Introduction to the Plane-Wave pseudopotential method

P. Giannozzi
Università di Udine and IOM-Democritos, Trieste, Italy

Penn State University, 16 June 2014

Many pictures courtesy of Shobhana Narasimhan, JNCASR

DEMOCRITOS
DEmocritos MOdeling Center for Research In aTOmistic Simulation

Outline

1. Solution of the DFT problem: Self-consistency, global minimization

2. Hellmann-Feynman forces, structural optimization, molecular dynamics

3. Plane waves basis set

4. Crystals: periodicity, direct and reciprocal lattice, unit cell, supercells

5. Pseudopotentials

6. Plane-wave-pseudopotential method and technicalities

1. Towards electronic ground state (fixed nuclei)

Possible methods to find the DFT ground state:

1. By the *self-consistent* solution of the Kohn-Sham equations

$$H_{KS}\psi_i \equiv \left(T + V + V_H[n] + V_{xc}[n]\right)\psi_i = \epsilon_i\psi_i$$

   where

   - $n(\mathbf{r}) = \sum_i f_i|\psi_i(\mathbf{r})|^2$ is the charge density, f_i are occupation numbers
   - V is the nuclear (pseudo-)potential acting on electrons (may be nonlocal)
   - $V_H[n]$ is the Hartree potential, $V_H(\mathbf{r}) = e^2\int\frac{n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}d\mathbf{r}'$
   - $V_{xc}[n]$ is the exchange-correlation potential. For the simplest case, LDA, $V_{xc}[n]$ is a *function* of the charge density at point $\mathbf{r}$: $V_{xc}(\mathbf{r}) \equiv \mu_{xc}(n(\mathbf{r}))$

   Orthonormality constraints $\langle\psi_i|\psi_j\rangle = \delta_{ij}$ automatically hold.

2. By *constrained global minimization* of the energy functional

$$E[\psi] = \sum_i f_i \langle \psi_i | T + V | \psi_i \rangle + E_H[n] + E_{xc}[n]$$

under orthonormality constraints $\langle \psi_i | \psi_j \rangle = \delta_{ij}$, i.e. minimize:

$$\widetilde{E}[\psi, \Lambda] = E[\psi] - \sum_{ij} \Lambda_{ij} \left(\langle \psi_i | \psi_j \rangle - \delta_{ij} \right)$$

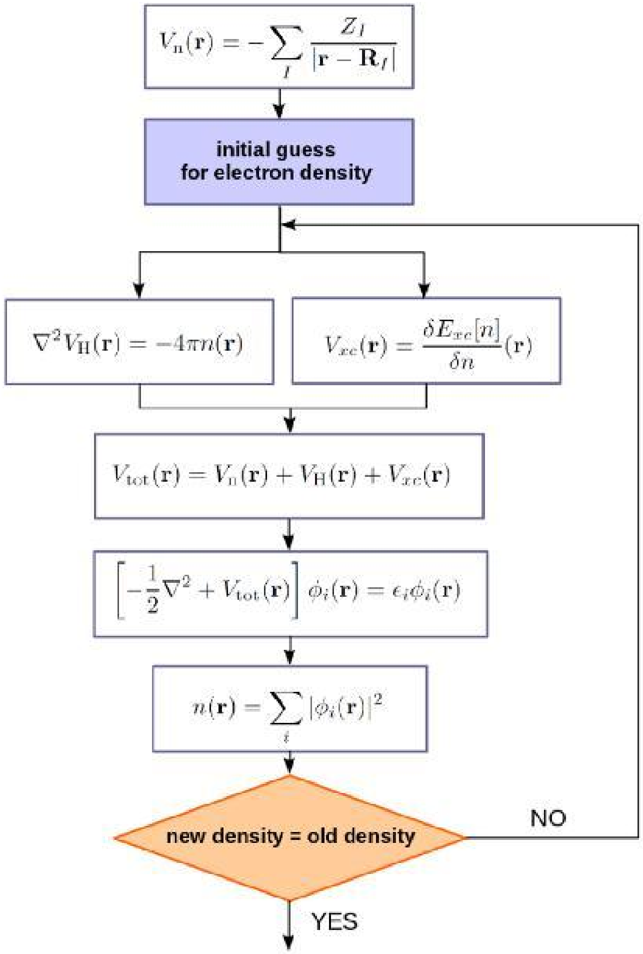
where

- V, $n(\mathbf{r})$ are defined as before, $\psi \equiv$ all occupied Kohn-Sham orbitals
- Λ_{ij} are Lagrange multipliers, $\Lambda \equiv$ all of them
- $E_H[n]$ is the Hartree energy, $E_H = \dfrac{e^2}{2} \displaystyle\int \dfrac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r} d\mathbf{r}'$
- $E_{xc}[n]$ is the exchange-correlation energy. For the simplest case, LDA, $E_{xc} = \displaystyle\int n(\mathbf{r}) \epsilon_{xc}(n(\mathbf{r})) d\mathbf{r}$ where ϵ_{xc} is a function of $n(\mathbf{r})$.

Towards electronic ground state II

In a self-consistent approach, we need to find the self-consistent charge density (or potential), performing the following operations:

- Calculate the potential from the charge density
- Solve (*diagonalize*) the Kohn-Sham equations at fixed potential
- Calculate the charge density from Kohn-Sham orbitals


$V_n(\mathbf{r}) = -\sum_I \frac{Z_I}{|\mathbf{r}-\mathbf{R}_I|}$
initial guess
for electron density
$\nabla^2 V_H(\mathbf{r}) = -4\pi n(\mathbf{r})$
$V_{xc}(\mathbf{r}) = \frac{\delta E_{xc}[n]}{\delta n}(\mathbf{r})$
$V_{tot}(\mathbf{r}) = V_n(\mathbf{r}) + V_H(\mathbf{r}) + V_{xc}(\mathbf{r})$
$\left[-\frac{1}{2}\nabla^2 + V_{tot}(\mathbf{r})\right]\phi_i(\mathbf{r}) = \epsilon_i\phi_i(\mathbf{r})$
$n(\mathbf{r}) = \sum_i |\phi_i(\mathbf{r})|^2$
new density = old density
NO
YES

In a global-minimization approach, the operations are basically the same, since one needs the *gradients* of the energy functional, that is, $H_{KS}\psi$ products:

$$\frac{\delta \widetilde{E}[\psi,\Lambda]}{\delta \psi_j^*} = H_{KS}\psi_j - \sum_i \Lambda_{ji}\psi_i$$

Towards electronic ground state III

Let us start from some guess of the input charge density $n^{in}(\mathbf{r})$:

$$n^{in} \longrightarrow (V_H + V_{xc})[n^{in}] \longrightarrow \psi_i(\mathbf{r}) \longrightarrow n^{out}(\mathbf{r}) = \sum_i f_i |\psi_i(\mathbf{r})|^2$$

Such procedure defines the output charge density as a *functional* of the input one: $n^{out} \equiv F[n^{in}]$. Assuming we have a black box producing $F[n]$, we have to reach self-consistency, that is, find n^{gs} such that $n^{gs} = F[n^{gs}]$

Simply re-inserting n^{out} as n^{in} *is not guaranteed to converge* (it seldom does!). Reason: there is no guarantee that such procedure leads to a reduction of all component of the error (in particular, in typical condensed-matter systems low-frequency, small-**G** components of the error are not reduced). One can use a simple **mixing algorithm**:

$$n^{new} = \alpha n^{out} + (1-\alpha) n^{in}, \qquad 0 < \alpha < 1$$

guaranteed to converge if α is small enough.

Practical, more sophisticated algorithms (Anderson, Broyden) use the input and output of several preceding steps to determine the next optimal input combination.

Calculation of the total energy

Once self-consistency (or the minimum) is reached, the **total energy** of the system can be calculated:

$$E = \sum_i f_i \langle \psi_i | T + V | \psi_i \rangle + E_H[n] + E_{xc}[n] + E_{ion-ion}$$

where $E_{ion-ion}$ is the repulsive contribution from nuclei to the energy:

$$E_{ion-ion} = \frac{e^2}{2} \sum_{\mu \neq \nu} \frac{Z_\mu Z_\nu}{\mid \mathbf{R}_\mu - \mathbf{R}_\nu \mid}$$

Equivalent expression for the energy, using Kohn-Sham eigenvalues:

$$E = \sum_i f_i \epsilon_i - E_H[n] + E_{xc}[n(\mathbf{r})] - \int n(\mathbf{r}) V_{xc}[n(\mathbf{r})] d\mathbf{r} + E_{ion-ion}$$

The total energy depends upon all atomic positions $\mathbf{R}_\mu$.

2. Hellmann-Feynman Forces

Forces on atoms are the derivatives of the total energy wrt atomic positions. The *Hellmann-Feynman theorem* tells us that forces are the expectation value of the derivative of the external potential only:

$$\mathbf{F}_\mu = -\frac{\partial E}{\partial \mathbf{R}_\mu} = -\sum_i f_i \langle \psi_i | \frac{\partial V}{\partial \mathbf{R}_\mu} | \psi_i \rangle = -\int n(\mathbf{r}) \frac{\partial V}{\partial \mathbf{R}_\mu} d\mathbf{r}$$

the rightmost expression being valid only for *local* potentials, $V \equiv V(\mathbf{r})$ (the one at the left is more general, being valid also for nonlocal potentials $V \equiv V(\mathbf{r}, \mathbf{r}')$).

Demonstration (simplified). In addition to the explicit derivative of the external potential (first term), there is an implicit dependency via the derivative of the charge density:

$$\frac{\partial E}{\partial \mathbf{R}_\mu} = \int n(\mathbf{r}) \frac{\partial V}{\partial \mathbf{R}_\mu} d\mathbf{r} + \int \frac{\delta E}{\delta n(\mathbf{r})} \frac{\partial n(\mathbf{r})}{\partial \mathbf{R}_\mu} d\mathbf{r}$$

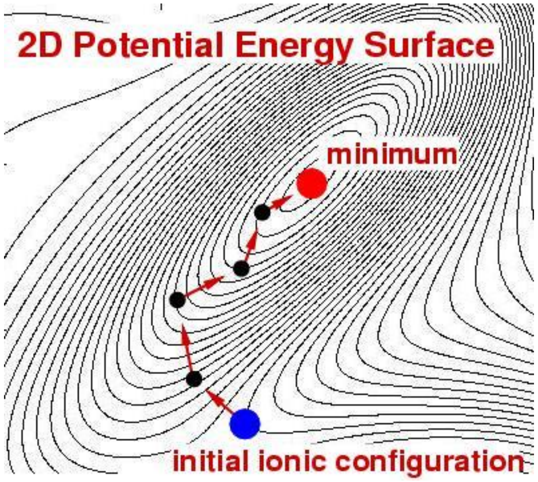
The green term cancels due to the *variational character* of DFT: $\delta E / \delta n(\mathbf{r}) = \mu$, constant.

The calculation of the Hellmann-Feynman forces is straightforward (in principle, not necessarily in practice!) once the self-consistent electronic structure is calculated.

Structural Optimization and Molecular Dynamics

Within the *Born-Oppenheimer*, or *adiabatic* approximation, the total energy as a function of atomic positions, or *Potential Energy Surface* (PES), determines the behaviour of nuclei.

The *global* ground state can be found by minimizing the function $E(\mathbf{R}_1, \mathbf{R}_2, ..., \mathbf{R}_N)$, depending upon the $3N$ atomic coordinates for a system of N atoms. This is a "standard" mathematical problem: finding the minimum of a function, knowing its derivatives, that is, the Hellmann-Feynman forces (in the picture, a cartoon of a PES in two dimensions with the path to the minimum).



Once forces are calculated, one can perform not only structural optimization, but also *molecular dynamics*. If a classical behaviour of the nuclei is assumed, all the machinery of classical MD can be recycled, with forces calculated from *first principles*.

3. Diagonalization of the KS Hamiltonian

The solution of the Kohn-Sham problem $H_{KS}\psi = \epsilon\psi$ at fixed potential is (usually) the toughest problem. How to proceed? By expanding ψ into some suitable **basis set** $\{\phi_i\}$ as

$$\psi(\mathbf{r}) = \sum_i c_i \phi_i(\mathbf{r}).$$

For an orthonormal basis set, solve the secular equations

$$\sum_j (H_{ij} - \epsilon\delta_{ij})c_j = 0$$

where $H_{ij} = \langle\phi_i|H_{KS}|\phi_j\rangle$ are the *matrix elements* of the Hamiltonian.
For a non-orthonormal basis set, solve the generalized problem:

$$\sum_j (H_{ij} - \epsilon S_{ij})c_j$$

where $S_{ij} = \langle\phi_i|\phi_j\rangle$ is the *overlap matrix*.

Diagonalization algorithms are well known in linear algebra, but in practice, one has to resort to smarter *iterative* algorithms, allowing *not* to store those matrices.

Most popular basis sets

We have to choose now a suitable basis set. Typical candidates include

- *Localized* basis sets:
  atom-centred functions such as
  - Linear Combinations of Atomic Orbitals (LCAO)
  - Gaussian-type Orbitals (GTO)
  - Linearized Muffin-Tin Orbitals (LMTO)
- *Delocalized* basis sets:
  - Plane Waves (PW)

One could also consider *mixed* basis sets. The Linearized Augmented Plane Waves (LAPW) could be classified in this category.

Advantages and disadvantages of various basis sets

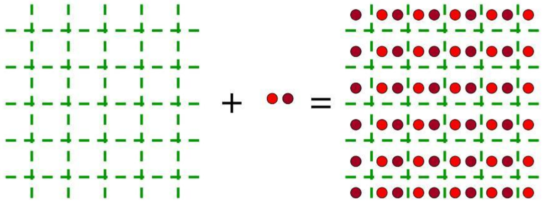
- Localized basis sets:
  - + fast convergence with respect to basis set size (just a few functions per atom needed)
  - + can be used in finite as well as in periodic systems (as Bloch sums: $\phi_{\mathbf{k}} = \sum_{\mathbf{R}} e^{-i\mathbf{k}\cdot\mathbf{R}}\phi(\mathbf{r}-\mathbf{R})$)
  - – difficult to evaluate convergence quality (no systematic way to improve convergence)
  - – difficult to use (two- and three-centre integrals)
  - – difficult to calculate forces (*Pulay forces* if basis set is not complete)

- Plane Waves:
  - – slow convergence with respect to basis set size (many more PWs than localized functions needed)
  - – require periodicity: in finite systems, *supercells* must be introduced
  - + easy to evaluate convergence quality (just increase a single parameter, the *cutoff*)
  - + easy to use (Fourier transform)
  - + easy to calculate forces (no Pulay forces even if the basis set is incomplete)

4. Periodicity

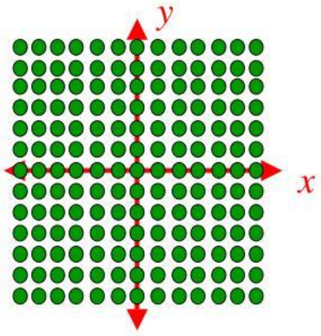
Let us focus on the case of the *infinite perfect crystals*, having translation symmetry. A perfect crystal is described in terms of

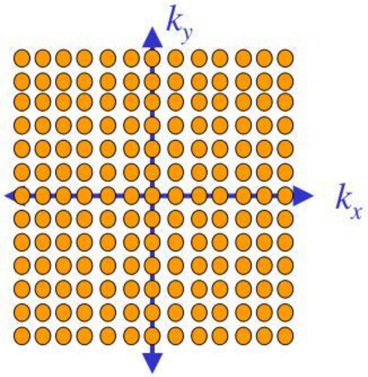
- a periodically repeated **unit cell** and a **lattice** of translation vectors $\mathbf{R} = n_1\mathbf{R}_1 + n_2\mathbf{R}_2 + n_3\mathbf{R}_3$, defined via three primitive vectors $\mathbf{R}_1, \mathbf{R}_2, \mathbf{R}_3$ and integer coefficients n_1, n_2, n_3.



- a **basis** of atomic positions $\mathbf{d}_i$ into the unit cell

- a **reciprocal lattice** of vectors $\mathbf{G}$ such that $\mathbf{G} \cdot \mathbf{R} = 2\pi l$, with l integer:
  $\mathbf{G} = m_1\mathbf{G}_1 + m_2\mathbf{G}_2 + m_3\mathbf{G}_3$ with $\mathbf{G}_i \cdot \mathbf{R}_j = 2\pi\delta_{ij}$ and m_1, m_2, m_3 integer.



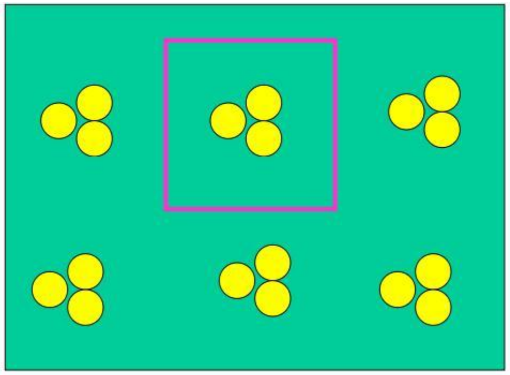


Non periodic systems: supercells

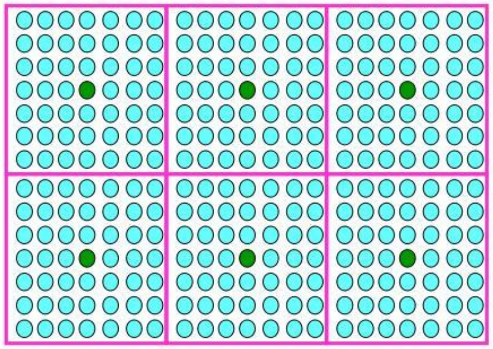
What about e.g. defects in crystals, surfaces, alloys, amorphous materials, liquids, molecules, clusters? none of these has perfect periodicity! One can use **supercells**, introducing an artificial periodicity.

The supercell geometry is dictated by the type of system under investigation:

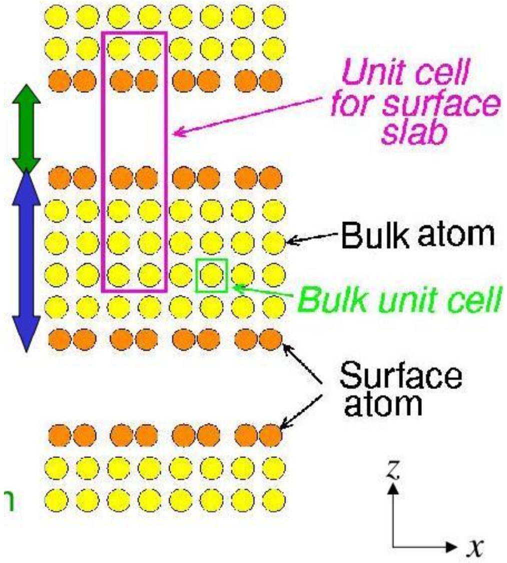
- Molecules, clusters:
  the supercell must allow a minimum distance of at least a few A (~ 6) between the closest atoms in different periodic replica.



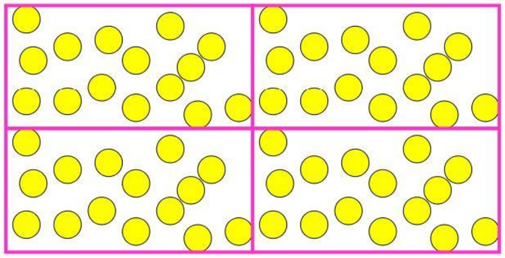
- Defects in crystals:
  the supercell is commensurate with the perfect crystal cell. The distance between periodic replica of the defect must be "big enough" to minimize spurious defect-defect interactions.



- Surfaces:
  slab geometry. The number of layers of the materials must be "big enough" to have "bulk behaviour" in the furthest layer from the surface. The number of empty layers must be "big enough" to have minimal interactions between layers in different regions.



- Alloys, amorphous materials, liquids:
  the supercell must be "big enough" to give a reasonable description of physical properties.



Conceptually there is no difference between a "supercell" and an ordinary unit cell: typically, "supercell" is used when the periodicity is not perfect or non-existent

Band Structure, Bloch states

The one-electron states $\psi(\mathbf{r})$ of a perfect crystal Hamiltonian $H = T + V$ are described by a **band index** i and a **wave vector** $\mathbf{k}$.

It is convenient to consider the *thermodynamic limit*: a slab of crystal composed of $\mathcal{N} = \mathcal{N}_1\mathcal{N}_2\mathcal{N}_3$ unit cells, $\mathcal{N} \to \infty$, obeying Periodic Boundary Conditions:

$$\psi(\mathbf{r} + \mathcal{N}_1\mathbf{R}_1) = \psi(\mathbf{r} + \mathcal{N}_2\mathbf{R}_2) = \psi(\mathbf{r} + \mathcal{N}_3\mathbf{R}_3) = \psi(\mathbf{r}).$$

There are $\mathcal{N}$ wave vectors $\mathbf{k}$ in the unit cell of the reciprocal lattice, called the **Brillouin Zone**. The one-electron states (energy bands) can be written as

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

where $u_{i,\mathbf{k}}(\mathbf{r})$ is translationally invariant:

$$u_{i,\mathbf{k}}(\mathbf{r} + \mathbf{R}) = u_{i,\mathbf{k}}(\mathbf{r}).$$

Plane-wave basis set

A PW basis set for states of wave vector $\mathbf{k}$ is defined as

$$\langle \mathbf{r}|\mathbf{k}+\mathbf{G}\rangle = \frac{1}{\sqrt{\mathcal{N}\Omega}} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}}, \qquad \frac{\hbar^2}{2m}|\mathbf{k}+\mathbf{G}|^2 \le E_{cut}$$

$\Omega =$ unit cell volume, $\mathcal{N}\Omega =$ crystal volume, $E_{cut} =$ cutoff on the kinetic energy of PWs (in order to have a finite number of PWs!). The PW basis set is *complete* for $E_{cut} \to \infty$ and *orthonormal*: $\langle \mathbf{k}+\mathbf{G}|\mathbf{k}+\mathbf{G}'\rangle = \delta_{\mathbf{G}\mathbf{G}'}$

The components on a PW basis set are the *Fourier transform*:

$$|\psi_i\rangle = \sum_{\mathbf{G}} c_{i,\mathbf{k}+\mathbf{G}} |\mathbf{k}+\mathbf{G}\rangle$$

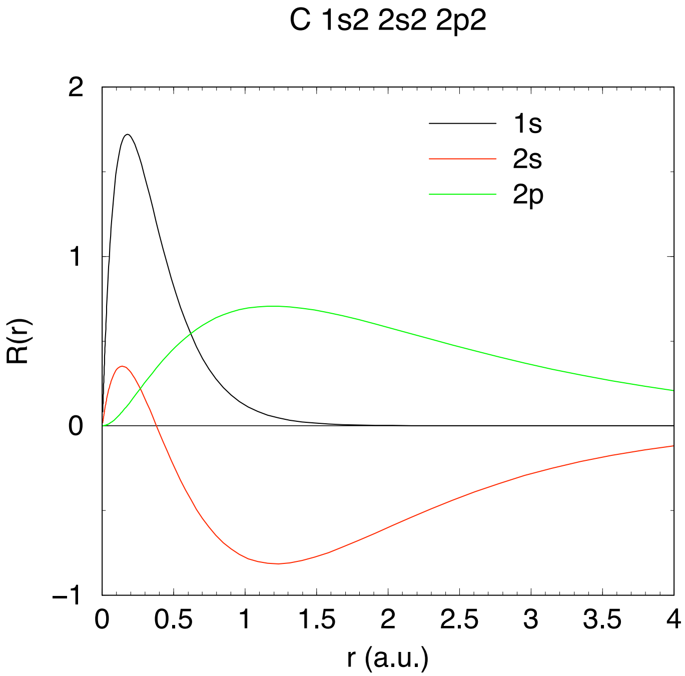
$$c_{i,\mathbf{k}+\mathbf{G}} = \langle \mathbf{k}+\mathbf{G}|\psi_i\rangle = \frac{1}{\sqrt{\mathcal{N}\Omega}} \int \psi_i(\mathbf{r}) e^{-i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}} d\mathbf{r} = \widetilde{\psi}_i(\mathbf{k}+\mathbf{G}).$$

5. The need for Pseudopotentials

Are PWs a practical basis set for electronic structure calculations? Not really!
From simple Fourier analysis: length scale $\delta \longrightarrow$ Fourier components up to $q \sim 2\pi/\delta$.
In a solid, this means $\sim 4\pi(2\pi/\delta)^3/3\Omega_{BZ}$ PWs (volume of the sphere of radius q divided by $\Omega_{BZ} = 8\pi^3/\Omega$, volume of the Brillouin Zone).

Estimate for diamond: $1s$ wavefunction has $\delta \simeq 0.1$ a.u., $\Omega = (2\pi)^3/(a_0^3/4)$ with lattice parameter $a_0 = 6.74$ a.u. $\longrightarrow 250,000$ PWs! We need to:

- get rid of core states
- get rid of orthogonality wiggles close to the nucleus



Solution: **Pseudopotentials** (PP). A smooth effective potential that reproduces the effect of the nucleus plus core electrons on valence electrons.

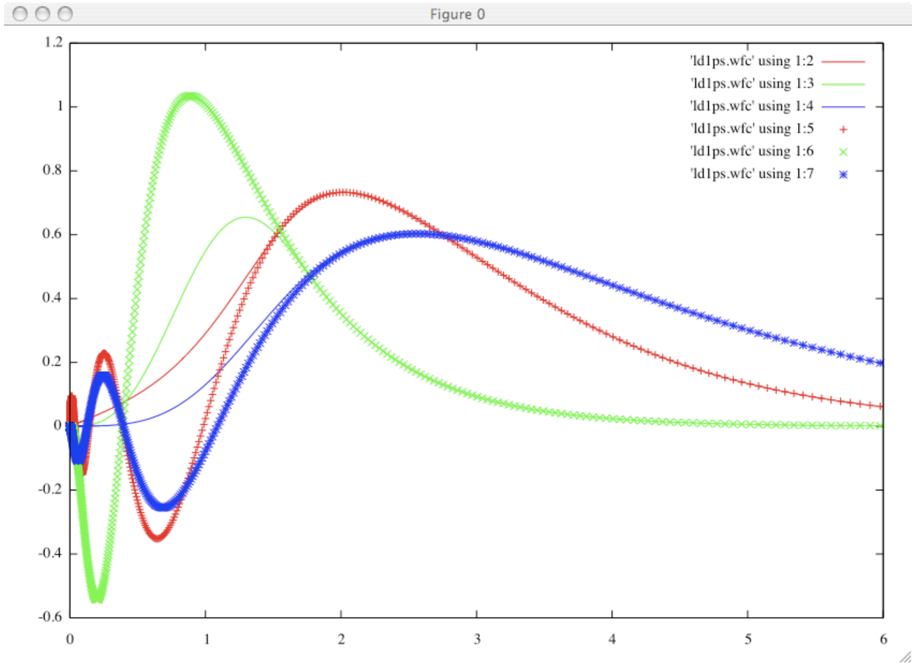
Understanding Pseudopotentials

Smoothness and *transferability* are the relevant keywords:

- We want our pseudopotential and pseudo-orbitals to be as *smooth* as possible so that expansion into plane waves is convenient (i.e. the required kinetic energy cutoff is small)

- We want our pseudopotential to produce pseudo-orbitals that are as close as possible to true ("all-electron") orbitals outside the core region, *for all systems containing a given atom* (in the figure: all-electron and pseudo-orbitals for Si)

Of course, the two goals are usually conflicting!



Pseudopotentials have a long story: let's start from the end.

Understanding PP: Projector-Augmented Waves

Let us look for a linear operator $\widehat{T}$ connecting all-electron orbitals $|\psi_i\rangle$ to pseudo-orbitals $|\widetilde{\psi}_i\rangle$ as in: $|\psi_i\rangle = \widehat{T}|\widetilde{\psi}_i\rangle$. Pseudo-orbitals will be our variational parameters. We write the charge density, energy, etc. using pseudo-orbitals and $\widehat{T}$ instead of all-electron orbitals.

The operator $\widehat{T}$ can be defined in terms of its action on atomic waves (i.e. orbitals at a given energy, not necessarily bound states):

- $|\phi_l\rangle$: set of atomic all-electron waves (bound or unbound states)
- $|\widetilde{\phi}_l\rangle$: corresponding set of atomic pseudo-waves. Beyond some suitable “core radius” R_l, $\widetilde{\phi}_l(r) = \phi_l(r)$; for $r < R_l$, $\widetilde{\phi}_l(r)$ are smooth functions.

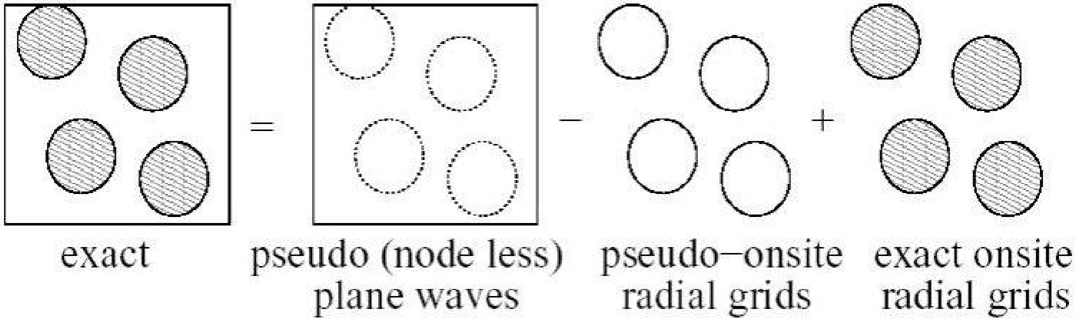
(P. E. Blöchl, PRB **50**, 17953 (1994))

Understanding PP: the PAW transformation

If the above sets are complete in the core region, the operator $\widehat{T}$ can be written as

$$|\psi_i\rangle = \widehat{T}|\widetilde{\psi}_i\rangle = |\widetilde{\psi}_i\rangle + \sum_l \left(|\phi_l\rangle - |\widetilde{\phi}_l\rangle\right)\langle\beta_l|\widetilde{\psi}_i\rangle$$

where the β_l "projectors" are atomic functions, having the properties $\langle\beta_l|\widetilde{\phi}_m\rangle = \delta_{lm}$ and $\beta_l(r) = 0$ for $r > R_l$. The logic is described in the picture below:



The pseudopotential itself is written as a *nonlocal* operator, $\hat{V}$, in terms of the β_l projectors:

$$\hat{V} \equiv V_{loc}(r) + \sum_{lm} |\beta_l\rangle D_{lm}\langle\beta_m|$$

(V_{loc} contains the long-range Coulomb part $-Ze^2/r$)

Understanding PP: Charge in PAW

The (valence) charge density is no longer the simple sum of $|\psi_i|^2$:

$$n(\mathbf{r}) = \sum_i f_i |\psi_i(\mathbf{r})|^2 + \sum_i f_i \sum_{lm} \langle\psi_i|\beta_l\rangle Q_{lm}(\mathbf{r}) \langle\beta_m|\psi_i\rangle,$$

and

$$Q_{lm}(\mathbf{r}) = \phi_l^*(\mathbf{r})\phi_m(\mathbf{r}) - \widetilde{\phi}_l^*(\mathbf{r})\widetilde{\phi}_m(\mathbf{r}).$$

The *augmentation charges* $Q_{lm}(\mathbf{r})$ are zero for $r > R_l$. A generalized orthonormality relation holds for pseudo-orbitals:

$$\langle\psi_i|S|\psi_j\rangle = \int \psi_i^*(\mathbf{r})\psi_j(\mathbf{r})d\mathbf{r} + \sum_{lm} \langle\psi_i|\beta_l\rangle q_{lm} \langle\beta_m|\psi_j\rangle = \delta_{ij}$$

where $q_{lm} = \displaystyle\int Q_{lm}(\mathbf{r})d\mathbf{r}$. The D_{lm} quantites and β_l, Q_{lm} functions are atomic quantities that define the PP (or PAW set).

PP taxonomy: PAW, Ultrasoft, norm-conserving

- In the full PAW scheme, the augmentation functions are calculated and stored on a *radial grid*, centred at each atom. The charge density is composed by a “smooth” term expanded into plane waves, and an “augmentation” term calculated on the radial grid (Kresse and Joubert, PRB59, 1759 (1999))

- In the Ultrasoft PP scheme (D.Vanderbilt, B 41, R7892 (1990)), the augmentation functions $Q_{lm}(r)$ are *pseudized*, i.e. made smoother: both “smooth” and “augmentation” terms can be calculated on a FFT grid, in either reciprocal or real space. The latter term usually requires a larger grid in G-space than the former

- If we set $Q_{lm}(r) = 0$, we obtain good old norm-conserving PPs (Hamann, Schlüter, Chiang 1982) in the separable, nonlocal form.

Which pseudopotentials are good for me?

- **Norm-conserving**:
  + are simple to generate and to use. Theory and methodological improvements are invariably implemented first (and often only) for norm-conserving PPs
  – are relatively *hard*: core radii R_l cannot exceed by much the outermost maximum of the valence atomic orbitals, or else the loss of transferability is large. For some atoms: 2p elements C, N, O, F, 3d transition metals, 4f rare earths, this restriction may lead to very high plane-wave cutoffs (70 Ry and up)
  – do not give any sensible information about the orbitals close to the nucleus (all-electron orbitals can be "reconstructed" using the PAW transformation)

This is usually your first choice and starting point.

Which pseudopotentials are good for me? (II)

- **Ultrasoft**:

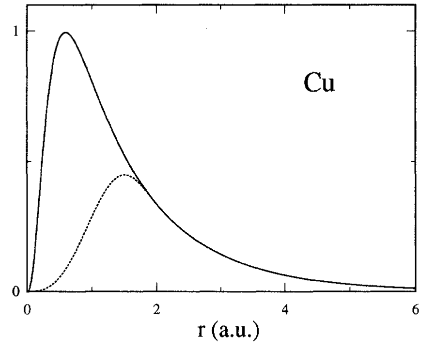
  + can be made *smooth* with little loss of transferability: core radii R_l can be pushed to larger values, even for "difficult cases". Cutoffs of 25 to 35 Ry are usually good for most cases. Note that you may need a second FFT grid for augmentation charges, with typical cutoff 8÷12× orbital cutoff (instead of 4)

  - are not simple to generate: the pseudization of augmentation charges is often a source of trouble (e.g. negative charge)

  - some calculations not available for Ultrasoft PPs

  – give even less information about the orbitals close to the nucleus (all-electron orbitals can be "reconstructed")

Ultrasoft PPs are typically used in all cases where norm-conserving PPs are too hard: C, N, O, F, 3d elements, "semicore" states



Which pseudopotentials are good for me? (III)

- **PAW**:
  - + most transferrable, even for atoms that are “difficult” for Ultrasoft PPs (e.g. magnetic materials): accuracy is comparable to all-electron techniques (e.g. FLAPW)
  - + give information about the orbital close to the nucleus
  - - less well-known and used, less experience available
  - - not all calculations are available for PAW

Which pseudopotentials are good for me? (IV)

There are a few more aspect to be considered in the choice of a pseudopotential:

- *PPs are bound to a specific XC functional*, at least in principle. xception: Hybrid and nonlocal (vdW-DF) functionals, for which very few (or no) PPs are available.
- The distinction between “core” and “valence” electrons is not always clear-cut. In some cases you may need to extend “valence” to include the so-called *semicore states* in order to achieve better (or less lousy) transferability. E.g.: 3d states in Zn and Ga; 3s and 3p states in 3d transition metals Fe, Co, Ni, ...

Inclusion of semicore states adds considerable complexity to both the generation and the practical usage of a PP: to be done only if needed.

Where do I find pseudopotentials?

There are many ready-to-use PPs tables around, but *there is not a single standard PP file format*: each code has its own format.

Quantum ESPRESSO accepts an XML-like format called UPF, plus some old formats. See the pseudopotential page on the web site `http://www.quantum-espresso.org/pseudopotentials/`, for more on

- **PSlibrary**: a project by A. Dal Corso to set up verified PPs for most elements
- other available PPs in UPF format (and their naming convention)
- other pseudopotential repositories, conversion from other formats

If everything else fails, PPs have to be generated.

Pseudopotential testing

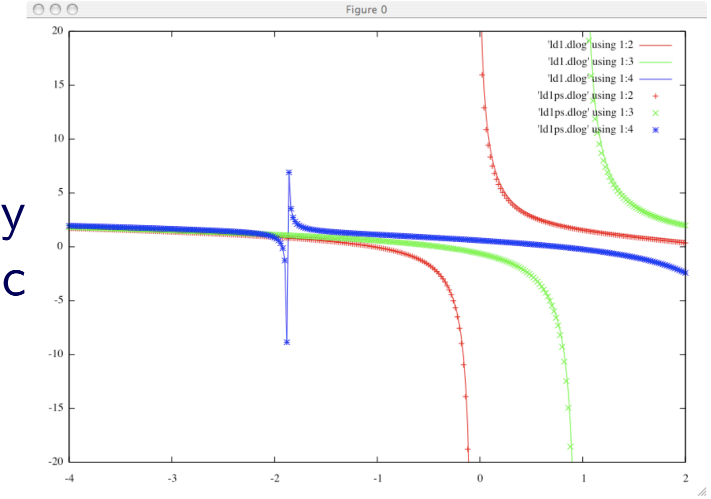
PPs must be *always* tested to check for

- absence of *ghost states*: spurious unphysical states in the valence region of energies, or close to it. All nonlocal PPs can be affected

- poor transferability

Testing can be performed

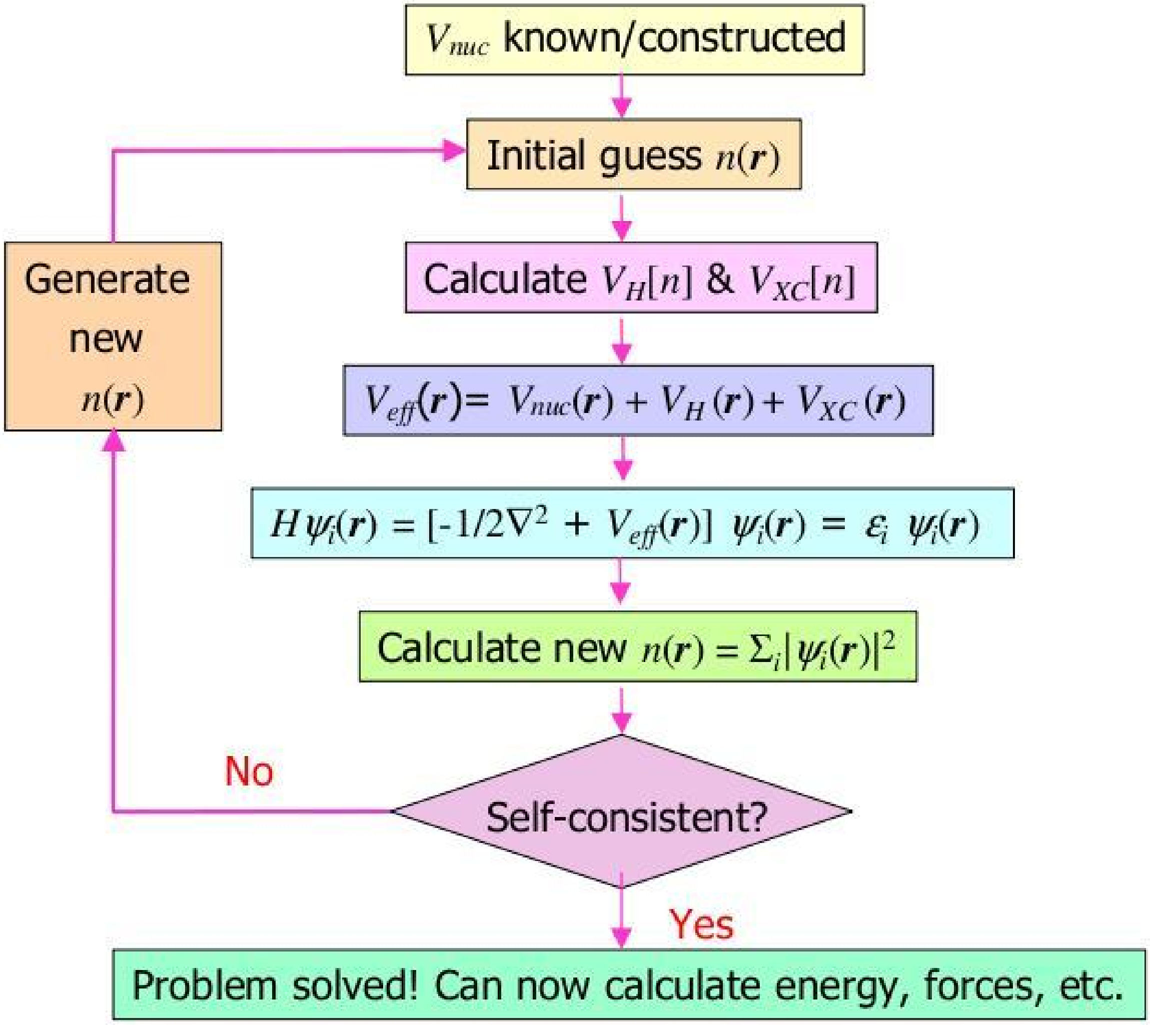
- by the PP generation code itself, by comparing *energy differences* between electronic configurations, and *logarithmic derivatives*;



- in simple molecular or solid-state systems, *ideally* by comparing with accurate all-electron results; less ideally, with other PP results; even less ideally, with experimental data

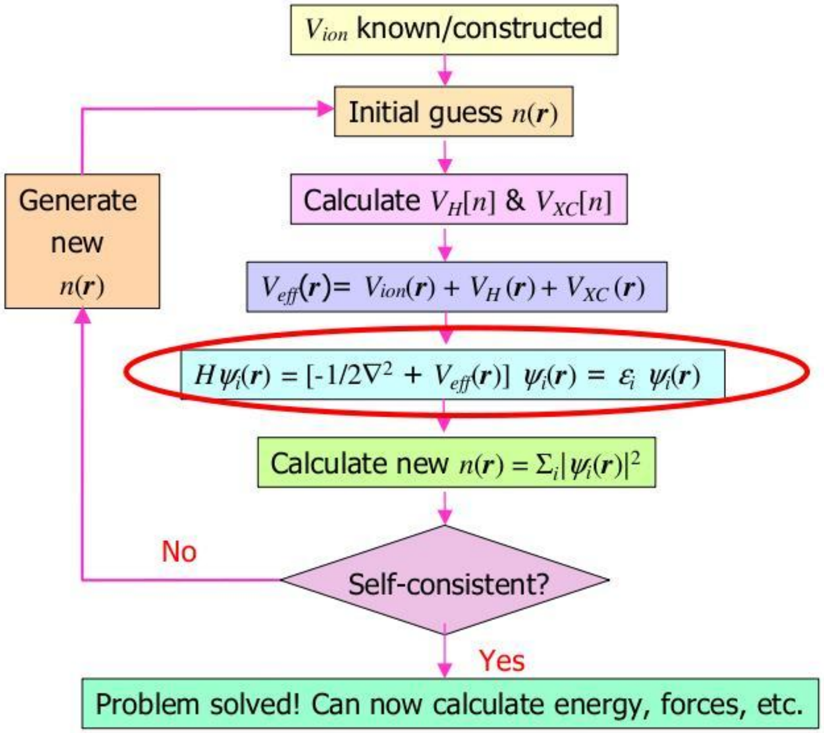
7. Plane-Wave Pseudopotential method, summary

- Supercell geometry: lattice vectors + atoms in the unit cell
- Plane-wave basis set, determined by the lattice and by a single parameter (*cutoff*)
- Atom-based pseudopotentials representing the electron-nuclei potential (V_{nuc} in the figure)
- Charge density with valence electrons only


V_{nuc} known/constructed
Initial guess $n(\mathbf{r})$
Calculate $V_H[n]$ & $V_{XC}[n]$
$V_{eff}(\mathbf{r}) = V_{nuc}(\mathbf{r}) + V_H(\mathbf{r}) + V_{XC}(\mathbf{r})$
$H\psi_i(\mathbf{r}) = [-1/2\nabla^2 + V_{eff}(\mathbf{r})]\ \psi_i(\mathbf{r}) = \varepsilon_i\ \psi_i(\mathbf{r})$
Calculate new $n(\mathbf{r}) = \Sigma_i|\psi_i(\mathbf{r})|^2$
Self-consistent?
No
Generate new $n(\mathbf{r})$
Yes
Problem solved! Can now calculate energy, forces, etc.

PW-PP method, iterative diagonalization

The solution of $(H-\epsilon)\psi_i = 0$ for a large $N \times N$ matrix costs $T_{CPU} = \mathcal{O}(N^3)$. Too much for most applications: N, the number of PWs, can be very large for large supercells...
...but we actually need only the lowest occupied $M << N$ eigenvectors. Solution: **Iterative diagonalization**. Based on iterative refinement of a trial solution. Refinement is stopped when the reached accuracy is deemed sufficient. Typical algorithm: *Block Davidson*.


V_{ion} known/constructed
Initial guess $n(\mathbf{r})$
Calculate $V_H[n]$ & $V_{XC}[n]$
$V_{eff}(\mathbf{r}) = V_{ion}(\mathbf{r}) + V_H(\mathbf{r}) + V_{XC}(\mathbf{r})$
$H\psi_i(\mathbf{r}) = [-1/2\nabla^2 + V_{eff}(\mathbf{r})]\ \psi_i(\mathbf{r}) = \varepsilon_i\ \psi_i(\mathbf{r})$
Calculate new $n(\mathbf{r}) = \Sigma_i|\psi_i(\mathbf{r})|^2$
Self-consistent?
No
Generate new $n(\mathbf{r})$
Yes
Problem solved! Can now calculate energy, forces, etc.

Iterative diagonalization is very convenient in conjunction with SCF iteration:

- high accuracy not needed in the first iterations
- starting trial wavefunctions available from previous iteration
- needed approximate inverse matrix easily calculated (H is diagonally dominated)

Very Technical Digression: Fast Fourier Transforms and Iterative Diagonalization

Let us consider first the simple case of a periodical function $f(x)$, with period L. Its Fourier components:

$$\widetilde{f}(q) = \frac{1}{L}\int f(x)e^{-iqx}dx$$

are nonzero over an *infinite* set of *discrete* values of q:

$$q_n = n\frac{2\pi}{L}, \qquad -\infty < n < \infty$$

The Fourier components decay to 0 for large q. The inverse Fourier transform has the form

$$f(x) = \sum_n \widetilde{f}(q_n)e^{iq_nx} = \sum_n \widetilde{f}_n e^{in(2\pi x/L)}$$

Our functions are however defined over a discrete but *finite* grid of q. How are they represented in x space?

Discrete Fourier Transform

We assume *both* x and q grids to be discrete, finite, and periodically repeated, and we write, for N large enough to accommodate all q components:

$$\begin{aligned} f(x) &\rightarrow f_m = f(x_m), \qquad x_m = m\frac{L}{N}, \qquad m = 0, .., N-1 \\ \widetilde{f}(q) &\rightarrow \widetilde{f}_n = \widetilde{f}(q_n), \qquad q_n = n\frac{2\pi}{L}, \qquad n = 0, .., N-1 \end{aligned}$$

(q components of negative value refold into those at the end of the box). The *Discrete Fourier Transform* can be written as

$$\begin{aligned} f_m &= \sum_{n=0}^{N-1} \widetilde{f}_n e^{i(2\pi nm/N)} \qquad (x-\text{space}) \\ \widetilde{f}_n &= \frac{1}{N}\sum_{m=0}^{N-1} f_m e^{-i(2\pi nm/N)} \qquad (q-\text{space}) \end{aligned}$$

Discrete Fourier Transform in 3D

Generalization of the Discrete Fourier Transform to 3 dimensions:

$$\mathbf{G} = n'_1\mathbf{G}_1 + n'_2\mathbf{G}_2 + n'_3\mathbf{G}_3$$

with $n_1 = 0, .., N_1-1$, $n_2 = 0, .., N_2-1$, $n_3 = 0, .., N_3-1$, and n'_1, n'_2, n'_3 are n_1, n_2, n_3 refolded so that they are centered around the origin (remember: the $\mathbf{G}-$space grid is also periodic!). $\mathbf{G}_1, \mathbf{G}_2, \mathbf{G}_3$ are the primitive translations of the unit cell of the reciprocal lattice.

$$\mathbf{r} = m_1\frac{\mathbf{R}_1}{N_1} + m_2\frac{\mathbf{R}_2}{N_2} + m_3\frac{\mathbf{R}_3}{N_3}$$

with $m_1 = 0, .., N_1 - 1$, $m_2 = 0, .., N_2 - 1$, $m_3 = 0, .., N_3 - 1$. $\mathbf{R}_1, \mathbf{R}_2, \mathbf{R}_3$ are the primitive translations of the unit cell. This grid spans the unit cell. N_1, N_2, N_3 define the *FFT grid*.

Discrete Fourier Transform in 3D (2)

Original Fourier transform:

$$f(\mathbf{r}) = \sum_{\mathbf{G}} \widetilde{f}(\mathbf{G}) e^{i\mathbf{G}\cdot\mathbf{r}} \rightarrow f(m_1, m_2, m_3)$$

$$\widetilde{f}(\mathbf{G}) = \frac{1}{\Omega} \int f(\mathbf{r}) e^{-i\mathbf{G}\cdot\mathbf{r}} d\mathbf{r} \rightarrow \widetilde{f}(n_1, n_2, n_3)$$

Discretized Fourier Transform:

$$f(m_1, m_2, m_3) = \sum_{n_1,n_2,n_3} \widetilde{f}(n_1, n_2, n_3) e^{i(2\pi n_1 m_1/N_1)} e^{i(2\pi n_2 m_2/N_2)} e^{i(2\pi n_3 m_3/N_3)}$$

$$\widetilde{f}(n_1, n_2, n_3) = \frac{1}{N} \sum_{m_1,m_2,m_3} \widetilde{f}(m_1, m_2, m_3) e^{-i(2\pi n_1 m_1/N_1)} e^{-i(2\pi n_2 m_2/N_2)} e^{-i(2\pi n_3 m_3/N_3)}$$

where $N = N_1 N_2 N_3$. Remember that $\mathbf{G}_i \cdot \mathbf{R}_j = 2\pi\delta_{ij}$.

PW-PP calculations and Discrete Fourier Transform

$$\psi_i(\mathbf{r}) = \frac{1}{V}\sum_{\mathbf{G}} c_{\mathbf{k}+\mathbf{G}} e^{i(\mathbf{k}+\mathbf{G})\cdot\mathbf{r}}, \qquad \frac{\hbar^2}{2m}|\mathbf{k}+\mathbf{G}|^2 \le E_{cut}$$

Which grid in **G**-space? Need to calculate the charge density. In principle (but not in practice):

$$n(\mathbf{G}') = \sum_{\mathbf{G}}\sum_{i,\mathbf{k}} f_{i,\mathbf{k}} c^*_{i,\mathbf{k}+\mathbf{G}} c_{i,\mathbf{k}+\mathbf{G}+\mathbf{G}'}$$

Fourier components $\mathbf{G}'$ with max$(|\mathbf{G}'|) = 2$max$(|\mathbf{G}|)$ appear. Or we need the product of the potential time a wavefunction:

$$(V\psi)(\mathbf{G}) = \sum_{\mathbf{G}'} V(\mathbf{G}-\mathbf{G}') c_{i,\mathbf{k}+\mathbf{G}'}$$

Again, max$(|\mathbf{G}-\mathbf{G}'|) = 2max(|\mathbf{G}|)$. We need a kinetic energy cutoff for the Fourier components of the charge and potentials that is four time larger as the cutoff for the PW basis set:

$$\frac{\hbar^2}{2m}|\mathbf{G}|^2 \le 4E_{cut}$$

In practice such condition may occasionally be relaxed. **Important:** for *ultrasoft* pseudopotentials, a different (larger) cutoff for augmentation charges may be needed!

The Fourier Transform grid is thus

$$\mathbf{G} = n_1'\mathbf{G}_1 + n_2'\mathbf{G}_2 + n_3'\mathbf{G}_3$$

with $n_1 = 0, .., N_1 - 1$, $n_2 = 0, .., N_2 - 1$, $n_3 = 0, .., N_3 - 1$. This grid must be big enough to include all $\mathbf{G}-$vectors up to a cutoff

$$\frac{\hbar^2}{2m}|\mathbf{G}|^2 \leq 4E_{cut}$$

and NOT up to the cutoff of the PW basis set! In general, the grid will also contain "useless" Fourier components (beyond the above-mentioned cutoff, so that $n(\mathbf{G}) = 0, V(\mathbf{G}) = 0$ etc.)

Fast Fourier Transform

Computational cost of a conventional Fourier Transform of order n: $T_{CPU} = \mathcal{O}(n^2)$.

Computational cost of a Fast Fourier Transform of order n: $T_{CPU} = \mathcal{O}(n \log n)$.

Difference: enormous in practical applications.

Advantages of the use of FFT in PW-PP calculations: enormous, especially in conjunction with iterative techniques and of the "dual-space" technique

Iterative matrix diagonalization

Basic ingredients: evaluation of products $\phi_i = (H - \epsilon_i)\psi_i$ on trial wavefunctions ψ_i. Same ingredient used in direct minimization, Car-Parrinello, etc.

If the product is calculated as a matrix-vector product: $T_{CPU} = \mathcal{O}(N^2)$ for a single product, $T_{CPU} = \mathcal{O}(MN^2)$ in total. Still much better than conventional diagonalization. But:

- if N becomes large, storing H becomes unpractical, if not impossible;
- much CPU could be spared if an economical way of calculating $H\psi$ was available

Solution: treat H as an operator, taking advantage of FFTs. There is no longer any need to store H as a matrix.

Dual space technique

$$H\psi \equiv (T + \hat{V}_{NL} + V_{loc} + V_H + V_{xc})\psi$$

$(T\psi)$: easy in **G**-space, $T_{CPU} = \mathcal{O}(N)$

$(V_{loc} + V_H + V_{xc})\psi$: easy in **r**-space, $T_{CPU} = \mathcal{O}(N)$

$(\hat{V}_{NL}\psi)$: easy in **G**-space (also in **r**-space) if $\hat{V}$ is written in separable form
$T_{CPU} = \mathcal{O}(mN)$, $m =$number of projectors

FFT is used to jump from real to reciprocal space. Operations are performed where it is easier.

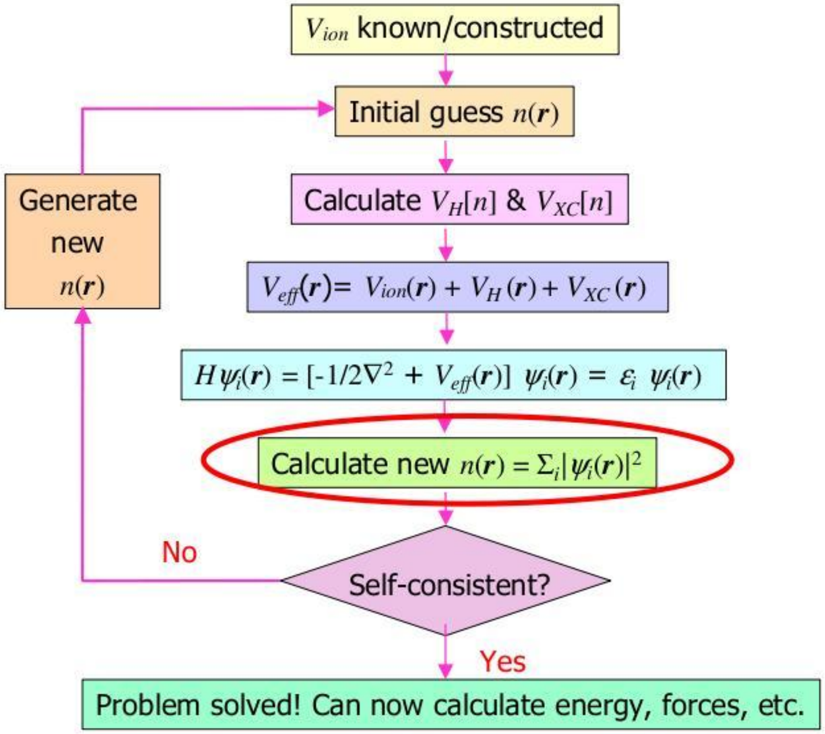
The same technique is used to calculate the charge density from Kohn-Sham orbitals, the exchange-correlation GGA potential from the charge density, etc.: in all cases, we move to the more convenient space to perform the required operations.

(End of the Very Technical Digression)

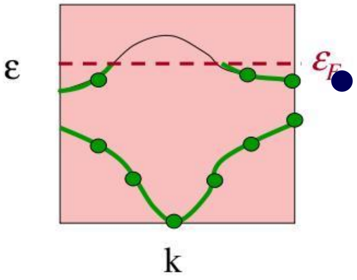
Brillouin Zone sampling

- The calculation of the charge density (and of many other quantities) requires sums over all **k**-points in the Brillouin Zone (BZ); in practice, some form of *BZ sampling* is needed. Convergence wrt **k**-point sampling *must be tested*!

- For insulators in large supercells, amorphous systems, liquids, molecules, sampling with Γ $(\mathbf{k} = 0)$ only is fine. For other insulators, a small number of **k**-points is usually sufficient.



V_{ion} known/constructed
Initial guess $n(\mathbf{r})$
Calculate $V_H[n]$ & $V_{XC}[n]$
$V_{eff}(\mathbf{r}) = V_{ion}(\mathbf{r}) + V_H(\mathbf{r}) + V_{XC}(\mathbf{r})$
$H\psi_i(\mathbf{r}) = [-1/2\nabla^2 + V_{eff}(\mathbf{r})]\ \psi_i(\mathbf{r}) = \varepsilon_i\ \psi_i(\mathbf{r})$
Calculate new $n(\mathbf{r}) = \Sigma_i|\psi_i(\mathbf{r})|^2$
Self-consistent?
No
Yes
Generate new $n(\mathbf{r})$
Problem solved! Can now calculate energy, forces, etc.



- For metals, a very fine sampling of the *Fermi surface*, together with some *broadening*, or *smearing*, technique, is needed. One could in principle use Fermi-Dirac occupations at finite T, but this would require either very high T or too fine sampling

Metals: broadening technique

The practical way to deal with metals uses a broadening σ in the following way:

$$\sum_i f_i \epsilon_i \to \int_{-\infty}^{\epsilon_F} \delta(\frac{\epsilon - \epsilon_i}{\sigma}) \epsilon d\epsilon = \sum_i \theta_i \epsilon_i + \sum_i \delta_i,$$

where $\delta(x)$ is a gaussian or similar function centered around $x = 0$,

$$\theta_i = \int_{-\infty}^{\epsilon_F} \delta(\frac{\epsilon - \epsilon_i}{\sigma}) d\epsilon, \quad \delta_i = \int_{-\infty}^{\epsilon_F} \delta(\frac{\epsilon - \epsilon_i}{\sigma})(\epsilon - \epsilon_i) d\epsilon = \sigma^2 \int_{-\infty}^{(\epsilon_F - \epsilon_i)/\sigma} x\delta(x) dx,$$

and ϵ_F is determined by the condition $\sum_i f_i =$ number of electrons. It is equivalent to introduce a fictitious "temperature" σ / k_B and the corresponding "free energy".

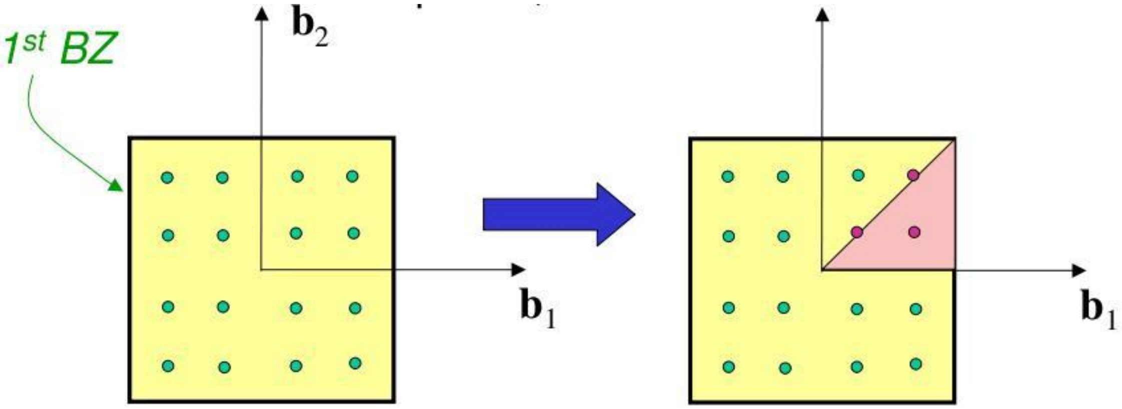
Specially tailored broadening functions (Marzari-Vanderbilt, Methfessel-Paxton), ensure fast convergence. This must be tested by performing several runs at different σ for increasingly dense k-point grids, until a suitable k-point grid and σ are found yielding satisfactorily converged results.

Grids of k-points for Brillouin Zone sampling

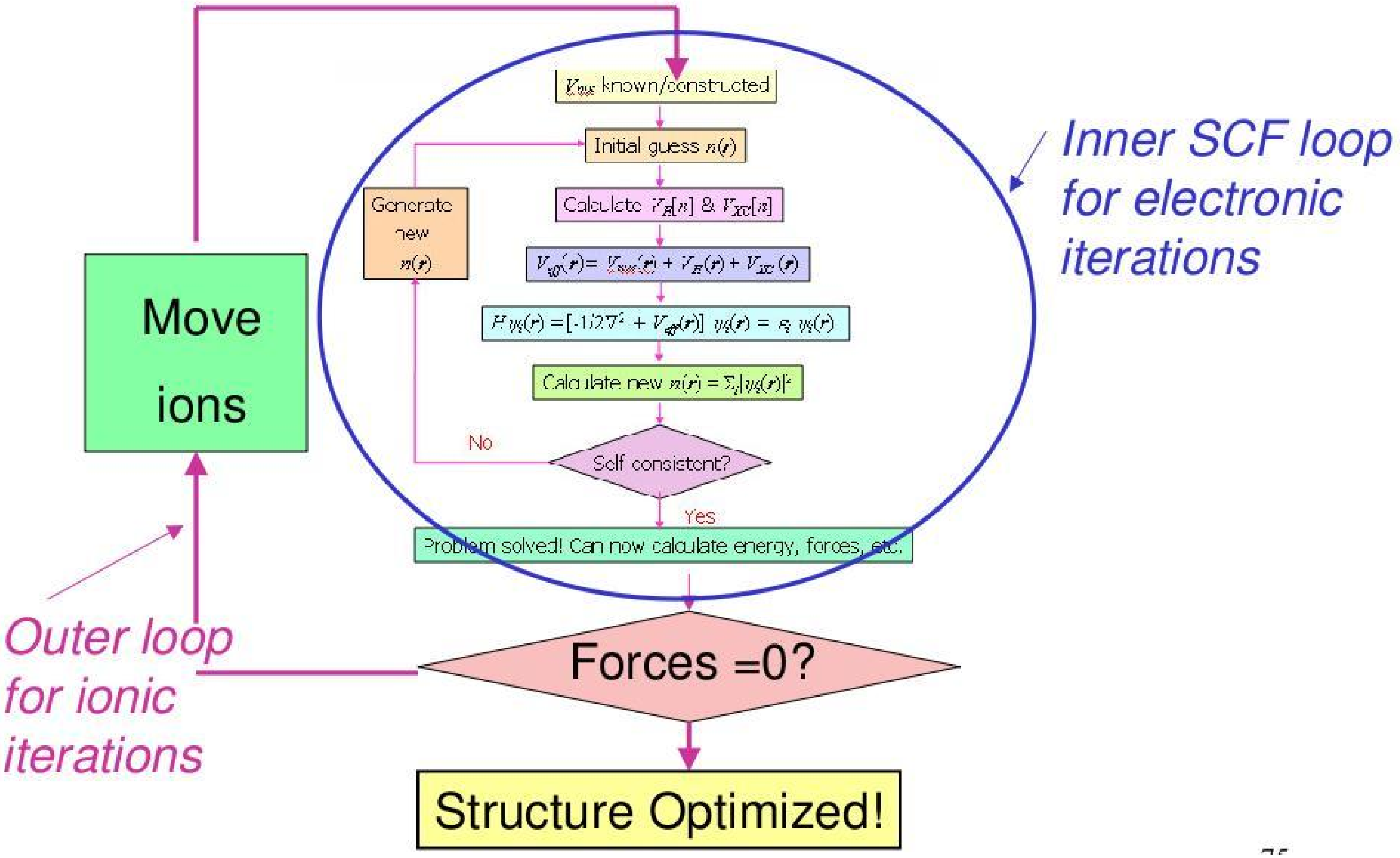
Typical ways of sampling the Brillouin Zone (BZ):

- **Special Points** (e.g. Baldereschi, Chadi and Cohen)
  Points designed to give quick convergence for particular crystal structures.
- **Uniform Grids** (e.g. Monkhorst-Pack)
  Equally spaced in reciprocal space. May be centred on origin ("non-shifted") or not ("shifted").

In presence of *symmetry*, only **k**-points in the *Irreducible* BZ, or IBZ, need to be computed: the charge density is reconstructed using symmetry. Appropriate *weights* for **k**-points must be specified (or can be calculated).



Structural optimization



V_{ext} known/constructed

Initial guess $n(\mathbf{r})$

Calculate $V_H[n]$ & $V_{XC}[n]$

$V_{eff}(\mathbf{r}) = V_{ext}(\mathbf{r}) + V_H(\mathbf{r}) + V_{XC}(\mathbf{r})$

$H\psi_i(\mathbf{r}) = [-1/2\nabla^2 + V_{eff}(\mathbf{r})]\ \psi_i(\mathbf{r}) = \varepsilon_i\ \psi_i(\mathbf{r})$

Calculate new $n(\mathbf{r}) = \Sigma_i|\psi_i(\mathbf{r})|^2$

Self consistent?

No

Generate new $n(\mathbf{r})$

Yes

Problem solved! Can now calculate energy, forces, etc.

Forces =0?

Structure Optimized!

Move ions

Inner SCF loop for electronic iterations

Outer loop for ionic iterations

Beware: in a periodic system, one must distinguish between

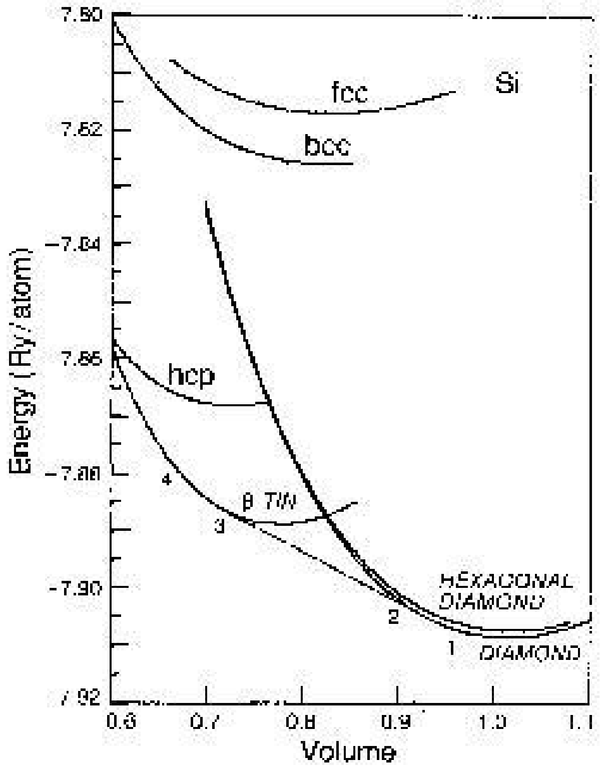
- atomic displacements *inside* the unit cell, determined by the *forces*
- elastic displacements changing the *shape* of the unit cell, determined by the *stresses*

Simple case: finding the equilibrium volume

For simple crystals, the global ground state can be found by calculating for a few values of the lattice parameter the $E(V)$ curve, fitting it to a phenomenological *equation of state* (EOS) like Murnaghan's:

$$P(V) = \frac{B}{B'}\left[\left(\frac{V_0}{V}\right)^{B'} - 1\right]$$

(the $E(V)$ formula can be obtained by observing that the pressure $P = -\partial E/\partial V$). Equilibrium volume V_0, bulk modulus B and its pressure derivative B' are the fit parameters.



In the picture from a famous Yin-Cohen 1980 paper, the energies of different candidate structures for Si can be compared, phase transitions under pressure found.

Note that in simple structures *the force on atoms can be zero* by symmetry even if the the system is *not at equilibrium*!

General case: variable-cell structural optimization

In general, the equilibrium in a crystal (or supercell) is reached when all forces on atoms in the unit cell/supercell are zero, *and* all **stresses** are zero.

If a *strain* $\epsilon_{\alpha\beta}$ is applied to all coordinates:

$$r_\alpha \longrightarrow (1 + \sum_\gamma \epsilon_{\alpha\gamma} r_\gamma)$$

the stress $\sigma_{\alpha\beta}$ is the derivative of the energy wrt the strain:

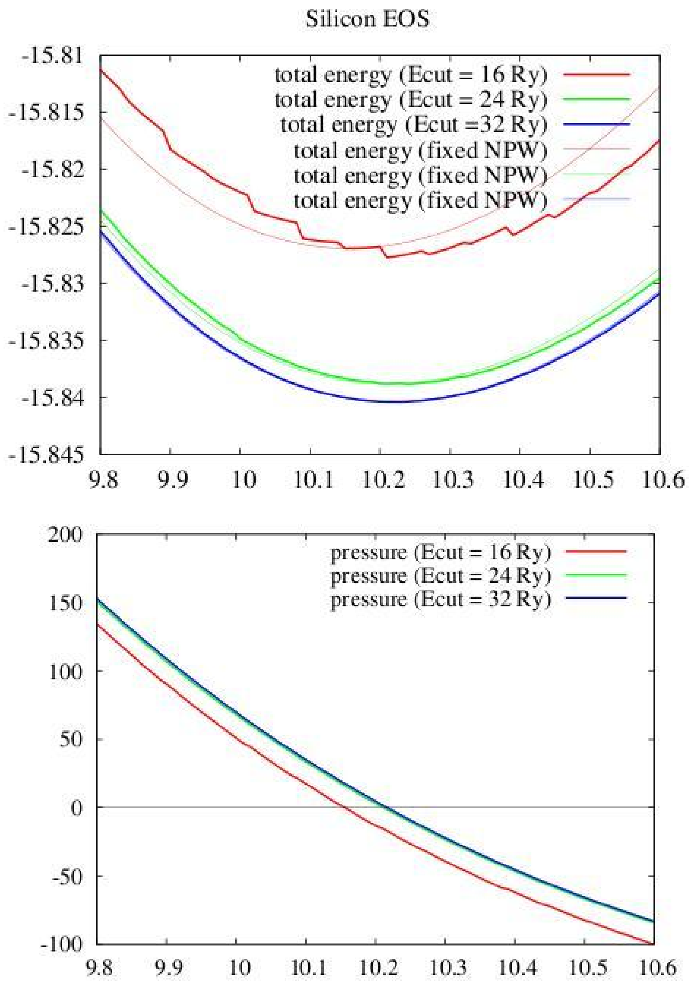
$$\sigma_{\alpha\beta} = -\frac{1}{\Omega}\frac{\partial E}{\partial \epsilon_{\alpha\beta}}$$

and it can be calculated from ground-state Kohn-Sham orbitals (Nielsen and Martin, PRB 3780 and 3792, 1985). Note that pressure is related to the stress via $P = -\mathrm{Tr}\sigma/3$.

Unlike the case of forces, there is an incomplete-basis-set (Pulay) correction to stresses: the basis set *depends upon the strain* via the size of the unit cell.

Note on incompleteness of the PW basis set

Practical calculations are invariable performed with a cutoff ”as low as possible”, sometimes quite far from convergence. The consequence are especially visible when comparing the $E(V)$ curve at fixed cutoff and the same at fixed number of plane waves: the curve at low cutoff is ”rigged”, the pressures calculated from the stress and from the equation of state do not match.
Note that the strategy ”fixed cutoff + fit to an EOS” converges much better than the ”fixed number of PWs” strategy, or equivalently to locating the zero of the calculated pressure.



The incompleteness is not as serious as it may look: energy *differences* between different structures, structural parameters such as lattice parameters and bond lengths, converge much quicker than absolute energies.

Quasi-Newton algorithms for structural optimization

The BFGS (Broyden-Fletcher-Goldfarb-Shanno) algorithm is the workhorse for structural minimization, either at fixed cell or with variable cell.

Close to an equilibrium point $\vec{X}^{(eq)}$, for which $\nabla E(\vec{X}^{(eq)}) = 0$ holds, a quadratic form is assumed for the function $E(\vec{X})$ (H is the Hessian matrix):

$$E(\vec{X}) \simeq E(\vec{X}^{(eq)}) + \frac{1}{2}(\vec{X} - \vec{X}^{(eq)})^T H(\vec{X} - \vec{X}^{(eq)})$$

Given two points $\vec{X}_1$ and $\vec{X}_0$ and corresponding gradients $\vec{g} = \nabla E(\vec{X})$, this means $\vec{g}_1 - \vec{g}_0 = H(\vec{X}_1 - \vec{X}_0)$, that is, $\vec{g}_1 = 0$ if $\vec{X}_1 = \vec{X}_0 - H^{-1}\vec{g}_0$ (*Newton-Raphson step*).

Practical algorithm: a sequence of calculations at positions $\vec{X}_i$

$$\vec{X}_{i+1} = \vec{X}_i + T_k^L \frac{\vec{s}_k^{NR}}{|s_k^{NR}|}, \qquad \vec{s}_k^{NR} = -H_k^{-1}\vec{g}_k$$

where T_k^L is called "trust radius".

The inverse Hessian matrix is updated at each step using the BFGS scheme:

$$H_{k+1}^{-1} = H_k^{-1} + \left(1 + \frac{\gamma_k^T H_k^{-1} \gamma_k}{s_k^T \gamma_k}\right) \frac{s_k s_k^T}{s_k^T \gamma_k} - \left(\frac{s_k \gamma_k^T H_k^{-1} + H_k^{-1} \gamma_k s_k^T}{s_k^T \gamma_k}\right)$$

$$\gamma_k = g_{k+1} - g_k$$

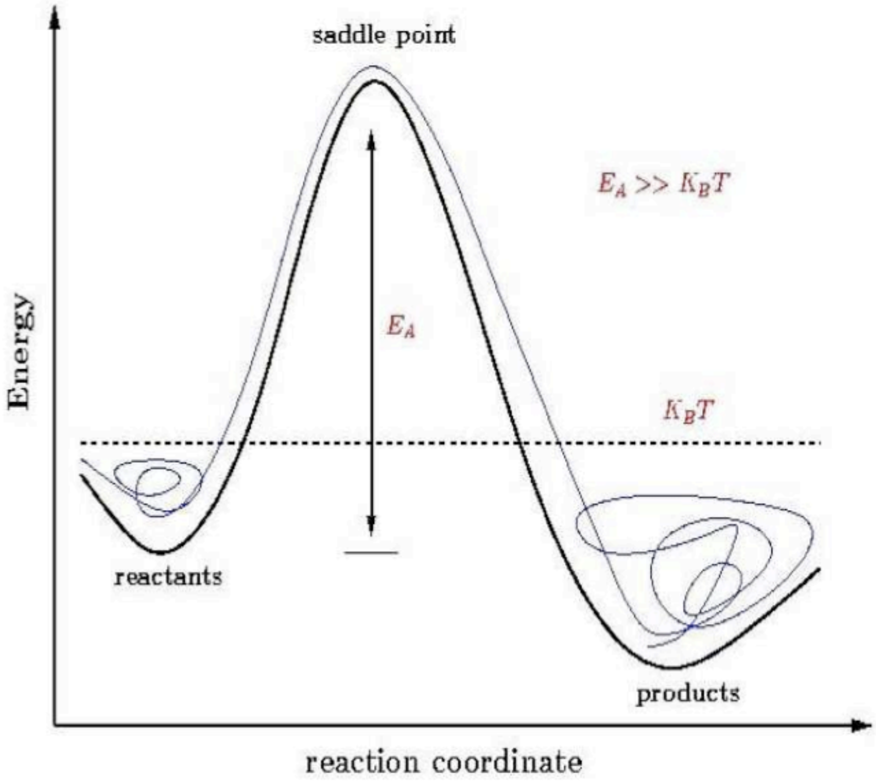
- At fixed cell:
  - $\vec{X} = (\vec{d}_1, \ldots, \vec{d}_N)$, atomic positions
  - $\vec{g} = -(\vec{f}_1, \ldots, \vec{f}_N)$, Hellmann-Feynman forces on atoms
- With variable cell:
  - $\vec{X} = (\vec{d}_1, \ldots, \vec{d}_N, \epsilon_{\alpha\beta})$, atomic positions and cell strains
  - $\vec{g} = -(\vec{f}_1, \ldots, \vec{f}_N, \sigma_{\alpha\beta})$, Hellmann-Feynman forces on atoms and stresses

Note that

- in variable-cell optimization, the PW basis set is kept fixed during optimization
- Structural optimization *does not break symmetry*, at least in principle, and
- it may only find *local minima* (it cannot overcome potential barriers).

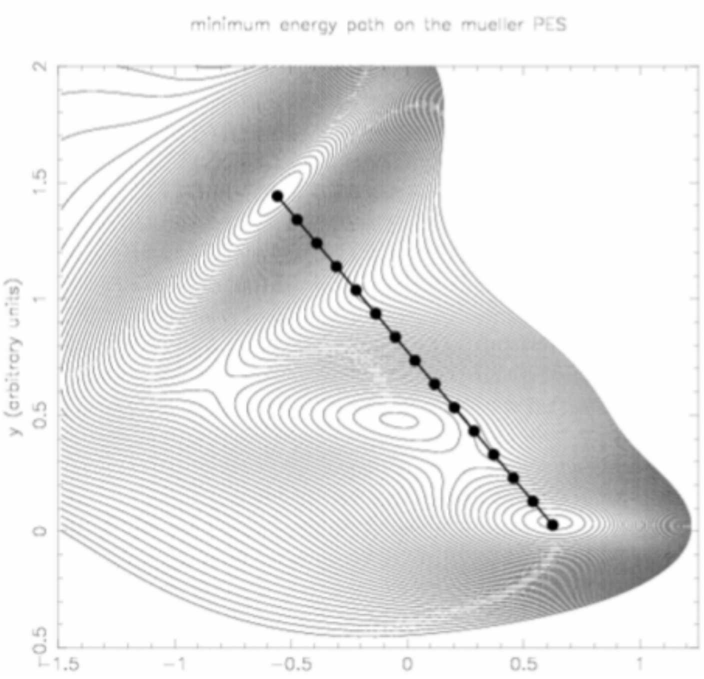
Reaction pathway calculation using NEB

The Nudged Elastic Band (NEB) method can be used to calculate reaction pathways, energy barriers, and via the harmonic Transition State Theory, transition rates. These quantites are in practice not accessible by Molecular Dynamics. The basic aspects of the theory are

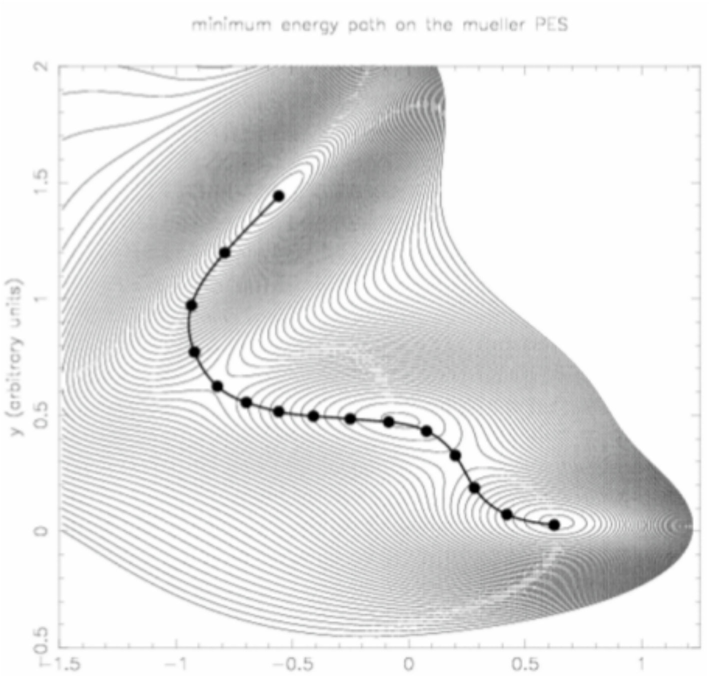


- The path is *discretized* into "images", i.e. points in the configuration space.
- Fictitious "springs" connecting the images keep them apart and prevent them from falling into the same local minima.
- The minimum-energy path is located by imposing that the component of the force orthogonal to the path vanishes.

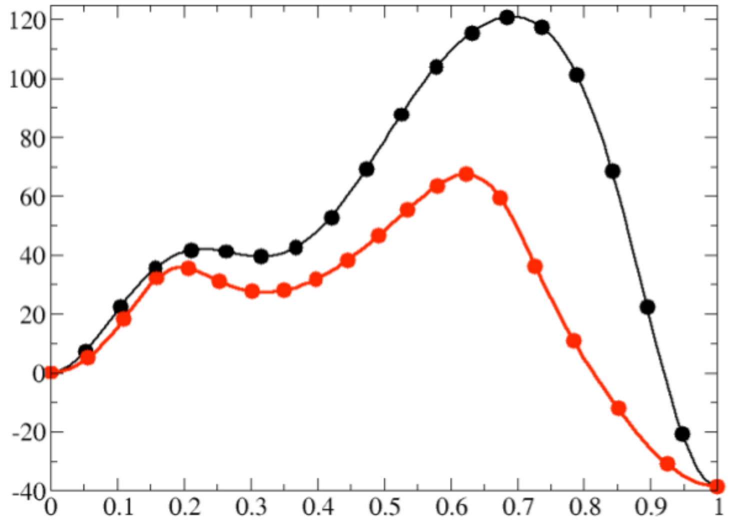
Reaction pathway calculation using NEB II



NEB



Above, a 2D example (the model Mueller potential energy surface). The starting path (black dots) is typically chosen by linear interpolation between initial and final positions. The final path (red dots) is found by an iterative procedure. Notice the two saddle points and a metastable state


120
100
80
60
40
20
0
-20
-40
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1