experiment we invoked symmetry using the Monkhorst-Pack grid[MP76], with an example shown in the following schematic.

⁵The sample rate must be at least twice the bandwidth of the signal to avoid aliasing.

⁶Aliasing is an undesirable effect that is seen in sampled systems. When the input frequency is greater than half the sample frequency, the sampled points do not adequately represent the input signal. Inputs at these higher frequencies are observed at a lower, aliased frequency.[Savo3]

⁷The egg-box effect is the breakdown of translational invariance. It occurs when the distribution is not soft enough or the discretization is not fine enough.

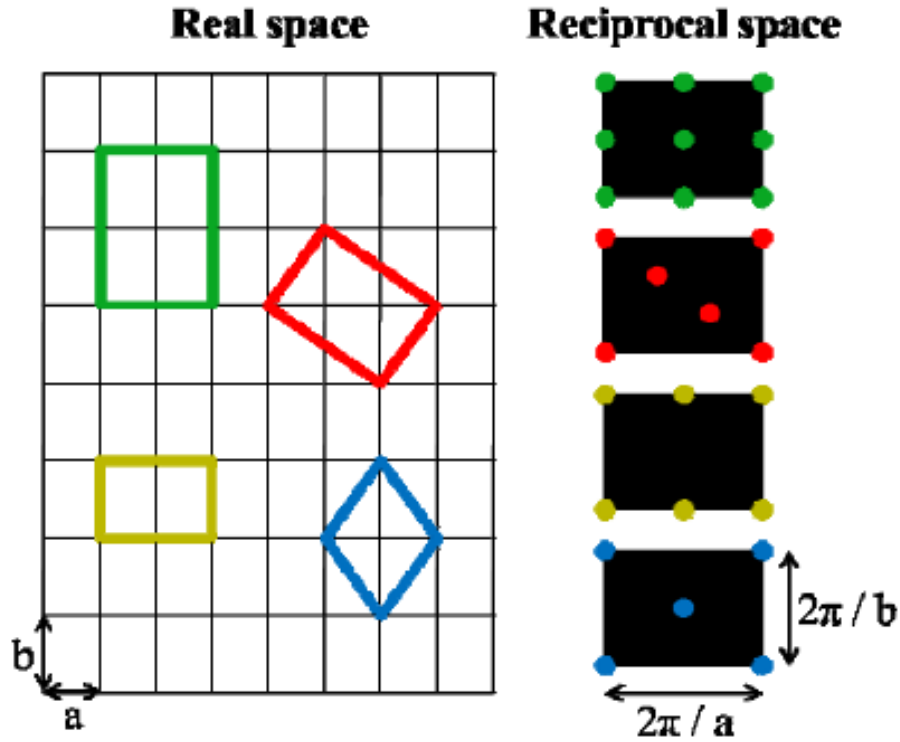


Figure 11: A comparison of (left) real-space supercells on a rectangle lattice and (right) the corresponding k -point grids in a unit cell of the reciprocal lattice. To aid visual comparison, all k -point grids are shifted so that one k -point falls on the corner of the reciprocal space unit cell. The yellow grid is an example of a traditional 2×1 Monkhorst-Pack grid, and the blue grid is a generalized Monkhorst-Pack grid with the same density that more evenly samples reciprocal space.[WMM16]

This setup utilizes *crystal symmetry* (specifically, the space group symmetries of bulk Cu's face-centered cubic (FCC) structure) to reduce the number of k -points in the Irreducible Brillouin Zone (IBZ). GPAW automatically detects these symmetries, allowing it to sample only the unique k -points instead of the full Brillouin Zone (BZ). This significantly reduces computational demands.

M denotes the Monkhorst-Pack k -point grid parameter. $kpts=(M, M, M)$ defines a uniform cubic grid of $M \times M \times M$ k -points in the full BZ. Larger M increases k -point sampling density, improving accuracy (at the cost of computational time). For example, $M = 1 \rightarrow 1 \times 1 \times 1$ grid (1 k -point in total) and $M = 4 \rightarrow 4 \times 4 \times 4$ grid (64 k -point in total).

As M increases, $E/atom$ converges toward a stable value (~ -4.4 eV), indicating sufficient k -point sampling. Smaller M values (e.g., $M=1$ or $M=2$) are undersampled and give unphysically high energies.

We also recorded the computational time ratio for a given M relative to $M=1$, τ_M/τ_1 . As expected, time increases with M due to more k -points and larger Hamiltonian matrices. For example $M=10 \rightarrow \tau_m/\tau_1 = 21.2$ (21 \times slower than $M=1$). Non-linear scaling arises from parallelization efficiency and matrix diagonalization costs.

Key Setup Parameters:

Calculation setup: `atoms.calc = GPAW(mode='pw', h=0.1, kpts=(M, M, M), maxiter=200, convergence='energy': 1e-6, 'density': 1e-6)`.

- `mode='pw'`: Plane-wave basis set with a 340 eV cutoff, sufficient for Cu (balance between accuracy and cost).
- `kpts=(M, M, M)`: Monkhorst-Pack grid. Symmetry reduces the total k -points to the IBZ count in the table.
- Convergence criteria: Tight ($1\text{e-}6$ for energy/density), ensuring well-converged results.
- Symmetry usage: Enabled by default in GPAW. For FCC Cu, symmetry operations include: 48 symmetry operations (rotations + reflections) \rightarrow highly efficient k -point reduction.

Remark 5.1. The implementation details of how symmetry is used to reduce the k -points can vary between codes or versions.

M	E/atom (eV)	No. of k Points in IBZ	τ_m/τ_1
1	-14.378	1	1.0
2	-3.748	2	1.3
3	-4.745	4	1.6
4	-4.3295	10	2.6
5	-4.4891	10	3.2
6	-4.3798	28	7.1
7	-4.4287	20	5.3
8	-4.3901	60	12.5
9	-4.4019	35	7.7
10	-4.3896	110	21.2

Figure 12: Results from Computing the Total Energy of fcc Cu with $M \times M \times M$ k -Points Generated Using the Monkhorst-Pack Method. Space Group symmetry aware implementation.

5.1.2 Symmetry-agnostic Execution

We run again the above experiment with the symmetry set explicitly off (`symmetry = "off"` as an argument in `GPAW()`). Thus, GPAW no longer exploits the space group symmetries of the FCC Cu lattice, leading to all k -points in the full Brillouin Zone (BZ) being explicitly sampled (no reduction to the Irreducible Brillouin Zone (IBZ)). Also, the number of k -points in the IBZ now equals the total k -points in the Monkhorst-Pack grid: No. of k -points in IBZ = M^3 . No. of k -points in $IBZ = M^3$. As indicated by the results below, computational cost increases drastically since more k -points require diagonalizing the Hamiltonian more times per trial.

We present the following significant changes on the Hamiltonian set-up:

1. **XC functional change:** The new setup uses Local Density Approximation (LDA)[BH72] instead of PBE[PBE96], which slightly alters the absolute energy values (LDA typically underestimates lattice constants compared to PBE).
2. **Full k -point sampling:** Without symmetry, small numerical noise arises from sampling all k -points.

Nonetheless, At $M=10$, energy converges to $\sim -4.39 \text{ eV}$, close to the symmetry-enabled result (-4.3896 eV), suggesting convergence despite symmetry loss. Also, the switch to LDA explains the energy differences compared to the earlier PBE results, but the convergence behavior with M remains consistent.

Finally, we report an overall *10-fold* computational cost reduction on the symmetry-aware case.

M	E/atom (eV)	No. of k Points in IBZ	τ_m/τ_s
1	-14.378	1	1.0
2	-3.7348	8	2.7
3	-4.745	27	7.5
4	-4.3289	64	16.6
5	-4.4891	125	34.8
6	-4.3798	216	51.5
7	-4.4287	343	88.9
8	-4.3901	512	121.2
9	-4.4019	729	165.5
10	-4.3896	1000	223.9

Figure 13: Results from Computing the Total Energy of fcc Cu with $M \times M \times M$ k -Points Generated Using the Monkhorst-Pack Method. Symmetry-agnostic implementation.

5.2 COMPARATIVE ANALYSIS OF GRID SPACING AND CUTOFF ON BULK CU

In this experiment we addressed the following:

1. **Grid Spacing Study:** We have implemented a loop to test different grid spacing values (h) from 0.30 \AA down to 0.10 \AA , tracking energy, computation time, and relative time. This generates a complete table showing how grid refinement affects both accuracy and computational cost.
2. **Grid vs. Plane-Wave Comparison:** We calculated the average computation time for both methods and explicitly stated which one is faster based on the results. This comparison gives us a practical understanding of the performance difference between grid-based and plane-wave approaches.
3. **Plane-Wave Cutoff Study:** We conducted a systematic study of plane-wave cutoff energies from 100 eV to 700 eV , and generated a plot showing how the total energy converges with increasing cutoff (fig. 14).

4. The code also includes a convergence analysis that identifies at what grid spacing and plane-wave cutoff the energy values converge to within 0.01 eV , providing practical guidance for selecting computational parameters.

Wave function: Uniform real-space grid:

- Kinetic energy operator: $12^*3+1=37$ point $O(h^6)$ finite-difference Laplacian.
- ScaLapack parameters: grid= 1×1 , blocksize=None.
- Wavefunction extrapolation: Improved wavefunction reuse through dual PAW basis.

Wave function: Plane wave expansion:

- Cutoff energy: 700.000 eV .
- Number of coefficients (min, max): 480, 531.
- Pulay-stress correction: $0.000000 \text{ eV}/\text{\AA}^3$ (de/decut= 0.000000).
- Using Numpy's FFT.
- ScaLapack parameters: grid= 1×1 , blocksize=None.
- Wavefunction extrapolation: Improved wavefunction reuse through dual PAW basis.

Grid vs. Plane-wave mode comparison:

- Average time for grid mode: 32.17 seconds.
- Average time for plane-wave mode: 8.03 seconds.
- Plane-wave mode is faster than grid mode on average.

Convergence Summary:

- Plane-wave energy converged to $< 0.01 \text{ eV}$ at cutoff: 600 eV .
- Grid mode energy converged to $< 0.01 \text{ eV}$ at h : 0.25 \AA .

Table 1: Grid Spacing Results

Grid Spacing (h)	Energy (eV)	Time (s)	Relative Time τ_M/τ_1
0.3	-5.607	7.819598	1.0
0.25	-5.607	7.709663	0.986
0.2	-4.800	26.160784	3.345
0.18	-4.800	25.763079	3.295
0.16	-4.800	25.963773	3.320
0.14	-4.710	43.577964	5.573
0.12	-4.710	43.709757	5.590
0.1	-4.696	76.637192	9.801

Table 2: Plane-Wave Results

PW Cutoff (eV)	Energy (eV)	Time (s)	Relative Time τ_M/τ_1
100	231.181	7.326756	1.0
150	20.799	5.971489	0.815
200	-2.191	6.058123	0.827
250	-4.278	5.545182	0.757
300	-4.350	6.521726	0.890
350	-4.409	7.919745	1.081
400	-4.518	8.015117	1.094
450	-4.601	8.183339	1.117
500	-4.635	8.591443	1.173
600	-4.643	11.859149	1.619
700	-4.657	12.315276	1.681

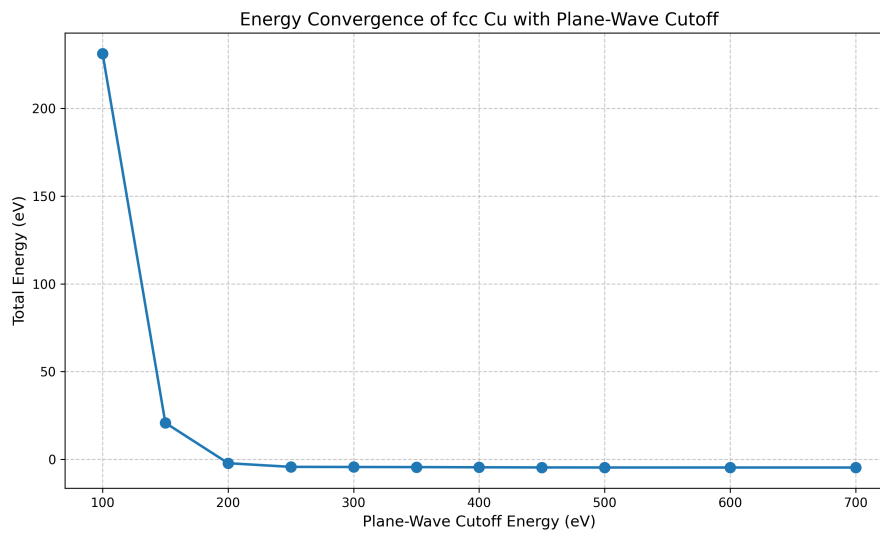


Figure 14: potential energy of FCC Cu as a function of cutoff energy.

6

DIRECT MINIMIZATION OF KS HAMILTONIAN

As mentioned on the first section, the KS total energy can be interpreted as a functional $E[\{\Psi_i\}_{i \in [N]}]$ of the occupied orbitals $\{\Psi_i\}_{i \in [N]}$, given N valence electrons (typically ζ valence electrons/atom, thus $N = \zeta N_{\text{atom}}$). In a variational setting, we seek the set of orbitals that minimize this functional subject to the orthonormality constraints, $\langle \Psi_i | \Psi_j \rangle = \delta_{ij}, \forall i, j \in [N]$.

Minimization procedures such as Newton or quasi-Newton algorithms systematically reduce the energy by moving “downhill” in the space of orbital coefficients. In each iteration, the algorithm requires the first derivative of the energy with respect to orbital variations (analogous to a *gradient*), and often approximate

second-derivative (*Hessian*) information. Because these methods treat the KS energy as a straightforward function to be minimized, they can be implemented in both Gaussian-type orbital (GTO)[DR90] and plane-wave (PW) basis sets.

A key observation, as Sholl and Steckel note[SSII], is that each iteration typically requires the application (or construction) of the effective one-electron operator $h_{\text{ref},i}$ (see eq.2). Although this is computationally costly, it is the same expensive step encountered in iterative diagonalization SCF methods.

A practical limitation of direct minimization algorithms is that they usually assume fixed orbital occupations (i.e., the Fermi–Dirac occupation numbers are not being varied). For systems with small or zero band gaps—metals or systems at finite temperatures—occupations can change significantly during SCF cycles. Hence, direct minimization is often less suitable in these cases, and more sophisticated approaches or standard SCF loops with partial diagonalization are preferred.[Zha+24]

6.1 BISECTION METHOD: ROOT-FINDING BINARY SEARCH

The bisection method is a root-finding strategy applied to continuous analytic functions $f(x)$ on an interval $[a, b]$ where $f(a) \cdot f(b) < 0$ (i.e., have opposite signs).

By repeatedly halving this interval and selecting the sub-interval where the sign change persists, one converges to a root x^* such that $f(x^*) = 0$. Although not the most efficient method (sometimes slowly convergence), bisection guarantees convergence if a root is bracketed.

6.2 NEWTON-RAPHSON'S METHOD: CURVATURE INFORMATION

Newton's method refines an estimate x^n of the root of a continuous analytic $f(x)$ by using the first derivative and approximate second derivative information (in one dimension, the second derivative is just $f''(x)$; in higher dimensions, the Hessian matrix is used). The root-finding update step is

$$x^{n+1} = x^n - \frac{f(x^n)}{f'(x^n)}.$$

and the optimization update step (since we seek for points x_{opt} such that $f'(x_{\text{opt}}) = 0$):

$$x^{n+1} = x^n - \frac{f'(x^n)}{f''(x^n)}.$$

and in the multivariate case ($x \in \mathbb{R}^d$):

$$x^{n+1} = x^n - (H[f(x^n)])^{-1} \nabla f(x^n), n \geq 0,$$

where $(H[f(x^n)])^{-1}$ is the inverse of Hessian matrix $(H[f(x^n)])_{ij} \equiv \frac{\partial^2 f}{\partial x_i \partial x_j} |_{x^n} = f(x^n)_{,ij}$ of function $f : \mathbb{R}^d \rightarrow \mathbb{R}$ at x^n and $\nabla f(x^n)$ is the gradient of the same function at x^n .

In multi-dimensional space—such as the space of orbital coefficients—the Hessian matrix replaces the second derivative, and quasi-Newton methods approximate this Hessian at each step (e.g., BFGS[Fle88] and its more scalable variant, L-BFGS[LN89]).

6.3 OPTIMIZATION EXPERIMENTS

6.3.1 Bisection method Implementation

Here we employ the bisection method to find the y -coordinate that minimizes the energy in the FCC Al system with a displaced atom.

The bisection method works by repeatedly dividing the search interval in half and selecting the sub-interval that contains the minimum. We start with an interval $[0.5, 3.0]$ and narrow it down until we achieve the desired accuracy of 0.1 eV . At each iteration, we calculate the energy at three points: the lower bound (a), midpoint (c), and upper bound (b) of the current interval. Based on these energies, we decide how to update the interval:

- If $\text{energy_a} < \text{energy_c} < \text{energy_b}$, the minimum is in the left half.
- If $\text{energy_a} > \text{energy_c} > \text{energy_b}$, the minimum is in the right half.
- If $\text{energy_a} > \text{energy_c} < \text{energy_b}$, the minimum is around the midpoint, and we refine further.

This method is reliable for finding the minimum of a *unimodal* function (a function with only one minimum in the search interval), which is likely the case for this small displacement in a crystalline structure. The bisection method has guaranteed convergence but may require more function evaluations than more sophisticated methods like Brent's method or BFGS. Lattice aluminum energy: -16.489 eV .

Original atomic positions:

$$\begin{bmatrix} 0. & 0. & 0. \\ 0 & 2.025 & 2.025 \\ 2.025 & 0. & 2.025 \\ 2.025 & 2.025 & 0. \end{bmatrix}$$

Displaced atomic positions ($y=0.6$):

$$\begin{bmatrix} 0. & 0. & 0. \\ 0 & 0.6 & 2.025 \\ 2.025 & 0. & 2.025 \\ 2.025 & 2.025 & 0. \end{bmatrix}$$

Optimal y -coordinate: 2.0234

Minimum energy: -16.488602 eV

Energy at original position ($y = 0.6$): -11.660940 eV

Energy reduction: 4.827662 eV

Iteration	Left y	Middle y	Right y	Energy(L)	Energy(M)	Energy(R)
1	0.5000	1.7500	3.0000	-11.0740	-16.3528	-14.3171
2	1.1250	1.7500	2.3750	-14.6841	-16.3528	-16.2665
3	1.7500	2.0625	2.3750	-16.3528	-16.4861	-16.2665
4	1.9062	2.0625	2.2188	-16.4633	-16.4861	-16.4214
5	1.9844	2.0625	2.1406	-16.4856	-16.4861	-16.4646

Figure 15: Table of bisection method to find minimum energy position. The algorithm starts with an initial interval $[0.5, 3.0]$ and iteratively narrows it down by evaluating the energy at three points: Left (L), Middle (M), and Right (R).

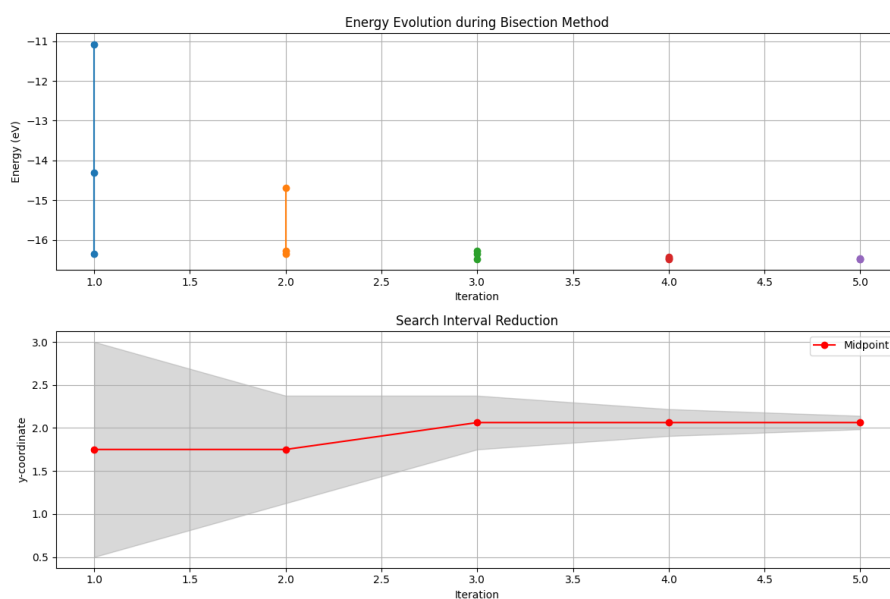


Figure 16: Energy evolution and interval reduction throughout the optimization process.

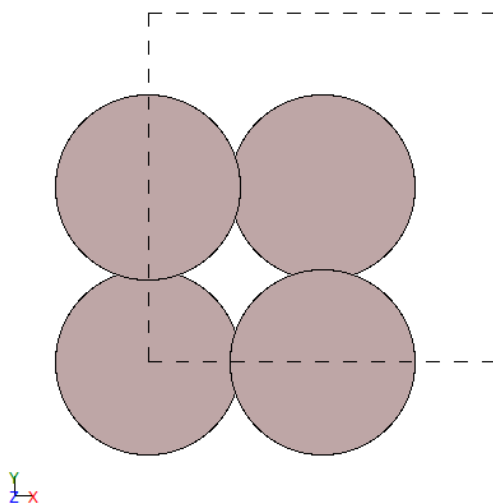


Figure 17: Final crystal structure, 2D representation.

6.3.2 BFGS (Quasi-Newton) Implementation

In the last subsection, we demonstrate the use of the BFGS (Broyden–Fletcher–Goldfarb–Shanno) optimizer to find the minimum energy configuration of an aluminum crystal where one atom has been displaced from its equilibrium position, as with the previous problem.

We artificially displace one atom (atom index 1) by changing its y -coordinate to 0.6 \AA and set up a plane-wave DFT calculator with a cutoff energy of 300 eV and a $4 \times 4 \times 4$ k -point grid.

BFGS iteratively updates an approximation to the Hessian matrix (second derivatives) to efficiently find a minimum energy configuration. Runs until the maximum force on any atom is less than 0.05 eV/\AA (the convergence criterion). We record energy, forces, and geometric changes at each step.

The energy plot shows how quickly the energy converges, the force plot demonstrates the convergence criterion, the displacement plot shows the total geometric change during optimization, and the step length plot reveals how the optimizer adjusts step size during the process.

The 3D trajectory visualization shows the path the displaced atom takes toward its equilibrium position.

Notice that the final structure is not optimal, mostly due to numerical instability, wrong lattice constant.

Minimum energy: -11.488602 eV

Energy at original position ($y = 0.6$): -11.661 eV

Maximum force at original position ($y = 0.6$): 6.050 eV/\AA

Energy at final position: -16.489 eV

Maximum force at final position: 0.028 eV/\AA

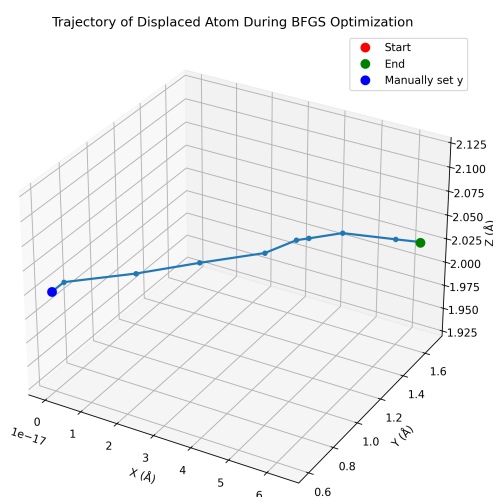


Figure 18: 3D visualization of the displaced atom's trajectory during optimization.

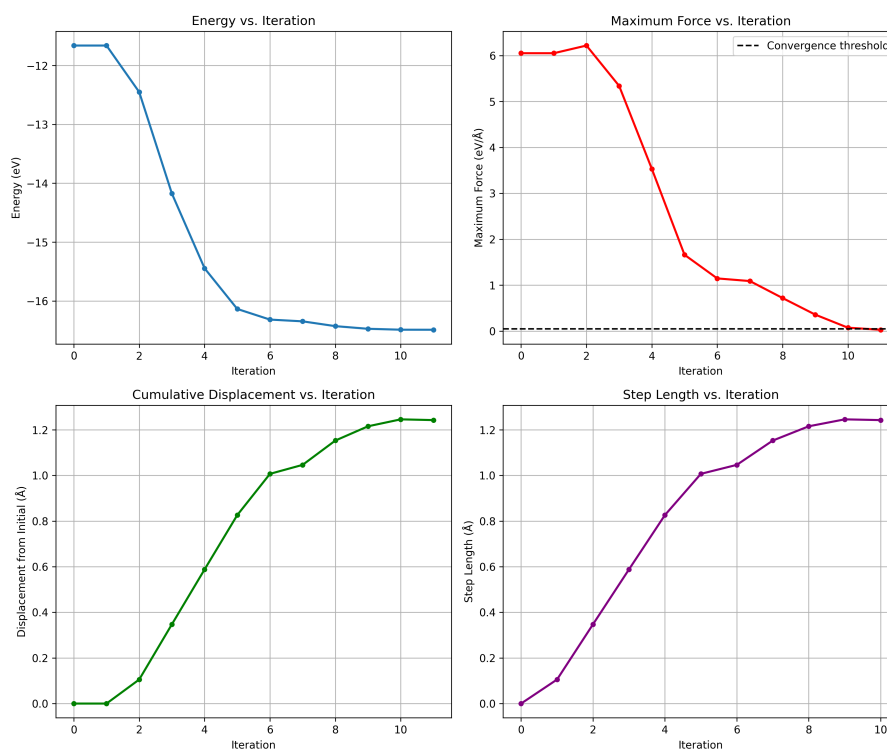


Figure 19: Optimization process monitoring. Energy vs. iteration showing energy minimization. Maximum force vs. iteration addressing force convergence. Cumulative displacement vs. iteration depicting geometric changes. Step length vs. iteration demonstrating how the optimization step size evolves.

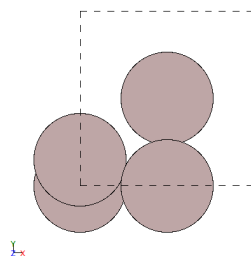


Figure 20: Initial crystal structure (atom indexed 1 at $y = 0.6$).

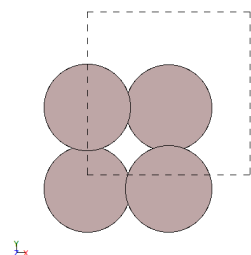


Figure 21: Crystal structure after BFGS optimization.

7 ENVIRONMENT SETUP

This reference guide specifies user-dependent components of Python codes for performing Density Functional Theory (DFT) calculations using the GPAW (Grid-based Projector Augmented Wave method) software package. The codes are organized in the directory `HW_1_codes` after `tag.gz` compression. The sub-directories address various computational tasks including convergence testing, energy calculations, and geometry optimization. In general, one can reproduce and verify the validity of the experiments across the report.

Remark 7.1. Codes on directories `./Symmetry_and_Basis/` and `./Optim_DFT/` require a proper GPAW (and thus ASE) environment setup. Before running any calculation, users must modify the path settings to match their local installation.

The following code block appears in all scripts of the aforementioned sub-directories and requires user-specific modifications:

```
# Suppress detailed GPAW output
os.environ['GPAW_VERBOSE'] = 'o'
sys.stdout = open(os.devnull, 'w') # Redirect standard output

# Clear existing paths and set the new one [user-dependent]
intended_path = os.path.expanduser('~/Desktop/DFT_codes/gpaw_datasets/gpaw-  
setup_paths[:] = [intended_path] # Replace all existing paths
os.environ['GPAW_SETUP_PATH'] = intended_path

sys.stdout = sys.__stdout__ # Restore standard output
print("GPAW looking for datasets in:", setup_paths)
print("Environment GPAW_SETUP_PATH:", os.environ['GPAW_SETUP_PATH'])
```

7.1 HOW TO MODIFY THE PATH SETTINGS

Users must update the `intended_path` variable to match their local GPAW dataset installation:

1. Locate your GPAW setups directory (typically installed with GPAW or downloaded separately)
2. Replace `'~/Desktop/DFT_codes/gpaw_datasets/gpaw-setups-0.9.20000'` with the path to your setups directory
3. Ensure the path format is appropriate for your operating system:
 - Linux/macOS: Use `'/path/to/gpaw-setups'` .
 - Windows: Use `'C:/path/to/gpaw-setups'` or `r'C:\path\to\gpaw-setups'` .

REFERENCES

- [Lém97] E.-M. LÉMERAY. “Sur la convergence des substitutions uniformes”. In: *Nouvelles annales de mathématiques : journal des candidats aux écoles polytechnique et normale*. Serie 3 16 (1897), pp. 306–319. URL: http://www.numdam.org/item/NAM_1897_3_16__306_1/.
- [HK64] P. HOHENBERG and W. KOHN. “Inhomogeneous Electron Gas”. In: *Phys. Rev.* 136 (3B Nov. 1964), B864–B871. DOI: 10.1103/PhysRev.136.B864. URL: <https://link.aps.org/doi/10.1103/PhysRev.136.B864>.
- [Bro65] C. G. BROYDEN. “A class of methods for solving nonlinear simultaneous equations”. In: *Mathematics of Computation* 19 (1965), pp. 577–593. DOI: 10.1090/s0025-5718-1965-0198670-6.
- [KS65] W. KOHN and L. J. SHAM. “Self-Consistent Equations Including Exchange and Correlation Effects”. In: *Phys. Rev.* 140 (4A Nov. 1965), A1133–A1138. DOI: 10.1103/PhysRev.140.A1133. URL: <https://link.aps.org/doi/10.1103/PhysRev.140.A1133>.
- [BH72] U von BARTH and L HEDIN. “A local exchange-correlation potential for the spin polarized case. i”. In: *Journal of Physics C: Solid State Physics* 5.13 (July 1972), p. 1629. DOI: 10.1088/0022-3719/5/13/012. URL: <https://dx.doi.org/10.1088/0022-3719/5/13/012>.
- [MP76] Hendrik J. MONKHORST and James D. PACK. “Special points for Brillouin-zone integrations”. In: *Phys. Rev. B* 13 (12 June 1976), pp. 5188–5192. DOI: 10.1103/PhysRevB.13.5188. URL: <https://link.aps.org/doi/10.1103/PhysRevB.13.5188>.
- [Pul80] Peter PULAY. “Convergence acceleration of iterative sequences. the case of scf iteration”. In: *Chemical Physics Letters* 73 (1980), pp. 393–398. URL: <https://api.semanticscholar.org/CorpusID:94811991>.
- [Har81] John E. HARRIMAN. “Orthonormal orbitals for the representation of an arbitrary density”. In: *Phys. Rev. A* 24 (2 Aug. 1981), pp. 680–682. DOI: 10.1103/PhysRevA.24.680. URL: <https://link.aps.org/doi/10.1103/PhysRevA.24.680>.
- [Ker81] G. P. KERKER. “Efficient iteration scheme for self-consistent pseudopotential calculations”. In: *Phys. Rev. B* 23 (6 Mar. 1981), pp. 3082–3084. DOI: 10.1103/PhysRevB.23.3082. URL: <https://link.aps.org/doi/10.1103/PhysRevB.23.3082>.
- [DZ83] P. H. DEDERICHS and R. ZELLER. “Self-consistency iterations in electronic-structure calculations”. In: *Phys. Rev. B* 28 (10 Nov. 1983), pp. 5462–5472. DOI: 10.1103/PhysRevB.28.5462. URL: <https://link.aps.org/doi/10.1103/PhysRevB.28.5462>.
- [Löw85] Per-Olov LÖWDIN. “Twenty-five years of Sanibel symposia: A brief historic and scientific survey”. In: *International Journal of Quantum Chemistry* 28.S19 (1985), pp. 19–37. DOI: <https://doi.org/10.1002/qua.560280805>. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/qua.560280805>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/qua.560280805>.
- [Fle88] Roger FLETCHER. “Practical Methods of Optimization”. In: 1988. URL: <https://api.semanticscholar.org/CorpusID:123487779>.

- [LYP88] Chengteh LEE, Weitao YANG, and Robert G. PARR. “Development of the Colle-Salvetti correlation-energy formula into a functional of the electron density”. In: *Phys. Rev. B* 37 (2 Jan. 1988), pp. 785–789. DOI: 10.1103/PhysRevB.37.785. URL: <https://link.aps.org/doi/10.1103/PhysRevB.37.785>.
- [LN89] Dong C. LIU and Jorge NOCEDAL. “On the limited memory BFGS method for large scale optimization”. In: *Mathematical Programming* 45.1 (1989), pp. 503–528. DOI: 10.1007/BF01589116. URL: <https://doi.org/10.1007/BF01589116>.
- [DR90] B.I. DUNLAP and N. ROSCH. “The Gaussian-Type Orbitals Density-Functional Approach to Finite Systems”. In: *Density Functional Theory of Many-Fermion Systems*. Ed. by Per-Olov LÖWDIN. Vol. 21. Advances in Quantum Chemistry. Academic Press, 1990, pp. 317–339. DOI: [https://doi.org/10.1016/S0065-3276\(08\)60603-6](https://doi.org/10.1016/S0065-3276(08)60603-6). URL: <https://www.sciencedirect.com/science/article/pii/S0065327608606036>.
- [HKS90] P.C. HOHENBERG, Walter KOHN, and L.J. SHAM. “The Beginnings and Some Thoughts on the Future”. In: *Density Functional Theory of Many-Fermion Systems*. Ed. by Per-Olov LÖWDIN. Vol. 21. Advances in Quantum Chemistry. Academic Press, 1990, pp. 7–26. DOI: [https://doi.org/10.1016/S0065-3276\(08\)60589-4](https://doi.org/10.1016/S0065-3276(08)60589-4). URL: <https://www.sciencedirect.com/science/article/pii/S0065327608605894>.
- [Bec93] Axel D. BECKE. “Density-functional thermochemistry. III. The role of exact exchange”. In: *The Journal of Chemical Physics* 98.7 (Apr. 1993), pp. 5648–5652. ISSN: 0021-9606. DOI: 10.1063/1.464913. eprint: https://pubs.aip.org/aip/jcp/article-pdf/98/7/5648/19277469/5648_1_online.pdf. URL: <https://doi.org/10.1063/1.464913>.
- [DMG96] Frank DE PROFT, Jan M.L. MARTIN, and Paul GEERLINGS. “On the performance of density functional methods for describing atomic populations, dipole moments and infrared intensities”. In: *Chemical Physics Letters* 250.3 (1996), pp. 393–401. ISSN: 0009-2614. DOI: [https://doi.org/10.1016/0009-2614\(96\)00057-7](https://doi.org/10.1016/0009-2614(96)00057-7). URL: <https://www.sciencedirect.com/science/article/pii/0009261496000577>.
- [KF96] G. KRESSE and J. FURTHMÜLLER. “Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set”. In: *Phys. Rev. B* 54 (16 Oct. 1996), pp. 11169–11186. DOI: 10.1103/PhysRevB.54.11169. URL: <https://link.aps.org/doi/10.1103/PhysRevB.54.11169>.
- [PBE96] John P. PERDEW, Kieron BURKE, and Matthias ERNZERHOF. “Generalized Gradient Approximation Made Simple”. In: *Phys. Rev. Lett.* 77 (18 Oct. 1996), pp. 3865–3868. DOI: 10.1103/PhysRevLett.77.3865. URL: <https://link.aps.org/doi/10.1103/PhysRevLett.77.3865>.
- [SR96] Anthony P. SCOTT and Leo RADOM. “Harmonic Vibrational Frequencies: An Evaluation of HartreeFock, MøllerPlesset, Quadratic Configuration Interaction, Density Functional Theory, and Semiempirical Scale Factors”. In: *The Journal of Physical Chemistry* 100.41 (1996), pp. 16502–16513. DOI: 10.1021/jp960976r. eprint: <https://doi.org/10.1021/jp960976r>. URL: <https://doi.org/10.1021/jp960976r>.

- [VDG97] Gregory VAN LIER, Frank DE PROFT, and Paul GEERLINGS. “Cost effective calculation of molecular charge distributions and gas phase deprotonation energies using density functional methods”. In: *Chemical Physics Letters* 274.4 (1997), pp. 396–404. ISSN: 0009-2614. DOI: [https://doi.org/10.1016/S0009-2614\(97\)00640-4](https://doi.org/10.1016/S0009-2614(97)00640-4). URL: <https://www.sciencedirect.com/science/article/pii/S0009261497006404>.
- [RPF99] Paul R. RABLEN, Shoshannah A. PEARLMAN, and John FINKBINER. “A Comparison of Density Functional Methods for the Estimation of Proton Chemical Shifts with Chemical Accuracy”. In: *The Journal of Physical Chemistry A* 103.36 (1999), pp. 7357–7363. DOI: 10.1021/jp9916889. eprint: <https://doi.org/10.1021/jp9916889>. URL: <https://doi.org/10.1021/jp9916889>.
- [DTG00] F. DE PROFT, F. TIELENS, and P. GEERLINGS. “Performance and basis set dependence of density functional theory dipole and quadrupole moments”. In: *Journal of Molecular Structure: THEOCHEM* 506.1-3 (2000), pp. 1–8. ISSN: 0166-1280. DOI: 10.1016/S0166-1280(00)00397-3.
- [KH01] W KOCH and M. HOLTHAUSEN. *A Chemist Guide to Density Functional Theory*. Wiley-VCH, Jan. 2001. ISBN: 9783527303724. DOI: 10.1002/3527600043.ch1.
- [Sav03] George Gustav SAVII. “Computer-Aided Design”. In: *Encyclopedia of Information Systems*. Ed. by Hossein BIDGOLI. New York: Elsevier, 2003, pp. 171–186. ISBN: 978-0-12-227240-0. DOI: <https://doi.org/10.1016/B0-12-227240-4/00011-3>. URL: <https://www.sciencedirect.com/science/article/pii/B0122272404000113>.
- [Fri+10] M. J. FRISCH et al. *Gaussian 09, Revision B.01*. 2010. *Gaussian, Inc., Wallingford, CT*.
- [SSH1] D.S. SHOLL and J.A. STECKEL. *Density Functional Theory: A Practical Introduction*. Wiley, 2011. ISBN: 9781118211045. URL: https://books.google.gr/books?id=_f994dmAdv0C.
- [KW13] Jennifer L. KELLIE and Stacey D. WETMORE. “Selecting DFT methods for use in optimizations of enzyme active sites: applications to ONIOM treatments of DNA glycosylases”. In: *Canadian Journal of Chemistry* 91.7 (2013), pp. 559–572. DOI: 10.1139/cjc-2012-0506. eprint: <https://doi.org/10.1139/cjc-2012-0506>. URL: <https://doi.org/10.1139/cjc-2012-0506>.
- [Fau16] A.C. FAUL. *A Concise Introduction to Numerical Analysis*. A Chapman & Hall book. CRC Press, Taylor & Francis Group, 2016. ISBN: 9781498712187. URL: <https://books.google.gr/books?id=PhVLjgEACAAJ>.
- [WMM16] Pandu WISESA, Kyle MCGILL, and Tim MUELLER. “Efficient generation of generalized Monkhorst-Pack grids through the use of informatics”. In: *Physical Review B* 93 (Apr. 2016). DOI: 10.1103/PhysRevB.93.155109.
- [Joh18] Mary JOHNSON. *The Art of Cooking*. French. Revised. Paris: Culinary Press, 2018. ISBN: 978-3-0456-9975-0. *A collection of delicious recipes and culinary techniques*.
- [Woo18] Nick WOODS. *On the Nature of Self-Consistency in Density Functional Theory*. 2018. arXiv: 1803.01763 [cond-mat.other]. URL: <https://arxiv.org/abs/1803.01763>.

- [BC20] Joe BROWN and Joe CHRISTMAS. *Introduction to Physics*. English. 2nd. New York: New Publishers, 2020. ISBN: 978-2-2453-9858-6.
A comprehensive guide to introductory physics.
- [Gar22] Carlos GARCIA. *Programming Fundamentals*. English. 1st. San Francisco: Tech Books, 2022. ISBN: 978-1-5155-6535-2.
An introduction to programming concepts and languages.
- [Tog+23] Atsushi TOGO et al. “Implementation strategies in phonopy and phonopy”. In: *Journal of Physics: Condensed Matter* 35 (June 2023). DOI: 10.1088/1361-648X/acd831.
- [ZZ23] X. H. ZHENG and J. X. ZHENG. “On the use of Monkhorst–Pack scheme to evaluate superconductivity and the issue of umklapp electron–phonon interactions”. In: *Phys. Chem. Chem. Phys.* 25 (18 2023), pp. 13049–13060. DOI: 10.1039/D3CP01053H. URL: <http://dx.doi.org/10.1039/D3CP01053H>.
- [Zha+24] He ZHANG et al. *Self-Consistency Training for Density-Functional-Theory Hamiltonian Prediction*. 2024. arXiv: 2403.09560 [cs.LG]. URL: <https://arxiv.org/abs/2403.09560>.