Hands-on: Band structure and DOS metals/insulators

Tutors: Alberto Carta, Fatema Mohamed, Bryce Rodrigue Malonda and George Manyali

Topics of Day3 hands-on session:

1. How to relax a structure (graphane and oxygen adsorption)
2. How to run post-processing calculations to plot DOS (dos.x), PDOS (projwfc.x) and band-structure (bands.x).
3. How to choose the smearing and the broadening parameters for metals.
4. How to compute forces and stresses.

Exercise 1

Forces and relaxation

Input file scf calculation

```
 &CONTROL
    calculation = 'scf',
    prefix      = 'Graphane',
    outdir      = '/tmp',
    pseudo_dir  = '../../pseudo',
/

 &SYSTEM
....
....


/

 &ELECTRONS
    conv_thr = 1.0d-8
/




ATOMIC_SPECIES
   C  12.0107 C.pbe-rrkjus.UPF
   H  1.00007 H.pbe-rrkjus.UPF

ATOMIC_POSITIONS alat
...

K_POINTS automatic
   9 9 1   0 0 0
```

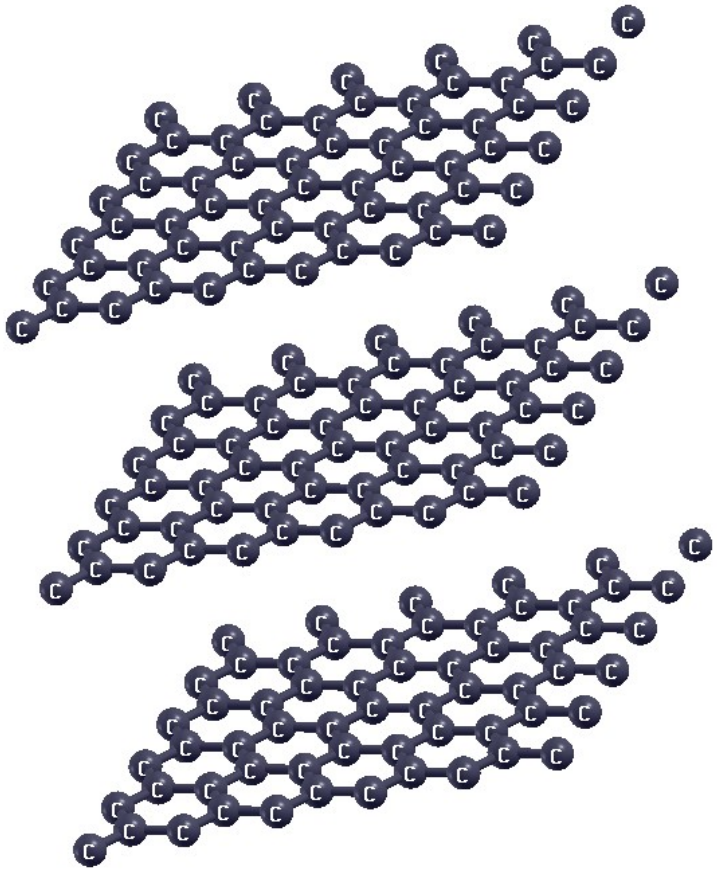
Input file for relaxation

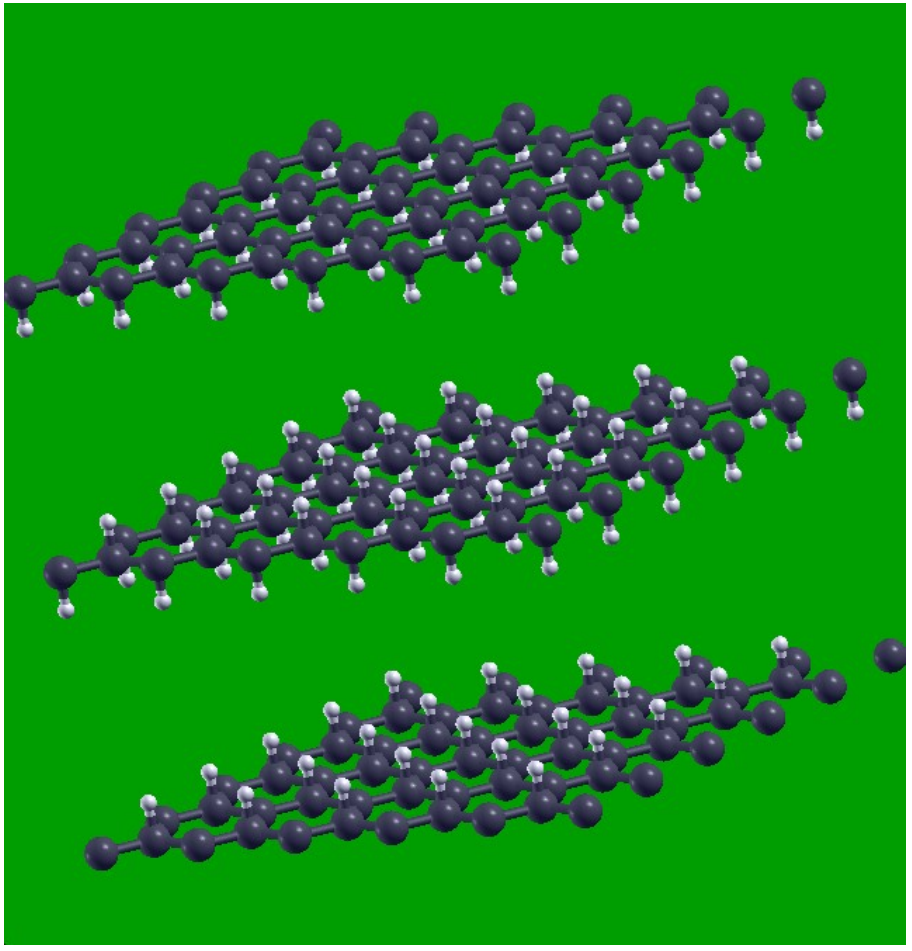
```
 &CONTROL
    calculation = 'relax',
    prefix      = 'Graphane',
    outdir      = '/tmp',
    pseudo_dir  = '../../pseudo',
/

 &SYSTEM
....
....


/

 &ELECTRONS
    conv_thr = 1.0d-8
/

 &IONS
/

ATOMIC_SPECIES
   C  12.0107 C.pbe-rrkjus.UPF
   H  1.00007 H.pbe-rrkjus.UPF

ATOMIC_POSITIONS alat
...

K_POINTS automatic
   9 9 1   0 0 0
```

Ions is jargon for atoms, this cell must appear!

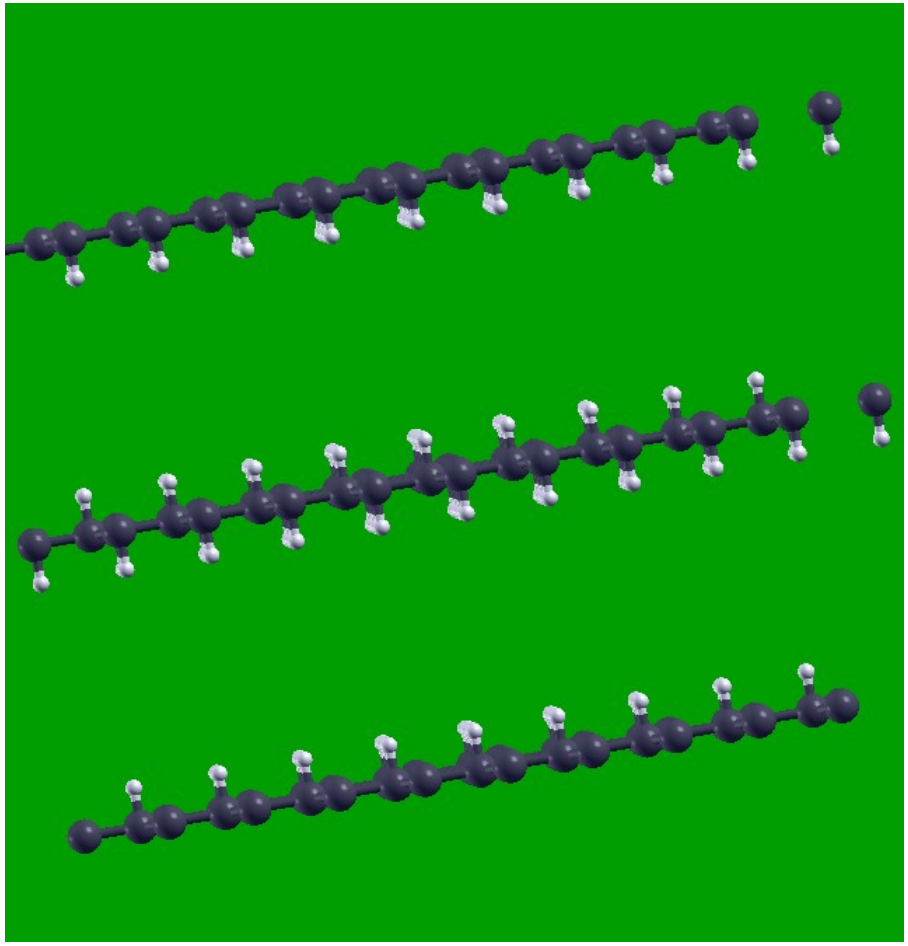
Buckling of graphane

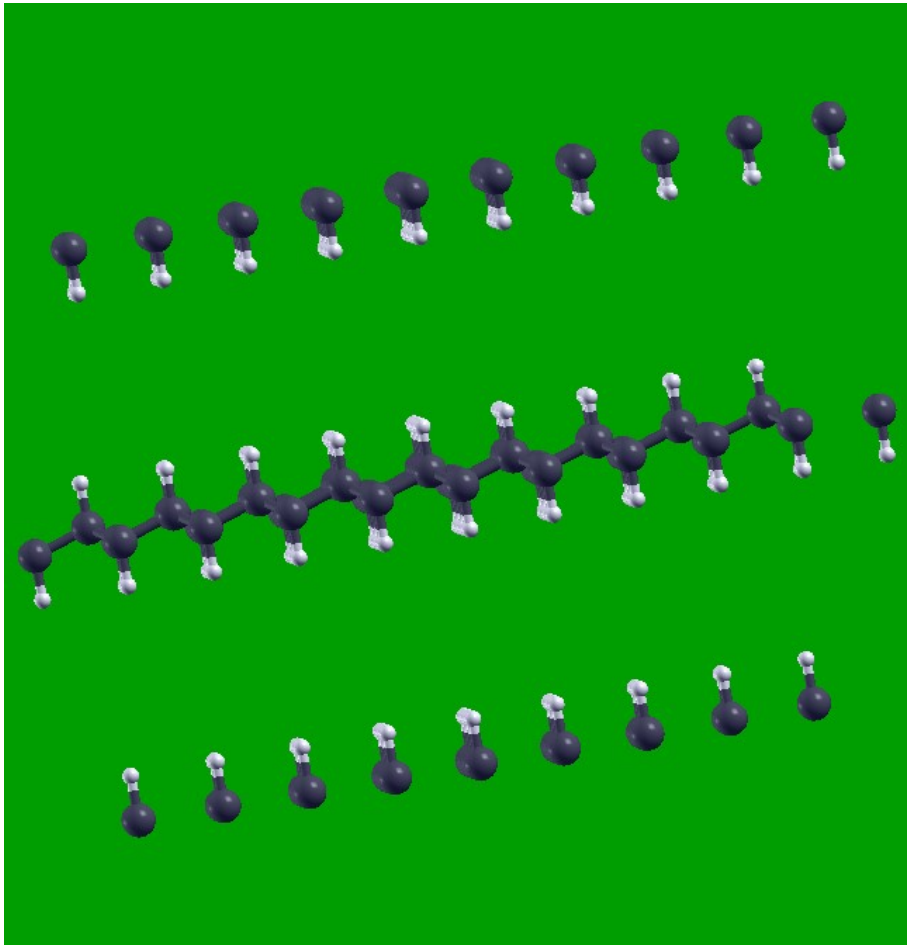
Graphane is hydrogenated graphene → hydrogen saturates double-bonds / pi-delocalized cloud





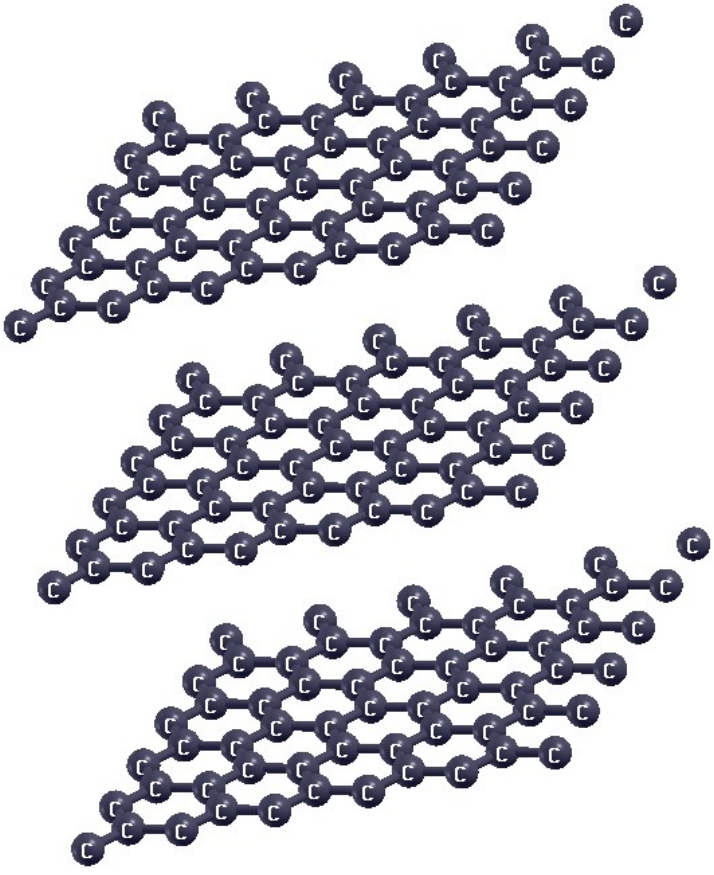
Buckling of graphane

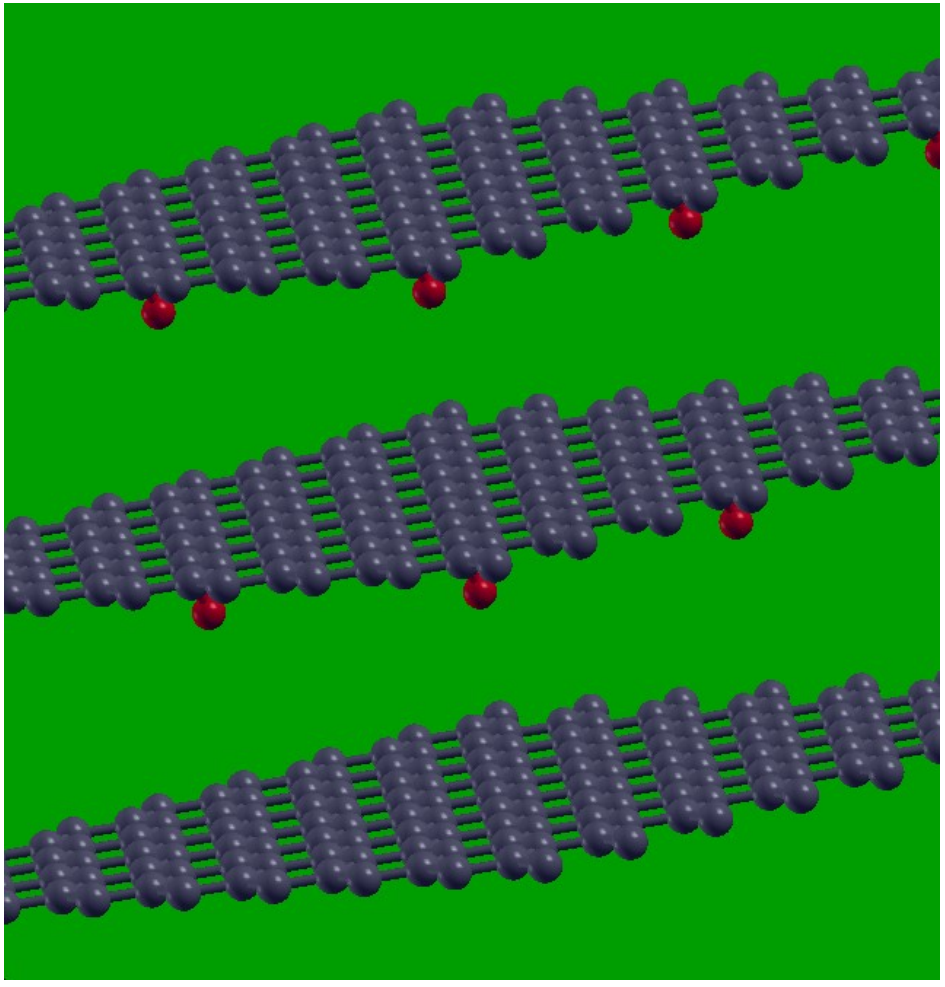




Oxygen adsorption on graphene

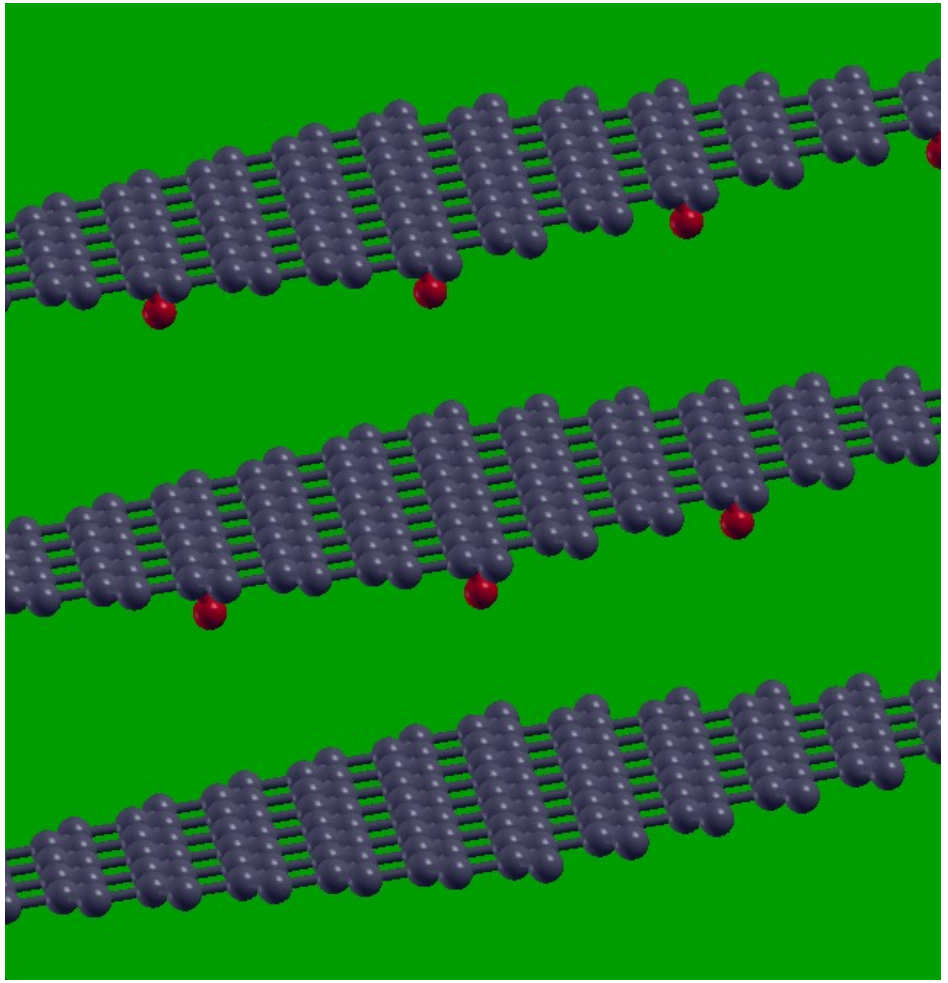
Oxygen is strongly electronegative → electron acceptor

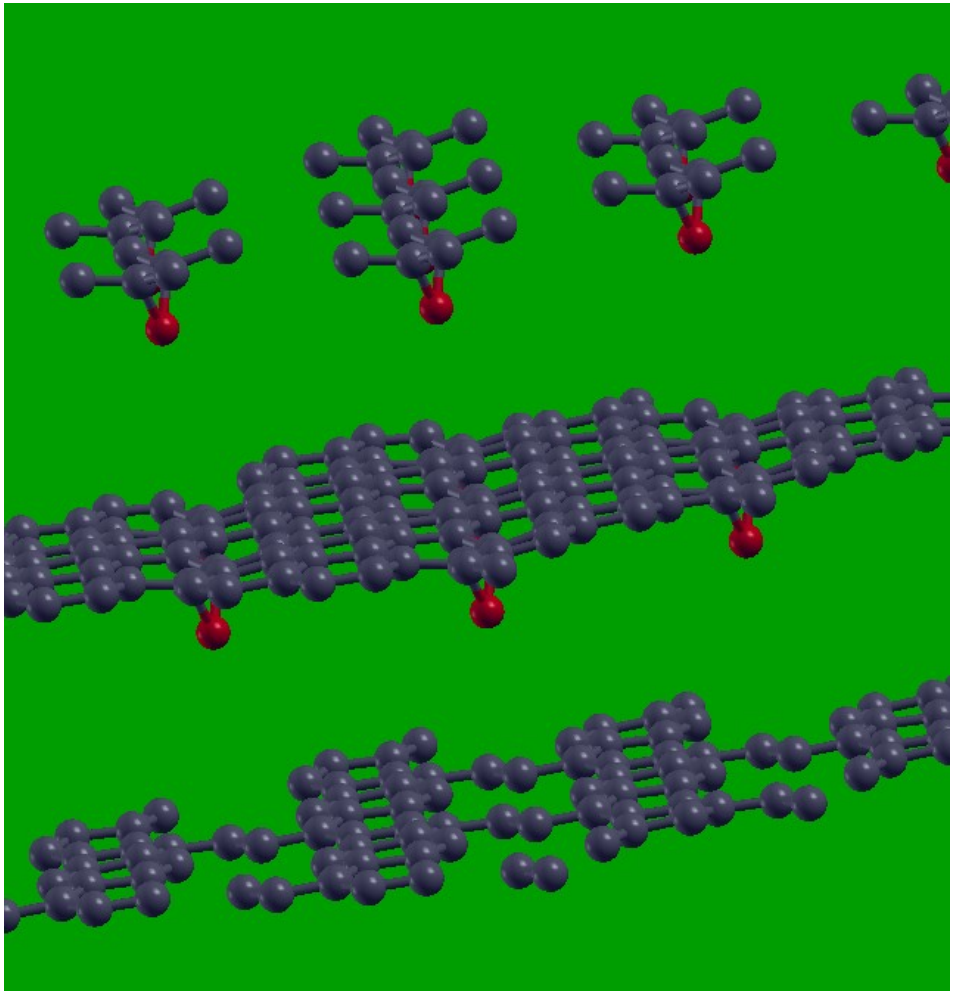




Oxygen adsorption on graphene

Oxygen is strongly electronegative → electron acceptor





Exercise 2

Follow up on silicon for Si

Input file scf calculation

```
&CONTROL
   calculation='scf',
   prefix='silicon',
   pseudo_dir='../../pseudo/',
   outdir='./tmp'
/
&SYSTEM
   ibrav =  2,
   celldm(1) = 10.6,
   nat =  2,
   ntyp = 1,
   nbnd=10,
   ecutwfc = 36,
/
&ELECTRONS
/

ATOMIC_SPECIES
   Si  28.086  Si.pz-vbc.UPF
ATOMIC_POSITIONS
   Si 0.00 0.00 0.00
   Si 0.25 0.25 0.25
K_POINTS automatic
   6 6 6   1 1 1
```

Input file for relaxation of unit cell

```
&CONTROL
   calculation='vc-relax',
   prefix='silicon',
   pseudo_dir='../../pseudo/',
   outdir='./tmp'
/
&SYSTEM
   ibrav =  2,
   celldm(1) = 10.6,
   nat =  2,
   ntyp = 1,
   nbnd=10,
   ecutwfc = 36,
/
&ELECTRONS
/
&IONS
/
&CELL
/
ATOMIC_SPECIES
   Si  28.086  Si.pz-vbc.UPF
ATOMIC_POSITIONS
   Si 0.00 0.00 0.00
   Si 0.25 0.25 0.25
K_POINTS automatic
   6 6 6   1 1 1
```

Slightly different calculation mode

We need also a CELL section

Relax lattice constant for Si

Run `pw.x < si.relax.in | tee si.relax.out` and consider the following:

- Compare the lattice volume to the one we optimized on Day 2, are they the same?
- What is the code doing from iteration to iteration?
- The new lattice constant is given in terms of the starting one, try to compute the relaxed lattice constant from the output and compare with the one we optimized on Day 2.

Density of State (DOS) for Si

The procedure to compute the DOS is the following:

- Use pw.x to calculate the density (calculation = ’scf’)
- Use pw.x to calculate the electronic eigenvalues on more k-point grids (calculation = ’nscf’)
- Use dos.x to compute the total DOS, will be produced in a data-file.

Important: we must keep the prefix same outdir and prefix for “nscf” and “scf” pw.x calculations and for the dos.x calculation.

Input file si.scf.in

```
&CONTROL
   calculation='scf'
   restart_mode='from_scratch',
   prefix = 'Si',
   pseudo_dir = '../../pseudo'
   outdir='./tmp'
/

&SYSTEM
   ibrav = 2,
   celldm(1) = 10.262,
   nat =  2,
   ntyp = 1,
   ecutwfc = 20.0,
   ecutrho = 200.0,
/

&ELECTRONS
/

ATOMIC_SPECIES
   Si  28.086  Si.pbe-rrkj.UPF

ATOMIC_POSITIONS alat
   Si 0.00 0.00 0.00
   Si 0.25 0.25 0.25

K_POINTS automatic
   6 6 6   1 1 1
```

Input file si.scf.in

```
&CONTROL
   calculation='nscf'
   restart_mode='from_scratch',
   prefix = 'Si',
   pseudo_dir = '../../pseudo'
   outdir='./tmp'
/

&SYSTEM
   ibrav = 2,
   celldm(1) = 10.262,
   nat =  2,
   ntyp = 1,
   ecutwfc = 20.0,
   ecutrho = 200.0,
   nbnd=-8
   occupations = 'tetrahedra
/

&ELECTRONS
/

ATOMIC_SPECIES
   Si  28.086  Si.pbe-rrkj.UPF

ATOMIC_POSITIONS alat
   Si 0.00 0.00 0.00
   Si 0.25 0.25 0.25

K_POINTS automatic
   12 12 12   1 1 1
```

to calculate unoccupied bands, look at number of Kohn-Sham states In the si.scf.out.

Input file si.dos.in

```
&dos

outdir = ’./tmp’,
prefix=’Si’,
fildos=’si.dos’,
/
```

The same as in the scf and nscf input

Data file si.dos

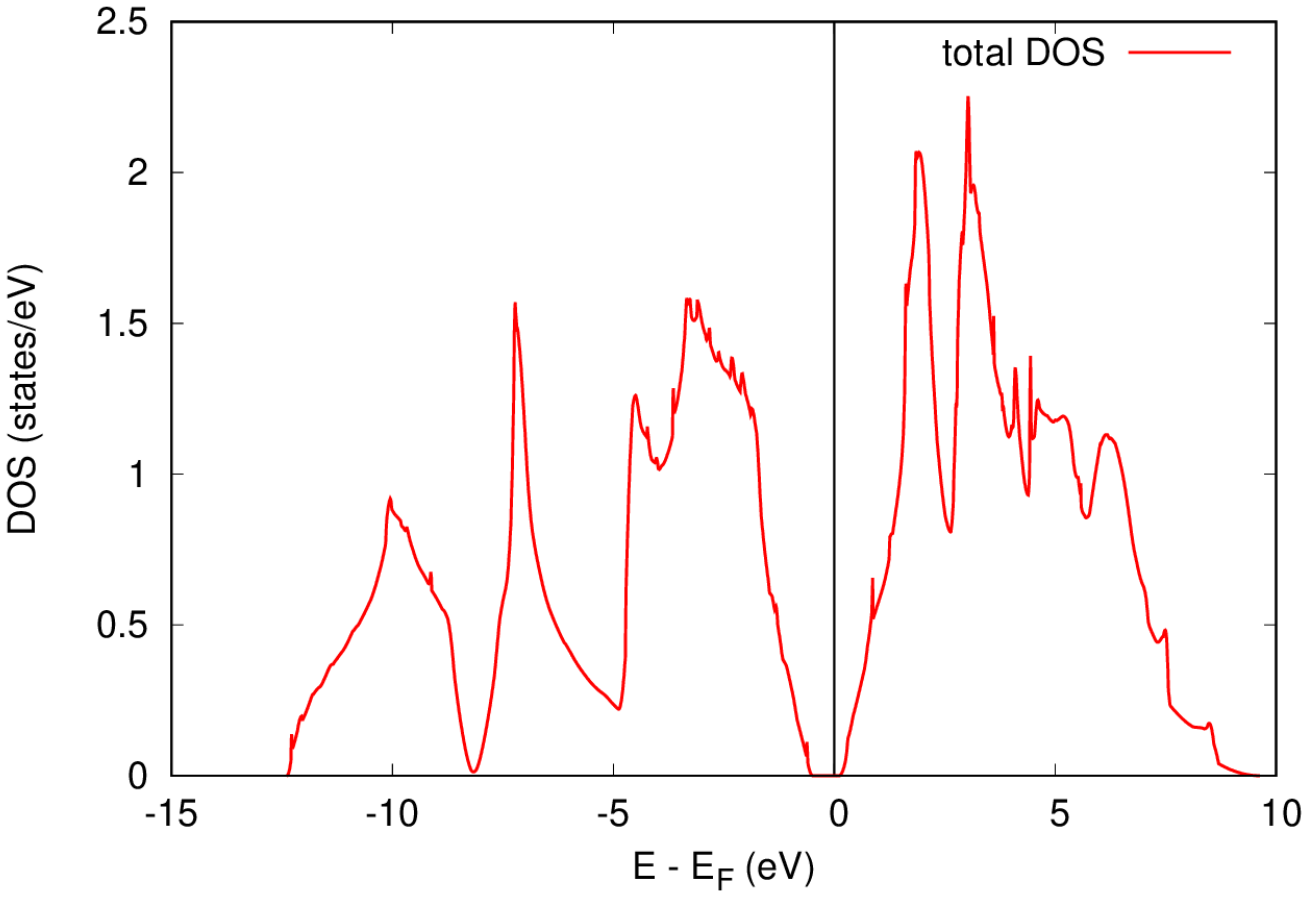
```
#  E (eV)   dos(E)     Int dos(E) EFermi =    6.724 eV
  -5.667  0.0000E+00  0.0000E+00
  -5.657  0.5876E-03  0.1959E-05
  -5.647  0.2351E-02  0.1567E-04
  -5.637  0.5289E-02  0.5289E-04
  -5.627  0.9402E-02  0.1254E-03
  -5.617  0.1469E-01  0.2448E-03
  -5.607  0.2115E-01  0.4231E-03
  -5.597  0.2879E-01  0.6719E-03
  -5.587  0.3761E-01  0.1003E-02
  -5.577  0.4760E-01  0.1428E-02
  -5.567  0.1183E+00  0.2264E-02
  -5.557  0.1378E+00  0.3596E-02
```

Gnuplot script: plot_dos.gp

DOS is shifted such that the Fermi energy corresponds to zero of energy. Use the Fermi energy from the file si.nscf.out.

After running the script, visualize the file DOS.eps using evince or okular command.

Density of State (DOS) for Si



Projected Density of State (PDOS) for Si

The PDOS allows us to get the contribution from each atom in the cell and/or each of their orbital contributions. We can achieve it using projwfc.x.

The calculation steps are similar, the differences are in the input files for dos and pdos.

Input file si.pdos.in

```
&PROJWFC

outdir = ’./tmp’,
prefix=’Si’,
filpdos=’si.dos’,
/
```

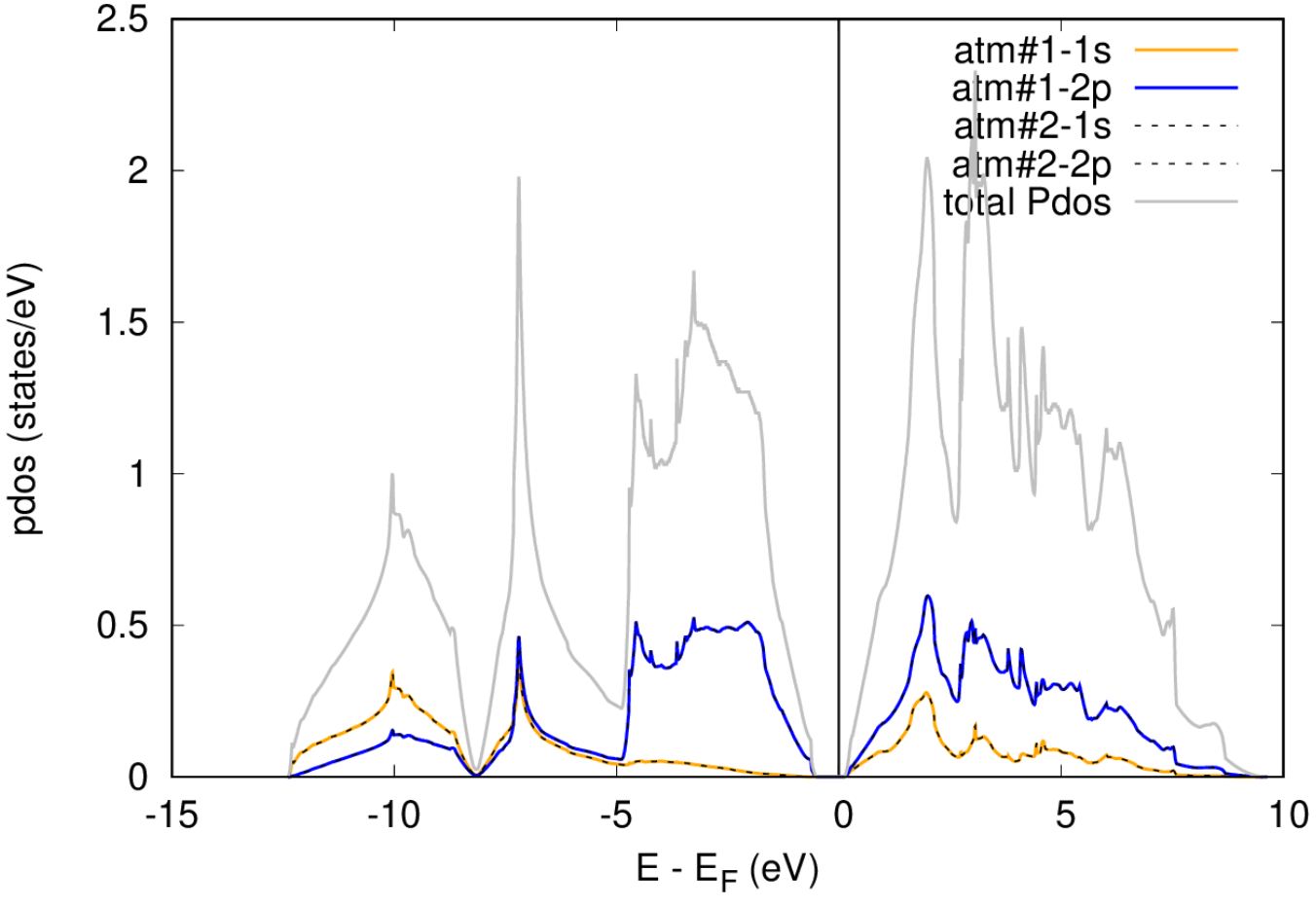
See the difference w.r.t dos.x

run the PDOS calculation, execute ‘projwfc.x’ as:

$ projwfc.x < si.pdos.in | tee si.pdos.out

Output: Have a look at si.pdos.out file. the PDOS values are written in the file {filpdos}.pdos_atm#N(Si)_wfc#M(l), where N, M, and l are the atom number, wfc number, and atomic wavefunction read from pseudopotential file.

Projected Density of State (PDOS) for Si



To Plot the data: use xFroggie or gnuplot script plot_pdos.gp on your directory

Exercise 3

Band structure for Si

Band Structure of Silicon

The scheme to compute the band structure is the following:

1. SCF pw.x calculation (calculation = 'scf')

2. "bands"-type non-SCF pw.x calculation (fixed-potential) with:

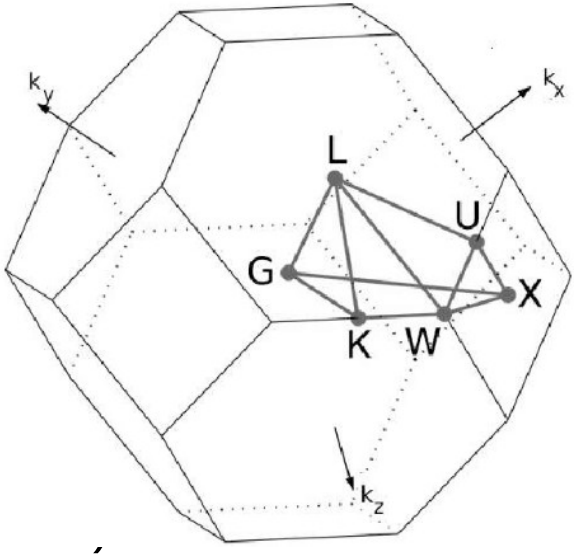
   - calculation = 'bands'
   - number of bands (variable nbnd) is specified
   - a suitable path of k-points is specified in K-POINTS card. The k-point path must be continuous in k-space.

3. bands.x calculation, which, among others, produces data-files for the plot.

Important: we must keep the prefix same outdir and prefix for "nscf" and "scf" pw.x calculations and for the bands.x calculation.

k-path for Band Structure calculation

The k-path in BZ for fcc crystal can be specified using xcrysden as follow:



1. Open the file si.scf.in with xcrysden.

2. Select: Tools → k-path selection

3. Select the path by clicking on a sequence of high symmetry points: W – Γ – X – W – L – Γ

4. Specify how many k points to be calculated (Ex:

5. save the k-path to file. (.pwscf extension is required for formatting the file for pw.x)

Input file si.bands.in

```
 &CONTROL
    calculation='bands'
    restart_mode='from_scratch',
    prefix = 'Si',
    pseudo_dir = '../../pseudo'
    outdir='./tmp'
    verbosity = 'high'
 /
&SYSTEM
    ibrav = 2,
    celldm(1) = 10.262,
    nat =  2,
    ntyp = 1,
    ecutwfc = 12.0,
    ecutrho = 200.0,
    nbnd=8
     /
 &ELECTRONS
 /
ATOMIC_SPECIES
   Si  28.086  Si.pbe-rrkj.UPF

ATOMIC_POSITIONS alat
   Si 0.00 0.00 0.00
   Si 0.25 0.25 0.25
K_POINTS {crystal_b}
  6
  0.50  0.25  0.75  30   !W
  0.00  0.00  0.00  30   !G
  0.50  0.00  0.50  30   !X
  0.50  0.25  0.75  30   !W
  0.50  0.50  0.50  30   !L
  0.00  0.00  0.00  30   !G
```

An input file for the data post processing program bands.x. This program print the eigenvalues to the file specified in ”filband”

```
&BANDS

outdir = ’./tmp’,
prefix=’Si’,
filband=’si_bands’,
/
```

The same as in the scf and bands input

Contains data for bands

Run the postprocessing calculation (bands.x):

```
$ bands.x < bands.in | tee  bands.out
```

bands data are stored in three files with different formats, among which we have bands.dat.gnu (can be plotted with gnuplot) and bands.dat can be visualized using the program plotband.x.

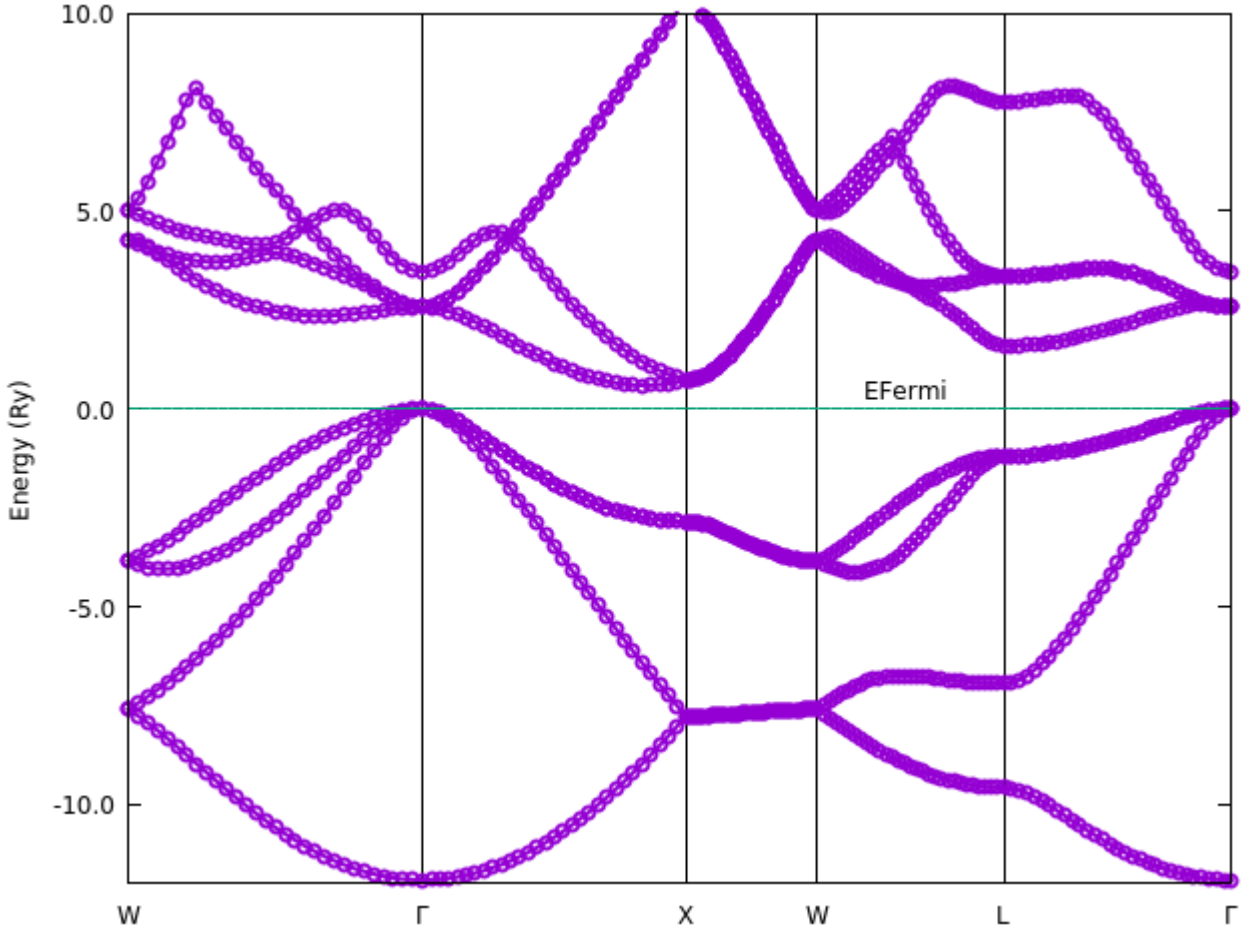
The program plotband.x can be to obtain a postscript file of the band structure and a set of data files that can be plotted with xmgr

plotband.x prompts for terminal input:

$ plotband.x

```
     Input file > bands.dat
Reading    8 bands at    151 k-points
Range:   -5.6670   16.3610eV  Emin, Emax, [firstk, lastk] > -5.6670 16.3610
high-symmetry point: -1.0000 0.5000 0.0000   x coordinate   0.0000
high-symmetry point:  0.0000 0.0000 0.0000   x coordinate   1.1180
high-symmetry point: -1.0000 0.0000 0.0000   x coordinate   2.1180
high-symmetry point: -1.0000 0.5000 0.0000   x coordinate   2.6180
high-symmetry point: -0.5000 0.5000 0.5000   x coordinate   3.3251
high-symmetry point:  0.0000 0.0000 0.0000   x coordinate   4.1912
output file (gnuplot/xmgr) > Si.bands.dat
bands in gnuplot/xmgr format written to file Si.bands.dat


output file (ps) > bands.ps
Efermi > 6.239
deltaE, reference E (for tics) 2, 6.239
bands in PostScript format written to file bands.ps
```

Band Structure of Silicon



Try to plot with xFroggie: https://xfroggie.com/

Exercise 2

A metallic example: Aluminum

A metallic example: Aluminum

Let us consider Aluminum, in the fcc phase. It is a metal, only valence bands and a few k-points will not suffice.

- move to the Day3/example2.Al directory
- read the pw.x input file al.scf.in
- notice the presence of new variables: occupations, smearing, degauss;
- run pw.x as:

```
$ pw.x < al.scf.in | tee al.scf.out
```

- in the output file notice that

  – the number of bands (Kohn-Sham states) is automatically set to a value larger than the number of electrons divided by 2

  – the Fermi energy is computed.

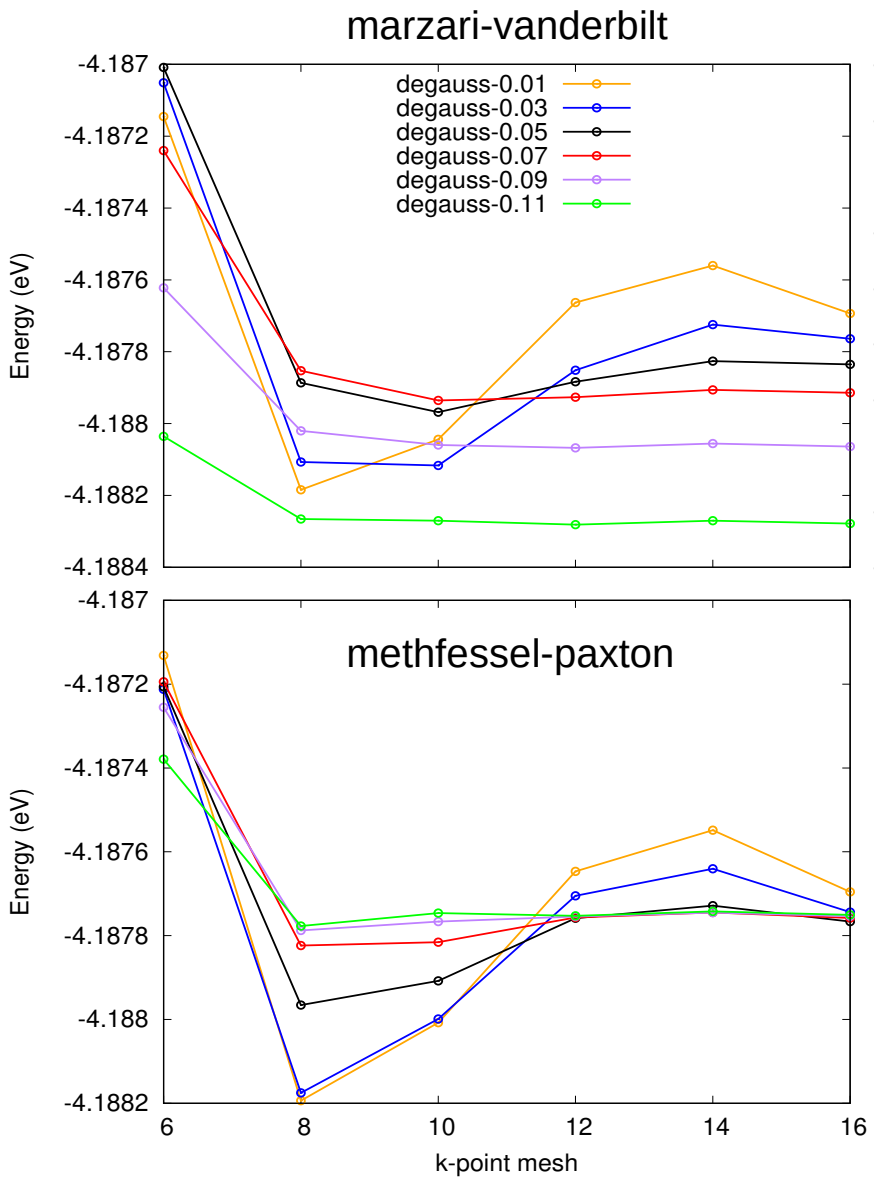
Convergence with respect to k-points, degauss, and smearing

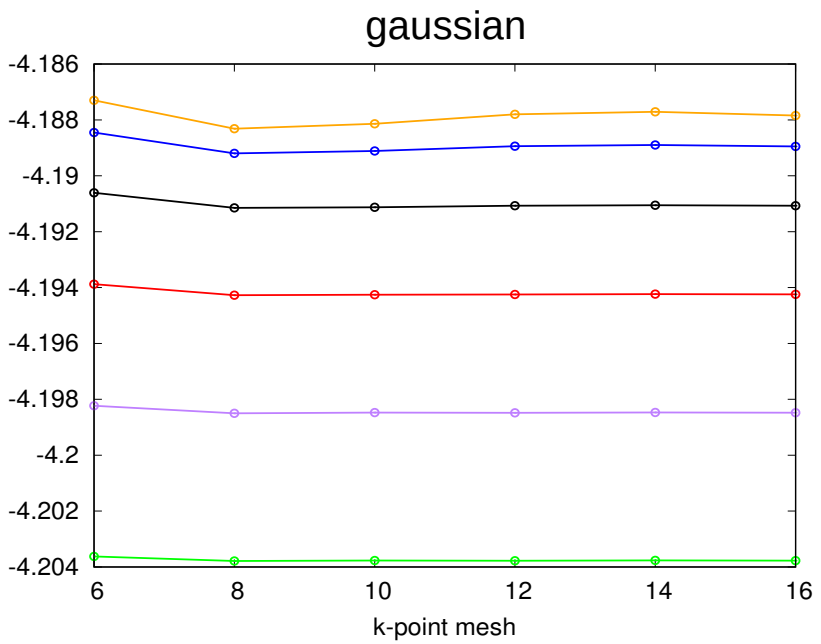
In this exercise we will compute the total energy for this fcc Al with the following values for degauss, k-point meshes and different smearing variables:

- degauss variable, in range from 0.01 to 0.11
- k-points: N×N×N (1×1×1) with N=6,8,10, …, 16
- smearing variable, possible values: 'gauss' (or 'g'), 'marzari-vanderbilt' (or 'm-v'), 'methfessel-paxton' (or 'm-p')
- run pw.x for run-al file at different directories in ex1-conv:

Plot the convergence of the total energy with respect to the number of k-points for the different smearing values.

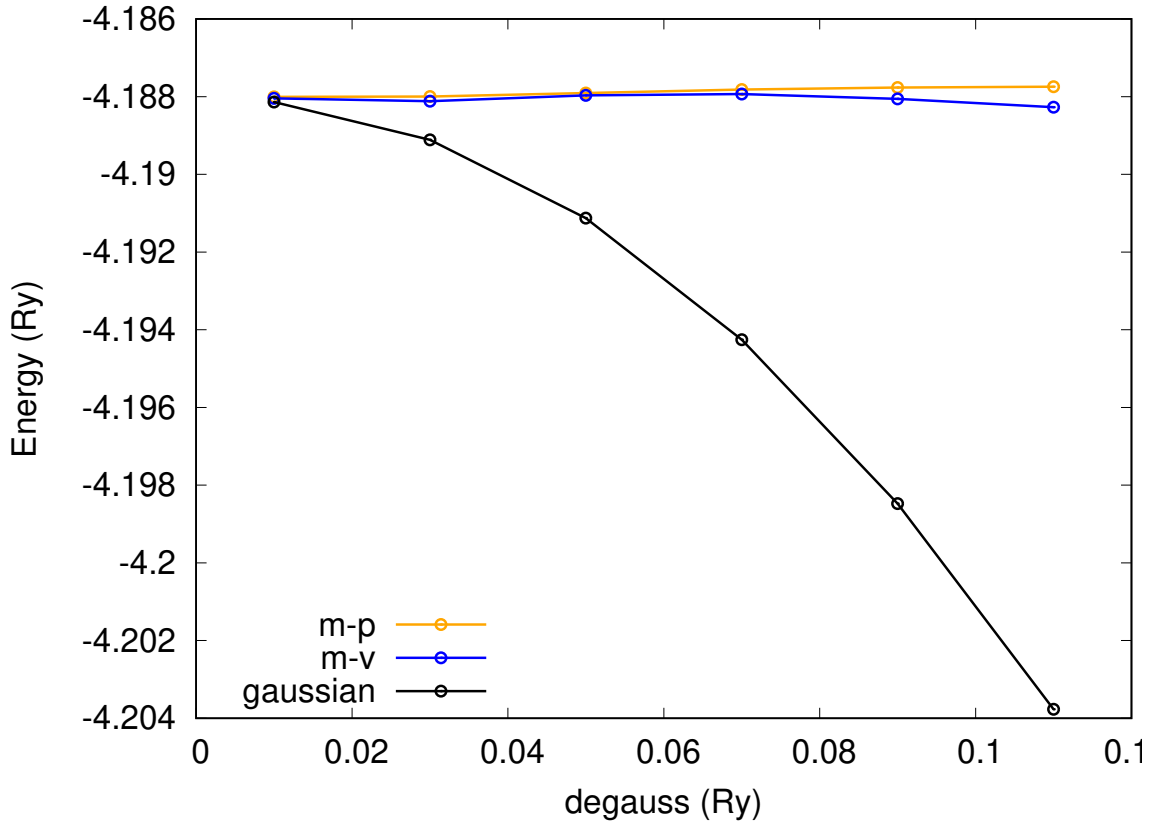
Convergence with respect to k-points, degauss, and smearing





Convergence with respect to k-points, degauss, and smearing

Choose a suitable k-point mesh and plot the total energy as a function of degauss for Marzari-Vanderbilt and Gaussean smearing.

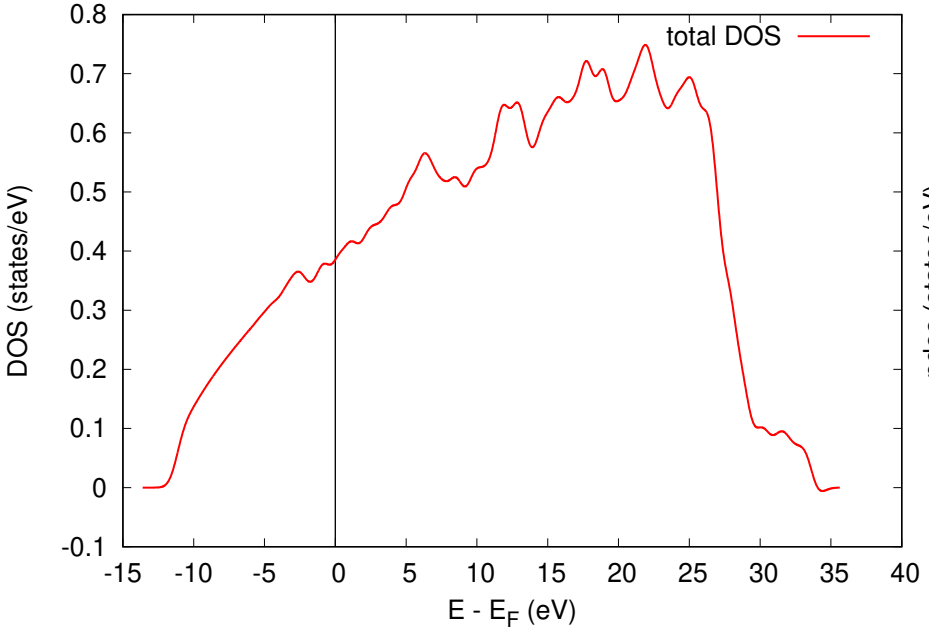


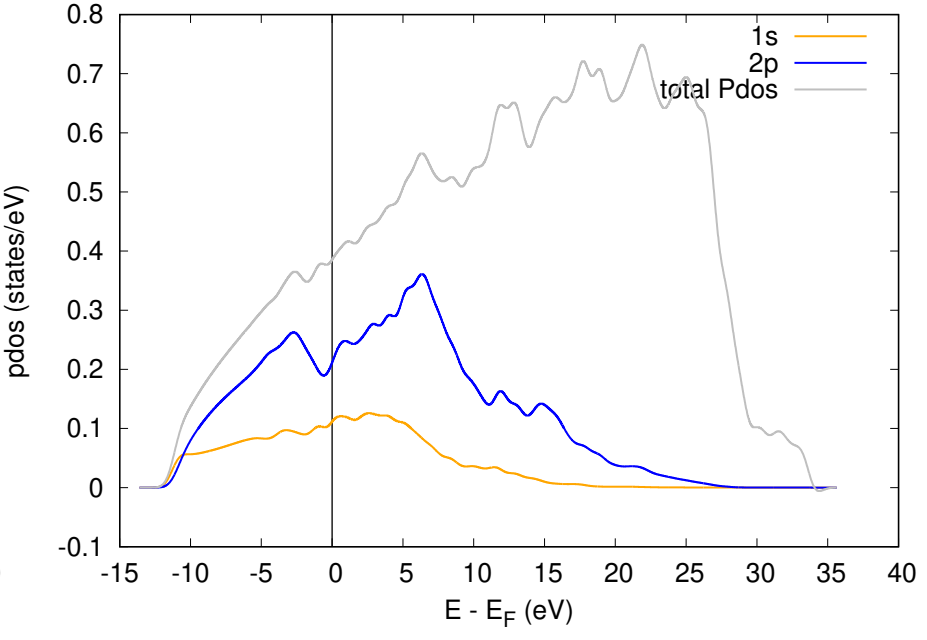
Density of state (DOS) and Projected DOS (PDOS) for Al

Use the procedure given in the instructions for example1:

- move to Day3/example2.Al/ex2.dos/ directory and
- Self-consistent calculation (pw.x)
- Select a finer mesh of k points by doing Non self-consistent calculation (pw.x)
- a dos.x calculation to calculate DOS
- a projwfc.x calculation to calculate PDOS projected to atomic states

Density of state (DOS) and Projected DOS (PDOS) for Al





Band structure for Al

Use the procedure given in the instructions for example1:

- move to Day3/example2.Al/ex2.bands/ directory and
- Self-consistent calculation (pw.x)
- Select a k-point path (i.e. by using XCrysDen, ...) and do “bands”-type non-SCF pw.x calculation
- a bands.x calculation print the eigenvalues and plot them using gnuplot or
- Use the program plotband.x

Band structure for Al