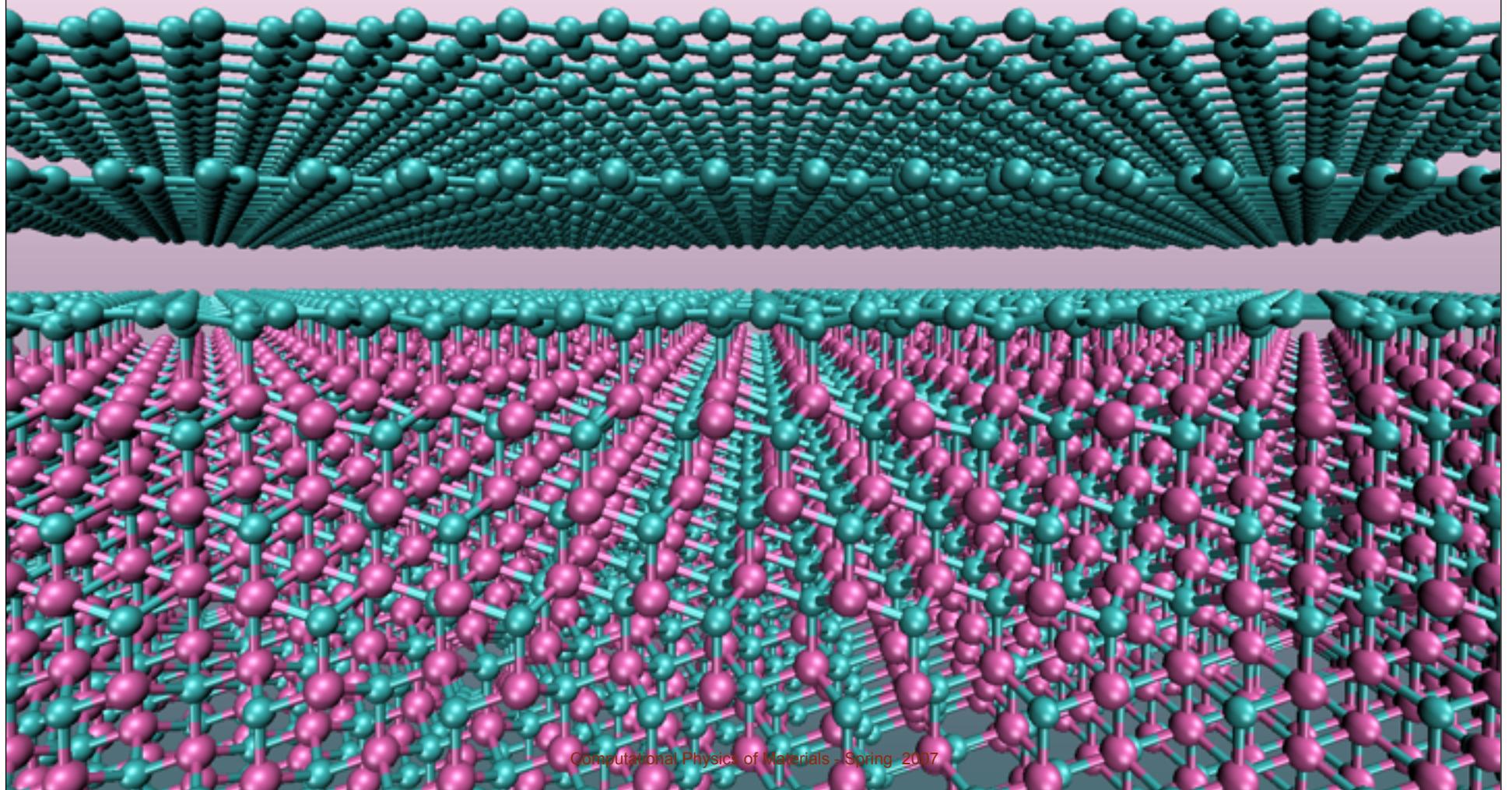


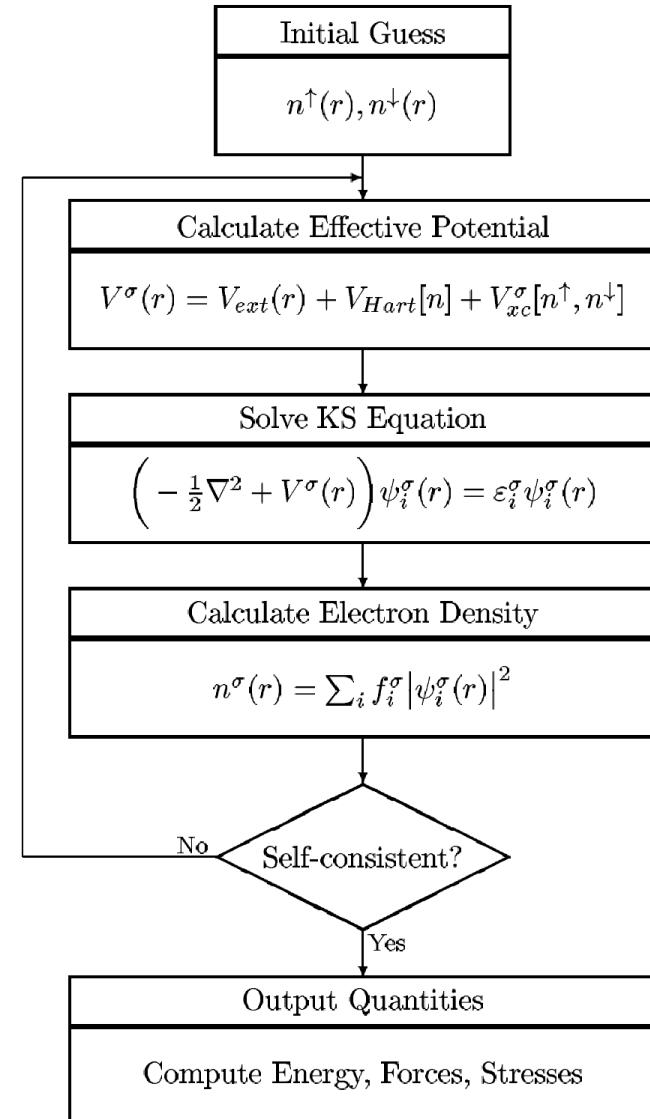
Computational Physics of Materials

How to do simple calculations with Quantum ESPRESSO



Kohn and Sham equations

- The set of K-S equations with LDA for exchange and correlation give us a formidable theoretical tool to study ground state properties of electronic systems
- Set of *self-consistent* equations that have to be solved simultaneously until convergence is achieved
- Note: K-S eigenvalues and energies are interpreted as true electronic wavefunction and electronic energies (electronic states in molecules or bands in solids)
- Note: K-S theory is a ground-state theory and as such is supposed to work well for ground state properties or small perturbations upon them
- Extremely successful in predicting materials properties - golden standard in research and industry



Numerical solution: plane waves

$$\hat{H}_{\text{eff}} \psi_i(\mathbf{r}) = \left[-\frac{\hbar^2}{2m_e} \nabla^2 + V_{\text{eff}}(\mathbf{r}) \right] \psi_i(\mathbf{r}) = \varepsilon_i \psi_i(\mathbf{r})$$

- Kohn-Sham equations are differential equations that have to be solved numerically
- To be tractable in a computer, the problem needs to be discretized via the introduction of a suitable representation of all the quantities involved
- Various discretization approaches. Most common are Plane Waves (PW) and real space grids.
- In periodic solids, plane waves of the form $e^{i\mathbf{q}\cdot\mathbf{r}}$ are most appropriate since they reflect the periodicity of the crystal and periodic functions can be expanded in the complete set of Fourier components through orthonormal PWs

$$\psi(\mathbf{r}) = \sum_{\mathbf{q}} c_{i,\mathbf{q}} \times \frac{1}{\Omega} e^{i\mathbf{q}\cdot\mathbf{r}} = \sum_{\mathbf{q}} c_{i,\mathbf{q}} \times |\mathbf{q}\rangle$$

- In Fourier space, the K-S equations become

$$\sum_{\mathbf{q}} \langle \mathbf{q}' | \hat{H}_{\text{eff}} | \mathbf{q} \rangle c_{i,\mathbf{q}} = \varepsilon_i \sum_{\mathbf{q}} \langle \mathbf{q}' | \mathbf{q} \rangle c_{i,\mathbf{q}} = \varepsilon_i c_{i,\mathbf{q}'}$$

- We need to compute the matrix elements of the effective Hamiltonian between plane waves

Numerical solution: plane waves

- Kinetic energy becomes simply a sum over \mathbf{q}

$$\sum_{\mathbf{q}'} \langle \mathbf{q}' | -\frac{1}{2} \nabla^2 | \mathbf{q} \rangle = \frac{1}{2} |q|^2 \delta_{\mathbf{q}\mathbf{q}'}$$

- The effective potential is periodic and can be expressed as a sum of Fourier components in terms of reciprocal lattice vectors

$$V_{\text{eff}}(\mathbf{r}) = \sum_m V_{\text{eff}}(\mathbf{G}_m) \exp(i\mathbf{G}_m \cdot \mathbf{r}) \text{ where } V_{\text{eff}}(\mathbf{G}_m) = \frac{1}{\Omega_{\text{cell}}} \int_{\Omega_{\text{cell}}} d\mathbf{r} V_{\text{eff}}(\mathbf{r}) \exp(-i\mathbf{G}_m \cdot \mathbf{r})$$

- Thus, the matrix elements of the potential are non-zero only if \mathbf{q} and \mathbf{q}' differ by a reciprocal lattice vector, or alternatively, $\mathbf{q} = \mathbf{k} + \mathbf{G}_m$ and $\mathbf{q}' = \mathbf{k} + \mathbf{G}_{m'}$
- The Kohn-Sham equations can be then written as matrix equations

$$\sum_{m'} H_{m,m'}(\mathbf{k}) c_{i,m'}(\mathbf{k}) = \varepsilon_i(\mathbf{k}) c_{i,m}(\mathbf{k})$$

- where:

$$H_{m,m'}(\mathbf{k}) = |\mathbf{k} + \mathbf{G}_m|^2 \delta_{m,m'} + V_{\text{eff}}(\mathbf{G}_m - \mathbf{G}_{m'})$$

- We have effectively transformed a differential problem into one that we can solve using linear algebra algorithms!

Numerical solution: plane waves

- In this representation both the potentials and the Bloch functions, solution of the K-S problem, are expanded on a set of plane waves

$$\psi_{i,\mathbf{k}}(\mathbf{r}) = e^{i\mathbf{k}\cdot\mathbf{r}} \sum_m c_{i,m}(\mathbf{k}) \exp(i\mathbf{G}_m \cdot \mathbf{r})$$

- In principle, the plane waves basis set is infinite, since I have an infinite number of reciprocal lattice vectors (or, in other words, Fourier components).
- In practice, we need to limit ourselves to a finite basis set for the practical solution of the linear equations: approximation!
- Remember: in Fourier space, that is in reciprocal space in a crystal, small \mathbf{q} components describe long-range features (wave-length), while large \mathbf{q} components, describe short-range features. Very sharp oscillations, for instance, need to be described by a large number of plane waves with large \mathbf{G} vectors
- Increasing the dimension of the basis (number of plane waves, so larger magnitudes of \mathbf{G}_m) allows for a better description of short-range features in either the potential or the density.

Numerical solution: plane waves

- Kohn-Sham equations are always self-consistent equations: the effective K-S potential depends on the electron density that is the solution of the K-S equations
- In reciprocal space the procedure becomes:

$$\sum_{m'} H_{m,m'}(\mathbf{k}) c_{i,m'}(\mathbf{k}) = \varepsilon_i(\mathbf{k}) c_{i,m}(\mathbf{k})$$

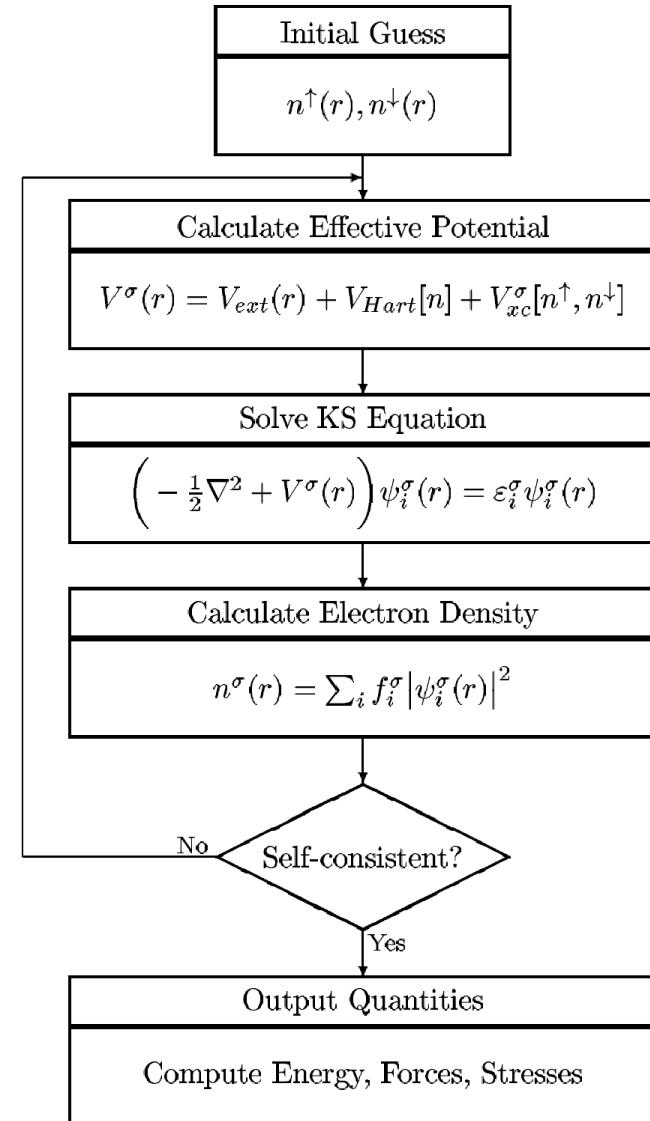
where

$$H_{m,m'}(\mathbf{k}) = |\mathbf{k} + \mathbf{G}_m|^2 \delta_{m,m'} + V_{\text{eff}}[n_{\mathbf{k},i}(\mathbf{G}_m - \mathbf{G}_{m'})]$$

and

$$n_{\mathbf{k},i}(\mathbf{G}) = \sum_m c_i^*(\mathbf{k}) c_i(\mathbf{k})$$

- Iterative solution of self-consistent equations



Numerical solution: plane waves

- The starting point for derivation of the full Kohn-Sham theory is the total energy that for a plane wave basis is written as

$$\begin{aligned} E_{tot}[V_{eff}] = & \frac{1}{N_k} \sum_{\mathbf{k}, i} w_{k,i} \left[\sum_{m,m'} c_{i,m}^*(\mathbf{k}) \left(\frac{\hbar^2}{2m_e} |\mathbf{K}_m|^2 \delta_{m,m'} + V_{ext}(\mathbf{K}_m, \mathbf{K}_{m'}) \right) c_{i,m'}(\mathbf{k}) \right] \\ & + \sum_{\mathbf{G}} \epsilon_{xc}(\mathbf{G}) n(\mathbf{G}) + \frac{1}{2} 4\pi e^2 \sum_{\mathbf{G} \neq 0} \frac{n(\mathbf{G})^2}{G^2} + \gamma_{Ewald} + \left(\sum_{\kappa} \alpha_{\kappa} \right) \frac{N_e}{\Omega}. \end{aligned}$$

- Correct treatment of the Coulomb terms is accomplished by separating out the $\mathbf{G} = 0$ components consistently in the potential and the total energy. The second term on the second line is the Coulomb interaction of the electrons with themselves excluding the divergent term due to the average electron density. Similarly, the $\mathbf{G} = 0$ Fourier component of the local potential of the ions (the local part of the pseudopotential) is defined to be zero
- Both these terms are included in the Ewald term γ_{Ewald} , which is the energy of point ions in a compensating background - this term includes the ion-ion terms as well as the interactions of the average electron density with the ions and with itself.
- The final term is a contribution due to the non-Coulombic part of the local pseudopotential where N_e/Ω is the average electron density - ions are not point charges.

Numerical solution: plane waves

- One of the most important operations is the calculation of the density n .
- The general form can be written as an average over the \mathbf{k} points:

$$n(\mathbf{r}) = \frac{1}{N_k} \sum_{\mathbf{k}, i} f(\varepsilon_{i,\mathbf{k}}) n_{i,\mathbf{k}}(\mathbf{r}), \text{ with } n_{i,\mathbf{k}}(\mathbf{r}) = |\psi_{i,\mathbf{k}}(\mathbf{r})|^2$$

- In plane waves, from the expression of the Bloch functions we get:

$$n_{i,\mathbf{k}}(\mathbf{r}) = \frac{1}{\Omega} \sum_{m,m'} c_{i,m}^*(\mathbf{k}) c_{i,m'}(\mathbf{k}) \exp(i(\mathbf{G}_{m'} - \mathbf{G}_m) \cdot \mathbf{r})$$

where

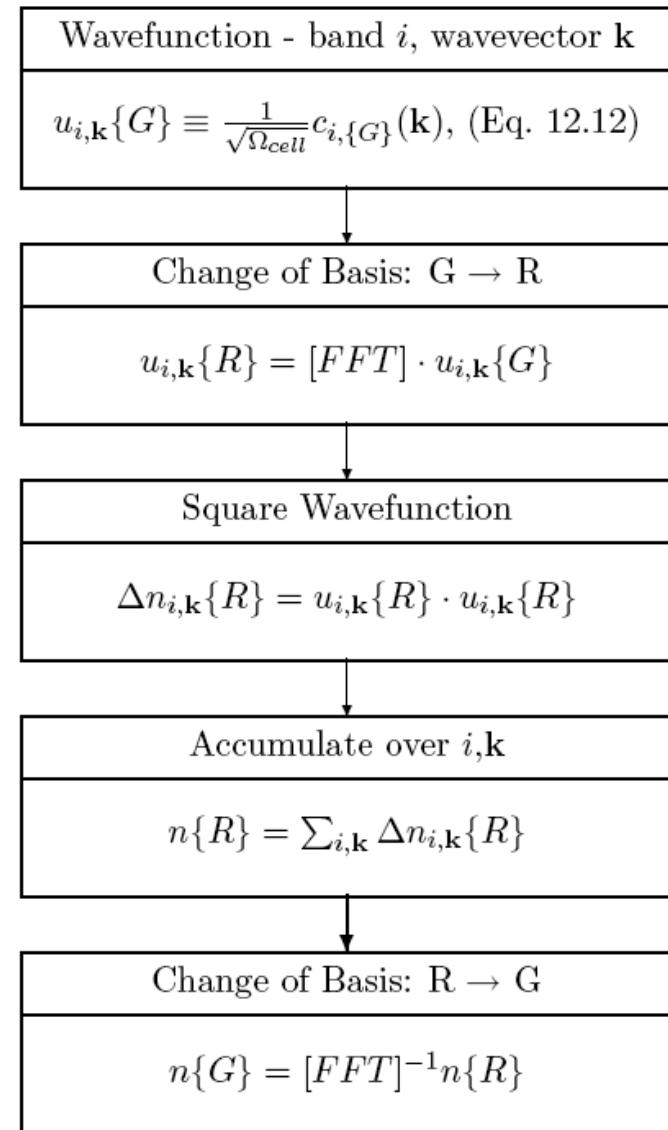
$$n_{i,\mathbf{k}}(\mathbf{G}) = \frac{1}{\Omega} \sum_m c_{i,m}^*(\mathbf{k}) c_{i,m}(\mathbf{k})$$

with $\mathbf{G}_{m''} = \mathbf{G}_m + \mathbf{G}$

- Further reduction of the computation can be achieved using the symmetry properties of the system and applying the symmetry group operations to a set of \mathbf{k} points in the irreducible Brillouin zone.
- However, such direct sum is not the most efficient way to calculate the density, since obtaining the Fourier components of $n_{i,\mathbf{k}}(\mathbf{G})$ involves a double sum that scales as N_G^2 . Very expensive for large systems.
- On the other hand, if the Bloch states are known on a finite grid in real space, the density is a sum of square wavefunctions - scales as N_R

Numerical solution: plane waves

- The trick is to use a Fast Fourier Transform (FFT) that allows one to transform from one space to the other in $N \log N$ operations, where $N = N_R = N_G$
- Moreover, $n(\mathbf{r})$ is needed to compute the exchange and correlation energy and potential!
- The inverse transform can be used to find $n(\mathbf{G})$ which can be used for solving the Poisson equation in Fourier space.



Numerical solution: plane waves

- Forces on atoms can be found by deriving the expression for the total energy with respect to atomic displacements (Hellmann-Feynman theorem):

$$\begin{aligned}\mathbf{F}_j^\kappa = -\frac{\partial E}{\partial \tau_{\kappa,j}} &= -\frac{\partial \gamma_{Ewald}}{\partial \tau_{\kappa,j}} - i \sum_m \mathbf{G}_m e^{i \mathbf{G}_m \cdot \tau_{\kappa,j}} V_{local}^\kappa(\mathbf{G}_m) n(\mathbf{G}_m) \\ &\quad \frac{-i}{N_k} \sum_{\mathbf{k}, i} w_{k,i} \sum_{m,m'} c_{i,m}^*(\mathbf{k}) \left[\mathbf{K}_{m,m'} e^{i (\mathbf{K}_{m,m'} \cdot \tau_{\kappa,j})} \delta V_{NL}^\kappa(\mathbf{K}_m, \mathbf{K}_{m'}) \right] c_{i,m}(\mathbf{k})\end{aligned}$$

where the pseudopotential terms have been split into a long range local part and a short range non-local part.

Quantum-ESPRESSO the home of innovation in electronic structure calculations



SEARCH

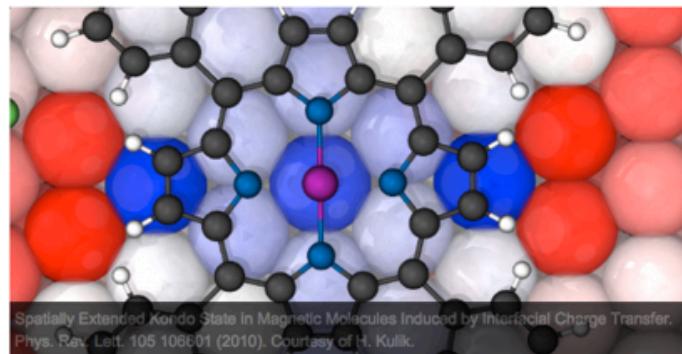
Forum

NEWS

28.08.13

ADVANCED QUANTUM ESPRESSO
DEVELOPER TRAINING, 9-20
DECEMBER 2013, ITALY

Building on the previous QE Developer training, we are now organizing an advanced developer training, which includes three months of dedicated distance learning sessions and ten days of hands-on coding with QE developers. As a result of this training marathon, we aim to implement two new features into QE distribution and publish one article detailing these advancements. Deadline for application: 9-9-2013. For more information and registration visit [here](#).



QUANTUM ESPRESSO

is an integrated suite of Open-Source computer codes for electronic-structure calculations and materials modeling at the nanoscale. It is based on density-functional theory, plane waves, and pseudopotentials.

[READ MORE >](#)

- Quantum-ESPRESSO is a state-of-the-art software package that we will use as an introduction to the actual procedure of modeling the electronic structure of a solid: <http://www.quantum-espresso.org>

Why “Quantum ESPRESSO”?!



Trieste, Tommaseo (ph. G. Crozzoli)



Shobhana Narasimhan, JNCASR

Quantum ESPRESSO: Organization



The distribution is maintained as a single CVS (Concurrent Version System) tree.
Available to everyone anytime via anonymous (read-only) access.

- *Web site:* <http://www.quantum-espresso.org>
- *Wiki:* http://www.quantum-espresso.org/index.php/Main_Page
contains the updated documentation
- *Developers' portal:* <http://www.qe-forge.org>
integrated developer environment, open to external contributions

Mailing lists:

- `pw_users`: used by developers for announcements about Quantum ESPRESSO
- `pw_forum`: for general discussions (all subscribed users can post)

P. Giannozzi



II. Doing a “Total Energy” Calculation with the PWscf Package of QE: The SCF Loop

The Kohn-Sham problem

- Want to solve the Kohn-Sham equations:

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

$\underbrace{\hspace{10em}}_H$

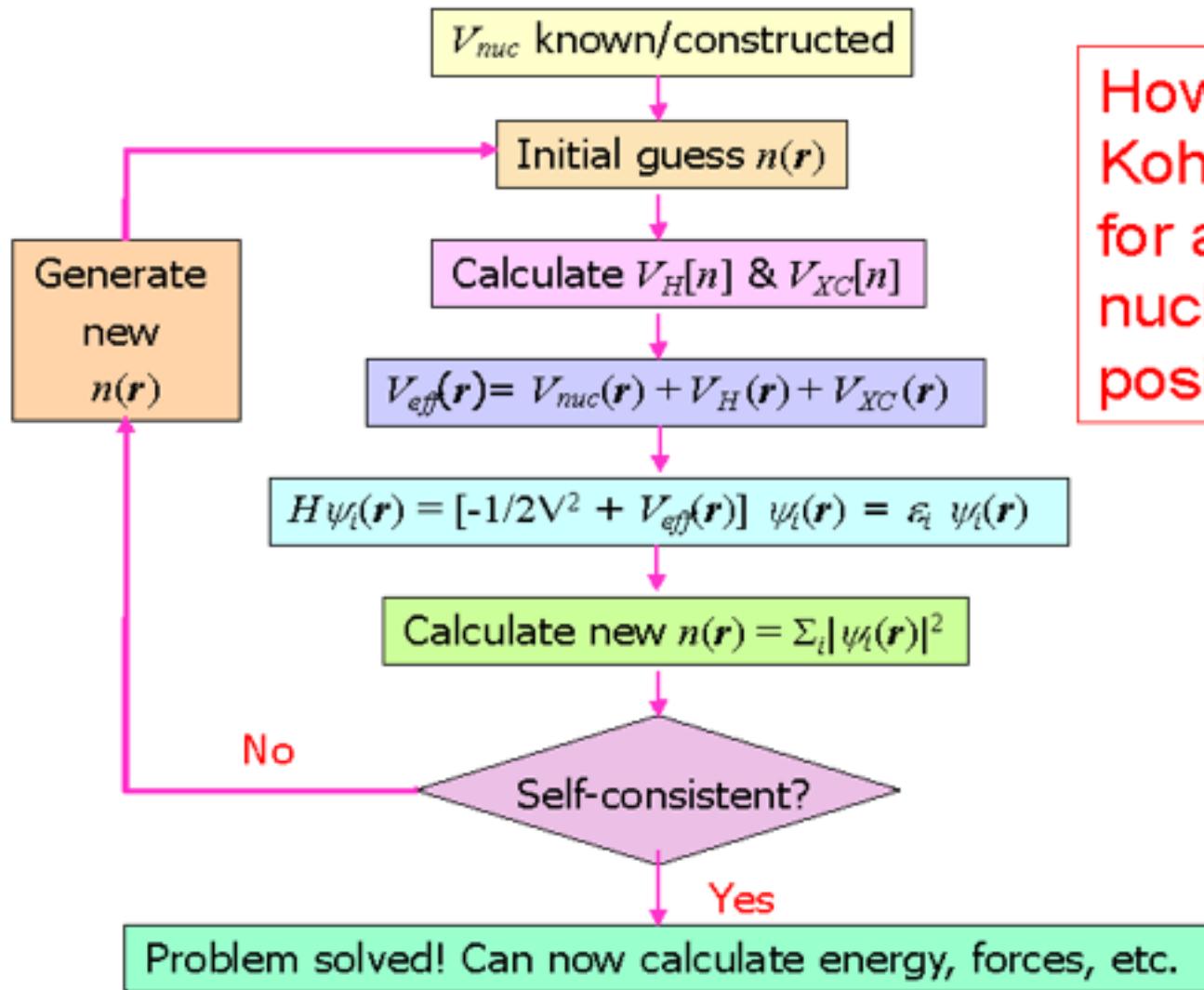
- Note that **self-consistent solution** necessary, as H depends on solution:

$$\{\psi_i\} \rightarrow n(r) \rightarrow H$$

- Convention:

$$e = \hbar = m_e = 1$$

Self-consistent Iterative Solution



How to solve the Kohn-Sham eqns. for a set of fixed nuclear (ionic) positions.

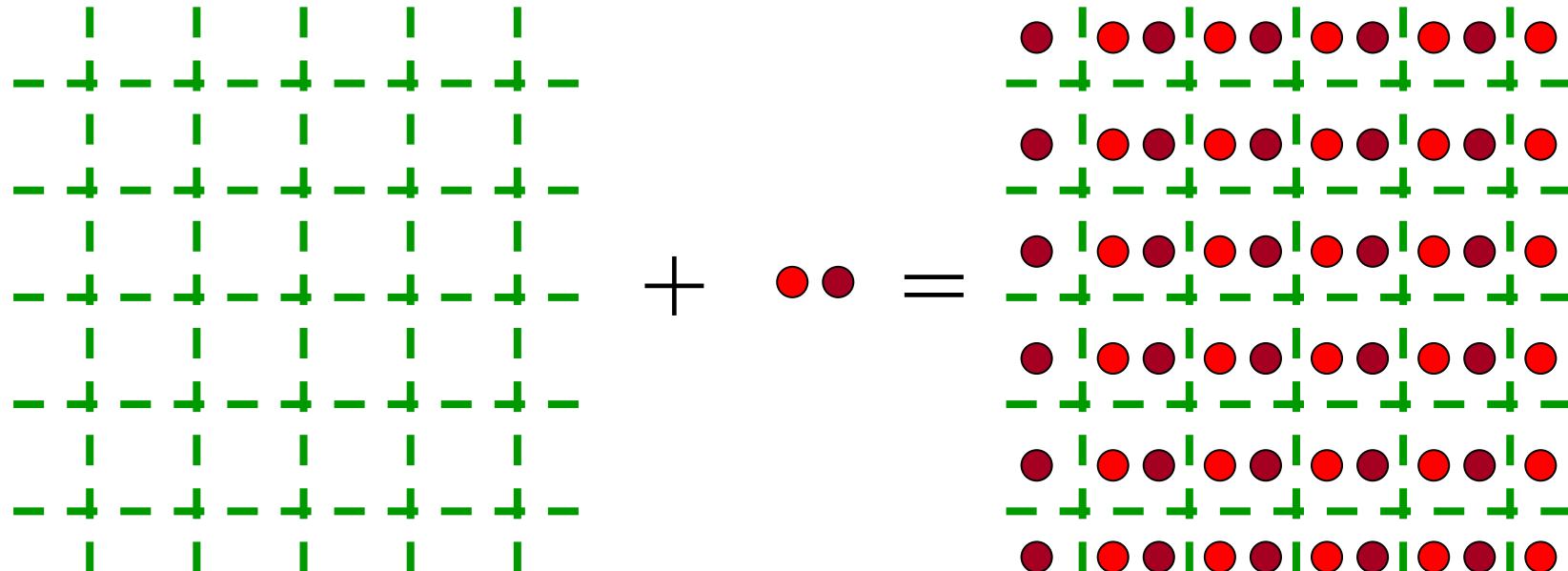
Step 0: Defining the (periodic) system



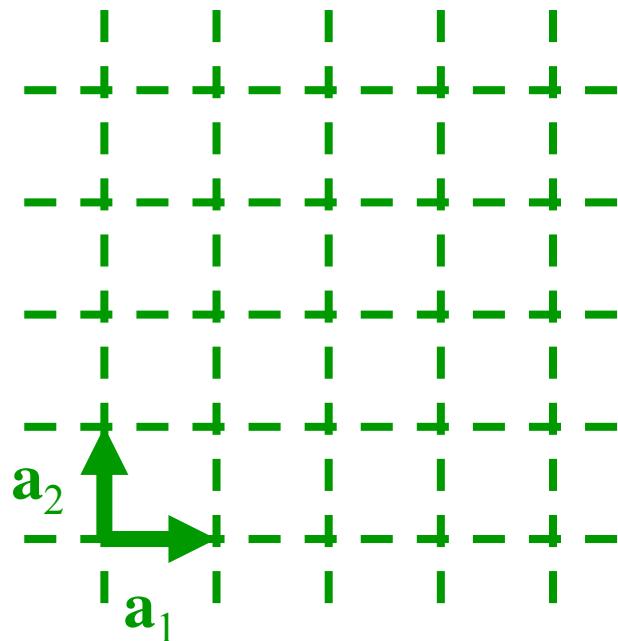
Namelist ‘**SYSTEM**’

How to Specify the System

- All periodic systems can be specified by a **Bravais Lattice** and an **atomic basis**.



How to Specify the Bravais Lattice / Unit Cell



Input parameter **ibrav**

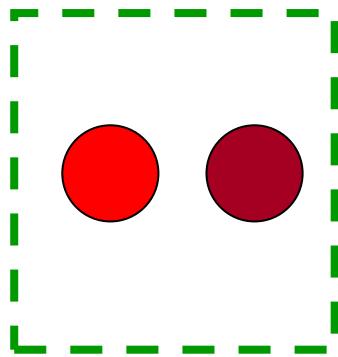
- Gives the type of Bravais lattice (SC, BCC, Hex, etc.)

Input parameters {**celldm(i)**}

- Give the lengths [& directions, if necessary] of the lattice vectors \mathbf{a}_1 , \mathbf{a}_2 , \mathbf{a}_3

- Note that one can choose a non-primitive unit cell (e.g., 4 atom SC cell for FCC structure).

Atoms Within Unit Cell – How many, where?



Input parameter **nat**

- Number of atoms in the unit cell

Input parameter **ntyp**

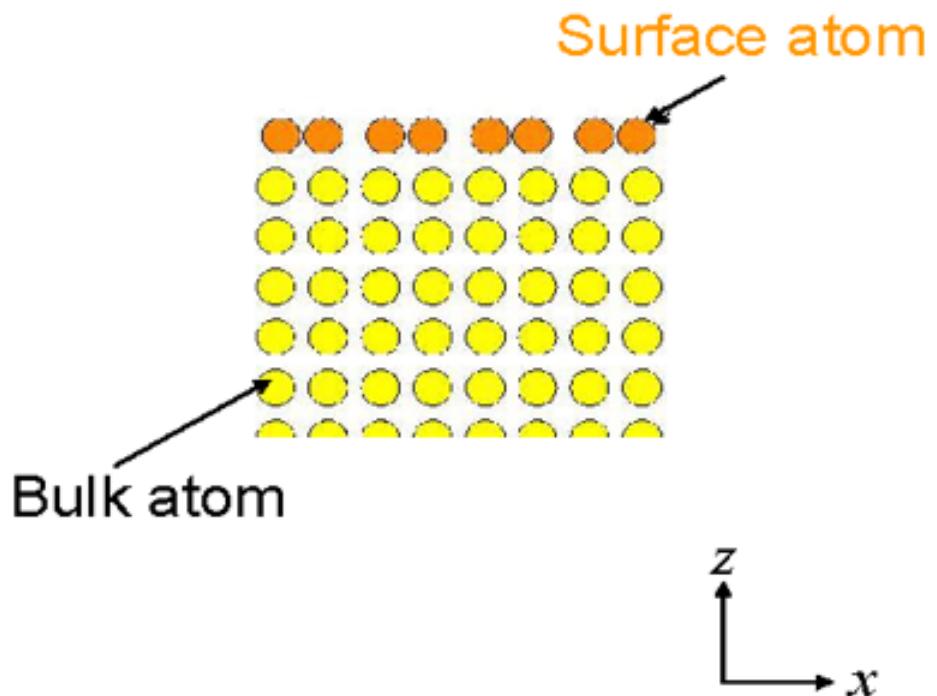
- Number of types of atoms

FIELD ATOMIC_POSITIONS

- Initial positions of atoms (may vary when “**relax**” done).
- Can choose to give in units of lattice vectors (“**crystal**”) or in Cartesian units (“**alat**” or “**bohr**” or “**angstrom**”)

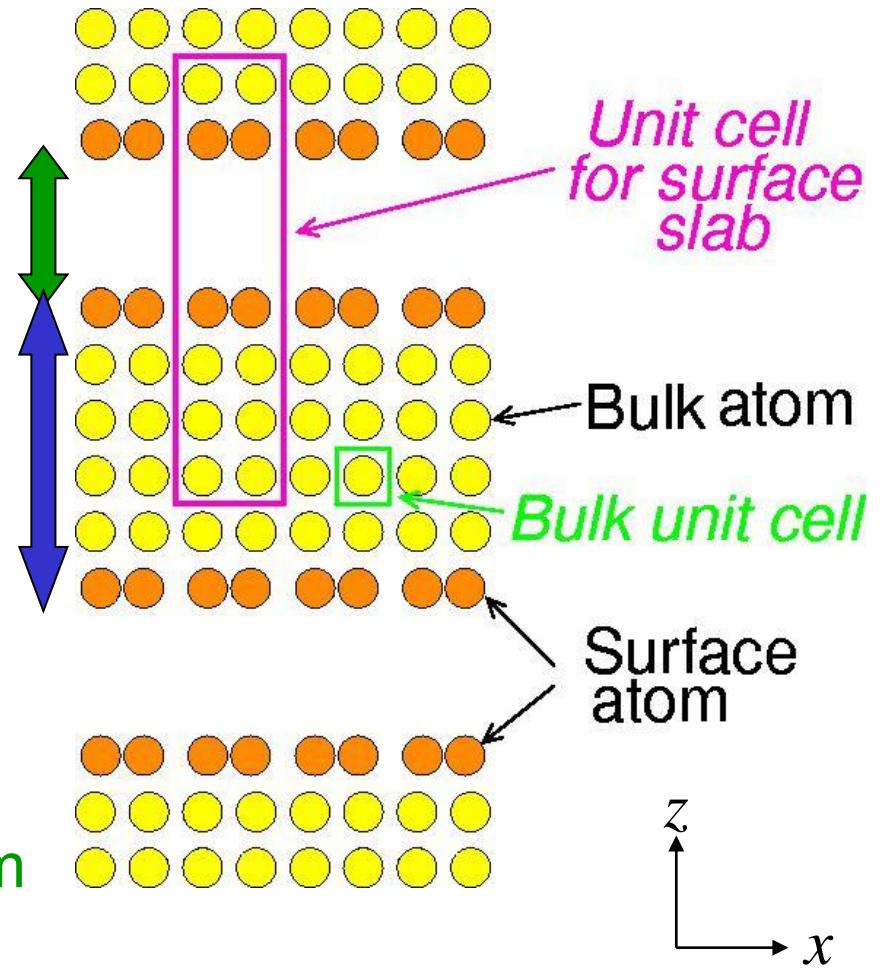
What if the system is not periodic?

- Example 1: Want to study properties of a system with a surface.
- Presence of surface \Rightarrow No periodicity along z .



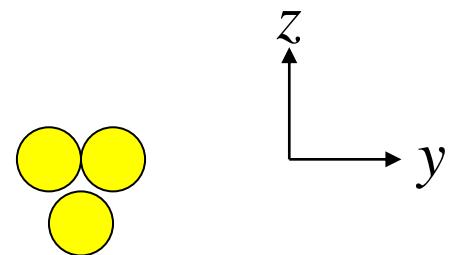
What if the system is not periodic?

- Example 1: Want to study properties of a system with a surface.
- Presence of surface \Rightarrow No periodicity along z .
- Use a **supercell**: **artificial periodicity along z** by repeating slabs separated by **vacuum**.
- Have to check convergence w.r.t. **slab thickness & vacuum thickness**.



What if the system is not periodic?

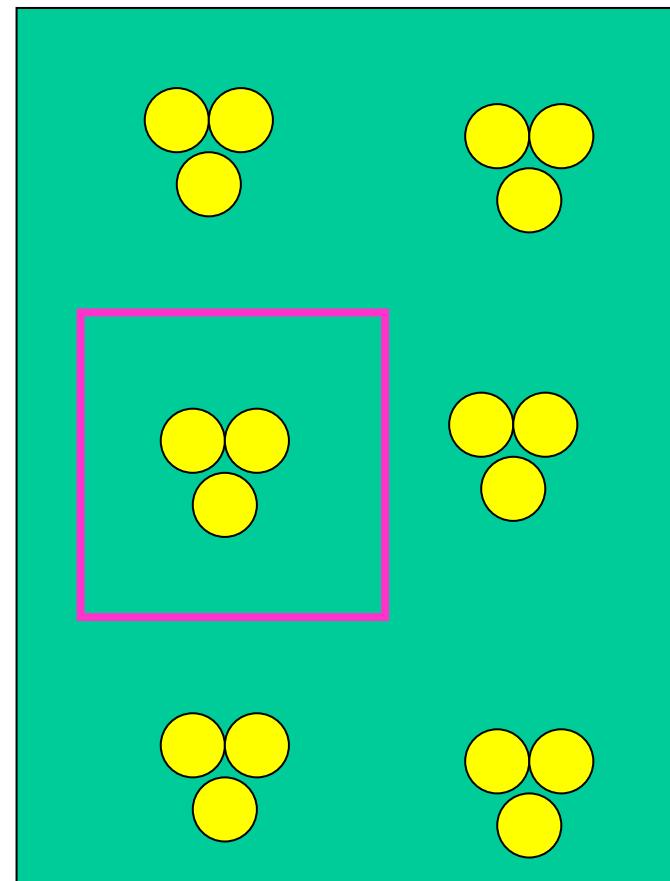
- Example 2: Want to study properties of a nanowire.



- Example 3: Want to study properties of a cluster

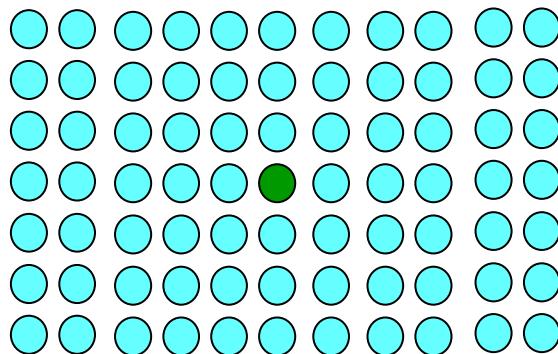
What if the system is not periodic?

- Example 2: Want to study properties of a nanowire \Rightarrow introduce **artificial periodicity along y & z .**
- Example 3: Want to study properties of a cluster \Rightarrow introduce **artificial periodicity along x, y & z .**



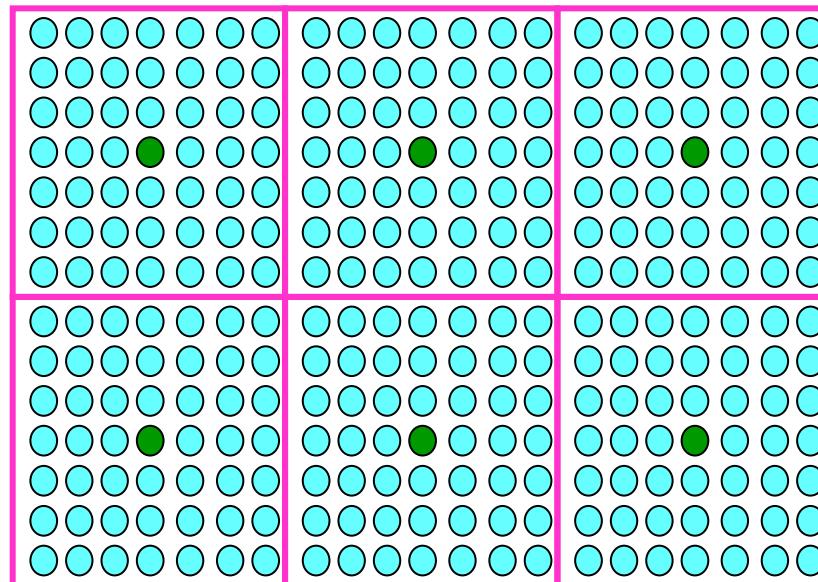
What if the system is not periodic?

- Example 4: Want to study a system with a defect, e.g., a **vacancy** or **impurity**:



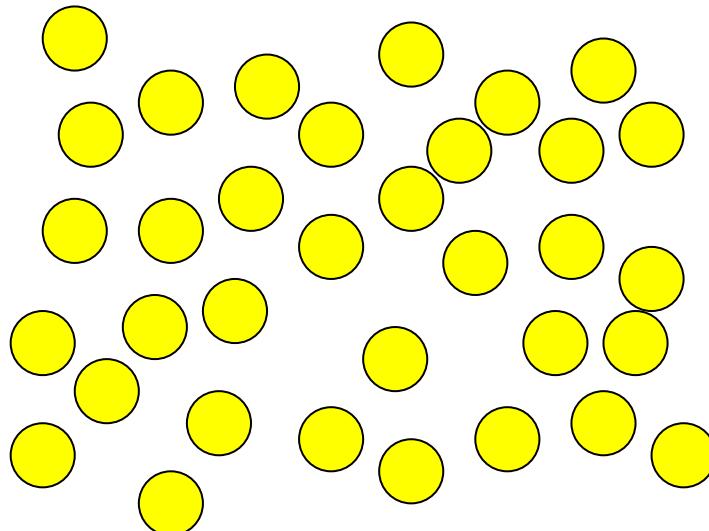
What if the system is not periodic?

- Example 4: Want to study a system with a defect, e.g., a **vacancy** or **impurity**:



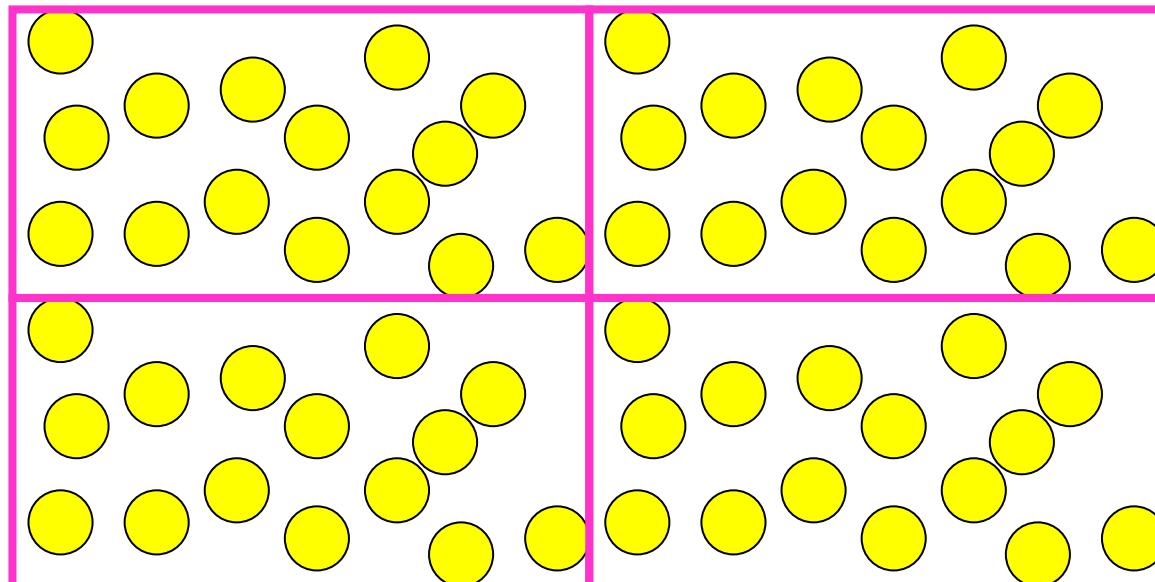
What if the system is not periodic?

- Example 5: Want to study an amorphous or quasicrystalline system.



What if the system is not periodic?

- Example 5: Want to study an amorphous or quasicrystalline system: approximate by a periodic system (with large **unit cell**).



Artificially Periodic Systems \Rightarrow Large Unit Cells



- Note: In all these cases, to minimize the effects of the artificially introduced periodicity, need a large unit cell.



- Long $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ (primitive lattice vectors)
 - Short $\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3$ (primitive reciprocal lattice vectors)
 - Many \mathbf{G} 's will fall within E_{cut} sphere!



PWscf input file

- The input data is organized as several namelists, followed by other fields introduced by keywords. The namelists are
 - &CONTROL: general variables controlling the run
 - &SYSTEM: structural information on the system under investigation
 - &ELECTRONS: electronic variables: self-consistency, smearing
 - &IONS (optional): ionic variables: relaxation, dynamics
 - &CELL (optional): variable-cell dynamics
 - &PHONON (optional): information needed to produce data for phonon calculations
- Optional namelist may be omitted if the calculation to be performed does not require them. This depends on the value of variable calculation in namelist &CONTROL. Most variables in namelists have default values. Only the following variables in &SYSTEM **MUST** be specified:
 - ibrav (integer): bravais-lattice index
 - celldm (real, dimension 6): crystallographic constants
 - nat (integer): number of atoms in the unit cell
 - ntyp (integer): number of types of atoms in the unit cell
 - ecutwfc (real): kinetic energy cutoff (Ry) for wavefunctions.

Description of all the input cards can be found in the file INPUT_PW

PWscf input file

```
&control
  calculation = 'scf'
  restart_mode='from_scratch',
  prefix='silicon',
  tstress = .true.
  tprnfor = .true.
  pseudo_dir = '/opt/marco/Si_test/'
 outdir='/opt/marco/Si_test/'
/
&system
  ibrav= 2, celldm(1) =10.20, nat= 2, ntyp= 1,
  ecutwfc =30.0,
/
&electrons
  mixing_mode = 'plain'
  mixing_beta = 0.7
  conv_thr = 1.0d-8
/
ATOMIC_SPECIES
Si 28.086 si.pbe-rrkj.UPF
ATOMIC_POSITIONS
Si 0.00 0.00 0.00
Si 0.25 0.25 0.25
K_POINTS
-10
  0.1250000  0.1250000  0.1250000  1.00
  0.1250000  0.1250000  0.3750000  3.00
  0.1250000  0.1250000  0.6250000  3.00
  0.1250000  0.1250000  0.8750000  3.00
  0.1250000  0.3750000  0.3750000  3.00
  0.1250000  0.3750000  0.6250000  6.00
  0.1250000  0.3750000  0.8750000  6.00
  0.1250000  0.6250000  0.6250000  3.00
  0.3750000  0.3750000  0.3750000  1.00
  0.3750000  0.3750000  0.6250000  3.00
```

- Input file for an electronic structure calculation for a bulk Si crystal
- **&control:**
 - **scf** = self-consistent solution of the Kohn-Sham equations (ground state energy, electron densities) from an arbitrary initial density ('**from_scratch**')
 - **prefix** = prefix for file names
 - **tstress, tprnfor** = compute also forces and stresses in the given geometry
 - Specify working directories (optional)

Input parameters: &system

- The `&system` namelist is the one where we specify the geometrical parameters of our simulation cell:
 - `ibrav` = specifies the Bravais lattice type
 - `celldm` = specifies the crystallographic constants (the dimension of the simulation cell)

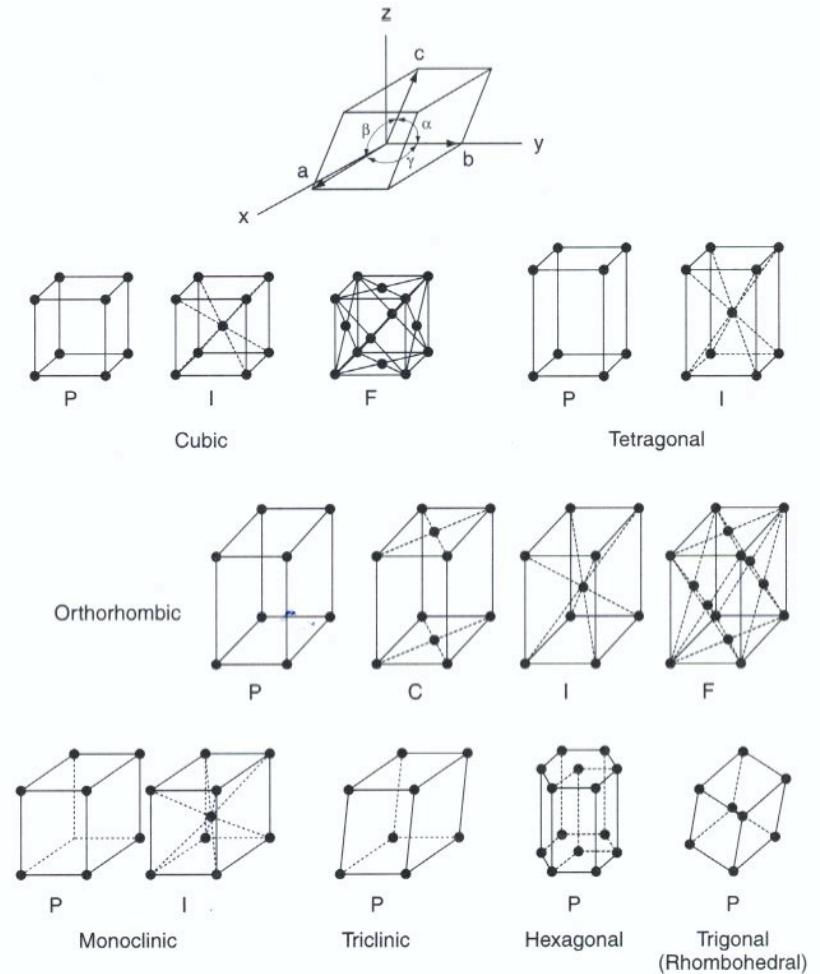
```

ibrav is the structure index:

  ibrav      structure          celldm(2)-celldm(6)

    0      "free", see above      not used
    1      cubic P (sc)          not used
    2      cubic F (fcc)          not used
    3      cubic I (bcc)          not used
    4      Hexagonal and Trigonal P
    5      Trigonal R            celldm(3)=c/a
    6      Tetragonal P (st)      celldm(4)=cos(aalpha)
    7      Tetragonal I (bct)      celldm(3)=c/a
    8      Orthorhombic P         celldm(3)=c/a
    9      Orthorhombic base-centered(bco)
   10      Orthorhombic face-centered
   11      Orthorhombic body-centered
   12      Monoclinic P           celldm(2)=b/a,celldm(3)=c/a,
                                celldm(4)=cos(ab)
   13      Monoclinic base-centered
   14      Triclinic P            celldm(2)= b/a,
                                celldm(3)= c/a,
                                celldm(4)= cos(bc),
                                celldm(5)= cos(ac),
                                celldm(6)= cos(ab)

```



Input parameters: &system

- nat and ntyp specify the number of atoms in the simulation cell and how many different atomic species are in the system we want to study
- Together with the information contained under the keywords ATOMIC SPECIES and ATOMIC POSITIONS they determine the basis in the primitive cell

```
ATOMIC_SPECIES
Si 28.086 Si.pbe-rrkj.UPF
ATOMIC_POSITIONS
Si 0.00 0.00 0.00
Si 0.25 0.25 0.25
```

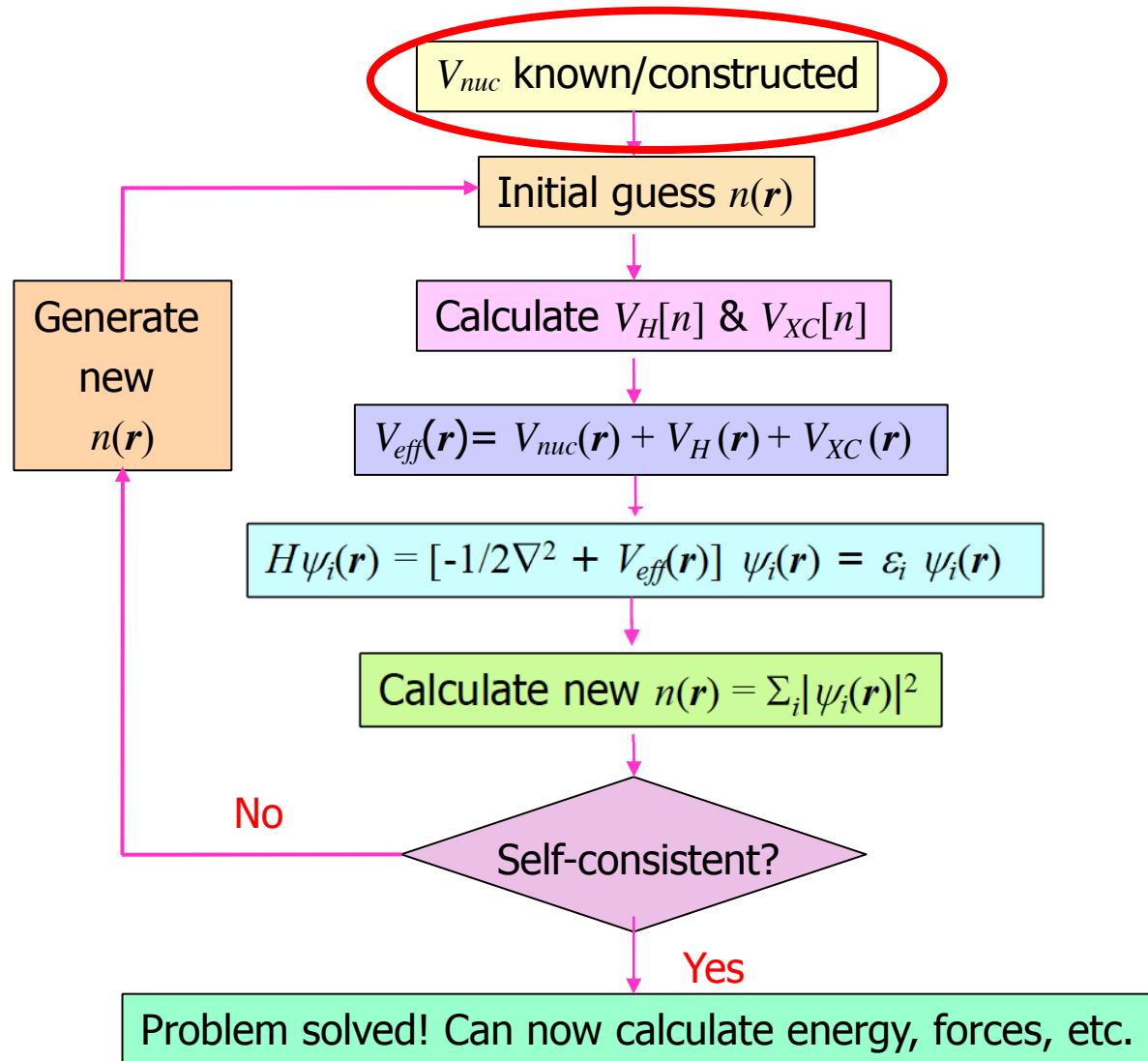
- ATOMIC SPECIES: list of all the atomic species in the system with information on the atomic mass and a link to an external file that contains the information necessary to describe that particular atom - pseudopotential file

```
ATOMIC_SPECIES
"ntyp" cards like the following
X Mass_X PseudoPot_X
X          Character: label of the atom
Mass_X    Real      : mass of the atomic species
                  not used if calculation='scf', 'nscf', 'phonon'
PseudoPot_X Character: file containing PP for this species
```

- ATOMIC POSITIONS: list of all the different atoms in the simulation cell with the specification of their atomic coordinates

```
ATOMIC_POSITIONS { alat | bohr | crystal | angstrom }
alat     : atomic positions are in units of alat (default)
bohr    : atomic positions are in a.u.
crystal : atomic positions are in crystal coordinates (see below)
angstrom: atomic positions are in A
```

Step 1: Obtaining V_{nuc}



Nuclear Potential



- Electrons experience a **Coulomb potential** due to the nuclei.
- This has a known and simple form:

$$V_{nuc} = -\frac{Z}{r}$$

- But this leads to computational problems!

Problem for Plane-Wave Basis

Core wavefunctions:
sharply peaked near
nucleus.

Valence wavefunctions:
lots of wiggles near
nucleus.

High Fourier components present

i.e., need large E_{cut} 

Solutions for Plane-Wave Basis

Core wavefunctions:
sharply peaked near
nucleus.

Valence wavefunctions:
lots of wiggles near
nucleus.

High Fourier components present

i.e., need large E_{cut}



Don't solve for the
core electrons!

Remove wiggles from
valence electrons.

Pseudopotentials



- Replace nuclear potential by pseudopotential
- This is a numerical trick that solves these problems
- There are different kinds of pseudopotentials
(Norm conserving pseudopotentials, ultrasoft pseudopotentials, etc.)
- Which kind you use depends on the element.

Pseudopotentials



QUANTUM ESPRESSO

HOME PROJECT DOWNLOAD RESOURCES PSEUDOPOTENTIALS CONTACTS NEWS & EVENTS

SEARCH

Search here...



Forum

PSEUDOPOTENTIALS

Admin PP Database

More about pseudopotentials

Naming convention for the pseudopotential

Unified Pseudopotential Format

PSEUDOPOTENTIALS

Ready-to-use pseudopotentials are available from the periodic table below. Choose the options you desire from the menus (pseudopotentials from PSlibrary are recommended), then press "Filter". Elements for which at least a pseudopotential is available will appear in red. Click on the element entry and follow the link to access the pseudopotentials and a minimal description of their characteristics.

More information about [pseudopotentials in general](#), the [naming convention](#) adopted for pseudopotential files, the [Unified Pseudopotential Format](#), and on other pseudopotential databases, can be found via the links of the menu at the left.

Important Note: although most of these pseudopotentials were published or used with satisfactory results in published work, we cannot give any warranty whatsoever that they fit your actual needs.

ANY FUNCTIONAL	ANY TYPE	Apply Filter
ANY PP LIBRARY	OTHER OPTIONS	
1 H		2 He
3 Li		5 B
4 Be		6 C
11 Na		7 N
12 Mg		8 O
19 K		9 F
20 Ca		10 Ne
37 Rb	21 Sc	13 Al
38 Sr	22 Tl	14 Si
	23 V	15 P
	24 Cr	16 S
	25 Mn	17 Cl
	26 Fe	18 Ar
	27 Co	
	28 Ni	
	29 Cu	
	30 Zn	
	31 Ga	
	32 Ge	
	33 As	
	34 Se	
	35 Br	
	36 Kr	
55 Cs	39 Y	5 B
56 Ba	40 Zr	6 C
57-70 -	41 Nb	7 N
	42 Mo	8 O
	43 Tc	9 F
	44 Ru	10 Ne
	45 Rh	
	46 Pd	
	47 Ag	
	48 Cd	
	49 In	
	50 Sn	
	51 Sb	
	52 Te	
	53 I	
	54 Xe	
71 Lu	72 Hf	73 Ta
74 W	75 Re	76 Os
77 Ir	78 Pt	79 Au
80 Hg	81 Tl	82 Pb
81 Bi	82 Po	83 At
84 Rn	85 Md	86 Rn
103 Fr	104 Ra	105 **
106 Lr	107 Rf	108 Db
109 Bh	109 Sg	109 Bh
		109 Mt
* Lanthanoids		
La	57 Ce	58 Pr
Ce	59 Nd	60 Pm
Pr	61 Sm	62 Eu
Nd	63 Gd	64 Tb
Pm	65 Dy	66 Ho
Sm	67 Er	68 Tm
Eu	69 Yb	
** Actinoids		
Ac	89 Th	90 Pa
Th	91 U	92 Np
Pa	93 Pu	94 Am
U	95 Cm	96 Bk
Np	97 Cf	98 Es
Pu	99 Fm	100 Md
Am	101 No	102 No

Upload a file
Download all Files

Search engine developed by [Erica Vidal](#)

Pseudopotentials for Quantum-ESPRESSO

Name: Oxygen
Symbol: O
Atomic number: 8
Atomic configuration: [He] 2s2 2p4
Atomic mass: 15.9994 (3)

Available pseudopotentials:

[O.blyp-mt.UPF](#) ([details](#))

Becke-Lee-Yang-Parr (BLYP) exch-corr
Martins-Troullier

[O.pbe-rrkjus.UPF](#) ([details](#))

Perdew-Burke-Ernzerhof (PBE) exch-corr
Rabe Rappe Kaxiras Joannopoulos (ultrasoft)

[O.pbe-van_bm.UPF](#) ([details](#))

Perdew-Burke-Ernzerhof (PBE) exch-corr
Vanderbilt ultrasoft
author: bm

[O.pz-mt.UPF](#) ([details](#))

Perdew-Zunger (LDA) exch-corr
Martins-Troullier

[O.pz-rrkjus.UPF](#) ([details](#))

Perdew-Zunger (LDA) exch-corr
Rabe Rappe Kaxiras Joannopoulos (ultrasoft)

[O.blyp-van_ak.UPF](#) ([details](#))

Becke-Lee-Yang-Parr (BLYP) exch-corr
Vanderbilt ultrasoft
author: ak

Pseudopotential's name gives information about :

- type of exchange-correlation functional
- type of pseudopotential
- e.g.:

[O.pbe-rrkjus.UPF](#) ([details](#))

→ Perdew-Burke-Ernzerhof (PBE) exch-corr
→ Rabe Rappe Kaxiras Joannopoulos (ultrasoft)

Element & V_{ion} for Quantum-ESPRESSO

e.g, for calculation on $BaTiO_3$:

ATOMIC_SPECIES

Ba 137.327 Ba.pbe-nsp-van.UPF

Ti 47.867 Ti.pbe-sp-van_ak.UPF

O 15.999 O.pbe-van_ak.UPF

- **ecutwfc**, **ecutrho** depend on type of pseudopotentials used (should test).
- When using ultrasoft pseudopotentials, set **ecutrho** = $8-12 \times \text{ecutwfc}$!!

Element & V_{ion} for Quantum-ESPRESSO

- Should have same exchange-correlation functional for all pseudopotentials.

input

```
mixing_beta = 0.7, conv_thr = 1.0
/
ATOMIC_SPECIES
Fe 55.85 Fe.pz-nd-rrkjus.UPF
Co 58.93 Co.pbe-nd-rrkjus.UPF
ATOMIC_POSITIONS (crystal)
Fe 0.00 0.00 0.00
```

oops!

output

```
Max angular momentum in pseudopotentials
from readpp : error #          2
inconsistent DFT read
stopping ..■
```

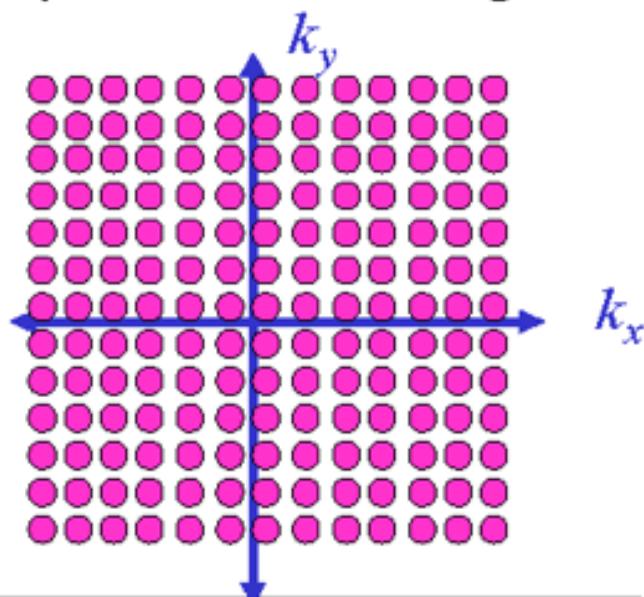
Plane Waves & Periodic Systems

- For a periodic system:

$$\psi_{\mathbf{k}}(\mathbf{r}) = \frac{1}{\Omega} \sum_{\mathbf{G}} c_{\mathbf{k},\mathbf{G}} e^{i(\mathbf{k} + \mathbf{G}) \cdot \mathbf{r}}$$

where \mathbf{G} = reciprocal lattice vector

- The plane waves that appear in this expansion can be represented as a grid in k-space:



- Only true for periodic systems that grid is discrete.
- In principle, still need infinite number of plane waves.

Input parameters: `ecutwfc`

- In this representation both the potentials and the Bloch functions, solution of the K-S problem, are expanded on a set of plane waves

$$\psi_{i,\mathbf{k}}(\mathbf{r}) = e^{i\mathbf{k} \cdot \mathbf{r}} \sum_m c_{i,m}(\mathbf{k}) \exp(i\mathbf{G}_m \cdot \mathbf{r})$$

- In principle, the plane waves basis set is infinite, since I have an infinite number of reciprocal lattice vectors (or, in other words, Fourier components).
- In practice, we need to limit ourselves to a finite basis set for the practical solution of the linear equations: approximation!
- Remember: in Fourier space, that is in reciprocal space in a crystal, small \mathbf{q} components describe long-range features (wave-length), while large \mathbf{q} components, describe short-range features. Very sharp oscillations, for instance, need to be described by a large number of plane waves with large \mathbf{G} vectors
- Increasing the dimension of the basis (number of plane waves, so larger magnitudes of \mathbf{G}_m) allows for a better description of short-range features in either the potential or the density.
- `ecutwfc` is the parameter that controls the number of PW's in the basis, so it affects directly the accuracy of the calculation: **convergence parameter**

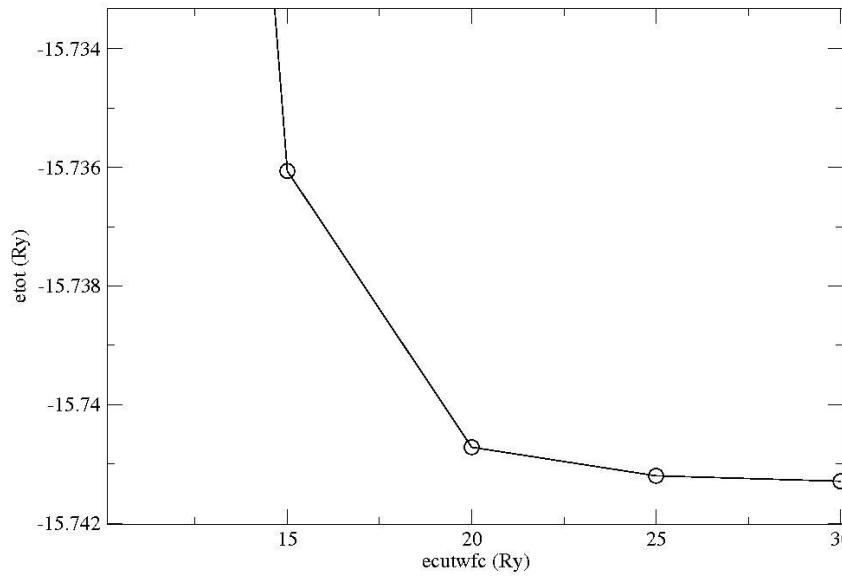
$$|\mathbf{G}_{max}|^2 \leq \text{ecutwfc}$$

Checking Convergence wrt ecutwfc

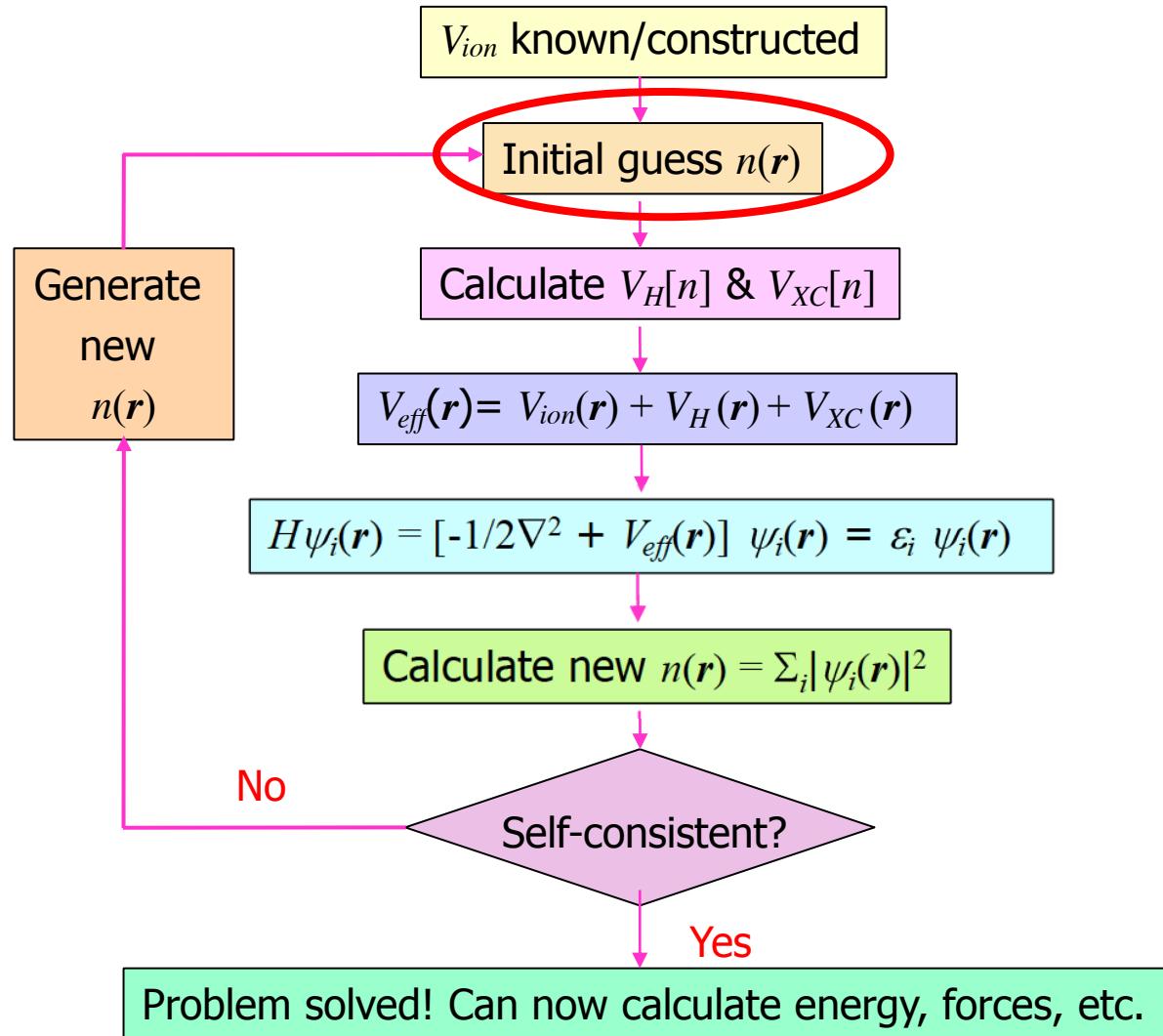


- Must always check.
- Monotonic (variational).

Silicon: Convergence wrt plane wave cutoff



Step 2: Initial Guess for $n(\mathbf{r})$



Starting Wavefunctions



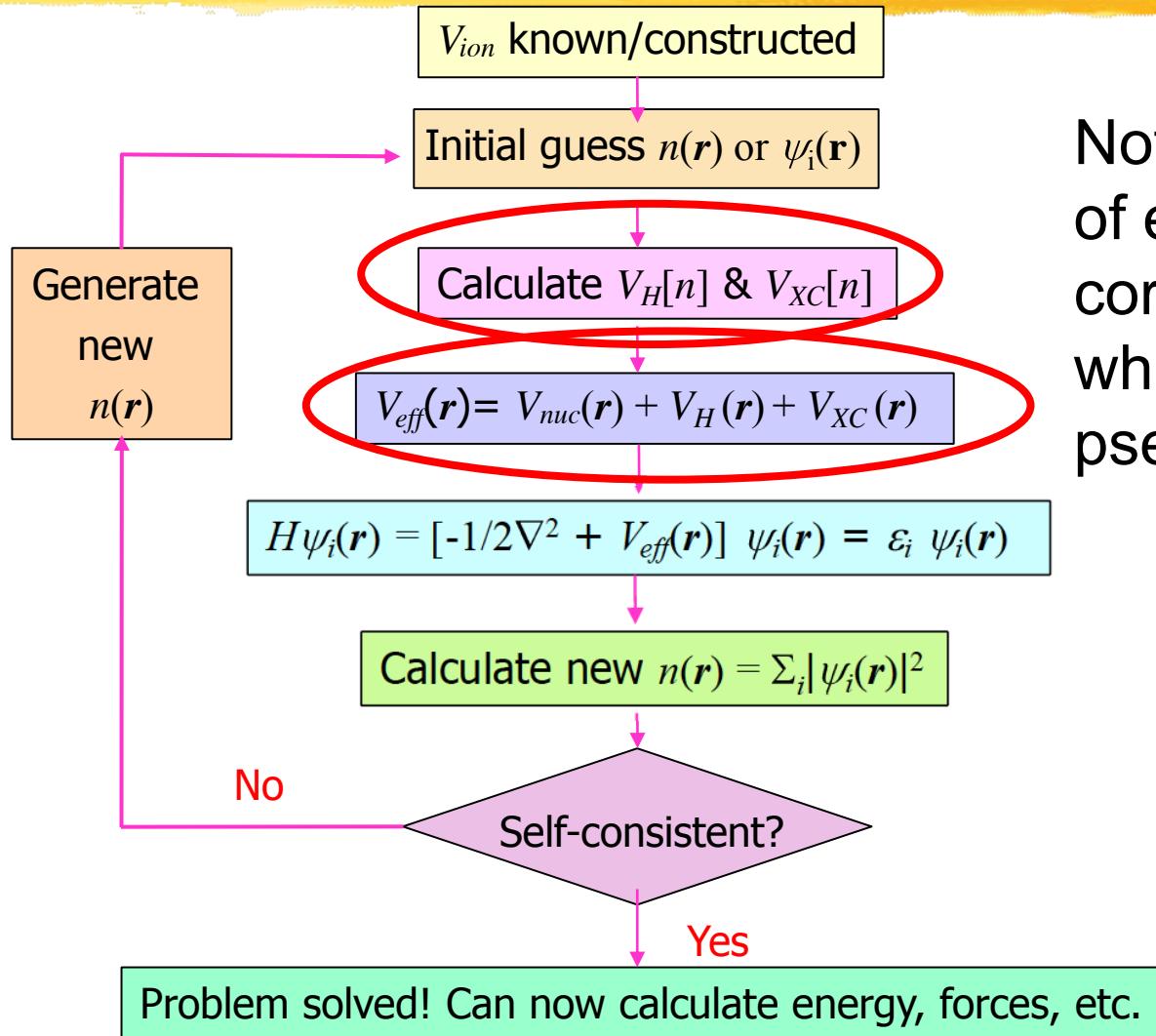
The closer your starting wavefunction is to the true wavefunction (which, of course, is something you don't necessarily know to start with!), the fewer the scf iterations needed.

```
startingwfc 'atomic'  
           'random'  
           'file'
```

Superposition of atomic orbitals

“The beginning is the most important part of the work” - Plato

Steps 3 & 4: Effective Potential



Note that type of exchange-correlation chosen while specifying pseudopotential



Exchange-Correlation Potential

- $V_{XC} \equiv \delta E_{XC} / \delta n$ contains all the many-body information.
- Known [numerically, from Quantum Monte Carlo ; various analytical approximations] for **homogeneous electron gas**.
- Local Density Approximation:

$$E_{xc}[n] = \int n(\mathbf{r}) V_{xc}^{\text{HOM}}[n(\mathbf{r})] d\mathbf{r}$$

-surprisingly successful!

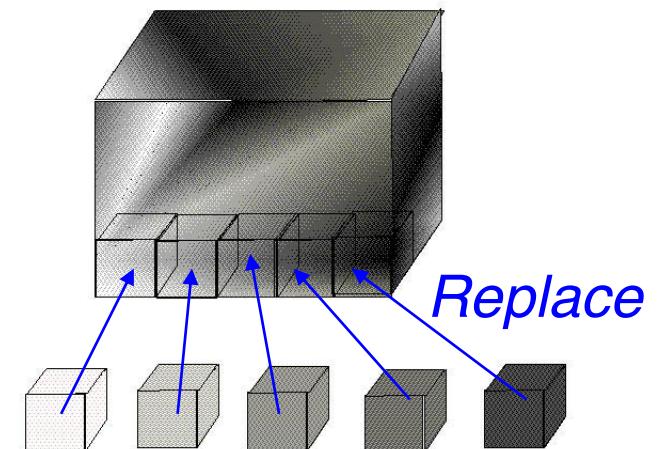
pz

(in name of pseudopotential)

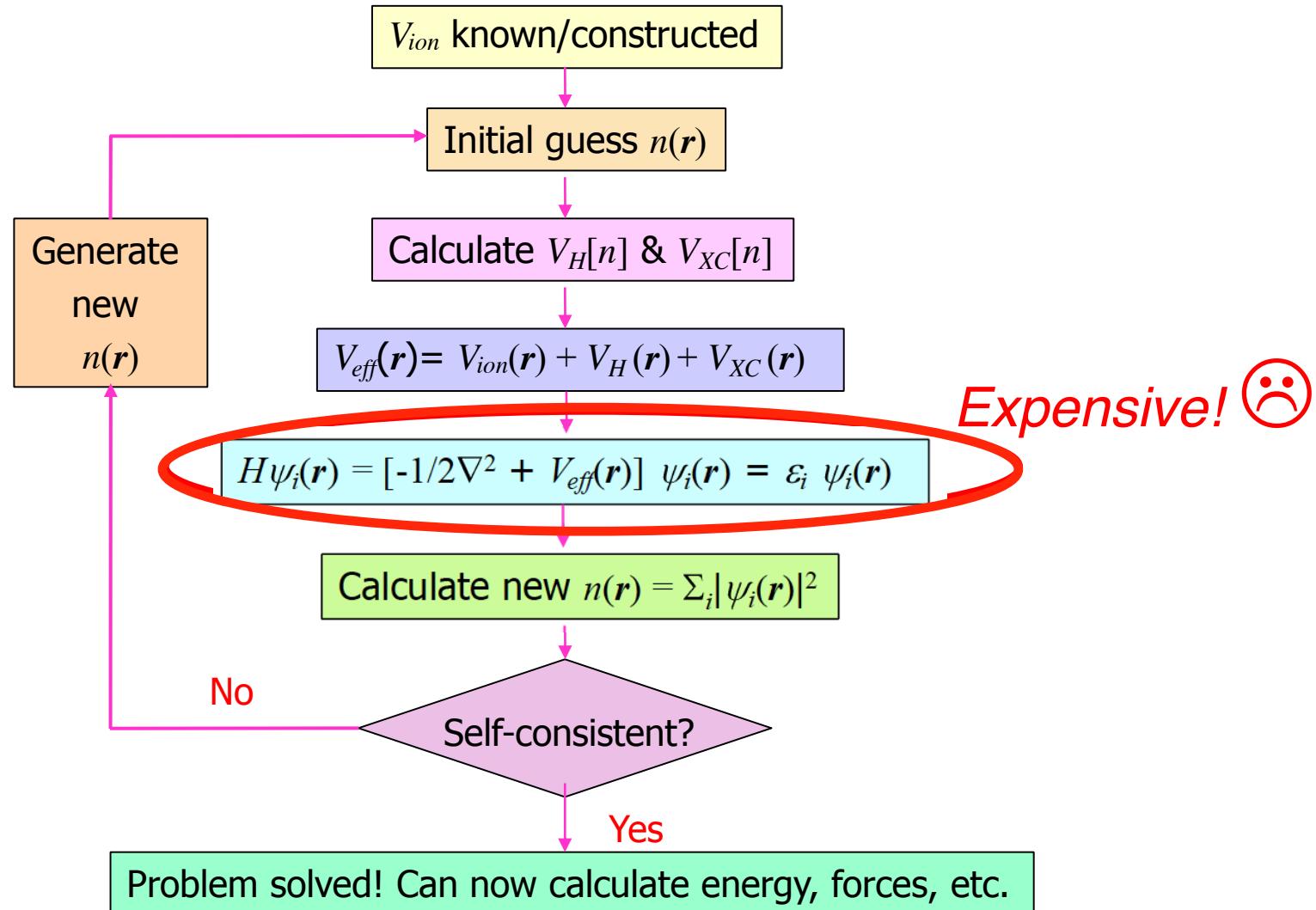
- Generalized Gradient Approximation(s): Include terms involving gradients of $n(\mathbf{r})$

pw91, pbe

(in name of pseudopotential)



Step 5: Diagonalization



Diagonalization

- Need to diagonalize a matrix of size $N_{PW} \times N_{PW}$
- $N_{PW} \gg N_b$ = number of bands required = $N_e/2$ or a little more (for metals).
- OK to obtain lowest few eigenvalues.
- Exact diagonalization is expensive!
- Use **iterative diagonalizers** that recast diagonalization as a minimization problem.

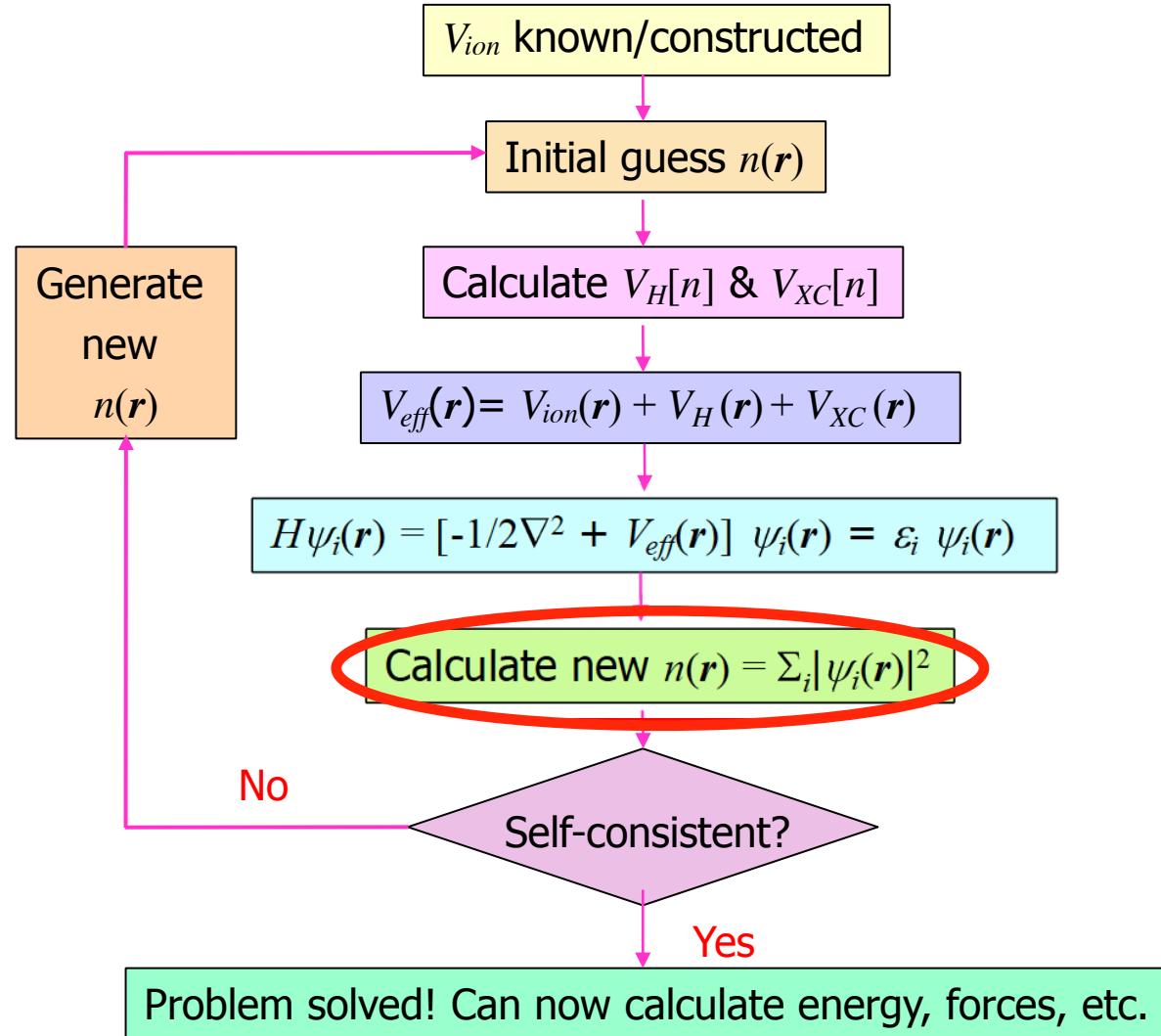
Input parameter **diagonalization**

-which algorithm used for iterative diagonalization

Input parameter **nbnd**

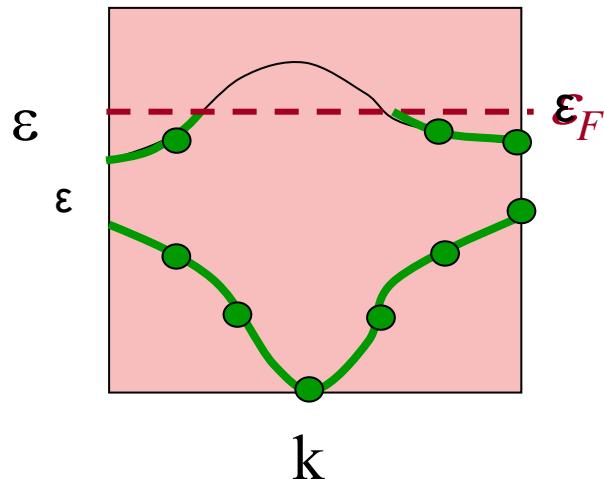
*-how many eigenvalues computed
for metals, choose depending on value of degauss*

Step 6: New Charge Density



Brillouin Zone Sums

- Many quantities (e.g., n , E_{tot}) involve sums over \mathbf{k} .
- In principle, need infinite number of \mathbf{k} 's.
- In practice, sum over a finite number: BZ “Sampling”.
- Number needed depends on band structure.
- Typically need more \mathbf{k} 's for metals.
- Need to test convergence wrt \mathbf{k} -point sampling.



$$\langle P \rangle = \frac{1}{N_{\mathbf{k}}} \sum_{\mathbf{k} \in BZ} P(\mathbf{k}) w_{\mathbf{k}}$$

Symmetry and special points

- Crystals are very symmetric systems whose geometrical properties are best described by the specification of their *space group*
- Space group of a crystal is the group composed by the whole set of translations and point symmetry operations (rotations, inversions, reflections and combinations of the above) that leave the system invariant (including a particular set of operations that combines a rotation with a non-integer translation, or glide, of a fraction of a crystal translation vector, called *non-symmorphic* operations)
- Any function that has the full symmetry of the crystal is invariant upon any operation of the space group: $S_n g(\mathbf{r}) = g(S_n \mathbf{r})$
- In particular, since the Hamiltonian is invariant upon any symmetry operation, any operation S_n leads to a new equation with $\mathbf{r} \rightarrow S_n \mathbf{r}$ and $\mathbf{k} \rightarrow S_n \mathbf{k}$
- The new solution of the transformed equation is still an eigenfunction of the Hamiltonian with the same eigenvalue:
 - A “high-symmetry” \mathbf{k} point is defined by the identity relation: $S_n \mathbf{k} = \mathbf{k}$. Helpful in the classification of electronic states.
 - One can define the Irreducible Brillouin Zone (IBZ), which is the smallest fraction of the Brillouin Zone that is sufficient to determine all the information on the electronic structure of the crystal. All the properties for \mathbf{k} outside the IBZ are obtainable via symmetry operations

Symmetry and special points

- Example in one dimension - the value of the integral

$$I_1 = \int_0^{2\pi} dk \sin(k) = 0$$

is given by the integrand at the mid-point π where $\sin(k)=0$.

- Let's now have an integrand that is the sum of two sine functions, $A_1 \sin(k) + A_2 \sin(2k)$. The exact value of this integral is given by a sum over two points:

$$I_2 = \int_0^{2\pi} dk f_2(k) = 0 = f_2(k = \pi/2) + f_2(k = 3\pi/2)$$

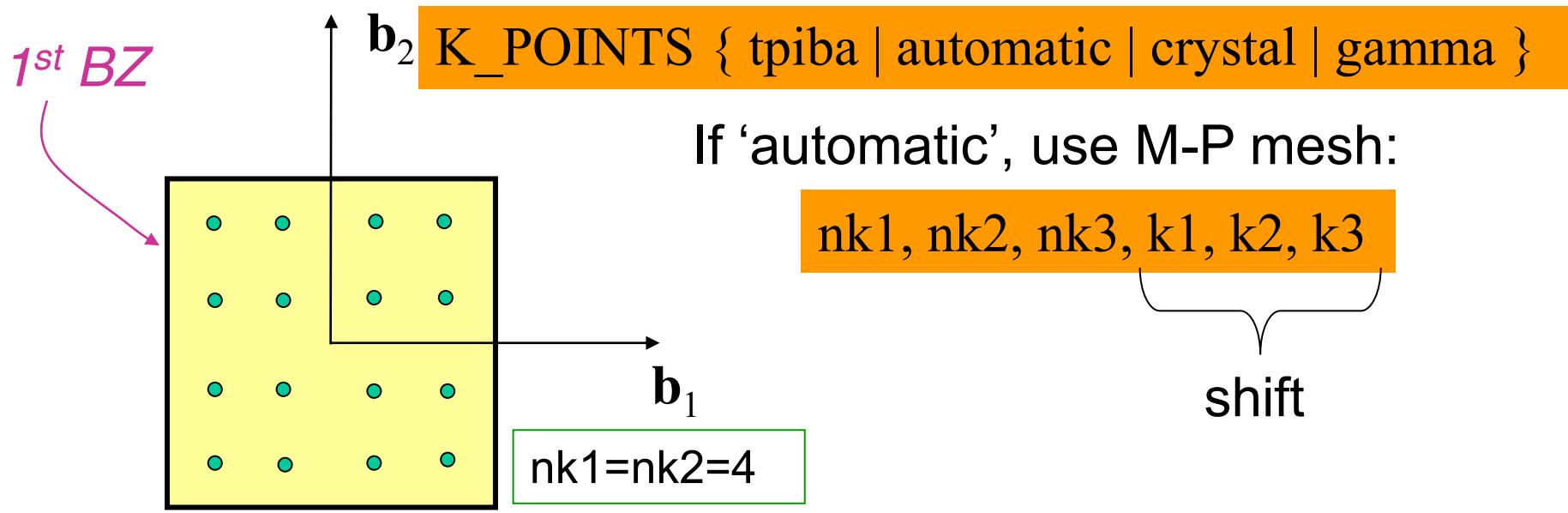
- Integrals over the BZ can be replaced by integrals over the IBZ, where integrals can be written as

$$\bar{f}_i = \sum_{\mathbf{k}}^{IBZ} w_{\mathbf{k}} f_i(\mathbf{k})$$

where we introduce the weights w to take into account the total number of distinguishable (not related by symmetry operations) \mathbf{k} points in the BZ.

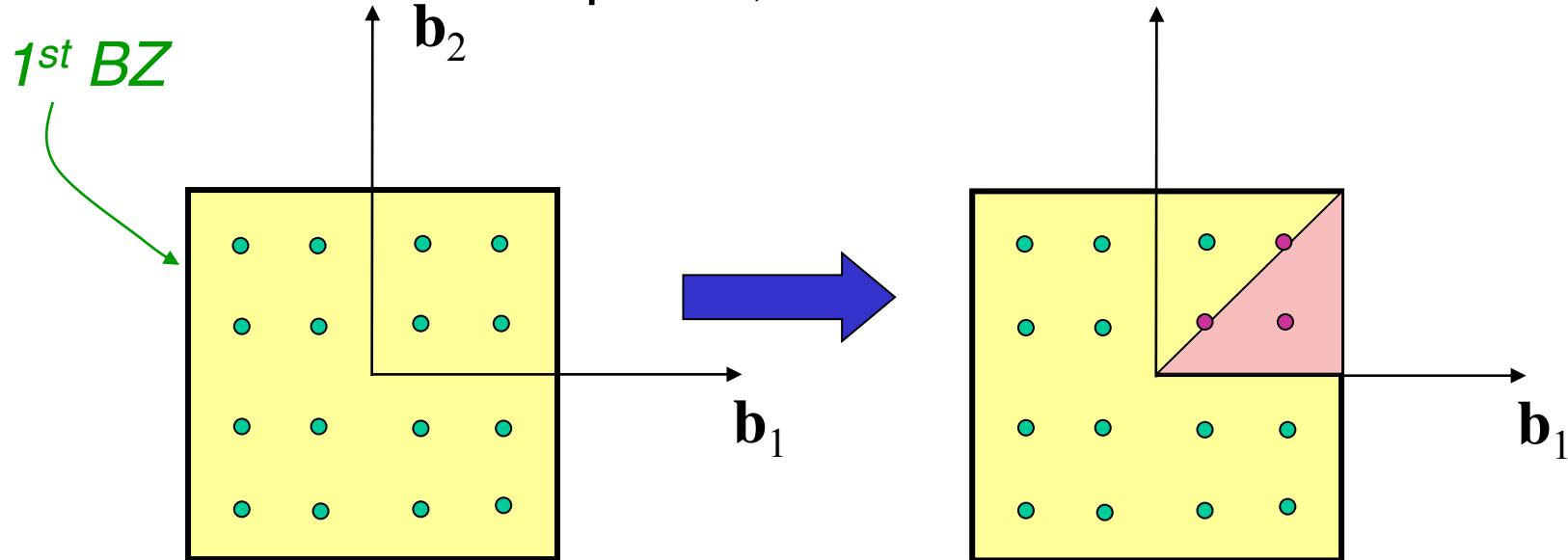
Types of k-point meshes

- **Special Points:** [Chadi & Cohen]
Points designed to give quick convergence for particular crystal structures.
- **Monkhorst-Pack:**
Equally spaced mesh in reciprocal space.
May be centred on origin ['non-shifted'] or not ['shifted']



Irreducible Brillouin Zone

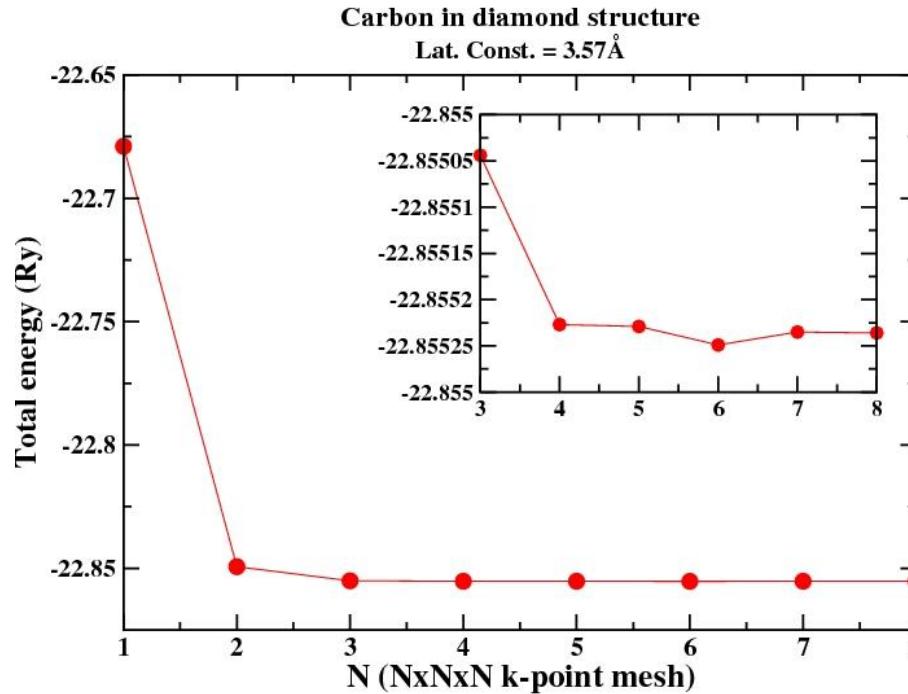
- IBZ depends on symmetries of system.
- Can save computational time by using appropriately weighted k-points from IBZ alone.
- For ‘automatic’ k-points, code will ‘reduce’ to IBZ.



- May not want to maintain symmetries in relaxation/MD.

Input parameter nosym

Convergence wrt BZ sampling



Madhura Marathe

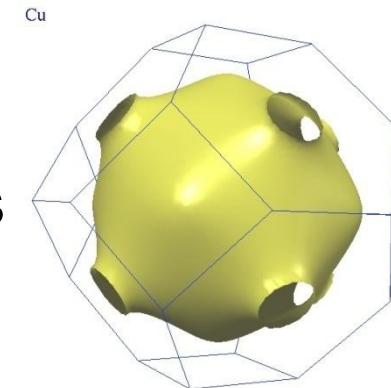
Note: Differences in energy usually converge faster than absolute value of total energy because of error cancellation (if supercells & k-points are identical or commensurate).

Problems with Metals

- Recall:

$$\langle P \rangle = \frac{\Omega}{(2\pi)^3} \sum_{n \text{ occ}_{BZ}} \int P_n(\mathbf{k}) d^3k$$

- For metals, at $T=0$, this corresponds to (for highest band) an integral over all **wave-vectors contained within the Fermi surface**, i.e., for highest band, **sharp discontinuity** in k-space between occupied and unoccupied states...need many k-points to reproduce this accurately.
- Also can lead to **scf convergence problems** because of band-crossings above/below Fermi level.
- Solve by “smearing”.



Cu
Fermi Surface of Cu
iramis.cea.fr

Metals

- Kohn-Sham equations are always self-consistent equations: the effective K-S potential depends on the electron density that is the solution of the K-S equations
- In reciprocal space the procedure becomes:

$$\sum_{m'} H_{m,m'}(\mathbf{k}) c_{i,m'}(\mathbf{k}) = \varepsilon_i(\mathbf{k}) c_{i,m}(\mathbf{k})$$

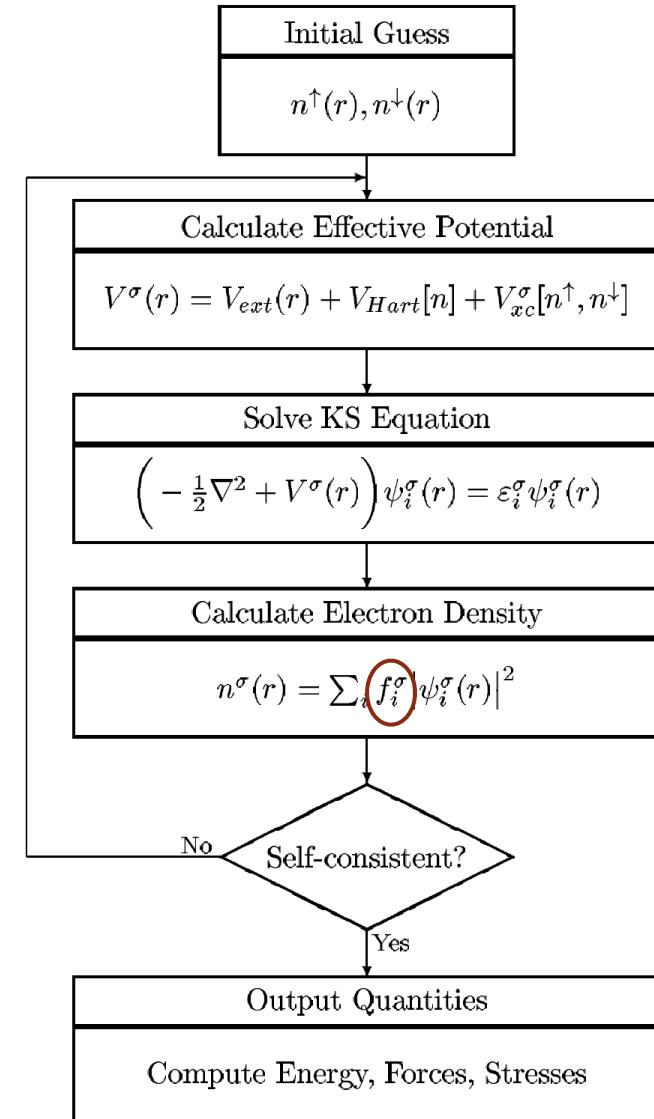
where

$$H_{m,m'}(\mathbf{k}) = |\mathbf{k} + \mathbf{G}_m|^2 \delta_{m,m'} + V_{\text{eff}}[n_{\mathbf{k},i}(\mathbf{G}_m - \mathbf{G}_{m'})]$$

and

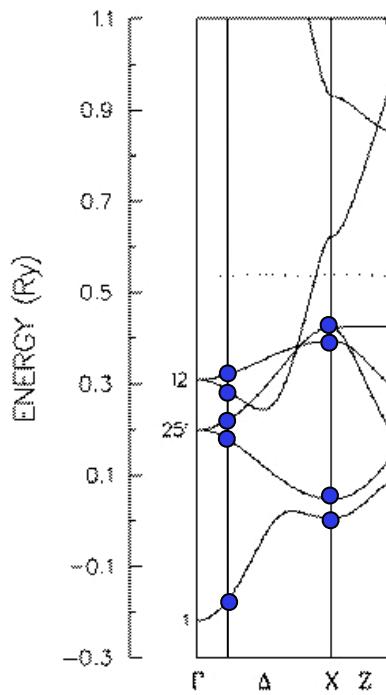
$$n_{\mathbf{k},i}(\mathbf{G}) = \sum_i f_{i,\mathbf{k}} c_i^*(\mathbf{k}) c_i(\mathbf{k})$$

- The density is the sum over the occupied states of the square of the wavefunctions. In general we need an occupation factor that populates the bands according to the Fermi statistics



Metals

- When dealing with metallic systems the occupation of the energy levels (bands) is a more delicate problem
- The number of occupied bands depends on the k -point!



- The numerical self-consistent procedure can be made unstable or incorrect if:
 - Too few k -points in the mesh are used
 - Levels that are close in energy are interchangeably occupied or not: numerical instability

Smeearing in Quantum-ESPRESSO



occupations 'smearing'

Instruction: use smearing

smearing 'gaussian'
'methfessel-paxton'
'marzari-vanderbilt'
'fermi-dirac'

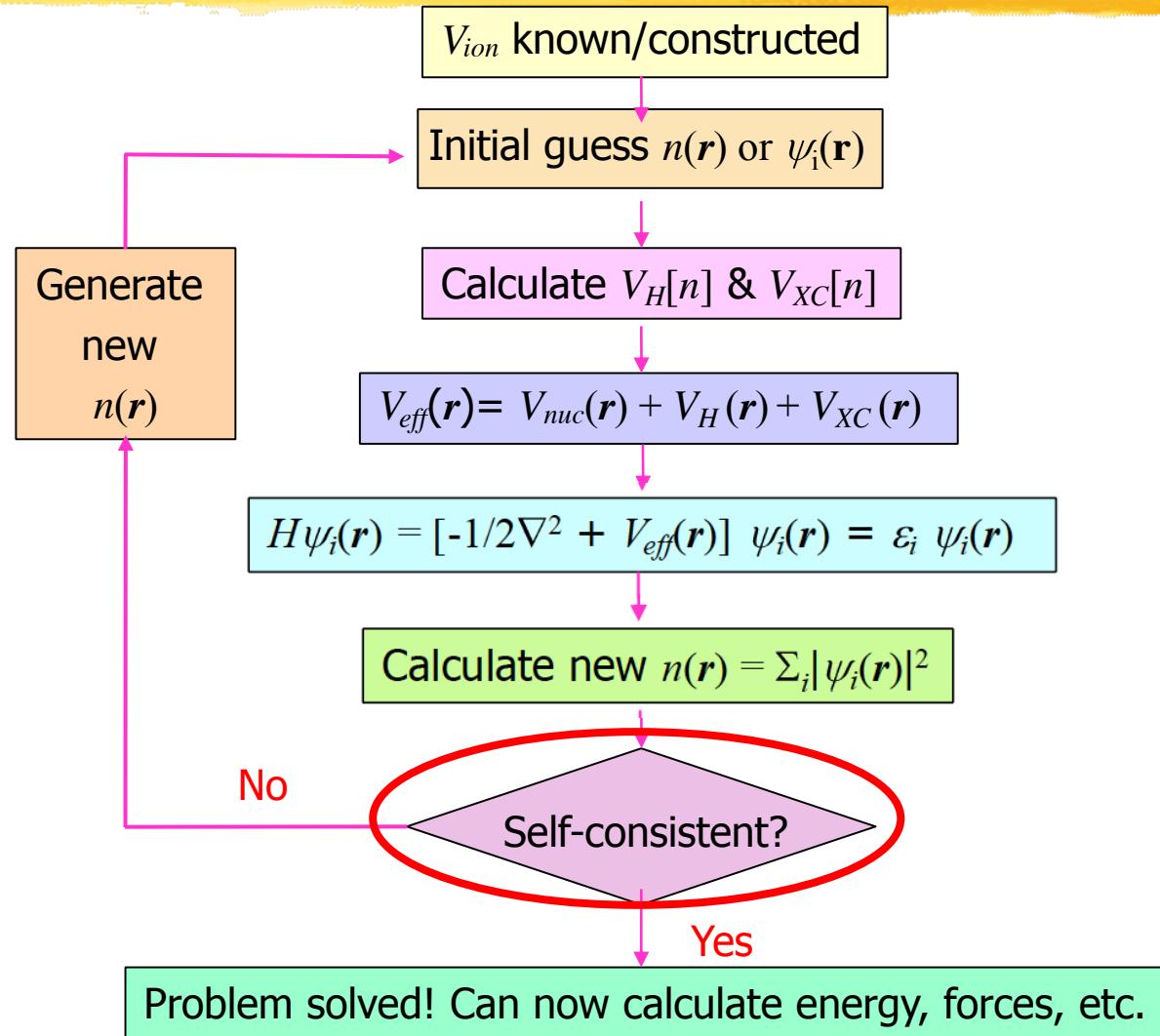
Type of
smearing

degauss

Smearing width

Methfessel & Paxton, Phys. Rev. B 40, 3616 (1989).
Marzari & Vanderbilt, Phys Rev. Lett. 82, 3296 (1999).

Step 7: Check if Convergence Achieved



Testing for scf convergence

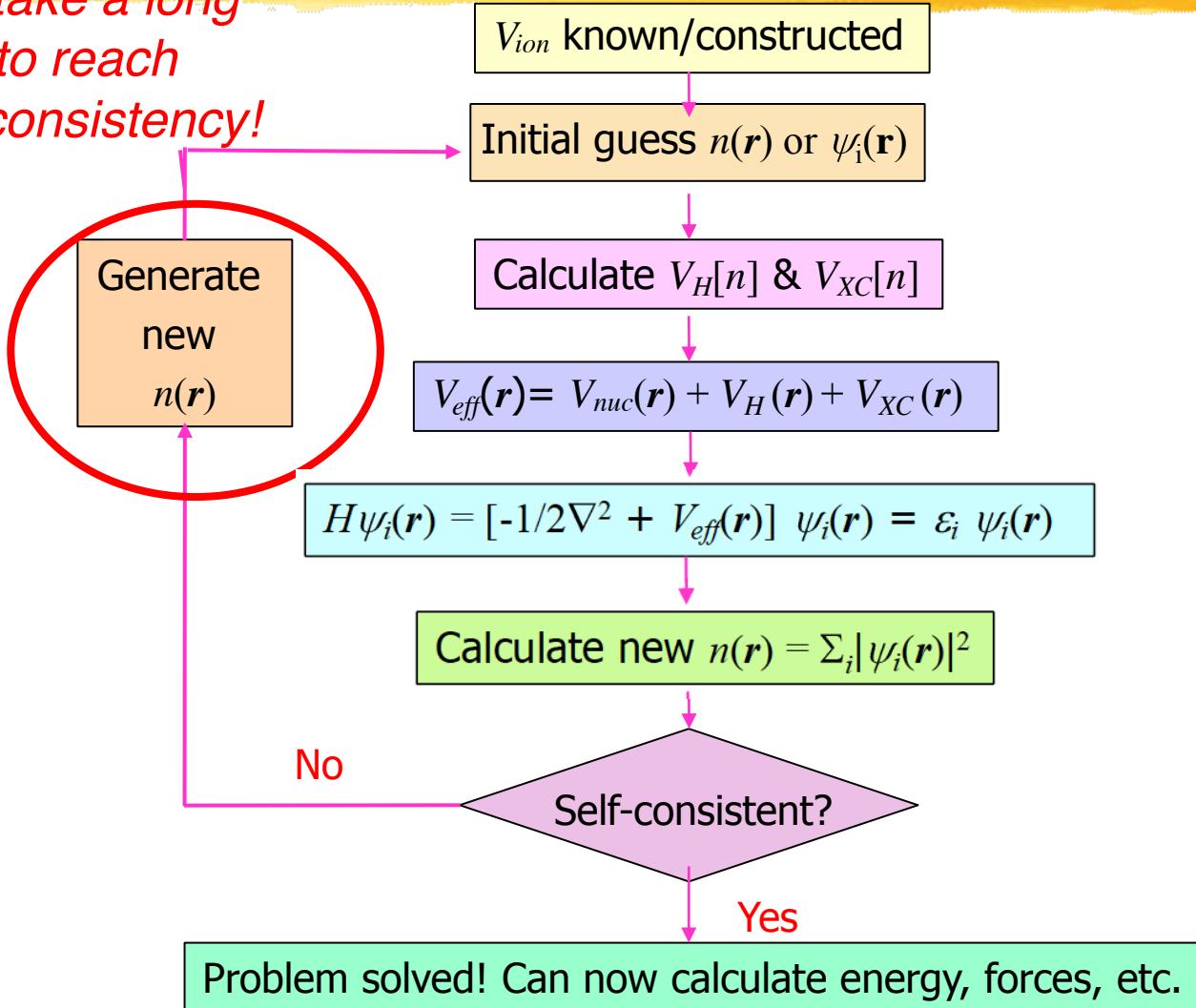
- Compare nth and (n-1)th approximations for density, and see if they are close enough that self-consistency has been achieved.
- Examine squared norm of difference between the charge density in two successive iterations...should be close to zero.



Input parameter **conv_thr**

Step 8: Mixing

Can take a long time to reach self-consistency!



Mixing



- Iterations n of self-consistent cycle:

- Successive approximations to density:

$$n_{in}(n) \rightarrow n_{out}(n) \rightarrow n_{in}(n+1).$$

- $n_{out}(n)$ fed directly as $n_{in}(n+1)$?? No, usually doesn't converge.
- Need to mix, take some combination of input and output densities (may include information from several previous iterations).
- Goal is to achieve self consistency ($n_{out} = n_{in}$) in as few iterations as possible.

Mixing in Quantum-ESPRESSO



Input parameter **mixing_mode**

-Prescription used for mixing.

Input parameter **mixing_beta**

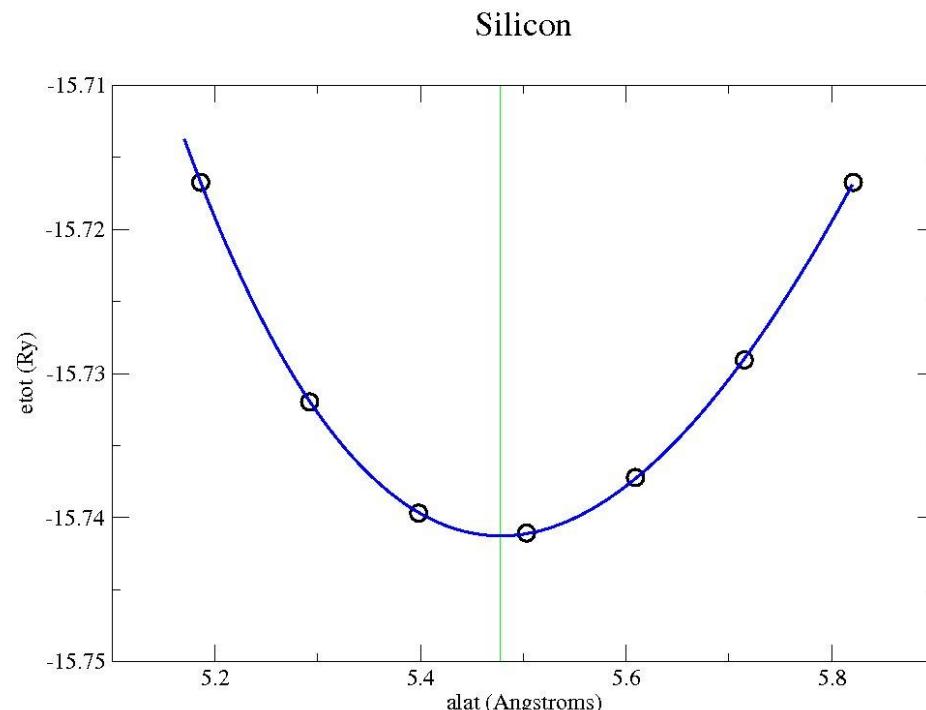
- How much of new density is used at each step*
- Typically use value between 0.1 & 0.7*

A few notes on “convergence”

- The accuracy of an electronic structure calculation depends on various approximation factors, both “physical” and numerical:
- Physical approximations:
 - description of nuclei/ions: all-electron vs. pseudopotential, type of pseudopotential used
 - choice of exchange and correlation functional (LDA vs. GGA. vs. hybrid functionals or mixed schemes (this is usually done in the pseudopotential input file)
- Numerical approximations:
 - The accuracy of the basis set:
 - In a plane wave basis, the extension of the reciprocal space mesh (Fourier basis)
 - In a real space calculation, the density of the grid in real space
 - The accuracy of the integrals in reciprocal space: density of the special \mathbf{k} -point mesh. Particularly important for metallic systems, where large sets of \mathbf{k} -points have to be used for a consistent description of the system
 - The accuracy of the self-consistent solution of the K-S equations: mixing schemes and convergence threshold

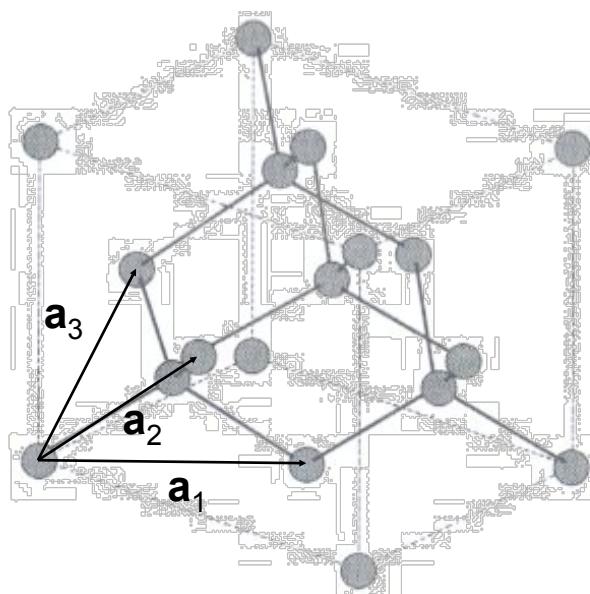
Output Quantities: Total Energy

- Perhaps the most important output quantity is the TOTAL ENERGY
- Can use, e.g., to optimize structure
- e.g., for a cubic crystal, where the structure can be specified by a single parameter (side of cube):



Results

- Calculation for a crystal of Si
 - Diamond structure
 - Equilibrium lattice parameter
 - FCC cell with two atoms in the basis - 48 point symmetry operation with non-symmorphic translations



Results

pw.x < si.scf.in > out

- Preliminaries

```
Program PWSCF      v.2.0      starts ...
Today is 27Aug2004 at 13:20:55

Ultrasoft (Vanderbilt) Pseudopotentials

Current dimensions of program pwscf are:
ntypx =10    npk =40000    lmax = 3
nchix = 6    ndim = 2000    nbrx = 8 nqfm = 8
```

- First part: geometry
 - Warning on the existence of non-symmorphic point group operations

```
warning: symmetry operation # 5 not allowed. fractional translation:
  0.2500000  0.2500000  0.2500000 in crystal coordinates
warning: symmetry operation # 6 not allowed. fractional translation:
  0.2500000  0.2500000  0.2500000 in crystal coordinates
warning: symmetry operation # 7 not allowed. fractional translation:
  0.2500000  0.2500000  0.2500000 in crystal coordinates
warning: symmetry operation # 8 not allowed. fractional translation:
  0.2500000  0.2500000  0.2500000 in crystal coordinates
warning: symmetry operation # 9 not allowed. fractional translation:
  0.2500000  0.2500000  0.2500000 in crystal coordinates
warning: symmetry operation # 10 not allowed. fractional translation:
  0.2500000  0.2500000  0.2500000 in crystal coordinates
```

Results

- More on geometry and computational parameters:

```
bravais-lattice index      =          2
lattice parameter (a_0)    =     10.3500  a.u.
unit-cell volume           =     277.1795 (a.u.)^3
number of atoms/cell       =          2
number of atomic types     =          1
kinetic-energy cutoff       =     30.0000  RY
charge density cutoff      =    120.0000  RY
convergence threshold      =   1.0E-08
beta                       =      0.7000
number of iterations used  =          8 plain      mixing
Exchange-correlation        =  SLA  PW  PBE  PBE (1434)
iswitch = 0

celldm(1)= 10.350000  celldm(2)= 0.000000  celldm(3)= 0.000000
celldm(4)= 0.000000  celldm(5)= 0.000000  celldm(6)= 0.000000

crystal axes: (cart. coord. in units of a_0)
a(1) = ( -0.500000  0.000000  0.500000 )
a(2) = (  0.000000  0.500000  0.500000 )
a(3) = ( -0.500000  0.500000  0.000000 )

reciprocal axes: (cart. coord. in units 2 pi/a_0)
b(1) = ( -1.000000 -1.000000  1.000000 )
b(2) = (  1.000000  1.000000  1.000000 )
b(3) = ( -1.000000  1.000000 -1.000000 )
```

lattice parameter and volume of the cell, atoms and species, kinetic energy cut-off (ecutwfc), convergence threshold and mixing parameters, exchange and correlation functional, real and reciprocal space basis vectors

Results

- Pseudopotential parameters

```
PSEUDO 1 is Si          zval = 4.0    lmax= 2    lloc= 0
(in numerical form:   883 grid points, xmin = 0.00, dx = 0.0000)

atomic species  valence  mass    pseudopotential
Si             4.00     28.08600   Si( 1.00)
```

- Symmetry operations and atomic coordinates

```
24 Sym.Ops. (no inversion)

Cartesian axes

site n.      atom      positions (a_0 units)
    1           Si      tau( 1) = (  0.0000000  0.0000000  0.0000000 )
    2           Si      tau( 2) = (  0.2500000  0.2500000  0.2500000 )
```

- k-mesh for BZ integration

```
number of k points= 10
                    cart. coord. in units 2pi/a_0
k( 1) = (  0.1250000  0.1250000  0.1250000), wk = 0.0625000
k( 2) = (  0.1250000  0.1250000  0.3750000), wk = 0.1875000
k( 3) = (  0.1250000  0.1250000  0.6250000), wk = 0.1875000
k( 4) = (  0.1250000  0.1250000  0.8750000), wk = 0.1875000
k( 5) = (  0.1250000  0.3750000  0.3750000), wk = 0.1875000
k( 6) = (  0.1250000  0.3750000  0.6250000), wk = 0.3750000
k( 7) = (  0.1250000  0.3750000  0.8750000), wk = 0.3750000
k( 8) = (  0.1250000  0.6250000  0.6250000), wk = 0.1875000
k( 9) = (  0.3750000  0.3750000  0.3750000), wk = 0.0625000
k( 10) = (  0.3750000  0.3750000  0.6250000), wk = 0.1875000
```

Results

- Iterate!

```

Initial potential from superposition of free atoms
starting charge = 7.99889
Starting wfc are atomic

total cpu time spent up to now is 1.58 secs

iteration # 1 ecut= 30.00 ryd beta=0.70
Davidson diagonalization (with overlap)
ethr = 1.00E-02, avg # of iterations = 2.0

Threshold (ethr) on eigenvalues was too large:
Diagonalizing with lowered threshold

Davidson diagonalization (with overlap)
ethr = 6.93E-03, avg # of iterations = 1.0

total energy = -15.73802140 ryd
estimated scf accuracy < 0.05493104 ryd

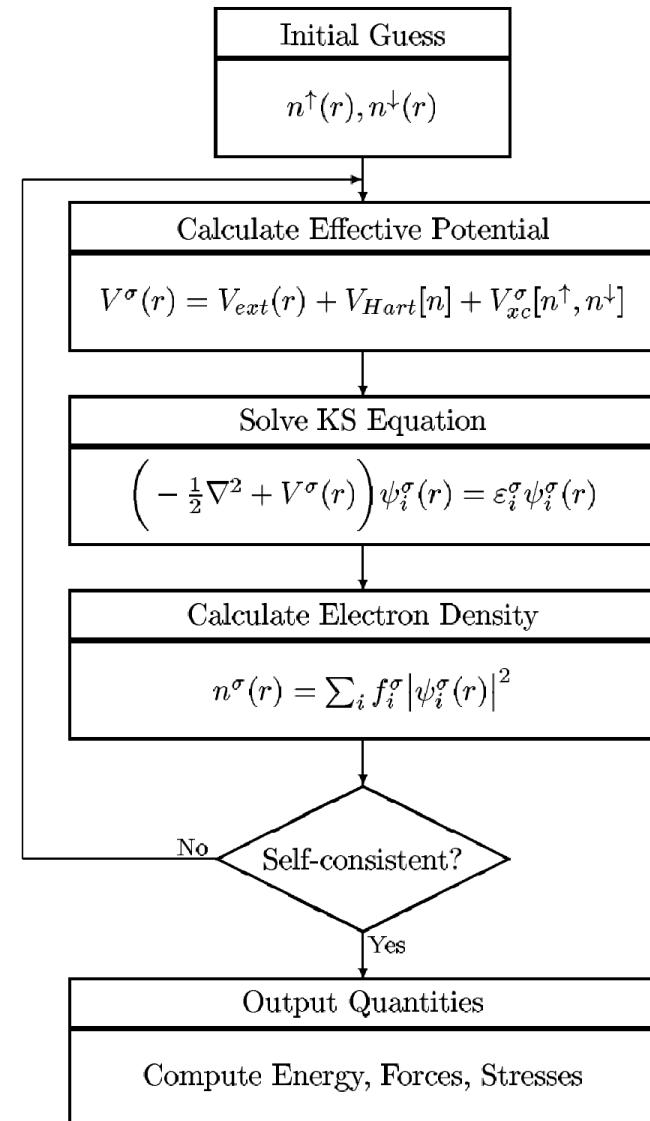
total cpu time spent up to now is 4.97 secs

iteration # 2 ecut= 30.00 ryd beta=0.70
Davidson diagonalization (with overlap)
ethr = 6.87E-04, avg # of iterations = 1.0

total energy = -15.74113145 ryd
estimated scf accuracy < 0.00282417 ryd

total cpu time spent up to now is 6.50 secs

```



Results

- Convergence!
- Electronic energies (eigenvalues of the K-S equation) for the occupied bands of the solid
- Total energy = ground state energy = minimum of the K-S functional
- Individual contributions to the ground state energy (given in an alternative way, that groups together various contributions and uses the sum of the eigenvalues of the K-S equations):
 - band energy
 - 1-electron contribution
 - Hartree energy
 - xc energy
 - Ewald (ionic) energy
- Implicitly, we clearly know the ground state electron density

```
iteration # 6      ecut=    30.00 ryd      beta=0.70
Davidson diagonalization (with overlap)
ethr =  3.58E-09,  avg # of iterations =  2.4

      k = 0.1250 0.1250 0.1250 ( 773 PWs)  bands (ev):
-5.5650   4.3943   5.6850   5.6850

      k = 0.1250 0.1250 0.3750 ( 774 PWs)  bands (ev):
-5.0396   2.7936   4.6850   4.7529

      k = 0.1250 0.1250 0.6250 ( 768 PWs)  bands (ev):
-4.0132   1.0990   3.3701   3.8164

      k = 0.1250 0.1250 0.8750 ( 763 PWs)  bands (ev):
-2.5486  -0.7523   2.6288   3.3521

      k = 0.1250 0.3750 0.3750 ( 776 PWs)  bands (ev):
-4.5430   1.4260   3.7181   5.2077

      k = 0.1250 0.3750 0.6250 ( 770 PWs)  bands (ev):
-3.5991   0.2286   2.7345   4.0905

      k = 0.1250 0.3750 0.8750 ( 757 PWs)  bands (ev):
-2.3805  -0.8498   2.0239   3.0924

      k = 0.1250 0.6250 0.6250 ( 769 PWs)  bands (ev):
-2.9142  -0.5528   2.0838   4.1273

      k = 0.3750 0.3750 0.3750 ( 763 PWs)  bands (ev):
-4.1115   0.1417   4.9030   4.9030

      k = 0.3750 0.3750 0.6250 ( 767 PWs)  bands (ev):
-3.4071  -0.6418   3.7401   4.4421

total energy          =     -15.74164440 ryd
estimated scf accuracy <      0.00000001 ryd

band energy sum       =      0.59060426 ryd
one-electron contribution =     4.58204631 ryd
hartree contribution   =     1.11702419 ryd
xc contribution        =    -4.78588032 ryd
ewald contribution     =   -16.65483457 ryd

convergence has been achieved
```

Results

- From the knowledge of the electron density, we can compute all the ground state properties of interest
- Forces (in this case are zero because of symmetry)

```
Forces acting on atoms (Ry/au):  
  
atom    1 type  1   force =      0.00000000      0.00000000      0.00000000  
atom    2 type  1   force =      0.00000000      0.00000000      0.00000000  
  
Total force =      0.000000      Total SCF correction =      0.000000
```

- Stress (zero, or negligible, since the calculation has been done at the equilibrium lattice parameter)

```
entering subroutine stress ...  
  
total stress (ryd/bohr**3)          (kbar)      P= -0.91  
-0.00000620  0.00000000  0.00000000  -0.91  0.00  0.00  
0.00000000 -0.00000620  0.00000000  0.00  -0.91  0.00  
0.00000000  0.00000000 -0.00000620  0.00  0.00  -0.91
```



III. Forces & Geometry Optimization

Forces

- Need for **geometry optimization** and **molecular dynamics**.
- Could get as **finite differences** of total energy - too expensive!
- Use **force (Hellmann-Feynman) theorem**:
 - Want to calculate the force on ion I :

$$\mathbf{F}_I = -\frac{d}{d\mathbf{R}_I} \langle \Psi | H | \Psi \rangle$$

- Get three terms:

$$\mathbf{F}_I = -\langle \Psi | \frac{\partial H}{\partial \mathbf{R}_I} | \Psi \rangle - \langle \frac{\partial \Psi}{\partial \mathbf{R}_I} | H | \Psi \rangle - \langle \Psi | H | \frac{\partial \Psi}{\partial \mathbf{R}_I} \rangle$$

When $|\Psi\rangle$ is an eigenstate, $H|\Psi\rangle = E|\Psi\rangle$

- Substitute this...

Forces (contd.)

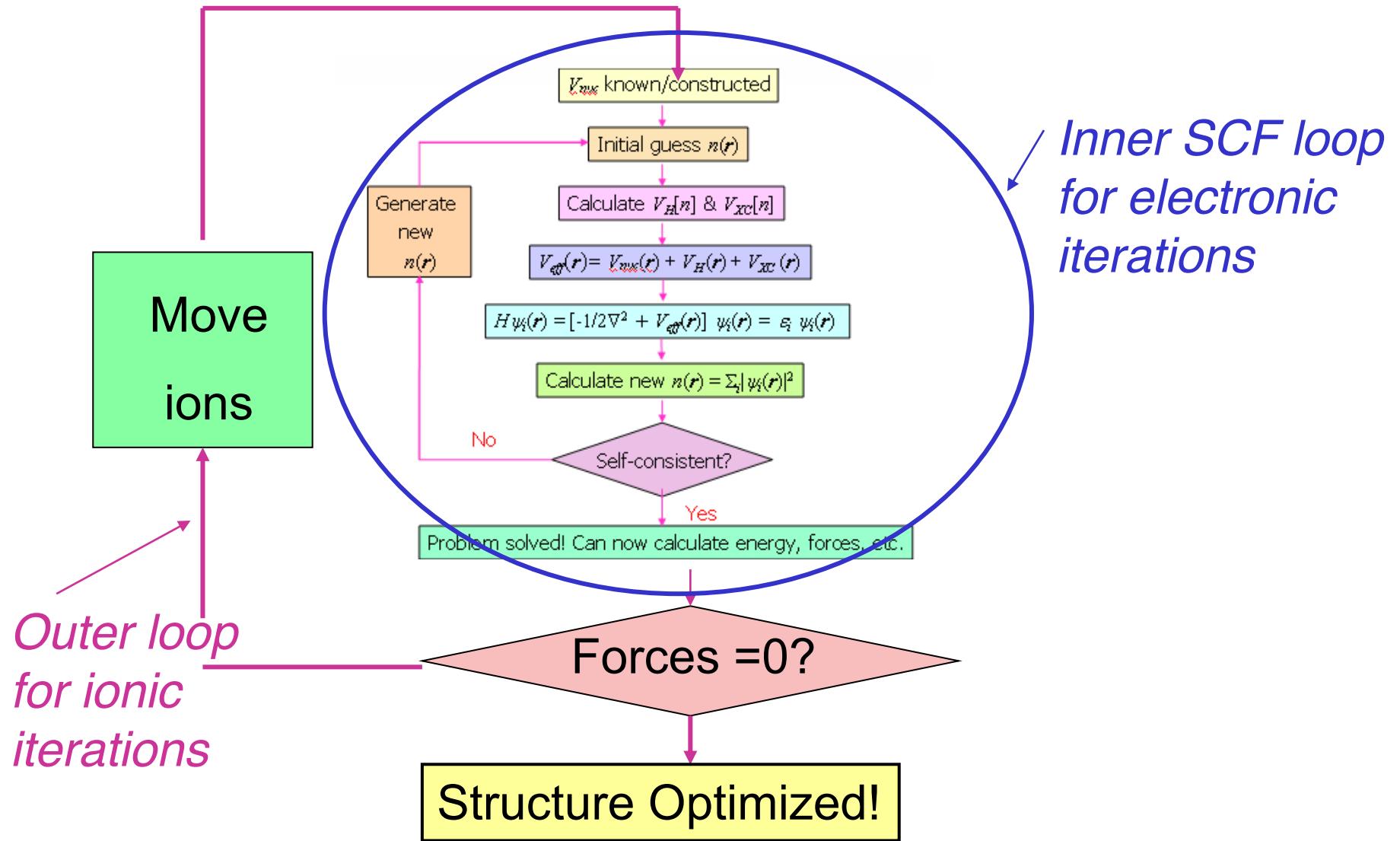
- The force is now given by

$$\begin{aligned}\mathbf{F}_I &= -\langle \Psi | \frac{\partial H}{\partial \mathbf{R}_I} | \Psi \rangle - E \langle \frac{\partial \Psi}{\partial \mathbf{R}_I} | \Psi \rangle - E \langle \Psi | \frac{\partial \Psi}{\partial \mathbf{R}_I} \rangle \\ &= -\langle \Psi | \frac{\partial H}{\partial \mathbf{R}_I} | \Psi \rangle - E \cancel{\frac{\partial}{\partial \mathbf{R}_I}} \langle \Psi | \Psi \rangle \xrightarrow{0}\end{aligned}$$

- Note that we can now calculate the force from a calculation at ONE configuration alone – huge savings in time.
- If the basis depends upon ionic positions (not true for plane waves), would have extra terms = Pulay forces.
- $|\Psi\rangle$ should be exact eigenstate, i.e., scf well-converged!

Input parameter tprnfor

An Outer Loop: Ionic Relaxation



Geometry Optimization With Forces



- Especially useful for optimizing internal degrees of freedom, surface relaxation, etc.

```
calculation = 'relax'
```

- Choice of algorithms for ionic relaxation, e.g., steepest descent, BFGS.

```
NAMELIST &IONS
```

```
Input parameter ion_dynamics
```

Results

- Collect run-time statistics and finish the run

```
PWSCF      :    14.95s CPU time

init_run   :    1.53s CPU
electrons  :   12.86s CPU
forces     :    0.07s CPU
stress     :    0.30s CPU

electrons  :   12.86s CPU
c_bands    :    9.22s CPU (      7 calls,  1.317 s avg)
sum_band   :    1.74s CPU (      7 calls,  0.249 s avg)
v_of_rho   :    1.81s CPU (     14 calls,  0.129 s avg)
mix_rho   :    0.20s CPU (      7 calls,  0.029 s avg)

c_bands    :    9.22s CPU (      7 calls,  1.317 s avg)
init_us_2  :    0.08s CPU (    170 calls,  0.000 s avg)
cegterg   :    9.11s CPU (     70 calls,  0.130 s avg)

sum_band   :    1.74s CPU (      7 calls,  0.249 s avg)

wfcrot    :    1.04s CPU (     10 calls,  0.104 s avg)
cegterg   :    9.11s CPU (     70 calls,  0.130 s avg)
h_psi      :    9.57s CPU (    217 calls,  0.044 s avg)
g_psi      :    0.01s CPU (    137 calls,  0.000 s avg)
overlap    :    0.00s CPU (    137 calls,  0.000 s avg)
cdiaghg   :    0.25s CPU (    147 calls,  0.002 s avg)
update     :    0.13s CPU (    137 calls,  0.001 s avg)
last       :    0.06s CPU (     70 calls,  0.001 s avg)

h_psi      :    9.57s CPU (    217 calls,  0.044 s avg)
init       :    0.02s CPU (    217 calls,  0.000 s avg)
firstfft   :    4.79s CPU (   829 calls,  0.006 s avg)
secondfft  :    4.16s CPU (   829 calls,  0.005 s avg)
add_vuspsi :    0.15s CPU (   217 calls,  0.001 s avg)

General routines
ccalbec   :    0.11s CPU (   237 calls,  0.000 s avg)
cft3      :    1.28s CPU (   175 calls,  0.007 s avg)
cft3s     :   10.03s CPU (  1938 calls,  0.005 s avg)
davcio    :    0.05s CPU (   240 calls,  0.000 s avg)
```