

# The ABINIT code



X. Gonze

Thanks to the > 50 ABINIT contributors, and especially to GM Rignanese for contributions to the slides

# ABINIT software project

Ideas (**1997**) :

- 1) Softwares for first-principles simulations are more and more sophisticated : one needs a worldwide collaboration, of specialized, complementary, groups
- 2) Linux software development : ‘free software’ model

Now (**2016**) :

- >1600 registered people on the forum
- >800 kLines of F90 + many python scripts (abipy)
- about 50 contributors to ABINITv7,
- last release v8.0.6 used in this school

# ABINIT milestones

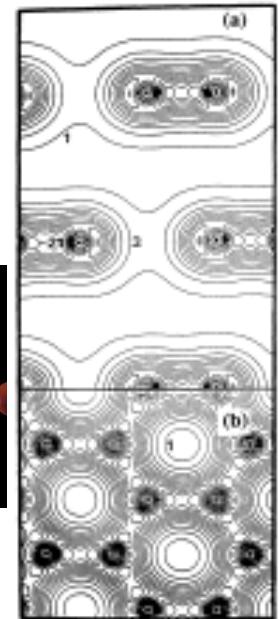
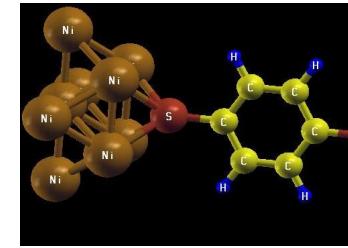
- Precursor : the Corning PW code (commercialized 1992-1995 by Biosym)
- 1997 : beginning of the ABINIT project
- Dec 2000 : release of ABINITv3 under the GNU General Public License (GPL)
- Nov 2002 – April 2015 : **7 international ABINIT developer workshops**  
(between 40 and 60 participants each)
- Jan 2010 : launch of the Forum



# Properties from DFT+MBPT+ ...

Computation of ...

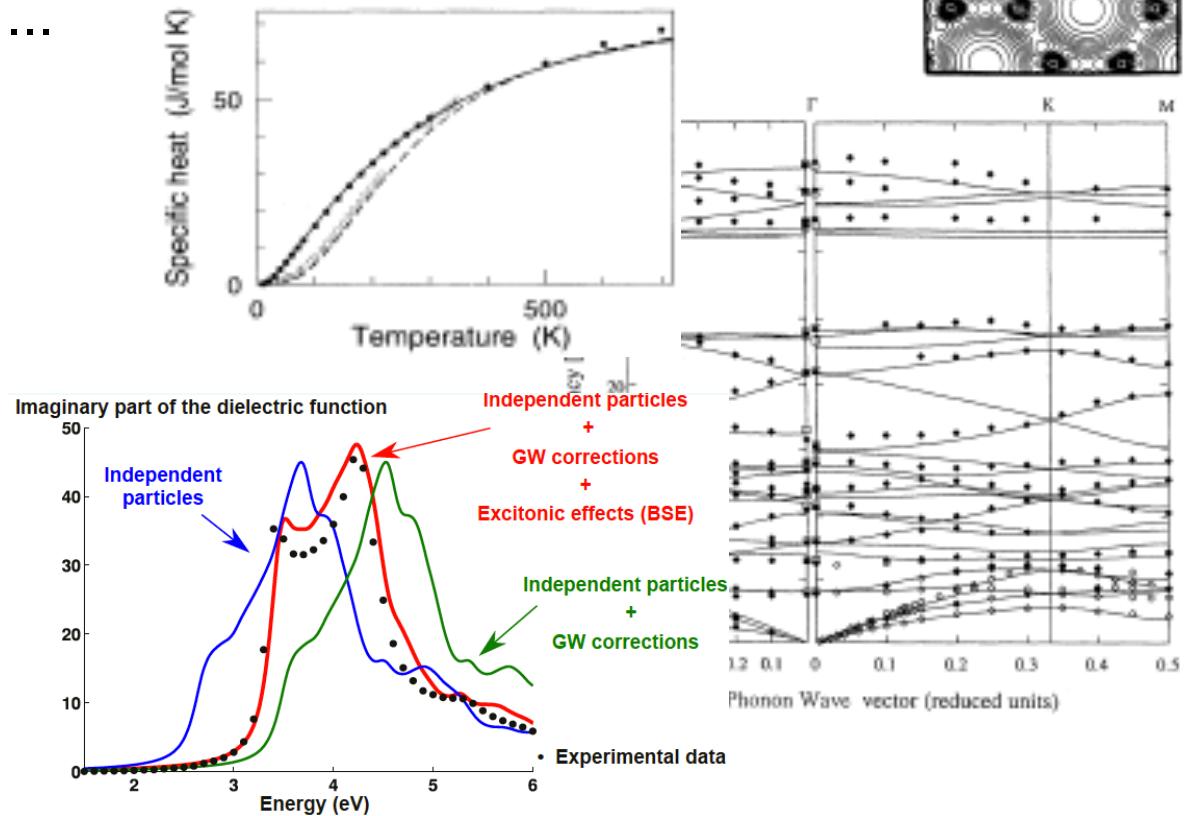
interatomic distances, angles, total energies  
electronic charge densities, electronic energies



A basis for the computation of ...

chemical reactions  
electronic transport  
vibrational properties  
thermal capacity  
dielectric behaviour  
optical response  
superconductivity  
surface properties  
spectroscopic responses

...



# Documentation

Web site <http://www.abinit.org>

- User's guide
- Installations notes
- List of input variables
- >30 tutorial lessons (each 1-2 hours)  
<http://www.abinit.org/documentation/helpfiles/for-v8.10/tutorial/welcome.html>
- + Forum Web site <http://forum.abinit.org>

## Videos

How to install on Linux ?

<http://www.youtube.com/watch?v=DppLQ-KQA68>

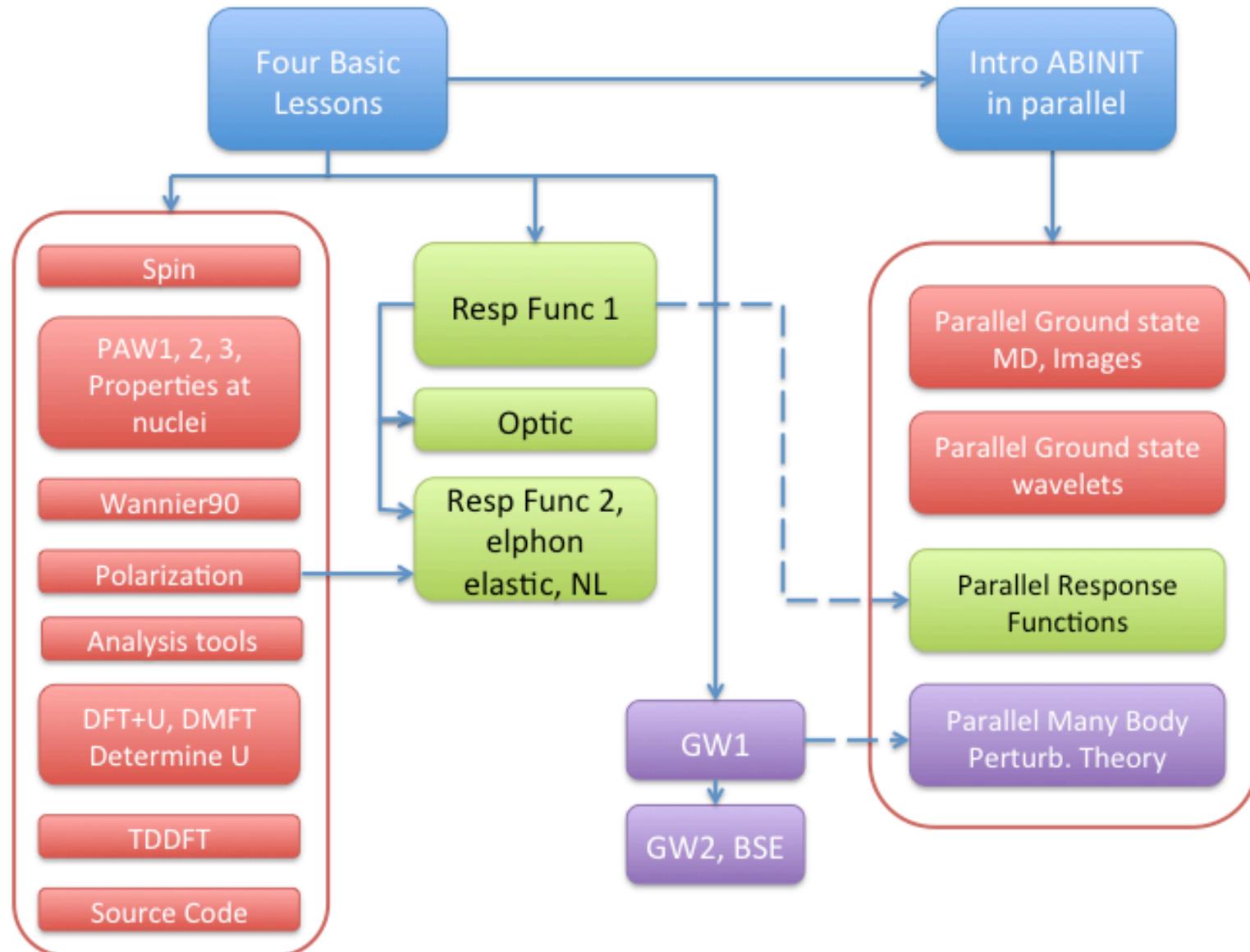
How to install on Windows 64bits ?

<http://www.youtube.com/watch?v=EfJcYi1MNBg>

First ABINIT tutorial ( $H_2$  molecule)

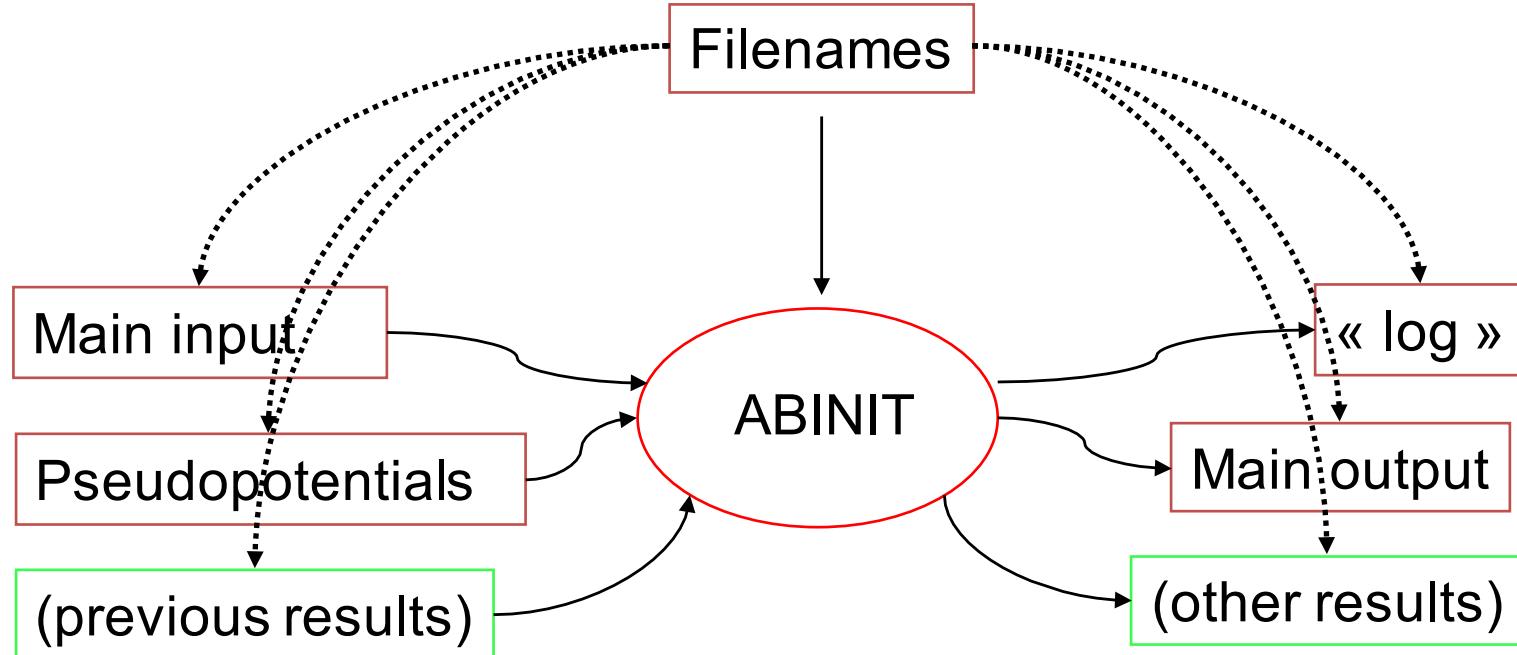
[http://www.youtube.com/watch?v=gcbfb\\_Mteo4](http://www.youtube.com/watch?v=gcbfb_Mteo4)

# ABINIT tutorial : layout + dependencies



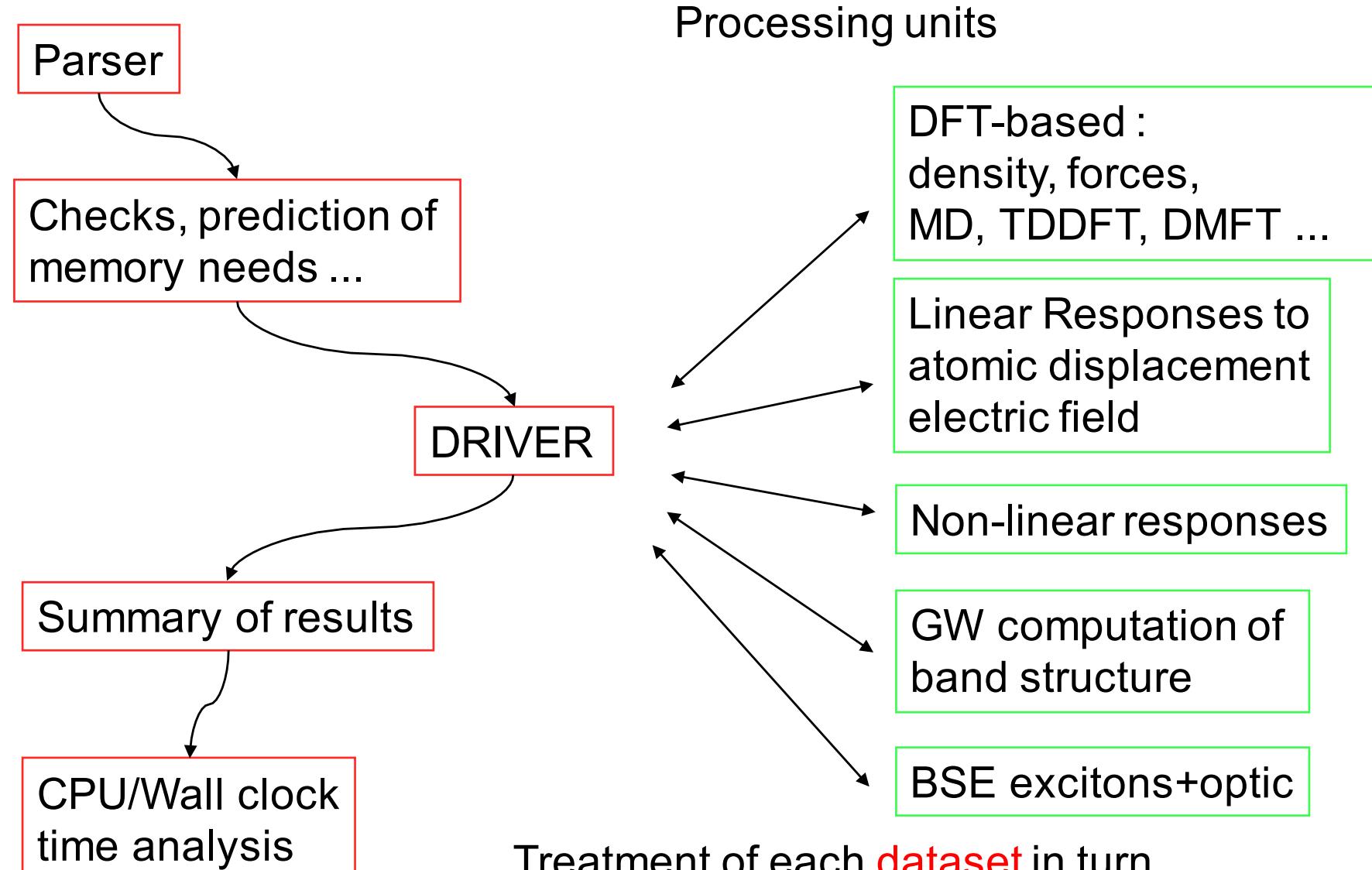
# Running ABINIT : basics

# External files in a ABINIT run



Results : density (\_DEN), potential (\_POT),  
wavefunctions (\_WFK), ...

# ABINIT : the pipeline and the driver



# Density Functional Theory calculations

In ABINIT ...

Representation of mathematical formalism  
with a **Plane Wave** basis set :

- wavefunctions
- density, potential

**Periodic boundary** conditions

=> wavefunctions characterized by a **wavevector** (k-vector)

**PseudoPotentials** (or Projector Augmented Waves – PAW)

**Iterative techniques** to solve the equations  
(Schrödinger equation ; DFT Self-consistency ; optimisation  
of atomic positions)

# A basic input file : dihydrogen (I)

```
# H2 molecule in a big box
```

```
# Definition of the unit cell
```

```
acell 10 10 10
```

# The keyword "acell" refers to the  
# lengths of the primitive vectors (default in Bohr)

```
# Definition of the atom types
```

```
ntypat 1
```

# There is only one type of atom

```
znucl 1
```

# The keyword "znucl" refers to the atomic number of the  
# possible type(s) of atom. The pseudopotential(s)  
# mentioned in the "filenames" file must correspond  
# to the type(s) of atom. Here, the only type is Hydrogen.

```
# Definition of the atoms
```

```
natom 2
```

# There are two atoms

```
typat 1 1
```

# They both are of type 1, that is, Hydrogen

```
xcart
```

# This keyword indicate that the location of the atoms  
# will follow, one triplet of number for each atom

```
-0.7 0.0 0.0
```

# Triplet giving the cartesian coordinates of atom 1, in Bohr

```
0.7 0.0 0.0
```

# Triplet giving the cartesian coordinates of atom 2, in Bohr

# A basic input file : dihydrogen (II)

```
# Definition of the planewave basis set
ecut 10.0          # Maximal plane-wave kinetic energy cut-off, in Hartree

# Definition of the k-point grid
kptopt 0           # Enter the k points manually
nkpt 1             # Only one k point is needed for isolated system,
                   # taken by default to be 0.0 0.0 0.0

#Definition of the SCF (self-consistent field) procedure
nstep 10           # Maximal number of SCF cycles
toldfe 1.0d-6       # Will stop when, twice in a row, the difference
                     # between two consecutive evaluations of total energy
                     # differ by less than toldfe (default in Hartree)
diemac 2.0          # Although this is not mandatory, it is worth to
                     # precondition the SCF cycle. The model dielectric
                     # function used as the standard preconditioner
                     # is described in the "dielng" input variable section.
                     # Here, we follow the prescriptions for molecules
                     # in a big box
```

# Specification of the atomic geometry

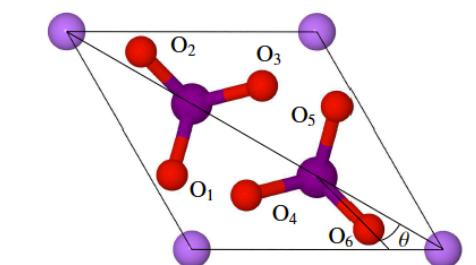
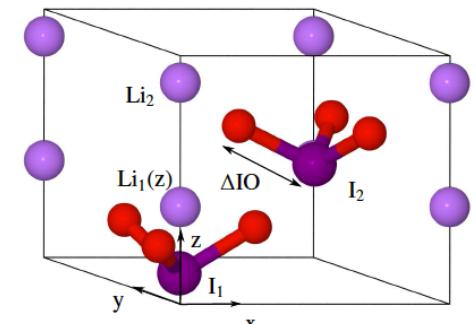
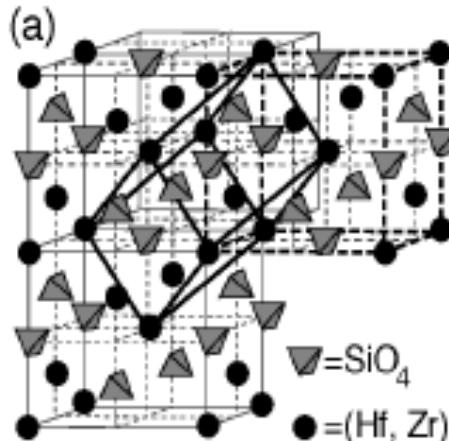
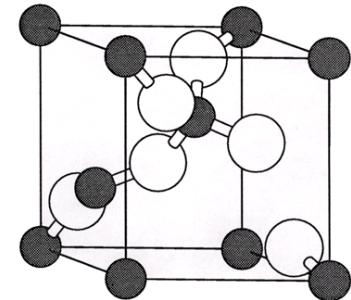
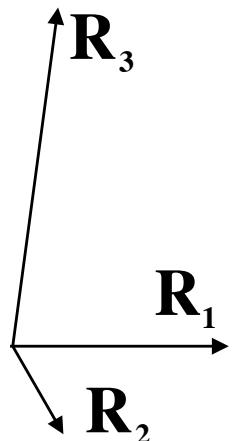
# Prerequisites of plane waves

Plane waves  $e^{i\mathbf{K}\mathbf{r}}$  : particularly simple and efficient (when used with pseudopotentials), but infinite spatial extent.

Cannot use a finite set of planewaves for finite systems !

Need periodic boundary conditions.

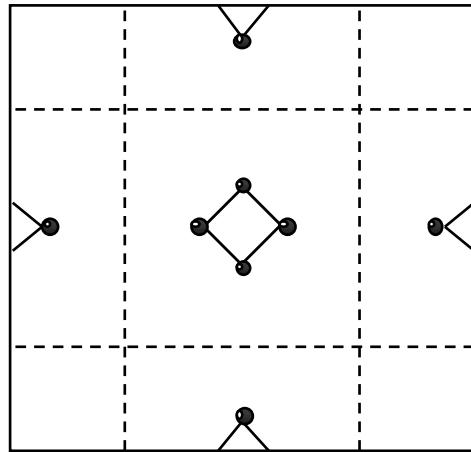
Primitive vectors  $\mathbf{R}_j$ , primitive cell volume  $\Omega_0$



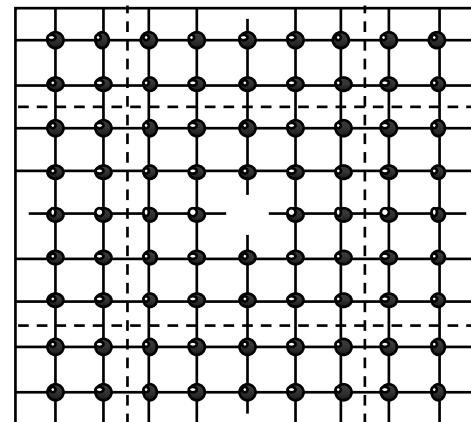
OK for crystalline solids

But : finite systems, surfaces, defects, polymers, nanosystems ... ?

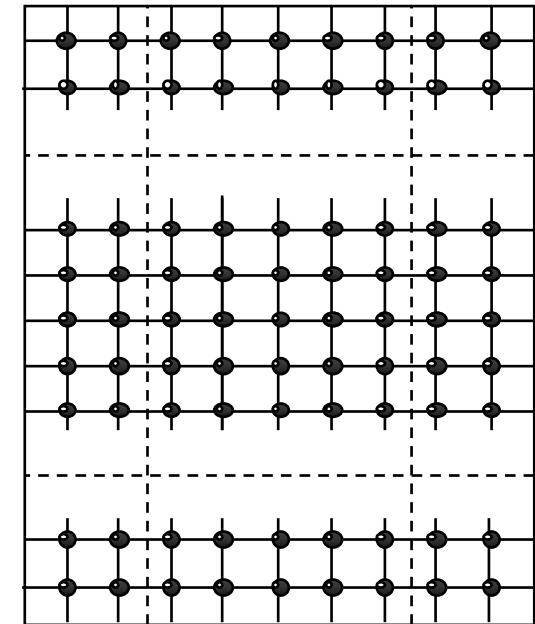
# Solution : the supercell technique



Molecule,  
cluster



Surface : treatment  
of a slab  
Interface

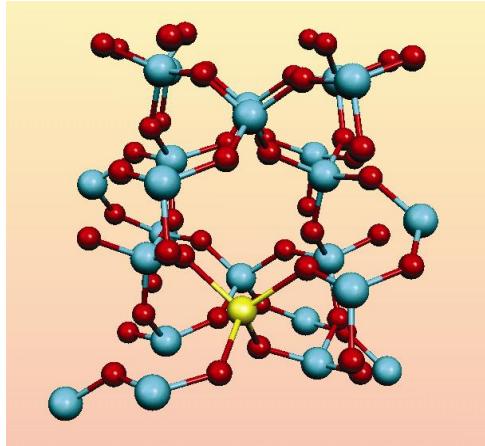


Point defect in a bulk solid

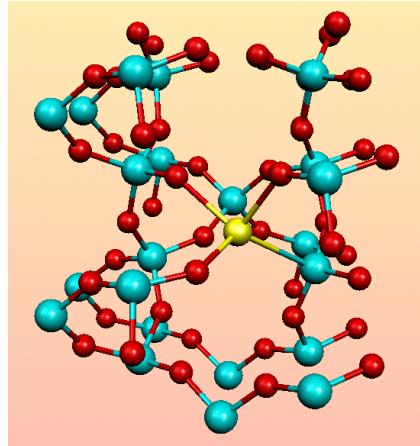
The supercell must be sufficiently big : convergence study

# Examples of defects SiO<sub>2</sub>-quartz : Pb

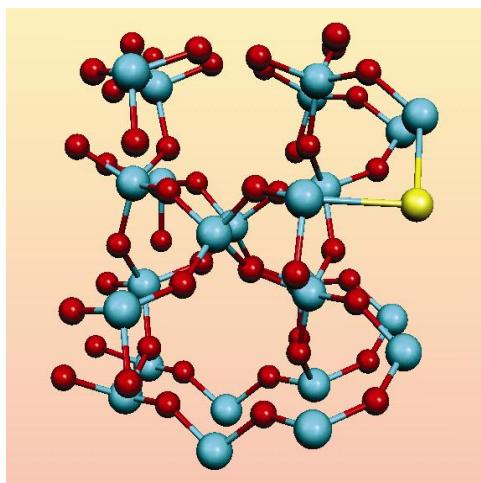
72-atom supercell of quartz



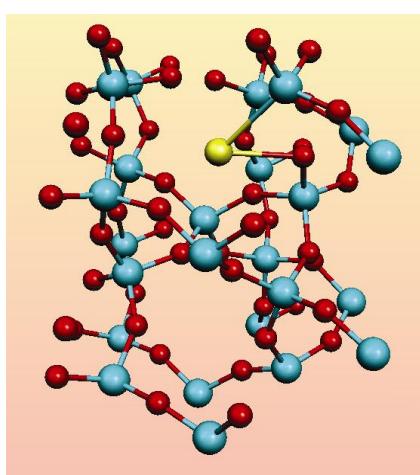
Pb<sup>Si</sup>



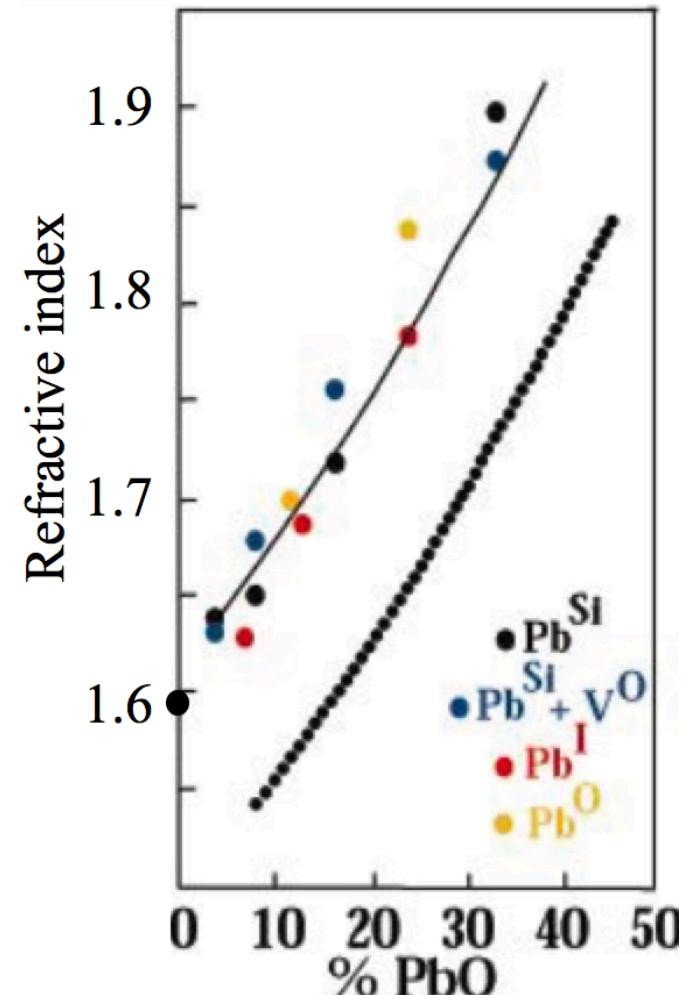
Pb<sup>Si</sup> + VO



Pb<sup>O</sup>



Pb<sup>I</sup>

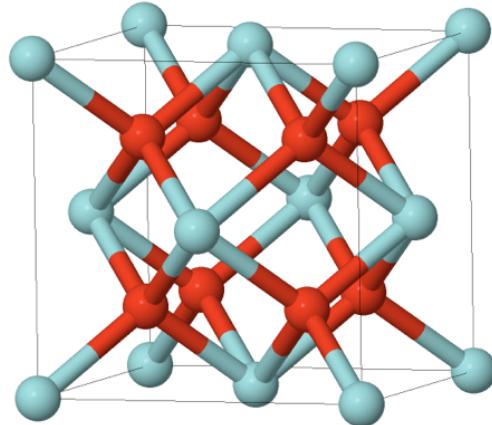


Comparison with amorphous SiO<sub>2</sub>

# Main input file : input variable flexibility

- cell primitive vectors → `rprim`
  - ... or angle (degrees) between primitive vectors → `angdeg`
- + scale cell vector lengths → `acell`
- + scale cartesian coordinates → `scalecart`
- number of atoms → `natom`
- reduced coordinates → `xred` (initial guess ... might be optimized)
  - ... or cartesian coordinates → `xcart` (in Bohr) / `xangst` (in Å)
- type of atoms → `typat`
- space group → `spgroup` + `natrd`
  - ... or number of symmetries → `nsym`
    - + symmetry operations → `symrel` + `tcons`

# Example : cubic zirconium dioxide



Fm-3m  
 $a=5.010\text{\AA}$   
 $b=5.010\text{\AA}$   
 $c=5.010\text{\AA}$   
 $\alpha=90.0^\circ$   
 $\beta=90.0^\circ$   
 $\gamma=90.0^\circ$

Bilbao Crystallographic Server → Assignment of Wyckoff Positions

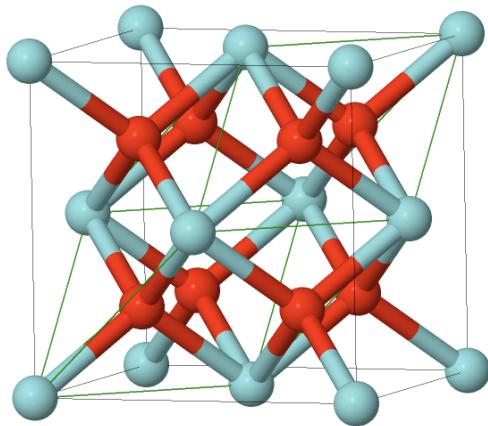
## Assignment of Wyckoff Positions

### Atoms Data:

AT.	WP	SS	Representative	Atomic orbit
Zr1	4a (0,0,0)	m-3m	(0.000000,0.000000,0.000000)	(0.000000,0.000000,0.000000) (0.000000,0.500000,0.500000) (0.500000,0.000000,0.500000) (0.500000,0.500000,0.000000)
O2	8c (1/4,1/4,1/4)	-43m	(0.250000,0.250000,0.250000)	(0.250000,0.250000,0.250000) (0.750000,0.750000,0.250000) (0.750000,0.250000,0.750000) (0.250000,0.750000,0.750000) (0.250000,0.250000,0.750000) (0.750000,0.750000,0.750000) (0.250000,0.750000,0.250000) (0.750000,0.250000,0.250000)

Face-centered cubic, with three atoms per primitive cell

# Example : cubic zirconium dioxide



natom 3

acell 3\*5.01 Angst

NOTE "\*" is a repeater

rprim 0.0 0.5 0.5

0.5 0.0 0.5

0.5 0.5 0.0

typat 1 2 2

xred 3\*0.0 3\*0.25 3\*0.75

=> symmetries are found automatically

OR

natom 3

acell 3\*5.01 Angst

rprim 0.0 0.5 0.5

0.5 0.0 0.5

0.5 0.5 0.0

typat 1 2 2

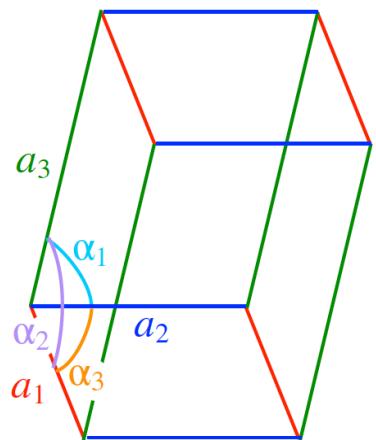
sppgroup 225 natrd 2

xred 3\*0.0 3\*0.25

=> the set of atoms is completed automatically

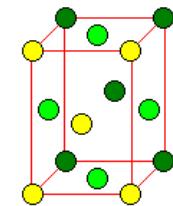
# Primitive vectors in ABINIT (rprimd)

- $R_i(j) \rightarrow r\text{primd}(j,i) = \text{scalecart}(j) \times r\text{prim}(j,i) \times a\text{cell}(i)$



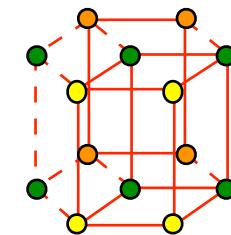
scalecart	9.5	9.8	10.0
rprim	0.0	0.5	0.5
	0.5	0.0	0.5
	0.5	0.5	0.0

face-centered  
orthorhombic



acell	9.5	9.5	10.0
rprim	0.8660254038E+00	0.5	0.0
	-0.8660254038E+00	0.5	0.0
	0.0	0.0	1.0

hexagonal



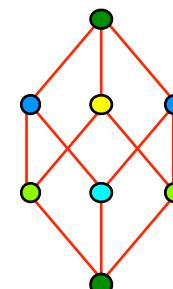
- $a_i \rightarrow a\text{cell}(i) / \alpha_i \rightarrow \text{angdeg}(i)$

acell	9.5	9.5	10.0
angdeg	120	90	90

hexagonal

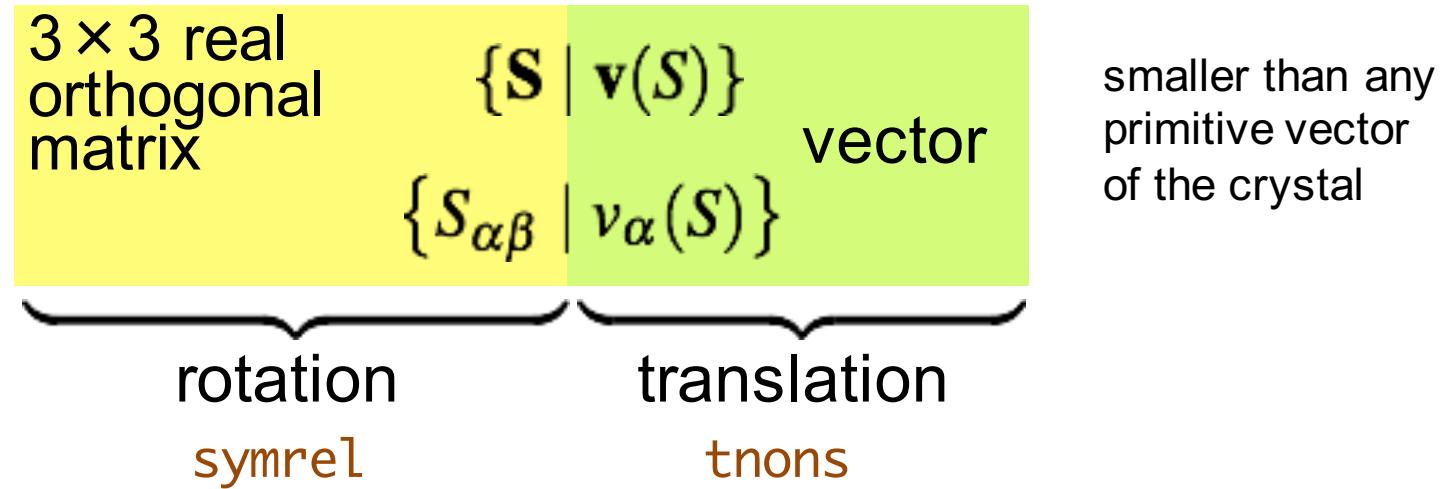
acell	9.0	9.0	9.0
angdeg	48	48	48

trigonal



# Symmetries in ABINIT

- Seitz notation for the symmetry operations of crystal :



- Applied to the equilibrium position vector of atom  $\kappa$  relative to the origin of the cell  $\tau_\kappa$ , this symmetry transforms it as:

$$\{\mathbf{S} | \mathbf{v}(S)\} \tau_\kappa = \mathbf{S} \tau_\kappa + \mathbf{v}(S) = \tau'_{\kappa'} + \mathbf{R}^a$$

$$\{S_{\alpha\beta} | v_\alpha(S)\} \tau_{\kappa\alpha} = S_{\alpha\beta} \tau_{\kappa\alpha} + v_\alpha(S) = \tau'_{\kappa'\alpha} + R_\alpha^a$$

where  $\mathbf{R}^a$  belongs to the real space lattice.

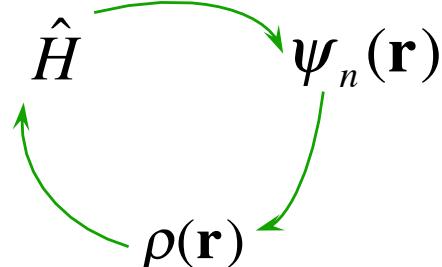
# The plane wave basis set

$$\psi_{\mathbf{k}}(\mathbf{r}) = \left(N\Omega_0\right)^{-1/2} \sum_{\mathbf{G}} u_{\mathbf{k}}(\mathbf{G}) e^{i(\mathbf{k}+\mathbf{G})\mathbf{r}}$$

# A reminder : basic equations in DFT

Solve self-consistently the Kohn-Sham equation

$$\left\{ \begin{array}{l} \hat{H} |\Psi_n\rangle = \varepsilon_n |\Psi_n\rangle \\ \hat{H} = \hat{T} + \hat{V} + \hat{V}_{Hxc}[\rho] \\ \rho(\vec{r}) = \sum_n^{occ} \Psi_n^*(\vec{r}) \Psi_n(\vec{r}) \end{array} \right.$$



$$\delta_{mn} = \langle \Psi_m | \Psi_n \rangle \text{ for } m, n \in \text{occupied set}$$

or minimize

$$E_{el}\{\Psi\} = \sum_n^{occ} \langle \Psi_n | \hat{T} + \hat{V} | \Psi_n \rangle + E_{Hxc}[\rho]$$

with

$$\hat{V}(\vec{r}) = \sum_{ak} -\frac{Z_k}{|\vec{r} - \vec{R}_k|}$$

# Periodic system : wavevectors

For a **periodic** Hamiltonian : wavefunctions characterized by a wavevector  $\mathbf{k}$  (crystal momentum) in Brillouin Zone

Bloch's theorem

$$\psi_{m,k}(\mathbf{r} + \mathbf{R}_j) = e^{i\mathbf{k}\cdot\mathbf{R}_j} \psi_{m,k}(\mathbf{r})$$

$$\psi_{m,k}(\mathbf{r}) = (N\Omega_0)^{-1/2} e^{i\mathbf{k}\cdot\mathbf{r}} u_{m,k}(\mathbf{r}) \quad u_{m,k}(\mathbf{r} + \mathbf{R}_j) = u_{m,k}(\mathbf{r})$$

Normalization ?

Born-von Karman supercell      supercell vectors  $\mathbf{N}_j \mathbf{R}_j$  with  $N=N_1 N_2 N_3$

$$\psi_{m,k}(\mathbf{r} + \mathbf{N}_j \mathbf{R}_j) = \psi_{m,k}(\mathbf{r})$$

# Planewave basis set

Reciprocal lattice : set of  $\mathbf{G}$  vectors such that  $e^{i\mathbf{GR}_j} = 1$   
 $e^{i\mathbf{Gr}}$  has the periodicity of the real lattice

$$u_{\mathbf{k}}(\mathbf{r}) = \sum_{\mathbf{G}} u_{\mathbf{k}}(\mathbf{G}) e^{i\mathbf{Gr}}$$

$$\psi_{\mathbf{k}}(\mathbf{r}) = (N\Omega_0)^{-1/2} \sum_{\mathbf{G}} u_{\mathbf{k}}(\mathbf{G}) e^{i(\mathbf{k}+\mathbf{G})\mathbf{r}}$$

$$u_{\mathbf{k}}(\mathbf{G}) = \frac{1}{\Omega_o} \int_{\Omega_o} e^{-i\mathbf{Gr}} u_{\mathbf{k}}(\mathbf{r}) d\mathbf{r} \quad (\text{Fourier transform})$$

Kinetic energy of a plane wave

$$-\frac{\nabla^2}{2} \rightarrow \frac{(\mathbf{k}+\mathbf{G})^2}{2}$$

The coefficients  $u_{\mathbf{k}}(\mathbf{G})$  for the lowest eigenvectors decrease exponentially with the kinetic energy

$$\frac{(\mathbf{k}+\mathbf{G})^2}{2}$$

Selection of plane waves determined by a cut-off energy  $E_{\text{cut}}$

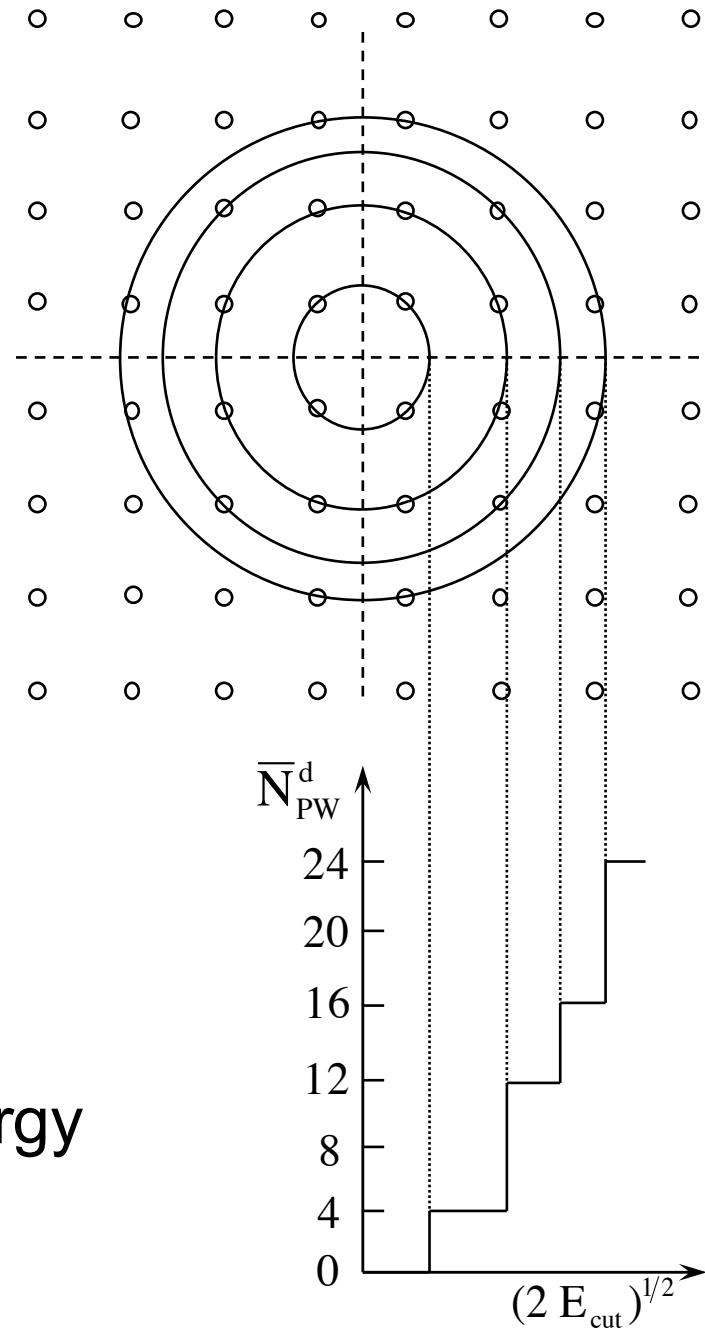
$$\frac{(\mathbf{k}+\mathbf{G})^2}{2} < E_{\text{cut}}$$

$\text{ecut}$

Plane wave sphere

# Number of planewaves

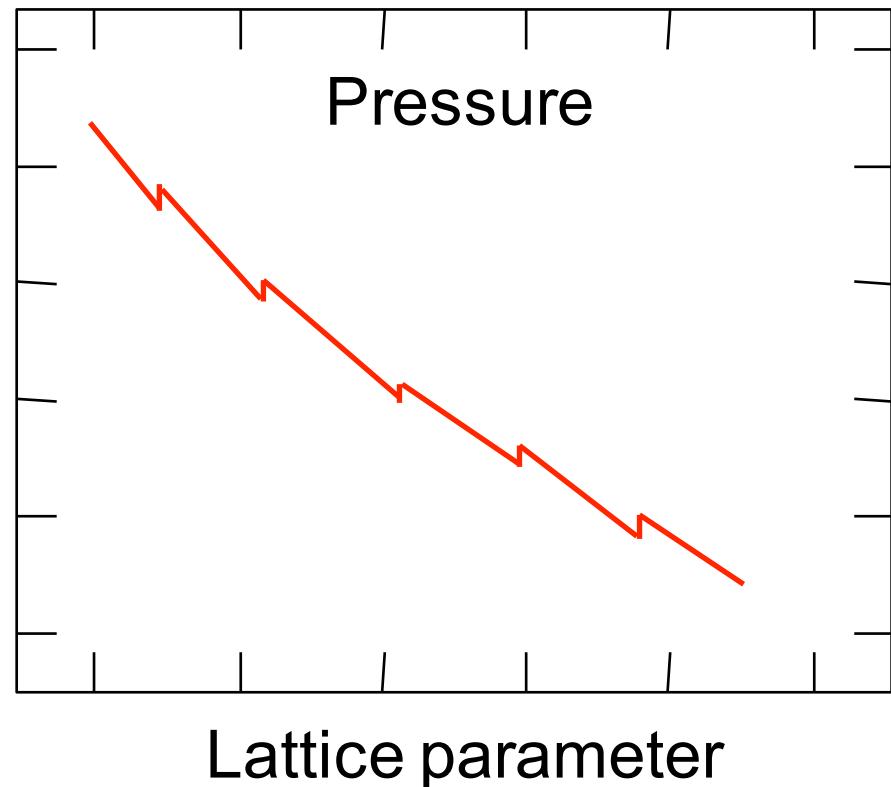
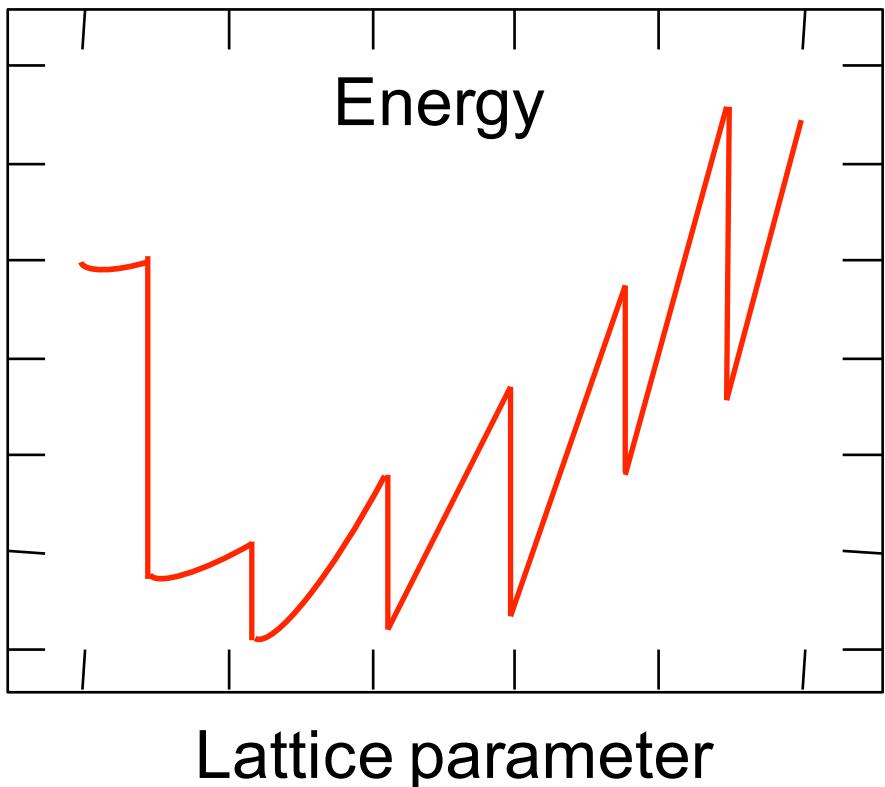
Number of plane waves  
= function of the  
kinetic energy cut-off  
... **not continuous**



Also, a (discontinuous) function of  
**lattice parameter** at fixed kinetic energy

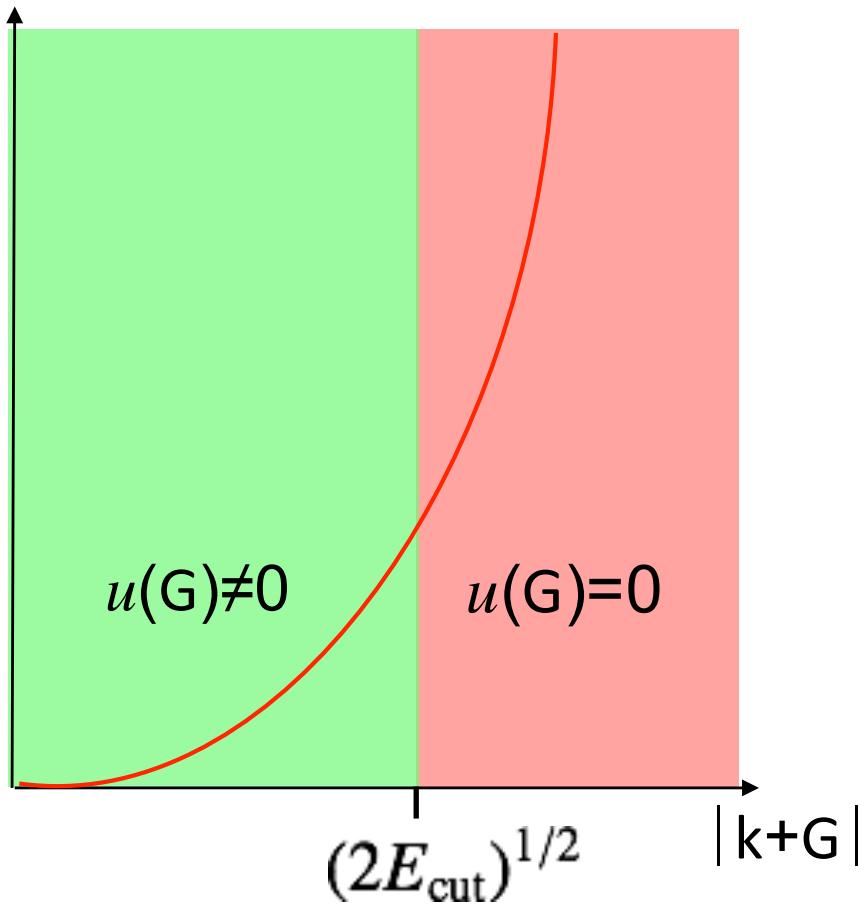
# Discontinuities in energy and pressure

=> Energy (and pressure) also (discontinuous) functions of lattice parameter at fixed kinetic energy

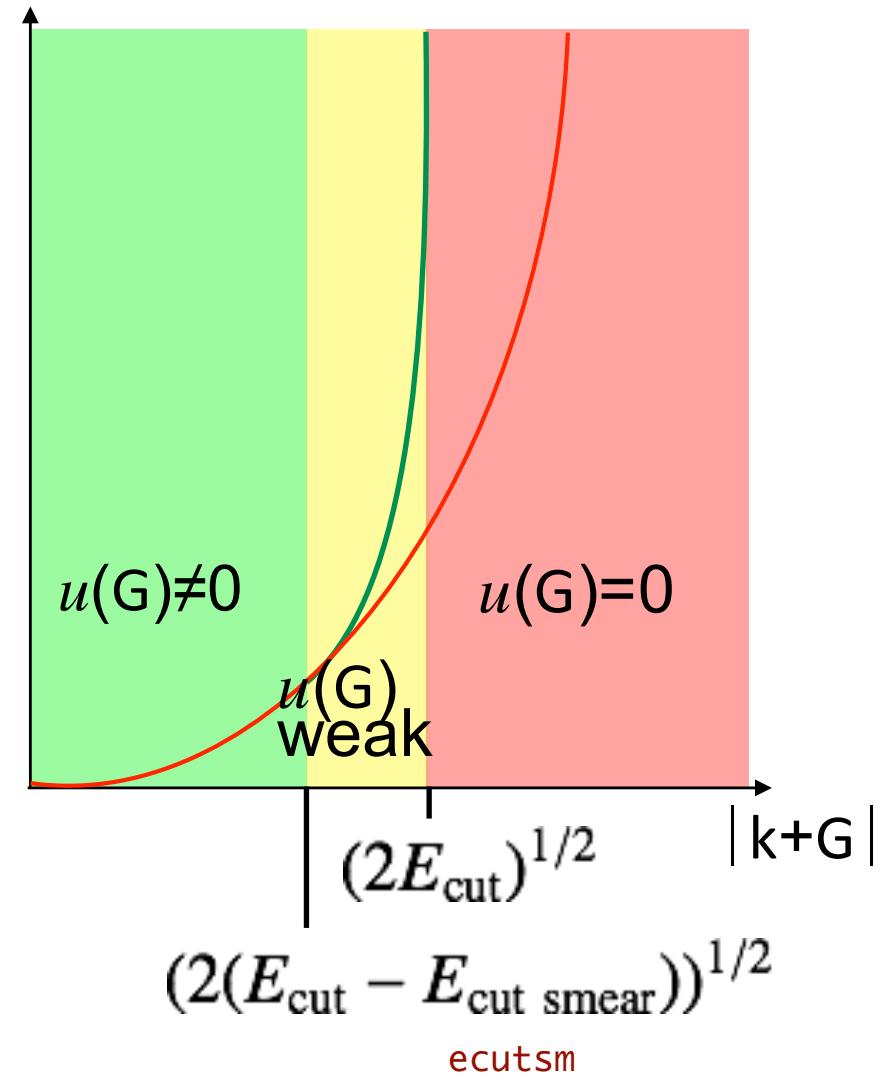


# Removing discontinuities

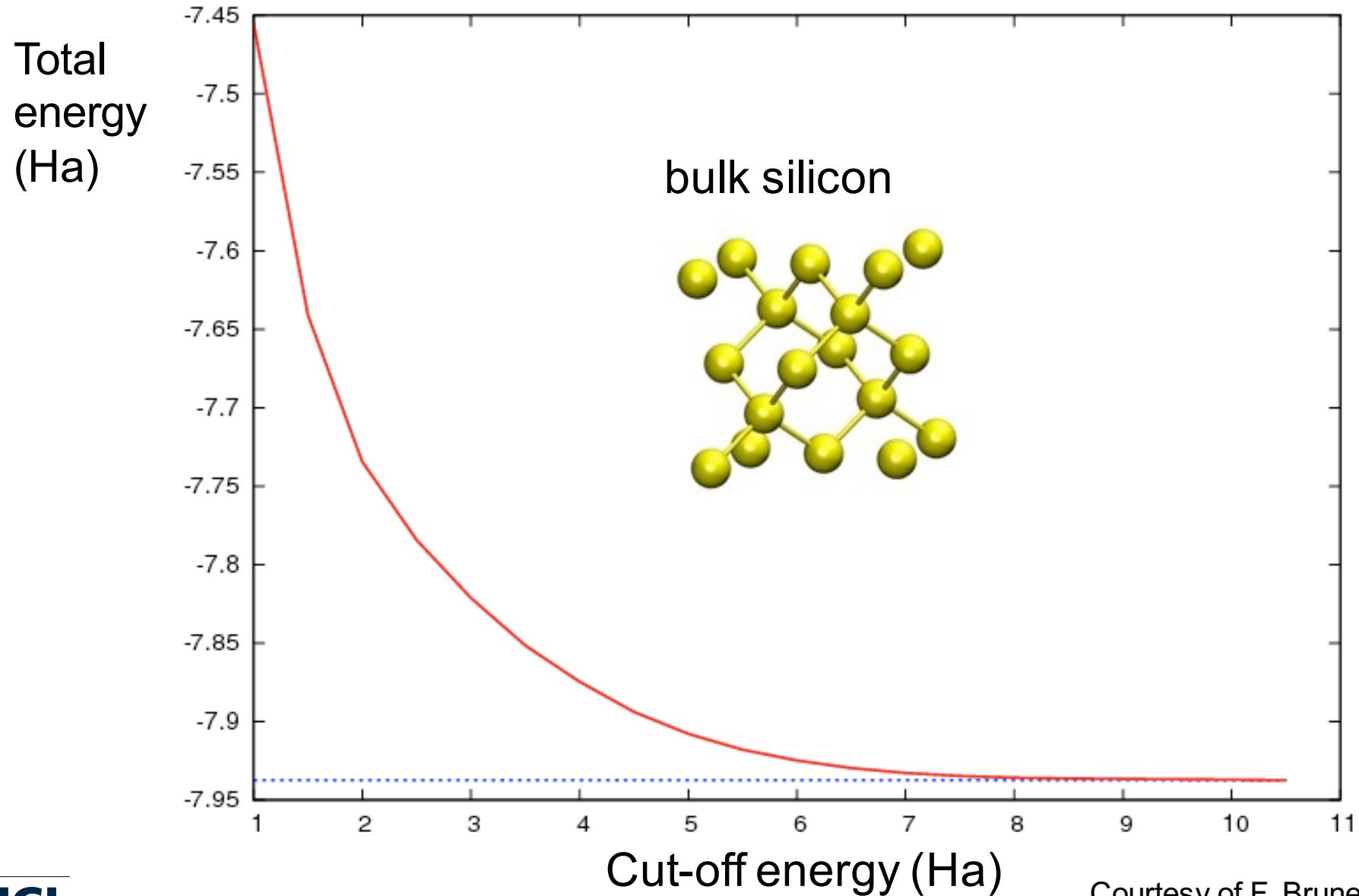
Kinetic energy



Kinetic energy



# Convergence wrt to kinetic energy cutoff



# Plane waves : the density and potential

Fourier transform of a periodic function  $f(\mathbf{r})$

$$f(\mathbf{G}) = \frac{1}{\Omega_{\text{or}}} \int_{\Omega_{\text{or}}} e^{-i\mathbf{Gr}} f(\mathbf{r}) d\mathbf{r} \quad f(\mathbf{r}) = \sum_{\vec{\mathbf{G}}} e^{i\mathbf{Gr}} f(\mathbf{G})$$

Poisson equation  $\Rightarrow n(\mathbf{G})$  and  $V_H(\mathbf{G})$

$$V_H(\mathbf{r}) = \int \frac{n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} d\mathbf{r}' \Leftrightarrow \nabla^2 V_H|_{\mathbf{r}} = -4\pi n(\mathbf{r})$$

Relation between Fourier coefficients:

$$G^2 V_H(\mathbf{G}) = 4\pi n(\mathbf{G}) \quad V_H(\mathbf{G}) = \frac{4\pi}{G^2} n(\mathbf{G})$$

For  $G^2 = 0$  ( $\mathbf{G}=0$ ) divergence of  $V_H$  ( $\mathbf{G}=0$ )

$$n(\mathbf{G}=0) = \frac{1}{\Omega_{\text{or}}} \int_{\Omega_{\text{or}}} n(\mathbf{r}) d\mathbf{r} \quad \text{Average}$$

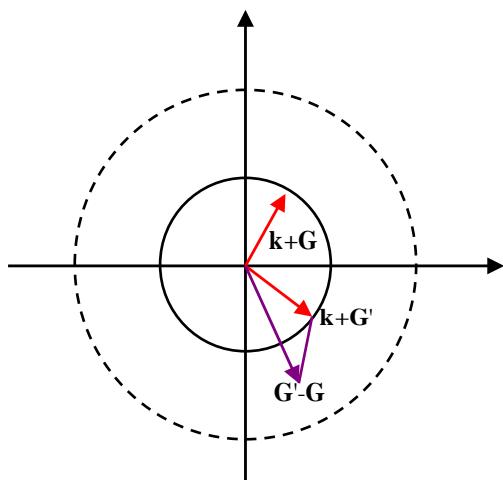
# Representation of the density

Density associated with one eigenfunction :

$$n_{nk}(\mathbf{r}) = u_{nk}^*(\mathbf{r}) u_{nk}(\mathbf{r})$$

Computation of

$$\begin{aligned} & u_{nk}^*(\mathbf{r}) u_{nk}(\mathbf{r}) \\ &= \left( \sum_{\mathbf{G}} u_{nk}^*(\mathbf{G}) e^{-i\mathbf{Gr}} \right) \left( \sum_{\mathbf{G}'} u_{nk}(\mathbf{G}') e^{-i\mathbf{G}'\mathbf{r}} \right) \\ &= \sum_{\mathbf{GG}'} [u_{nk}^*(\mathbf{G}) u_{nk}(\mathbf{G}')] e^{i(\mathbf{G}' - \mathbf{G})\mathbf{r}} \end{aligned}$$



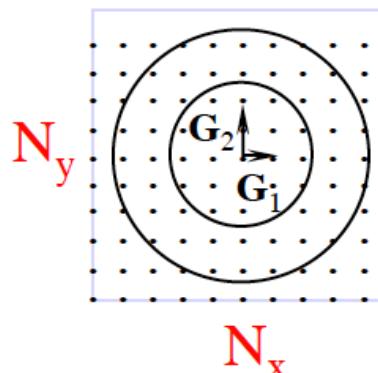
Non-zero coefficients for  $\mathbf{k} + \mathbf{G} \in$  sphere  
 $\mathbf{k} + \mathbf{G}' \in$  sphere

The sphere for  $n(\mathbf{G})$  has a double radius

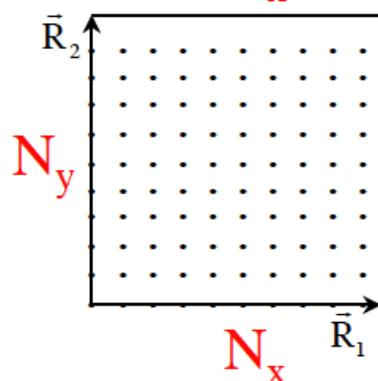
# From real space to reciprocal space

$$n(\mathbf{r}) = \sum_{\mathbf{G} \in \text{sphere}(2)} n(\mathbf{G}) e^{i\mathbf{Gr}}$$

Use of the discrete Fourier transform  $\{\mathbf{r}_i\} \leftrightarrow \{\mathbf{G}\}$



Reciprocal lattice



Real lattice: original cell

$$n(\mathbf{G}) = \frac{1}{N_{\mathbf{r}_i}} \sum_{\{\mathbf{r}_i\}} n(\mathbf{r}_i) e^{-i\mathbf{Gr}_i}$$

Fast Fourier Transform algorithm

# Representation : wrap-up

- Choice of a basis (e.g. Plane waves)
- Truncating of the basis -> finite basis

$$\frac{(\mathbf{k}+\mathbf{G})^2}{2} < E_{\text{cut}} \quad \text{Sphere of plane waves}$$

- Discontinuous increase of the number of plane waves ?
  - Smearing of  $u(\mathbf{G})$
  - > Progressive incorporation of new  $\mathbf{G}$  vectors
- Representation of the density
  - Sphere with a double radius in the reciprocal space
- Going from the real space to reciprocal space

Discrete Fourier transform  
Grid of points + Fast Fourier Transform

$$\{\mathbf{r}_i\} \leftrightarrow \{\mathbf{G}\}$$

# Sampling the Brillouin zone

# From discrete states to Brillouin zone

Discrete summations over states :

Total kinetic energy  $\sum_i \langle \psi_i | -\frac{1}{2} \nabla^2 | \psi_i \rangle$

Density  $n(\mathbf{r}) = \sum_i \psi_i^*(\mathbf{r}) \psi_i(\mathbf{r})$

In the periodic case : summation over energy bands +  
integration over the Brillouin zone

Total kinetic energy  $\sum_n \frac{1}{\Omega_{0k}} \int_{\Omega_{0k}} f(\varepsilon_F - \varepsilon_{nk}) \langle \psi_{nk} | -\frac{1}{2} \nabla^2 | \psi_{nk} \rangle dk$

Density  $n(\mathbf{r}) = \sum_n \frac{1}{\Omega_{0k}} \int_{\Omega_{0k}} f(\varepsilon_F - \varepsilon_{nk}) \psi_{nk}^*(\mathbf{r}) \psi_{nk}(\mathbf{r}) dk$

How to treat  $\frac{1}{\Omega_{0k}} \int_{\Omega_{0k}} X_k dk$  ?

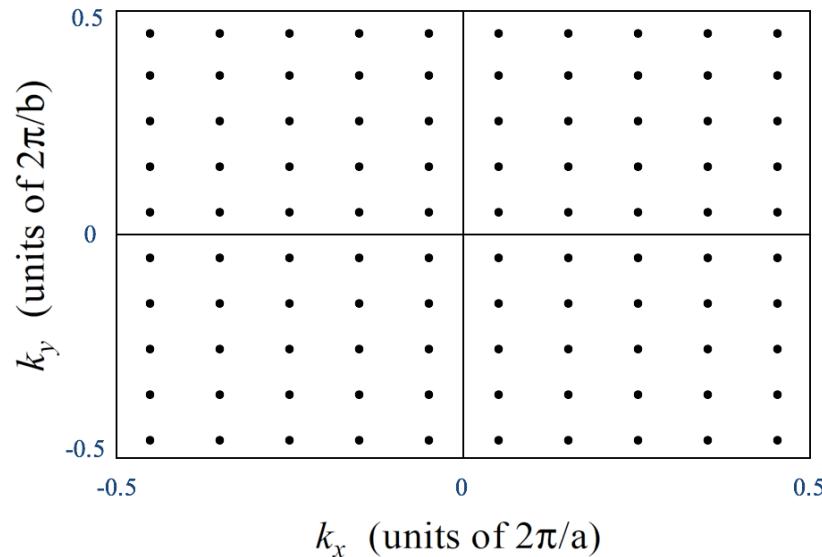
# Brillouin zone integration

$$\frac{1}{\Omega_{ok}} \int_{\Omega_{ok}} X_{\mathbf{k}} \, d\mathbf{k} \Rightarrow \sum_{\{\mathbf{k}\}} w_{\mathbf{k}} X_{\mathbf{k}} \quad [ \text{with } \sum_{\{\mathbf{k}\}} w_{\mathbf{k}} = 1 ]$$

How to chose  $\{\mathbf{k}\}$  and  $\{w_{\mathbf{k}}\}$  ?  
Special points      Weights

Simple answer : Homogeneous grid (1D - 2D - 3D) and equal weights

*Homogeneous sampling of the Brillouin zone*



# Brillouin zone integration

Theorem :

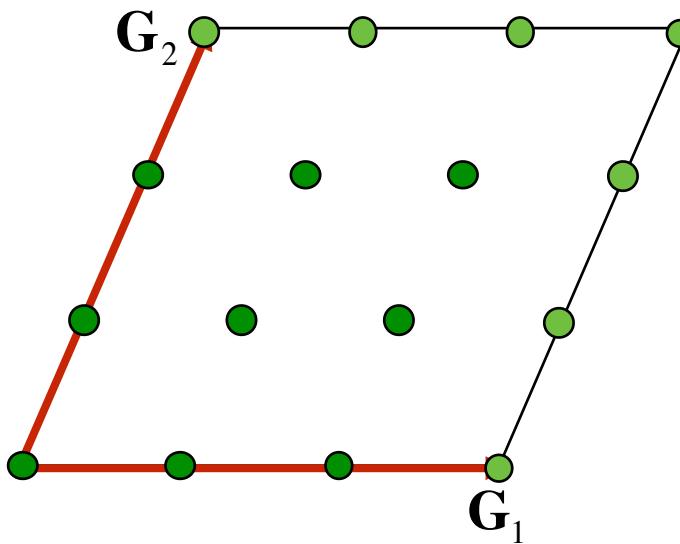
- If
- the integrand is periodic
  - the integrand is continuous + derivable at all orders ( $C^\infty D^\infty$ )
  - $\{\mathbf{k}\}$  homogeneous grid (1D - 2D - 3D) and  $w_{\mathbf{k}}$  all equal

Then exponential convergence, with respect to  $\Delta k$

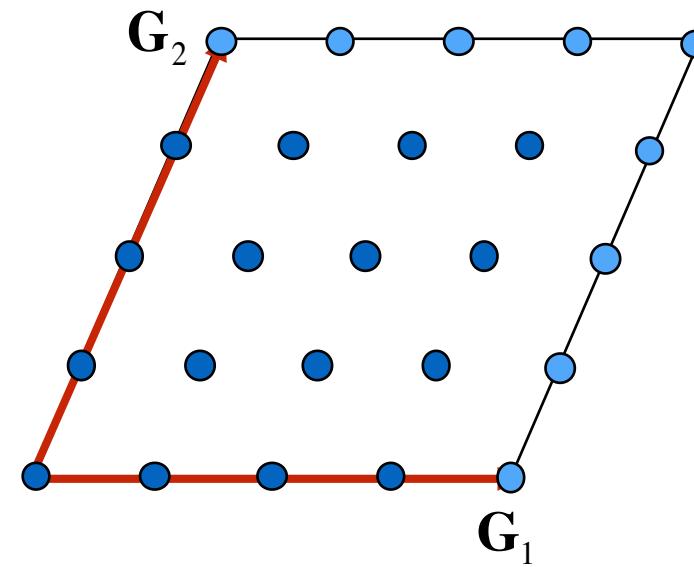
- OK for semiconductors/insulators where the occupation number is independent of  $k$  within a band
- Convergence : one ought to test several grids with different  $\Delta k$
- Monkhorst & Pack grids (Phys. Rev. B 13, 5188 (1976))  
 $k_1 \times k_2 \times k_3$  points + simple cubic, FCC, BCC ...
- Other techniques ... (tetrahedron method)

# BZ integration : Monkhorst-Pack grid

- Uniformly spaced grid of  $n_{k_1} \times n_{k_2} \times n_{k_3}$  points in the first Brillouin Zone  
[Monkhorst & Pack, Phys. Rev. B 13, 5188 (1976)]



$$n_{k1} = n_{k2} = 3$$

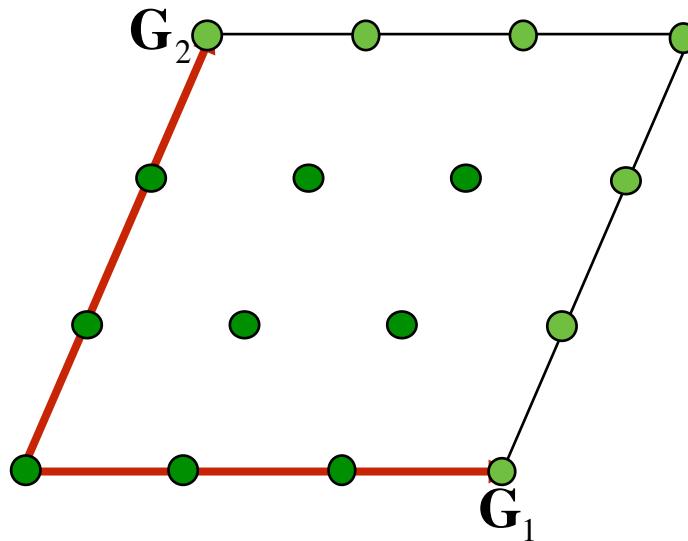


$$n_{k1} = n_{k2} = 4$$

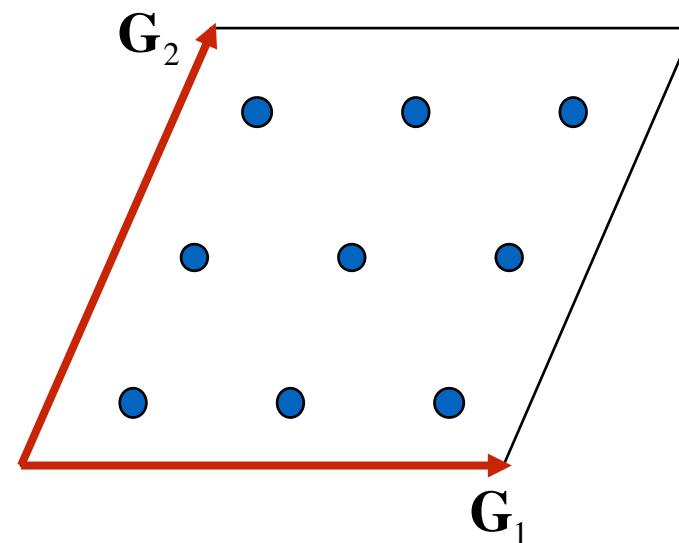
ngkpt nk1 nk2 nk3

# Unshifted and shifted grids

- k-points grid can be chosen to be shifted : not centered at  $\Gamma$ .
- Advantage : comparable accuracy can be obtained with fewer k-points in IBZ (especially for highly symmetric cases)



$n_{k1} = n_{k2} = 3$   
unshifted

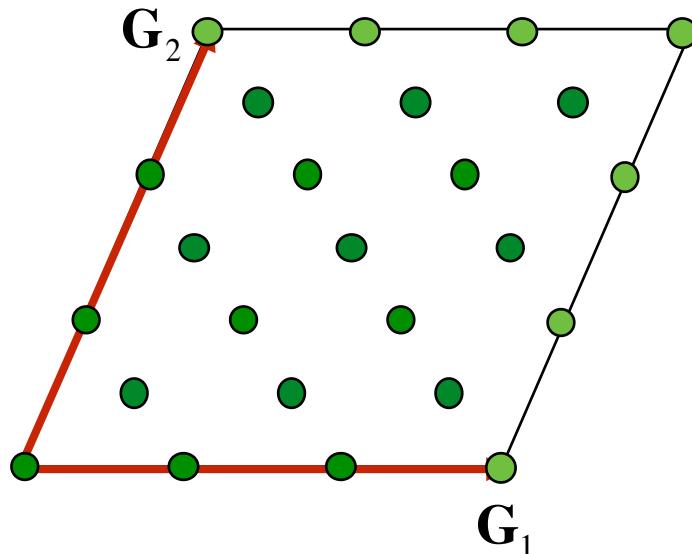


$n_{k1} = n_{k2} = 3$   
shifted by  $(1/2, 1/2)$

ngkpt nk1 nk2 nk3  
shiftk sk1 sk2 sk3    (default: 0.5 0.5 0.5)

# Combining grids with various shifts

- k-points grid with various shifts can also be combined.

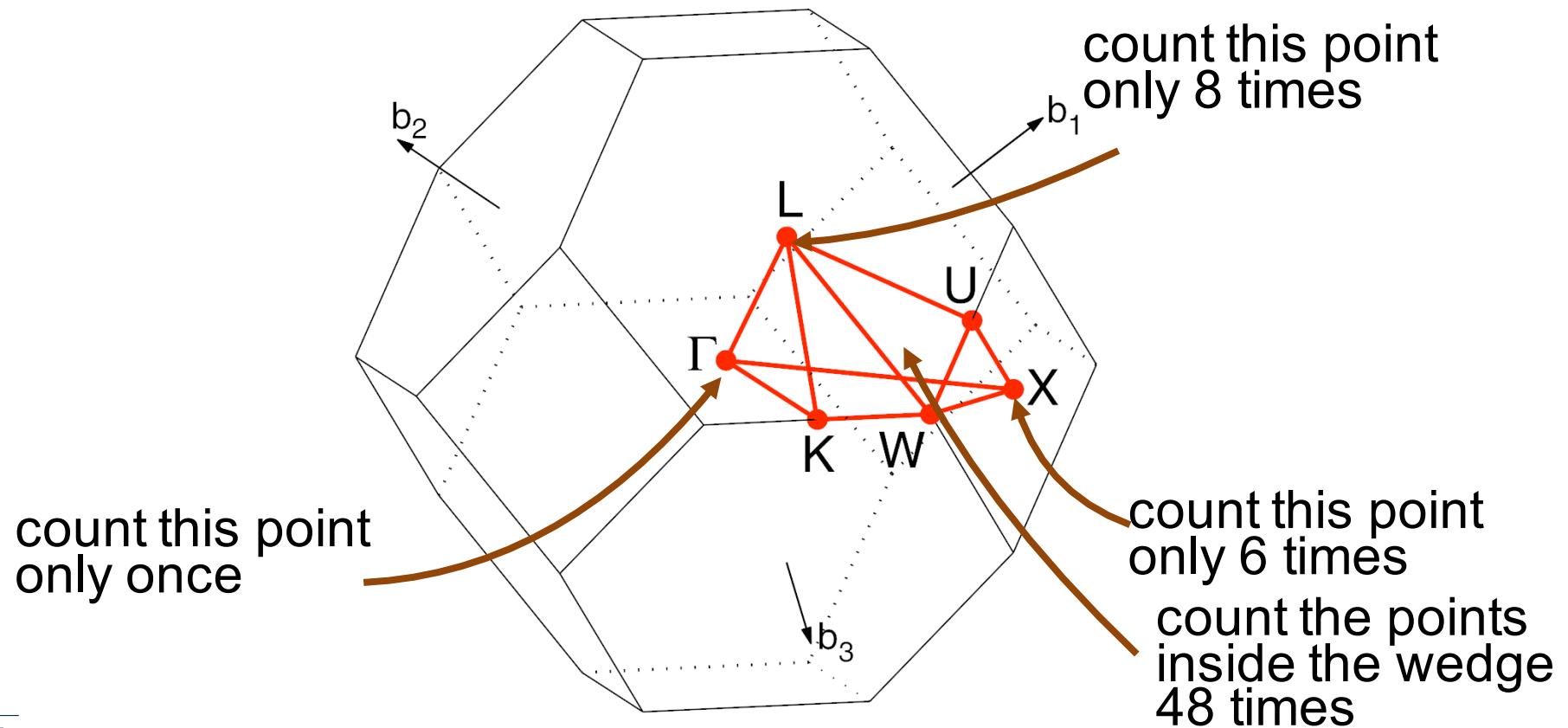


combining unshifted  
and shifted by  $(1/2, 1/2)$   
for  $n_{k1} = n_{k2} = 3$

ngkpt	nk1	nk2	nk3
nshiftk	nsk		
shiftk	sk1(1)	sk2(1)	sk3(1)
	sk1(2)	sk2(2)	sk3(2)
...	...	...	...
	sk1(nsk)	sk2(nsk)	sk3(nsk)

# Irreducible wedge

- Using symmetries to avoid summing entire BZ :
- Restrict the sum to the Irreducible Brillouin zone (IBZ) provided that weights are adapted.

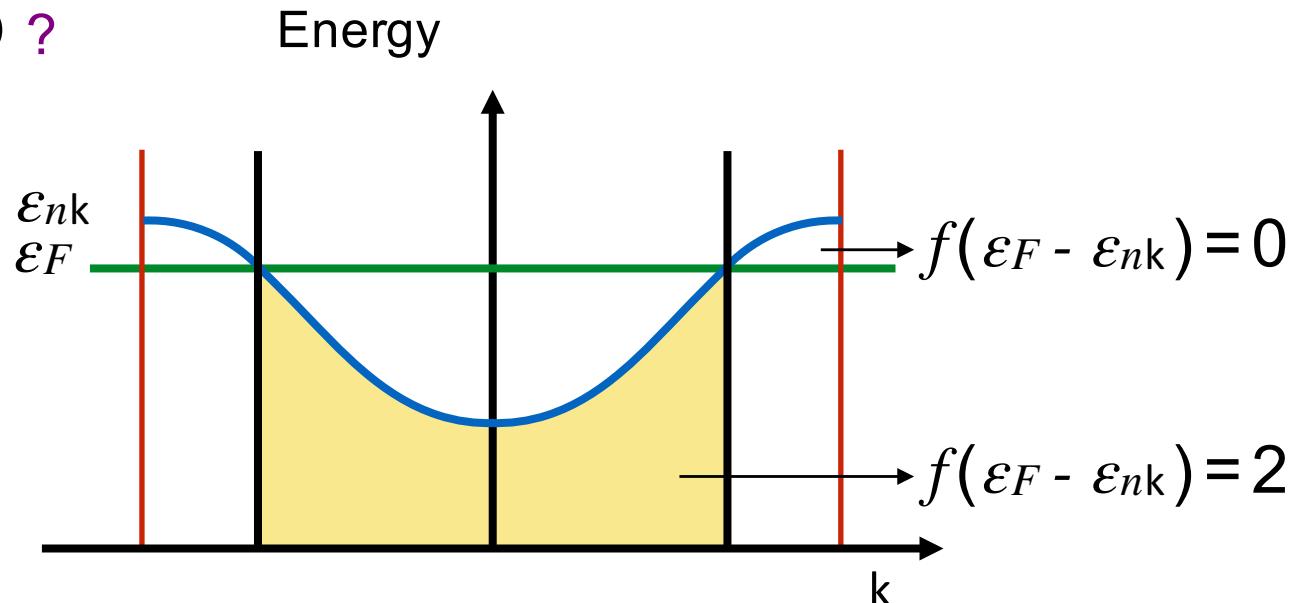


# Treatment of metals (I)

Behaviour of  $f(\epsilon_F - \epsilon_{nk})$  ?

Discontinuity of integrand at Fermi level

Smearing technique



First trial : generalisation of DFT to finite temperature

$$f(\epsilon_{nk}) = \frac{1}{1+e^{(\epsilon_{nk}-\epsilon_F)/kT}}$$

$f$  goes from 0 to 2 in an energy range

$$\sigma = k_B T$$

$$E(T) \approx E(T=0) + \alpha T^2 + \dots$$

$$F(T) = E - TS$$

occpt 3  
tsmear  $\sigma$

Problem :  $T$  needed to recover the same convergence as for semiconductors is very high ( $>> 2000$  K )

# Treatment of metals (II)

Better technique : obtain  $E(\sigma = 0)$  from total energy expression  $E(\sigma)$  with **modified occupation numbers**, and  $\sigma$  **similar** to a temperature

$$E(\sigma) = E(\sigma = 0) + \alpha\sigma^2 + O(\sigma^3) \quad \text{with } \alpha \text{ small}$$

or  $E(\sigma) = E(\sigma = 0) + \alpha\sigma^n + O(\sigma^{n+1}) \quad \text{with } n > 2$

$$f_{nk}(\varepsilon_{nk}) = s \cdot \underbrace{\int_{t=\frac{\varepsilon_{nk}-\varepsilon_F}{\sigma}}^{\infty} \tilde{\delta}(t) dt}_{\text{Spin factor}} \quad [\text{with } \int_{-\infty}^{\infty} \tilde{\delta}(t) dt = 1]$$

Gaussian smearing  $\tilde{\delta}(x) = \frac{1}{\sqrt{\pi}} e^{-x^2} \Rightarrow \alpha \text{ small}$  occopt 7

Gauss-Hermite smearing  $\tilde{\delta}(x) = \frac{1}{\sqrt{\pi}} \left( \frac{3}{2} - x^2 \right) e^{-x^2}$

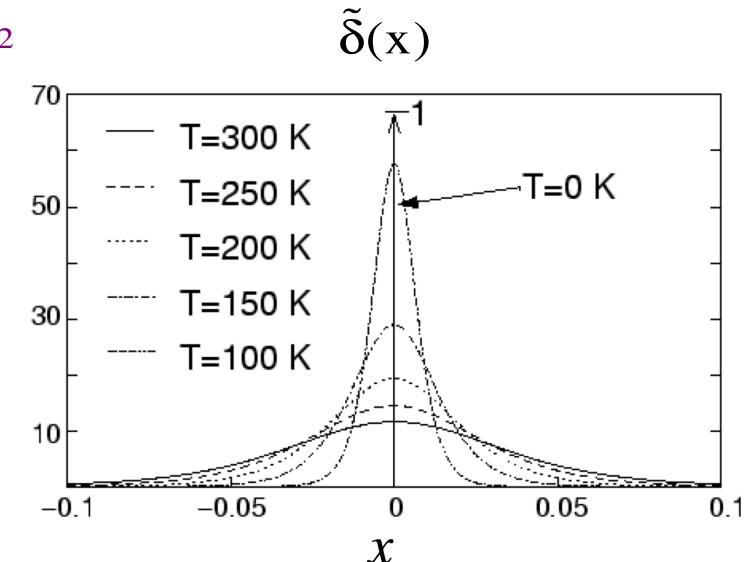
$\Rightarrow n = 4$  but occupations can be negative

...

'Cold Smearing' occopt 4/5

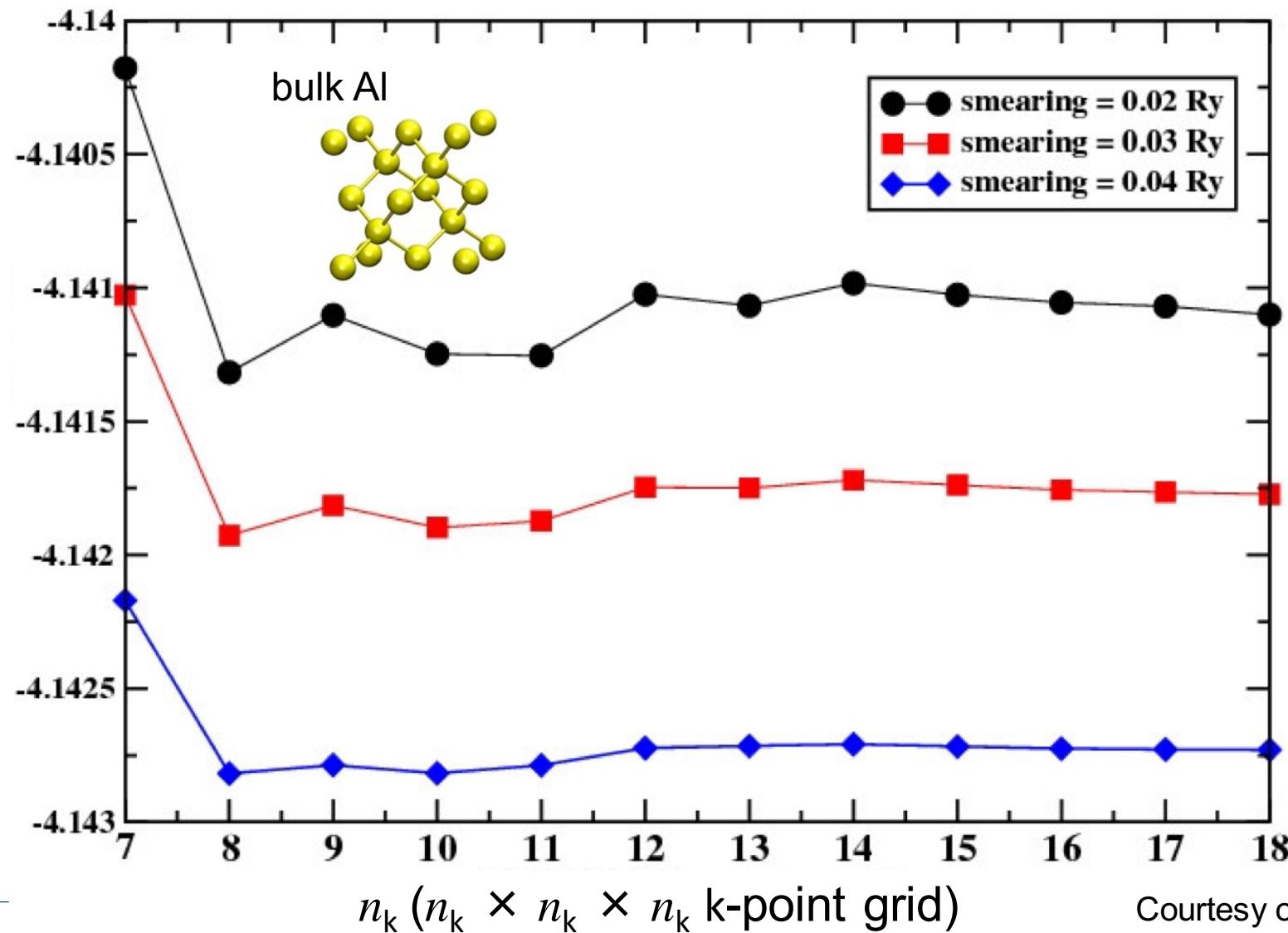
(Marzari et al, Phys. Rev. Lett. 82, 3296 (1999))

$\Rightarrow n = 3$  with positive occupations



# Convergence wrt k-points and smearing

Total energy (Ry)



# How many k points ? Smearing width ?

Rule of thumb ! Goal : lattice parameter converged better than 0.5 %

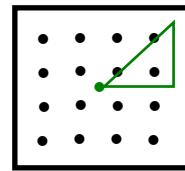
Semiconductors - Insulators       $\# \mathbf{k} \times N_{atoms}$       50 ... 500

Metals       $\# \mathbf{k} \times N_{atoms}$       1000 ... 2000



Use symmetries  $\Rightarrow$  integration in the **irreducible** Brillouin zone

2D Example



grid  $4 \times 4 = 16$

$\Downarrow$   
3 points in the irreducible Brillouin Zone

Smearing : depends on the density of electronic states (DOS) at the Fermi level

s-p Metal (Al, Na ...)  $\sim$  0.04 Ha

d Metal (Cu, Ag...)  $\sim$  0.01 Ha



magnetism needs small  $\sigma$

# Pseudopotentials

# Core and valence electrons (I)

Core electrons occupy orbitals that are « the same » in the atomic environment or in the bonding environment

It depends on the accuracy of the calculation !

Separation between core and valence orbitals : the density...

$$\begin{aligned} n(\mathbf{r}) &= \sum_i^N \psi_i^*(\mathbf{r}) \psi_i(\mathbf{r}) \\ &= \sum_{i \in \text{core}}^{N_{\text{core}}} \psi_i^*(\mathbf{r}) \psi_i(\mathbf{r}) + \sum_{i \in \text{val}}^{N_{\text{val}}} \psi_i^*(\mathbf{r}) \psi_i(\mathbf{r}) = n_{\text{core}}(\mathbf{r}) + n_{\text{val}}(\mathbf{r}) \end{aligned}$$

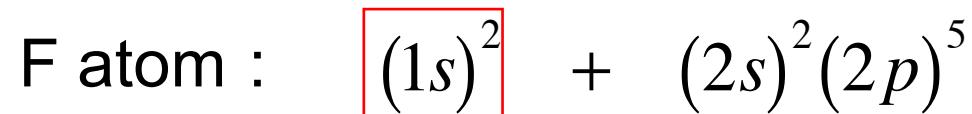
« Frozen core » for  $i \in \text{core}$  :  $\psi_i = \psi_i^{\text{atom}}$

# Small core / Large core

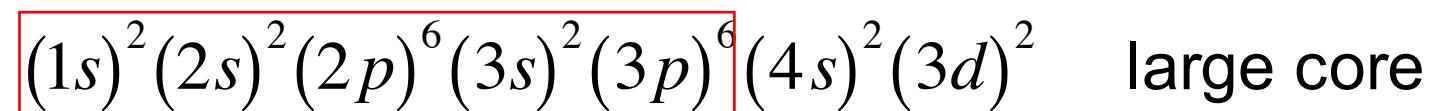
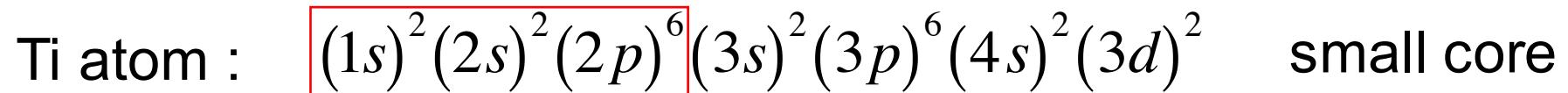
It depends on the target accuracy of the calculation !

(remark also valid for pseudopotentials, with similar cores)

For some elements, the core/valence partitioning is obvious,  
for some others, it is not.



IP 1keV 10-100 eV



IP 99.2 eV 43.3eV

Gd atom : small core with n=1,2,3 shells , might include  
4s, 4p, and 4d in the core. 4f partially filled

# Core and valence electrons (II)

Separation between core and valence orbitals : the energy ...

$$E_{\text{KS}}[\{\psi_i\}] = \sum_i \langle \psi_i | -\frac{1}{2} \nabla^2 | \psi_i \rangle + \int V_{ext}(\mathbf{r}) n(\mathbf{r}) d\mathbf{r} + \frac{1}{2} \int \frac{n(\mathbf{r}_1) n(\mathbf{r}_2)}{|\mathbf{r}_1 - \mathbf{r}_2|} d\mathbf{r}_1 d\mathbf{r}_2 + E_{\text{xc}}[n]$$

$$\begin{aligned} E_{\text{KS}}[\{\psi_i\}] &= \sum_{i \in \text{core}}^{N_{\text{core}}} \langle \psi_i | -\frac{1}{2} \nabla^2 | \psi_i \rangle + \int V_{ext}(\mathbf{r}) n_{\text{core}}(\mathbf{r}) d\mathbf{r} + \frac{1}{2} \int \frac{n_{\text{core}}(\mathbf{r}_1) n_{\text{core}}(\mathbf{r}_2)}{|\mathbf{r}_1 - \mathbf{r}_2|} d\mathbf{r}_1 d\mathbf{r}_2 \\ &\quad + \sum_{i \in \text{val}}^{N_{\text{val}}} \langle \psi_i | -\frac{1}{2} \nabla^2 | \psi_i \rangle + \int V_{ext}(\mathbf{r}) n_{\text{val}}(\mathbf{r}) d\mathbf{r} + \frac{1}{2} \int \frac{n_{\text{val}}(\mathbf{r}_1) n_{\text{val}}(\mathbf{r}_2)}{|\mathbf{r}_1 - \mathbf{r}_2|} d\mathbf{r}_1 d\mathbf{r}_2 \\ &\quad + \int \frac{n_{\text{val}}(\mathbf{r}_1) n_{\text{core}}(\mathbf{r}_2)}{|\mathbf{r}_1 - \mathbf{r}_2|} d\mathbf{r}_1 d\mathbf{r}_2 + E_{\text{xc}}[n_{\text{core}} + n_{\text{val}}] \end{aligned}$$

# Removing core electrons (I)

From the previous construction : valence orbitals must still be orthogonal to core orbitals  
( => oscillations, slope at the nucleus ...)

Pseudopotentials try to remove completely the core orbitals from the simulation

Problem with the number of nodes  
This is a strong modification of the system ...

Pseudopotentials confine the strong changes within a « cut-off radius »

# Removing core electrons (II)

Going from  $\left(-\frac{1}{2} \nabla^2 + v\right) |\psi_i\rangle = \varepsilon_i |\psi_i\rangle$

To  $\left(-\frac{1}{2} \nabla^2 + v_{ps}\right) |\psi_{ps,i}\rangle = \varepsilon_{ps,i} |\psi_{ps,i}\rangle$

Possible set of conditions (norm-conserving pseudopotentials)

NCPP - Hamann D.R., Schlüter M., Chiang C, Phys.Rev.Lett. 43, 1494 (1979)

$$\varepsilon_i = \varepsilon_{ps,i}$$

$$\psi_i(\mathbf{r}) = \psi_{ps,i}(\mathbf{r}) \quad \text{for } r > r_c$$

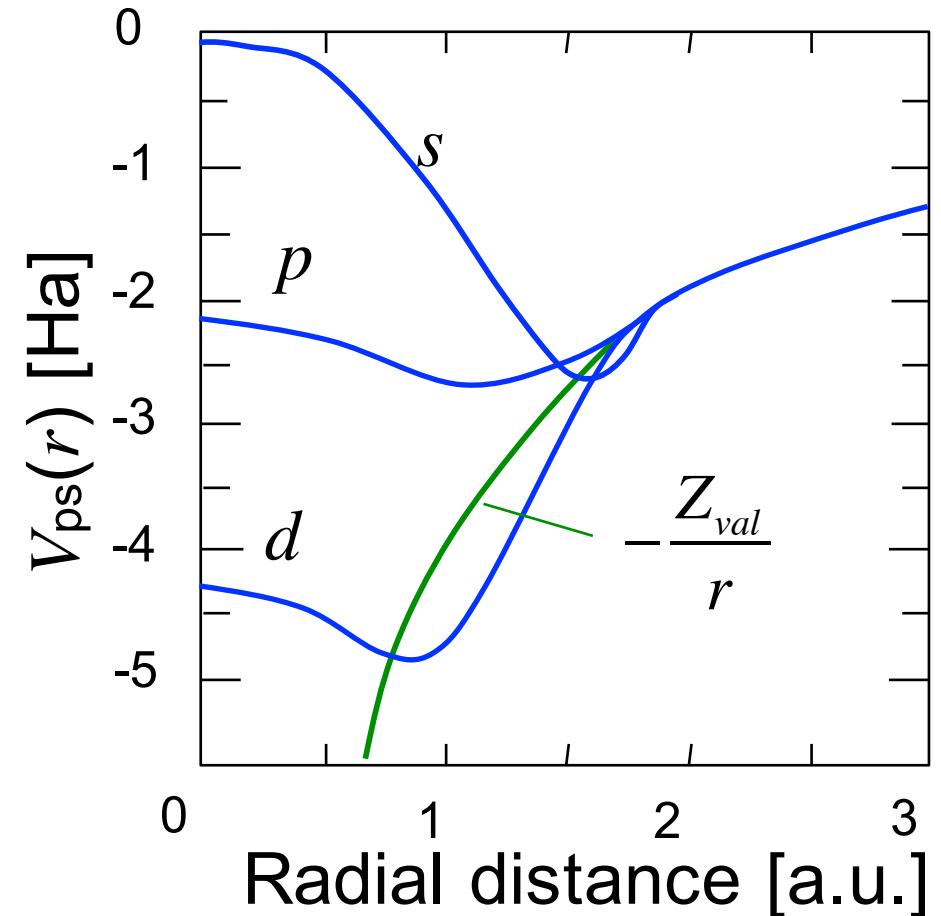
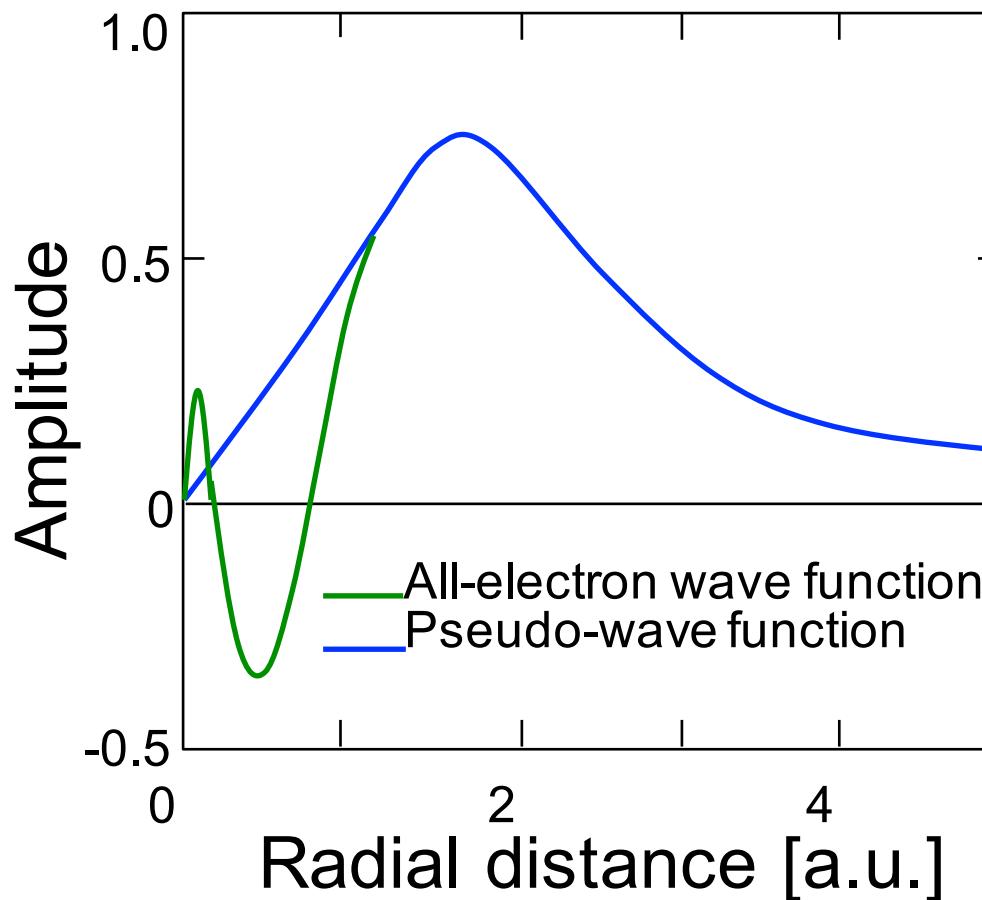
$$\int_{r < r_c} |\psi_i(r)|^2 dr = \int_{r < r_c} |\psi_{ps,i}(r)|^2 dr$$

For the lowest angular momentum channels (s + p ... d ...f)

Generalisation : ultra-soft pseudopotentials (USPP), projector-augmented plane waves (PAW)

# Example of pseudopotential

3s Radial wave function of Si

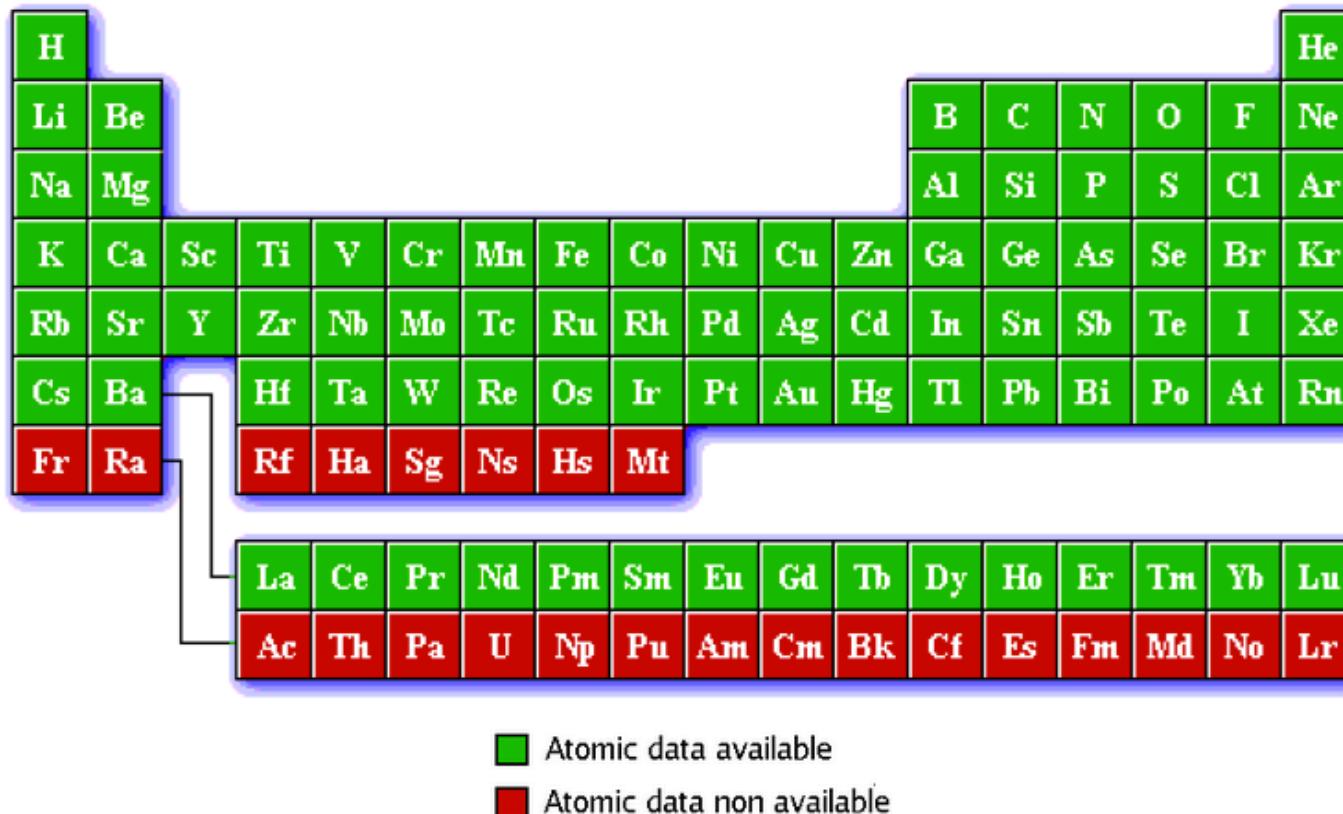


# Pseudopotentials/PAW data in ABINIT

- Preferred PAW atomic dataset table : JTH v0.2

*Jollet, Torrent, Holzwarth, Computer Physics Comm. 185, 1246 (2014)*

<http://www.abinit.org/downloads/PAW2>



Also, possibility to use : GPAW table, GBRV v1.0 table, or norm-conserving pseudopotentials (e.g. ONCVPSP pseudo generator), or many other pseudos

# Computing the forces

# Computing the forces (I)

Born - Oppenheimer approx.  $\Rightarrow$  find electronic ground state in potential created by nuclei.

A starting configuration of nuclei  $\{ R_{\kappa} \}$  is usually NOT in equilibrium geometry.

$$F_{\kappa,\alpha} = - \frac{\partial E}{\partial R_{\kappa,\alpha}} \Bigg|_{\{ \vec{R}_{\kappa} \}} \quad (\text{principle of virtual works})$$

Forces can be computed by finite differences.

Better approach : compute the response to a perturbation

$\Rightarrow$  What is the energy change ?

$$\{ R_{\kappa,\alpha} \} \rightarrow \{ R_{\kappa,\alpha} + \lambda \delta R_{\kappa,\alpha} \}$$

*Small parameter*

# Computing the forces (II)

To simplify, let's compute the derivative of an electronic eigenvalue

Perturbation theory : Hellmann - Feynman theorem

$$\frac{d\epsilon_n}{d\lambda} = \left\langle \Psi_n^{(0)} \left| \frac{d\hat{H}}{d\lambda} \right| \Psi_n^{(0)} \right\rangle$$

$$\frac{d\Psi_n}{d\lambda} \text{ not needed !}$$

Application to the derivative with respect to an atomic displacement :

$$\hat{H} = \hat{T} + \hat{V}_{ext}\{\hat{R}\} \Rightarrow \frac{\partial \hat{H}}{\partial R_{\kappa,\alpha}} = \frac{\partial \hat{V}_{ext}}{\partial R_{\kappa,\alpha}}$$

$$\frac{\partial \epsilon_n}{\partial R_{\kappa,\alpha}} = \left\langle \Psi_n \left| \frac{\partial \hat{H}}{\partial R_{\kappa,\alpha}} \right| \Psi_n \right\rangle = \int n(\mathbf{r}) \frac{\partial \hat{V}_{ext}(\mathbf{r})}{\partial R_{\kappa,\alpha}} d\mathbf{r}$$

# Computing the forces (III)

Generalisation to density functional theory

Reminder :  $E[\psi_i] = \sum_n \langle \psi_i | \hat{T} | \psi_i \rangle + \int n(\mathbf{r}) V_{\text{ext}}(\mathbf{r}) d\mathbf{r} + E_{\text{Hxc}}[n]$

If change of atomic positions ...

$$V_{\text{ext}}(\vec{r}) = \sum_{k'} -\frac{Z_{k'}}{|\vec{r} - \vec{R}_{k'}|} \quad (\text{can be generalized to pseudopotential case})$$

$$\frac{\partial V_{\text{ext}}(\vec{r})}{\partial R_{k,\alpha}} = +\frac{Z_{k'}}{|\vec{r} - \vec{R}_k|^2} \cdot \frac{\partial |\vec{r} - \vec{R}_k|}{\partial R_{k,\alpha}} = -\frac{Z_{k'}}{|\vec{r} - \vec{R}_k|^3} \cdot (\vec{r} - \vec{R}_k)_\alpha$$

$$\boxed{\frac{\partial E}{\partial R_{k,\alpha}} = \int n(r') \frac{\partial V_{\text{ext}}(r')}{\partial R_{k,\alpha}} dr' = - \int \frac{n(r')}{|\vec{r}' - \vec{R}_k|^3} \cdot (\vec{r}' - \vec{R}_k)_\alpha d\vec{r}'}$$

Forces can be computed directly from the density !

# Iterative algorithms

# Algorithmics : problems to be solved

(1) Kohn - Sham equation

$$\underline{\underline{A}} \underline{x}_i = \lambda_i \underline{x}_i$$

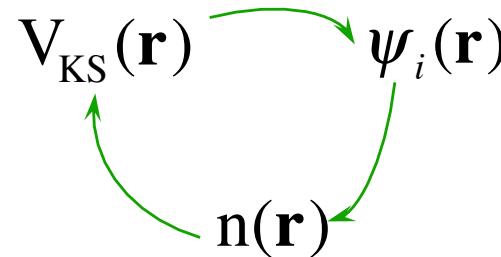
$$\left[ -\frac{1}{2} \nabla^2 + V_{KS}(\mathbf{r}) \right] \psi_i(\mathbf{r}) = \epsilon_i \psi_i(\mathbf{r})$$
$$\{\mathbf{G}_j\} \quad \{\mathbf{r}_j\}$$

Size of the system [2 atoms... 600 atoms...] + vacuum ?

Dimension of the vectors  $\underline{x}_i$  300... 100 000... (if planewaves)

# of (occupied) eigenvectors 4... 1200...

(2) Self-consistency



(3) Geometry optimization

Find the positions  $\{\mathbf{R}_\kappa\}$  of ions such that the forces  $\{\mathbf{F}_\kappa\}$  vanish  
[ = Minimization of energy ]

Current practice : iterative approaches

# The ‘steepest-descent’ algorithm

Forces are gradients of the energy : moving the atoms along gradients is the steepest descent of the energy surface.

=> Iterative algorithm.

Choose a starting geometry, then a parameter  $\lambda$  , and iterately update the geometry, following the forces :

$$\mathbf{R}_{\kappa,\alpha}^{(n+1)} = \mathbf{R}_{\kappa,\alpha}^{(n)} + \lambda \mathbf{F}_{\kappa,\alpha}^{(n)}$$

Equivalent to the simple mixing algorithm of SCF (see later)

# Energy+forces around equilib. geometry

Let us denote the equilibrium geometry as  $\underline{\mathbf{R}}_{\kappa,\alpha}^*$

Analysis of forces close to the equilibrium geometry,  
at which forces vanish, thanks to a Taylor expansion :

$$\underline{\mathbf{F}}_{\kappa,\alpha}(\underline{\mathbf{R}}_{\kappa',\alpha'}) = \cancel{\underline{\mathbf{F}}_{\kappa,\alpha}(\underline{\mathbf{R}}_{\kappa',\alpha'}^*)} + \sum_{\kappa',\alpha'} \frac{\partial \underline{\mathbf{F}}_{\kappa,\alpha}}{\partial \underline{\mathbf{R}}_{\kappa',\alpha'}} \Bigg|_{\{\underline{\mathbf{R}}^*\}} (\underline{\mathbf{R}}_{\kappa',\alpha'} - \underline{\mathbf{R}}_{\kappa',\alpha'}^*) + O((\underline{\mathbf{R}}_{\kappa',\alpha'} - \underline{\mathbf{R}}_{\kappa',\alpha'}^*)^2)$$

Moreover,  $\underline{\mathbf{F}}_{\kappa,\alpha} = -\frac{\partial \mathbf{E}^{\text{BO}}}{\partial \underline{\mathbf{R}}_{\kappa,\alpha}}$        $\frac{\partial \underline{\mathbf{F}}_{\kappa',\alpha'}}{\partial \underline{\mathbf{R}}_{\kappa,\alpha}} = -\frac{\partial^2 \mathbf{E}^{\text{BO}}}{\partial \underline{\mathbf{R}}_{\kappa,\alpha} \partial \underline{\mathbf{R}}_{\kappa',\alpha'}}$

Vector and matrix notation

$$\underline{\mathbf{R}}_{\kappa,\alpha}^* \rightarrow \underline{\mathbf{R}}^*$$

$$\underline{\mathbf{F}}_{\kappa,\alpha} \rightarrow \underline{\mathbf{F}}$$

$$\underline{\mathbf{R}}_{\kappa,\alpha} \rightarrow \underline{\mathbf{R}}$$

$$\frac{\partial^2 \mathbf{E}^{\text{BO}}}{\partial \underline{\mathbf{R}}_{\kappa,\alpha} \partial \underline{\mathbf{R}}_{\kappa',\alpha'}} \Bigg|_{\{\underline{\mathbf{R}}_{\kappa,\alpha}^*\}} \rightarrow \underline{\underline{\mathbf{H}}} \quad (\text{the Hessian})$$

# Steepest-descent : analysis (I)

$$\underline{R}_{\kappa,\alpha}^{(n+1)} = \underline{R}_{\kappa,\alpha}^{(n)} + \lambda \underline{F}_{\kappa,\alpha}^{(n)}$$

Analysis of this algorithm, in the linear regime :

$$\underline{F}(\underline{R}) = \underline{F}(\underline{R}^*) - \underline{\underline{H}}(\underline{R} - \underline{R}^*) + O((\underline{R} - \underline{R}^*)^2)$$
$$\underline{R}^{(n+1)} = \underline{R}^{(n)} + \lambda \underline{F}^{(n)} \rightarrow (\underline{R}^{(n+1)} - \underline{R}^*) = (\underline{R}^{(n)} - \underline{R}^*) - \lambda \underline{\underline{H}}(\underline{R}^{(n)} - \underline{R}^*)$$
$$(\underline{R}^{(n+1)} - \underline{R}^*) = (1 - \lambda \underline{\underline{H}})(\underline{R}^{(n)} - \underline{R}^*)$$

For convergence of the iterative procedure, the "distance" between trial geometry and equilibrium geometry must decrease.

- 1) Can we predict conditions for convergence ?
- 2) Can we make convergence faster ?

Need to understand the action of  
the matrix (or operator)

$$1 - \lambda \underline{\underline{H}}$$

# Steepest-descent : analysis (II)

What are the eigenvectors and eigenvalues of  $\underline{\underline{H}}$  ?

$\underline{\underline{H}}$  symmetric,  
positive definite matrix

$$\left( = \frac{\partial^2 E^{BO}}{\partial R_{\kappa,\alpha} \partial R_{\kappa',\alpha'}} \Bigg|_{\{R_{\kappa,\alpha}^*\}} \right)$$

$\underline{\underline{H}} \underline{f}_i = h_i \underline{f}_i$  where  $\{\underline{f}_i\}$  form a complete, orthonormal, basis set

Discrepancy decomposed as  $(\underline{R}^{(n)} - \underline{R}^*) = \sum c_i^{(n)} \underline{f}_i$

and  $(\underline{R}^{(n+1)} - \underline{R}^*) = (1 - \lambda \underline{\underline{H}}) \sum_i c_i^{(n)} \underline{f}_i = \sum_i c_i^{(n)} (1 - \lambda h_i) \underline{f}_i$

The coefficient of  $\underline{f}_i$  is multiplied by  $1 - \lambda h_i$

Iteratively :  $(\underline{R}^{(n)} - \underline{R}^*) = \sum_i c_i^{(0)} (1 - \lambda h_i)^{(n)} \underline{f}_i$

# Steepest-descent : analysis (III)

$$\left( \underline{R}^{(n)} - \underline{R}^* \right) = \sum_i c_i^{(0)} (1 - \lambda h_i)^{(n)} \underline{f}_i$$

The size of the discrepancy decreases if  $|1 - \lambda h_i| < 1$

Is it possible to have  $|1 - \lambda h_i| < 1$ , for all eigenvalues ?

H positive definite  $\Rightarrow$  all  $h_i$  are positive

Yes ! If  $\lambda$  positive, sufficiently small ...

# Steepest-descent : analysis (IV)

$$(\underline{R}^{(n)} - \underline{R}^*) = \sum_i c_i^{(0)} (1 - \lambda h_i)^{(n)} f_i$$

How to determine the optimal value of  $\lambda$  ?

The maximum of all  $|1 - \lambda h_i|$  should be as small as possible.

At the optimal value of  $\lambda$ , what will be the convergence rate ?

( = by which factor is reduced the worst component of  $(\underline{R}^{(n)} - \underline{R}^*)$  ? )

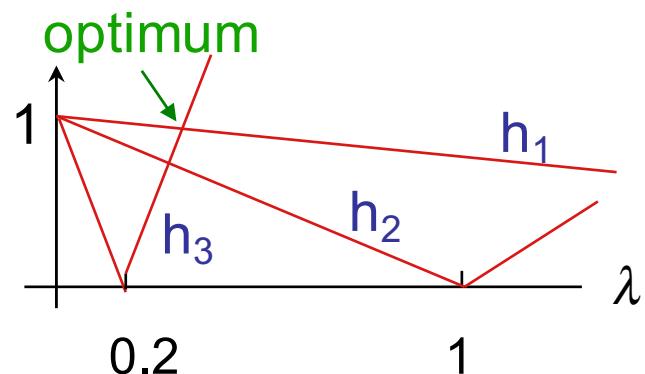
As an exercise : suppose  $\begin{cases} h_1 = 0.2 \\ h_2 = 1.0 \\ h_3 = 5.0 \end{cases}$  => what is the best value of  $\lambda$  ?  
+ what is the convergence rate ?

Hint : draw the three functions  $|1 - \lambda h_i|$  as a function of  $\lambda$ . Then, find the location of  $\lambda$  where the largest of the three curves is the smallest.  
Find the coordinates of this point.

# Steepest-descent : analysis (V)

Minimise the maximum of  $|1 - \lambda h_i|$

$$\begin{array}{ll} h_1 = 0.2 & |1 - \lambda .0.2| \\ h_2 = 1.0 & |1 - \lambda .1| \\ h_3 = 5.0 & |1 - \lambda .5| \end{array}$$



$$\begin{aligned} \text{optimum} &\Rightarrow \lambda = 5 \\ \text{optimum} &\Rightarrow \lambda = 1 \\ \text{optimum} &\Rightarrow \lambda = 0.2 \end{aligned} \quad \left. \right\} ?$$

$$\mu = |1 - \underbrace{\lambda}_{\text{positive}} 0.2| = |1 - \underbrace{\lambda}_{\text{negative}} 5|$$

$$1 - \lambda . 0.2 = -(1 - \lambda . 5)$$

$$2 - \lambda (0.2 + 5) = 0 \Rightarrow \lambda = 2/5.2$$

$$\mu = 1 - 2 \cdot (0.2 / 5.2)$$

Only  $\sim 8\%$  decrease of the error, per iteration ! Hundreds of iterations will be needed to reach a reduction of the error by 1000 or more.

Note : the second eigenvalue does not play any role.

The convergence is limited by the extremal eigenvalues : if the parameter is too large, the smallest eigenvalue will cause divergence, but for that small parameter, the largest eigenvalue lead to slow decrease of the error...

# The condition number

In general,  $\lambda_{\text{opt}} = 2 / (h_{\min} + h_{\max})$

$$\mu_{\text{opt}} = 2 / [1 + (h_{\max}/h_{\min})] - 1 = [(h_{\max}/h_{\min}) - 1] / [(h_{\max}/h_{\min}) + 1]$$

Perfect if  $h_{\max} = h_{\min}$ . Bad if  $h_{\max} \gg h_{\min}$ .

$h_{\max}/h_{\min}$  called the "condition" number. A problem is "ill-conditioned" if the condition number is large. It does not depend on the intermediate eigenvalues.

Suppose we start from a configuration with forces on the order of 1 Ha/Bohr, and we want to reach the target 1e-4 Ha/Bohr. The mixing parameter is optimal.

How many iterations are needed ?

For a generic decrease factor  $\Delta$ , with "n" the number of iterations.

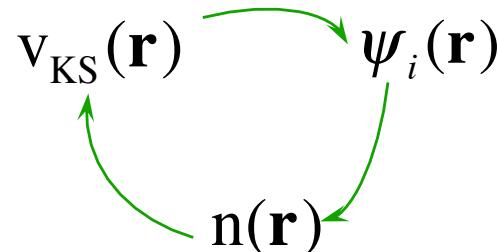
$$\|\underline{F}^{(n)}\| \approx \left( \frac{h_{\max}/h_{\min} - 1}{h_{\max}/h_{\min} + 1} \right)^n \|\underline{F}^{(0)}\|$$

$$\Delta \approx \left( \frac{h_{\max}/h_{\min} - 1}{h_{\max}/h_{\min} + 1} \right)$$

$$n \approx \left[ \ln \left( \frac{h_{\max}/h_{\min} + 1}{h_{\max}/h_{\min} - 1} \right) \right]^{-1} \ln \Delta \approx 0.5 \left( h_{\max}/h_{\min} \right) \ln \frac{1}{\Delta}$$

(The latter approximate equality supposes a large condition number)

# Analysis of self-consistency



Natural iterative methodology (KS : in => out) :

$$v_{in}(\mathbf{r}) \rightarrow \psi_i(\mathbf{r}) \rightarrow n(\mathbf{r}) \rightarrow v_{out}(\mathbf{r})$$

Which quantity plays the role of a force, that should vanish at the solution ?

The difference

$v_{out}(\mathbf{r}) - v_{in}(\mathbf{r})$  (generic name : a "residual")

Simple mixing algorithm  
( $\approx$  steepest- descent)

$$\underline{v}_{in}^{(n+1)} = \underline{v}_{in}^{(n)} + \lambda \left( \underline{v}_{out}^{(n)} - \underline{v}_{in}^{(n)} \right)$$

Analysis ...  $\underline{v}_{out} \left[ \underline{v}_{in} \right] = \underline{v}_{out} \left[ \underline{v}^* \right] + \frac{\delta \underline{v}_{out}}{\delta \underline{v}_{in}} \left( \underline{v}_{in} - \underline{v}^* \right)$

$\searrow \underline{H}$

Like the steepest-descent algorithm, this leads to the requirement to minimize  $|1 - \lambda h_i|$  where  $h_i$  are eigenvalues of

$$\frac{\delta \underline{v}_{out}}{\delta \underline{v}_{in}}$$

# Modify the condition number (II)

$$\underline{R}^{(n+1)} = \underline{R}^{(n)} + \lambda \left( \underline{\underline{H}}^{-1} \right)_{\text{approx}} \underline{F}^{(n)}$$

$$\underline{F}(\underline{R}) = -\underline{\underline{H}} \left( \underline{R} - \underline{R}^* \right) \implies \left( \underline{R}^{(n+1)} - \underline{R}^* \right) = \left( 1 - \lambda \left( \underline{\underline{H}}^{-1} \right)_{\text{approx}} \underline{\underline{H}} \right) \left( \underline{R}^{(n)} - \underline{R}^* \right)$$

- Notes :
- 1) If approximate inverse Hessian perfect, optimal geometry is reached in one step, with  $\lambda = 1$ .  
Steepest-descent NOT the best direction.
  - 2) Non-linear effects not taken into account. For geometry optimization, might be quite large. Even with perfect hessian, need 5-6 steps to optimize a water molecule.
  - 3) Approximating inverse hessian by a multiple of the unit matrix is equivalent to changing the  $\lambda$  value.
  - 4) Eigenvalues and eigenvectors of  $\left( \underline{\underline{H}}^{-1} \right)_{\text{approx}} \underline{\underline{H}}$  govern the convergence : the condition number can be changed.  
 $\left( \underline{\underline{H}}^{-1} \right)_{\text{approx}}$  often called a "pre-conditioner".
  - 5) Generalisation to other optimization problems is trivial.  
(The Hessian is referred to as the Jacobian if it is not symmetric.)

# Modify the condition number (III)

Approximate Hessian can be generated on a case-by-case basis.

Selfconsistent determination of the Kohn-Sham potential :

Jacobian = dielectric matrix.

Lowest eigenvalue **close to 1**.

Largest eigenvalue :

= **1.5 ... 2.5** for small close-shell molecules, and small unit cell solids

(Simple mixing will sometimes converge with parameter set to 1 !)

= the **macroscopic dielectric constant** (e.g. 12 for silicon),

for larger close-shell molecules and large unit cell insulators,

= **diverge** for large-unit cell metals, or open-shell molecules !

**Model** dielectric matrices known for rather homogeneous systems.

Knowledge of approx. macroscopic dielectric constant

=> efficient preconditioner

Work in progress for inhomogeneous systems

(e.g. metals/vacuum systems).

# Advanced algorithms : using the history

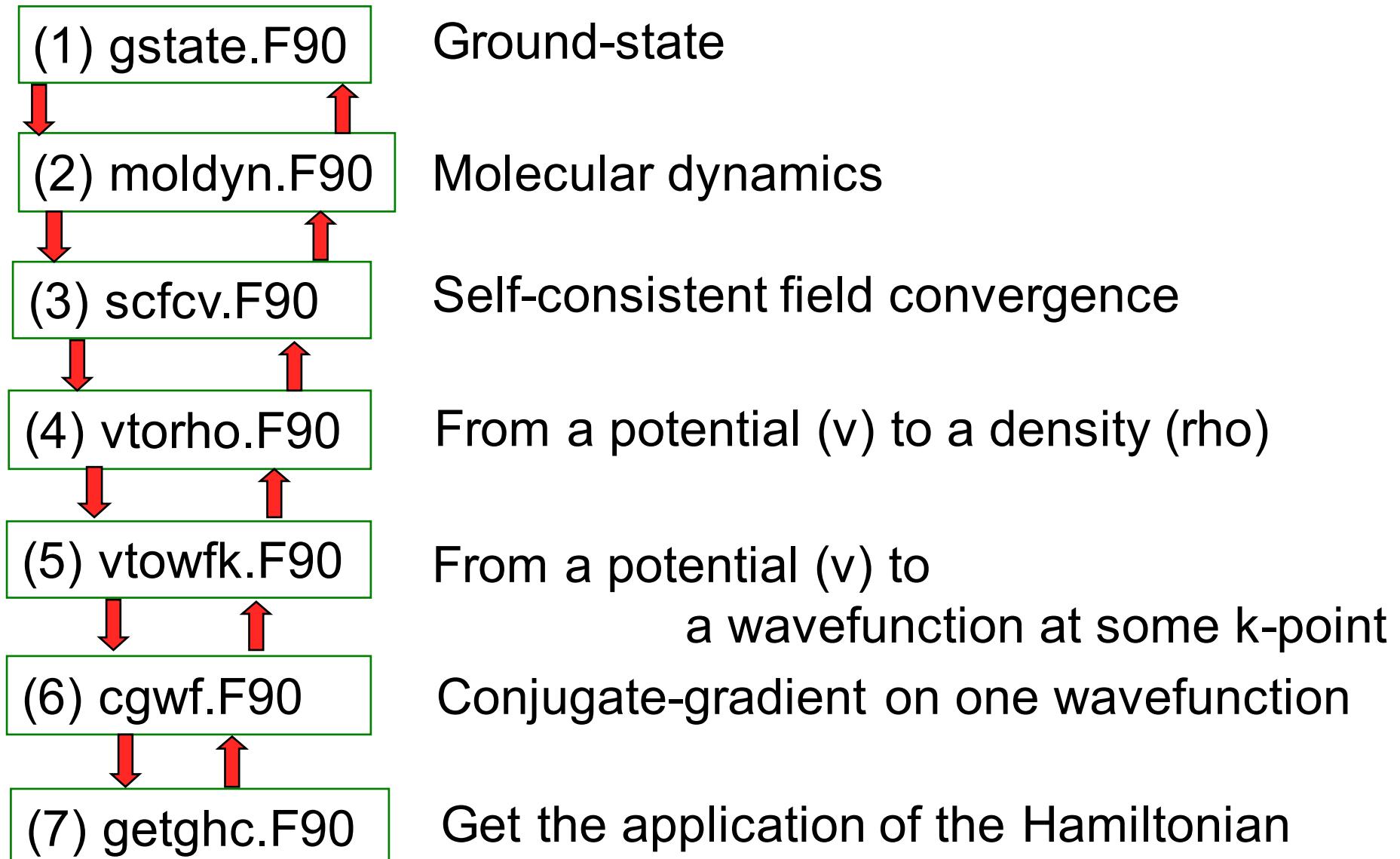
Instead of using only previously computed forces,  
take into account past forces for past positions

Large class of methods :

- Broyden (quasi-Newton-type),
- Davidson,
- conjugate gradients,
- Lanczos ...

Approximate Hessian can be combined with usage of history

# Stages in the main processing unit



# In practice ...

## (1) Kohn - Sham equation

$$\left[ -\frac{1}{2} \nabla^2 + V_{KS}(\mathbf{r}) \right] \psi_i(\mathbf{r}) = \varepsilon_i \psi_i(\mathbf{r})$$

Details are usually hidden to the user  
online

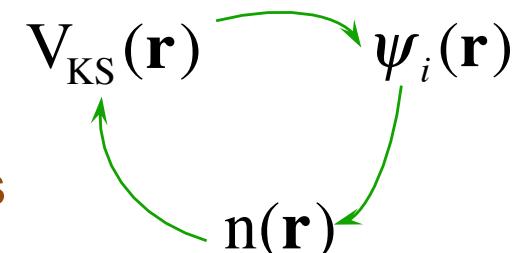
Note that scaling with size of system is quadratic or even cubic

## (2) Self-consistency

Target tolerance toldfe, toldff, tolfff, tolvs

+ Maximal number of loops nstep

Preconditioner diemac, diemix, ...



## (3) Geometry optimization / molecular dynamics

Find the positions  $\{\mathbf{R}_\kappa\}$  of ions such that the forces  $\{\mathbf{F}_\kappa\}$  vanish

Target tolerance tolmxsf

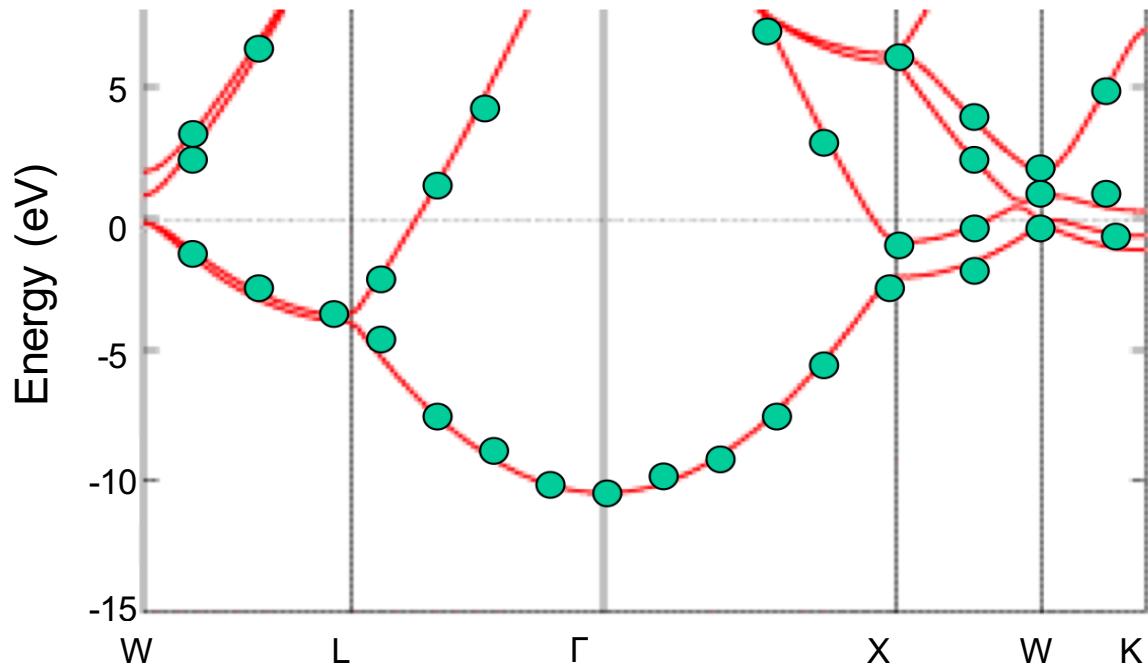
+ Maximal number of loops ntime

Algorithm ionmov

# Computing band structure and density of states

# Non-self consistent calculations

- Once the density has been determined self-consistently, it is possible to compute the eigenenergies/eigenfunctions rapidly for a large number of wavevectors, at fixed KS potential
- Band structure :
  - non self-consistent calculation (`iscf -2`)
  - k-points along high-symmetry directions (`kpto $\leq$ 0;kptbounds;ndivk`).



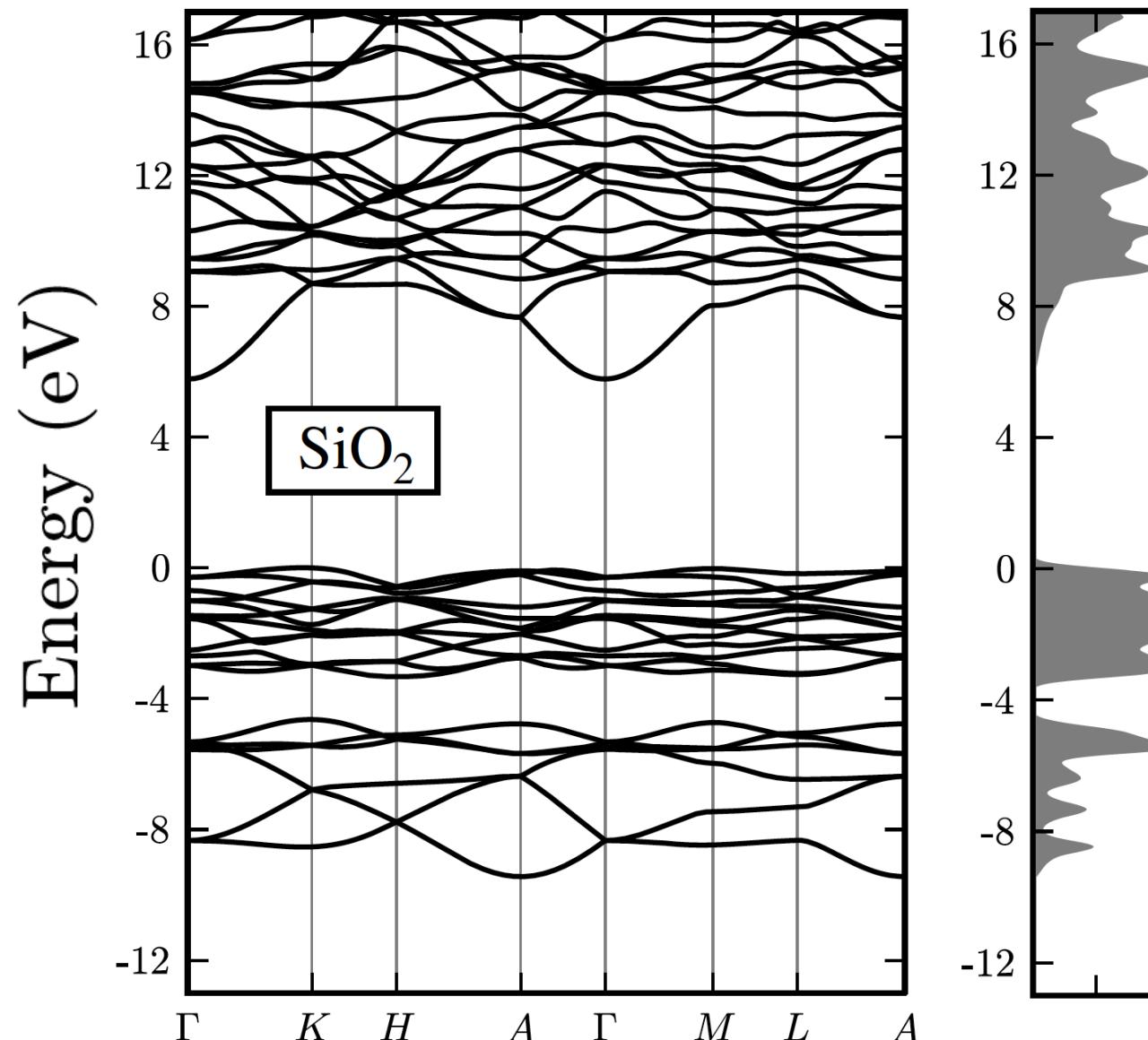
# Density of states

- Density of states (DOS) defined as number of states available in energy range  $E$  to  $E+dE$  :

$$g(E) = \sum_n \frac{1}{\Omega_{0k}} \int_{\Omega_{0k}} \delta(E - \varepsilon_{nk}) d\mathbf{k}$$

- Recipe : determine  $\varepsilon_{nk}$  on a grid of k-points in the BZ using a non self-consistent procedure (**iscf -3**).
- The  $\delta$ -function is approximated by smeared-out function (typically a Gaussian) with a width  $\sigma$  (**prtdos 1**).
- Very fine grid of k-points needed to get sharp features accurately.  
For faster convergence, use tetrahedron method to interpolate between k-points (**prtdos 2**).

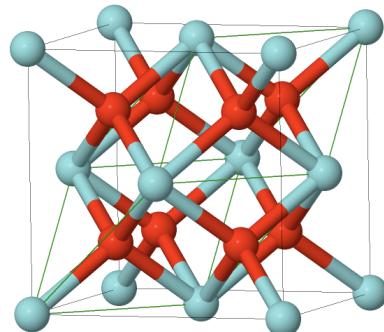
# Example : SiO<sub>2</sub>-quartz



# Summary

- Plane waves basis set
- Brillouin zone integration
- PW need pseudopotentials
- Easy computation of forces

=> relaxation of geometry, or molecular dynamics



$$\psi_{\mathbf{k}}(\mathbf{r}) = (N\Omega_0)^{-1/2} \sum_{\mathbf{G}} u_{\mathbf{k}}(\mathbf{G}) e^{i(\mathbf{k}+\mathbf{G})\mathbf{r}}$$

