

Excercise sheet 3

General remark: If you get an error that a libmkl or similar cannot be found, run the command

```
module load intel64
```

in putty (or the terminal) while being logged in to the HPC. This should load the MKL linear algebra libraries, which pp.x and similar programs from Quantum Espresso depend on.

The topic of this week's exercise sheet will be the simulation of the electronic structure from DFT. Some additional programs that are necessary for this week's exercises but were missing in qe-hpc.tar.gz have been added to StudOn (qe-hpc-additional-files.tar.gz). I noticed (a bit too late) that the file permissions are not retained if one opens the archive in Windows (e.g. in WinRAR) and drags the archived files onto the HPC. It also does not work if the archive is in .zip format. For this reason, put the file onto the bin/qe folder in your home folder, where the other QE files are, and uncompress the archive with the command

```
tar -xvzf qe-hpc-additional-files.tar.gz
```

The following figures show typical input files for density-of-states calculations

```
&control
  calculation = 'nscf'
  prefix='Cu',
  pseudo_dir = './',
  outdir='TMP_DIR/'
/
&system
 ibrav=0, nat= 1, ntyp= 1,ecutwfc =90.0, input_dft = 'pbesol',
occupations='smearing', smearing='cold', degauss=0.01,
  nbnd=50
/
&electrons
  conv_thr = 1.0d-10
/
ATOMIC_SPECIES
Cu 63.55 Cu.nc.UPF
ATOMIC_POSITIONS (crystal)
Cu 0.00000000 0.00000000 0.00000000
K_POINTS {automatic}
25 25 25 0 0 0
CELL_PARAMETERS (angstrom)
-0.000000000 1.793927308 1.793927308
1.793927308 -0.000000000 1.793927308
1.793927308 1.793927308 -0.000000000
```

← ,nscf: non-self-consistent calculation, weights for each k-point are taken into account

← Number of bands that are calculated at each k-point.

← Set of k-points that are calculated. For DOS, this is usually a regular k-point grid

And for bandstructure calculations

```

&control
  calculation = 'bands'
  restart_mode = 'from_scratch',
  prefix = 'Si',
  pseudo_dir = './',
  outdir = 'TMP_DIR/',
/
&system
 ibrav=0, nat= 2, ntyp= 1,
  ecutwfc =50.0, input_dft = 'pbe', nbnd=20
/
&electrons
  conv_thr = 1.0d-8
/
ATOMIC_SPECIES
Si 14.0 Si.UPF
ATOMIC_POSITIONS (crystal)
Si 0.00 0.00 0.00
Si 0.25 0.25 0.25
K_POINTS (crystal_b)
3
0.500 0.00 0.500 15
0.000 0.00 0.000 15
0.500 0.00 0.000 1
CELL_PARAMETERS (angstrom)
-0.0000000000 2.737747197 2.737747197
2.737747197 2.737747197 -0.0000000000
2.737747197 -0.0000000000 2.737747197

```

← ,bands': non-self-consistent calculation, weights for each k-point are not taken into account

← Convergence threshold for the electron *eigenvalues* that are calculated

← Definition of k-points that are to be calculated for the bandstructure plot

Here as well, the input files have the features of the SCF calculations from the last two exercises. The difference is, however, that these calculations using the input files above will be non-selfconsistent and thus require an existing converged electron density, i.e. have to be performed after a preceding SCF calculation.

In general, the only two significant differences compared to the input file for a SCF calculation are the calculation type as defined by the calculation keyword, and potentially the definition of the k-points used in the calculation.

- `calculation='nscf'` tells Quantum Espresso to perform a non-selfconsistent calculation using the parameters in the input file. For this, quantum espresso will attempt to read the converged electron density, which was stored in the folder defined by `outdir` by a previous SCF run. If no density is found, the calculation will be aborted with an error message.

A related run type is `calculation='bands'`. The only difference between 'bands' and 'nscf' appears to be that 'nscf' considers and stores the weights of the k-points used in the calculation (useful for, e.g. DOS calculations), while 'bands' does not and implicitly uses the same weight for all k-points. Thus, DOS calculations should be performed with 'nscf', while calculations of the bandstructure can be done with both 'nscf' and 'bands'

- As for SCF calculations, the KPOINTS block is used to define the k-points used for the calculation. For SCF calculations, one pretty much always uses an automatically generated uniform Monkhorst-Pack grid. This is also a reasonable choice for DOS calculations, where we are interested in an integral over the whole Brillouin zone.

Another possibility for both SCF and non-selfconsistent calculations is giving an explicit list of k-points (and potentially weights) that are to be considered. The latter usually is too tedious to make much sense for SCF calculations (unless one wants to use, e.g., Baldereschi or Chadi-Cohen special points.), while it could be used to calculate a small, discrete set of k-points of interest using an NSCF calculation.

For bandstructure calculations, QE offers the possibility of automatically generating k-points along a path between points defined in the KPOINTS block. This useful functionality is invoked by `KPOINTS (crystal_b)` and follows the following format:

```

KPOINTS (crystal_b)
Nk

```

```

X1(1) X1(2) X1(3) N1
X2(1) X2(2) X2(3) N2
...
Xk(1) Xk(2) Xk(3) 1

```

This generates k-points along a path with N_k-1 segments between the N_k explicitly given points. Here, $X1(1), X1(2), X1(3)$ are the coordinates in units of reciprocal lattice vectors of the starting point. QE then generates $N1$ points between $X1$ and the point $X2$, then $N2$ k-points between $X2$ and $X3$ etc.

Other formats exist as well and are described in https://www.quantum-espresso.org/Doc/INPUT_PW.html

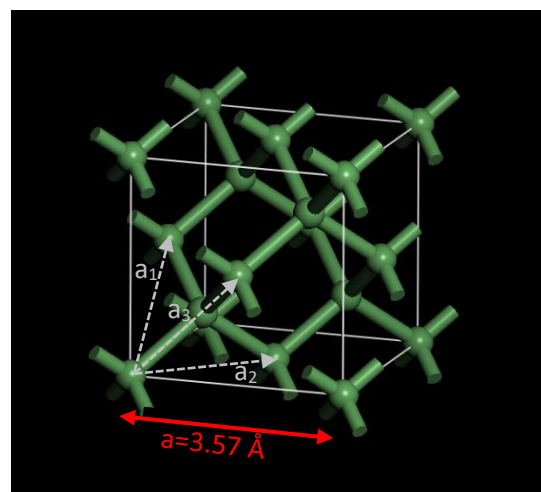
- Through the nscf/bands calculation, quantum espresso will calculate the band energies at the k-points defined in the KPOINTS block. The number of bands can be freely chosen using the keyword nbnd¹. QE will then calculate the nbnd eigenvalues of the Hamiltonian with the lowest energy. The same keyword can, of course, also be used in SCF calculations, but has to be at least as large as the number of occupied bands in the system (the calculation would not make much sense otherwise).
- As for the SCF calculations, the criterion for convergence is defined through conv_thr. In variance to SCF calculations, however, does conv_thr here (for nscf/bands calculations) not define the maximum allowed change in total energy for convergence, but the maximum allowed change of the single-particle eigenvalues between two steps in the iterative diagonalization procedure. While it is explicitly defined in the examples above, the default value defined by Quantum Espresso is also quite reasonable and one does not necessarily have to give a value for conv_thr.

Exercise 1: Electronic bandstructure of diamond

In this exercise, we will look at the electronic band structure of diamond, both from the view of local density functional theory and hybrid functionals.

- First relax the geometry of the system. Diamond is a zincblende material with the same structure as silicon, but with a cubic lattice constant of 3.57 Å. Use PBEsol as exchange-correlation functional to make sure that the optimized lattice constants is close to experiments.

Use a shifted uniform sampling of 6x6x6 k-points and a cutoff energy of 80 Ry (or do your



¹ nbnd cannot be larger than the number of plane waves used for the basis set, as this determines the size of the Kohn-Sham Hamiltonian. This is, however, never really a problem, as this would mean calculating to tens of thousands of bands, which one usually is not interested in doing (it also takes extremely long).

own convergence tests). A reference input file (C.in) is supplied in the folder "Exercise1/a" of .zip file for this exercise.

- (b) Based on the geometry obtained in (b), we will now calculate the electronic band structure of diamond within the PBEsol approximation along a high-symmetry path in the Brillouin zone. As laid out in the lecture, this means first obtaining the electron density from an SCF calculation and then perform the non-selfconsistent calculation on top of that.

A crucial information that we have to include for the non-selfconsistent calculation is of course the path that we want to plot the electronic bandstructure along. This path, the order of path segments, etc., are sufficiently arbitrary and depend a bit on one's preferences and areas of the Brillouin zone we are interested in. The input file includes a possible path that visits several high-symmetry points and is not that long, but you can play around with it, if you are interested. As the bandstructure calculation is quite fast, 50 k-points per path segment are calculated. This can also be increased or lowered, the calculation time scales linearly with the number of k-points.

After the 'bands' calculation finished successfully, plot the calculated bandstructure. For this, you have to first extract the eigenvalues to something that is plotable. Quantum Espresso contains the program bands.x, which does exactly that. It can be invoked by

```
$PATH_TO_BANDS.X/bands.x -i bands.in
```

where \$PATH_TO_BANDS.X is the folder, where bands.x is located and bands.in is a simple input file, which is included in the .zip file for this exercise. bands.x will write the bands into a format that can be plotted by the program plotbands.x that is part of Quantum Espresso (bands.out). It will also write a file (bands.out.gnu) that can be plotted by gnuplot or similar programs.

Note that the band energies are not shifted and the Fermi energy of the system is the value that is reported in the output of the SCF calculation. For ease of interpretation, you could opt to shift the bands such that either the Fermi energy or the valence band maximum is at $E=0$.

According to experiments, diamond is an insulator with an indirect fundamental band gap of 5.4 eV. Does the PBEsol functional (or alternatively functionals such as LDA or PBE) reproduce these experimental observations well?

- (c) Now, we will treat diamond using a hybrid functional ansatz for describing the exchange-correlation interaction. Quantum Espresso has several such hybrid functional implemented, we will here opt for the HSE06 functional, which has been shown to work very well for predicting electronic band gaps. Use of this hybrid functional is achieved by simply defining `input_dft='hse'` in the input file. However, as described in the lecture, non-selfconsistent calculations and calculations using non-uniform k-point samplings are not implemented in Quantum Espresso. We will thus take a different route here:

In this part, we will do an SCF calculation with a number of unoccupied bands. As hybrid functional calculations are much more expensive than PBEsol calculations, we will also reduce

the k-point sampling to a 5x5x5 grid, which is still sufficient to give good results for the electronic structure.

Run the calculation. Even on the HPC, this will take a few minutes. Reduce the cutoff energy, if you do the calculations on your home computer and want them to be faster.

Examine the output file. As laid out in the lecture, Quantum Espresso first does a (very fast) SCF calculation using the LDA functional. After this, the hybrid functional (HSE06) is switched on and the SCF procedure repeated. First, Quantum Espresso will use the previously calculated LDA orbitals as the φ_{jq} orbitals entering the Equation for the Hartree-Fock-type contribution to the total energy given in Lecture 9 and minimize the Kohn-Sham (total) energy through a number of SCF loops. It then updates the φ_{jq} orbitals entering the HF-type contributions by the new 'best' single-particle orbitals and then does another SCF loop - this continues until the change in total energy is small enough. Note that only the first SCF step after each 'restart' takes long to finish (because the HF-type contribution has to be evaluated), while the following steps take only as long as those from the initial LDA calculation. This is by virtue of smart mathematical tricks proposed in *J. Chem. Theory Comput.* 2016, 12, 2242 (2016), the so-called Adaptively Compressed Exchange (ACE) method.

- (d) Obviously, the SCF calculation in (c) does not give us the electronic bandstructure. What we will do now is using Wannier interpolation to obtain the bandstructure from the eigenvalues calculated in (d) on a uniform Monkhorst-Pack grid. We will use the code Wannier90 to do this. A version that runs on the HPC is supplied in the .zip file. Wannier90 is also installed in Quantum Mobile and can be downloaded as a package in Ubuntu and similar Linux systems. Unfortunately, there is no Windows-compatible version I am aware of.

To be able to use Wannier90, we first need to expand the symmetry-reduced grid to a 'full' non-reduced grid. This can be done conveniently using `open_grid.x`, which is part of Quantum Espresso. This requires a simple input file, which is supplied in the .zip file.

Expand the k-point grid by running

```
$PATH_TO_OPENGRID.X/opengrid.x -i opengrid.in
```

(or similar on Windows) This will add a folder `C_open.save` to `TMP_DIR`, which contains the desired orbitals and eigenvalues on the full grid.

We then need to extract the necessary data, i.e. single-particle energies and orbital overlaps, from the DFT calculation.

Examine the supplied wannier90 input file, `C.win`. Here, the geometry of our system (it has to be exactly the same as the atomic positions and lattice vectors used in part (c)) and the used k-point sampling are defined, the latter both in terms of a `NXMXL` values and using the k-point coordinates of the full grid. We also tell the system, how many Wannier functions we want (10) and how many bands we obtained from the DFT calculation (12). We need to have at least as many DFT bands available as we want Wannier functions. The keyword `exclude_bands` indicates that the top two bands will be neglected in the generation of the Wannier functions. This will give us 10 bands that are interpolated along the k-point path defined in the input file (block `kpoint_path`).

We also can (and should) give an initial guess for the shape of the resulting Wannier functions. Typically, atomic orbitals or LCAOs are good starting guesses. As the interatomic bonds in Ge are of sp^3 character, using this as initial guess might be even better. We have two Ge atoms in the system, each of which contributes 4 sp^3 orbitals, 2 occupied and two unoccupied. This gives is altogether 8 bands. For the remaining two bands, we use a random distribution as initial guess.

The following three steps can be done automatically by using the submission script provided in the .zip file. Make sure to update the paths in the submission script to point to where your pw2wannier90.x and wannier90.x are. If you run the following commands by hand on the HPC, make sure that the intel64 module is loaded.

We first need to do a preprocessing run, where Wannier90 collects the info it has and what it needs from the input file. Run

```
$PATH_TO_WANNIER90.X/wannier90.x -pp C
```

to do this. If everything goes well, it will write a .nnkp file.

Now we extract the necessary DFT data. Quantum Espresso offers the program pw2wannier90.x for this purpose. Again, an input file is provided in the .zip file.

Run

```
$PATH_TO_PW2WANNIER90.X/pw2wannier90.x -i pw2wan.in
```

After a short time, this will have finished writing a number of files. The C.eig file contains all the eigenvalues obtained in the DFT calculation.

Finally, with all the preparations done, we can interpolate. Run

```
$PATH_TO_WANNIER90.X/wannier90.x C
```

This might take a while.

After it is finished, you should have a file C_bands.dat. which contains the interpolated bandstructure. Plot it with gnuplot or a similar program. Does HSE06 work better than PBEsol for the description of the electronic structure of diamond?

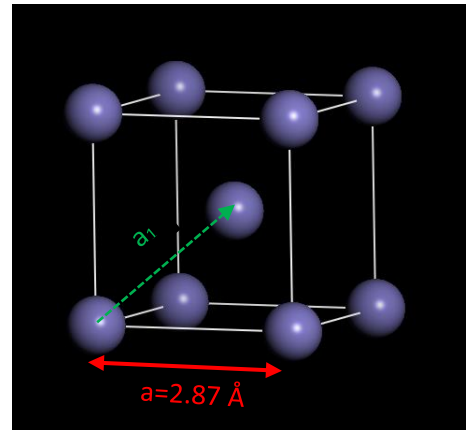
Exercise 2: Electronic bandstructure and DOS of iron

In this exercise, we will calculate the band structure and density-of-states of bcc-iron and extract the Fermi surface.

- (a) In the first step, as always, do a convergence test for the k-point sampling that should be used to ensure converged and meaningful results. As Fe is (of course) a metal, use cold smearing of the band occupations with a smearing width of 0.01 Ry. It is sufficient to test shifted $N \times N \times N$ grids, with N even and between 6 and 16.

The atomic positions and lattice vectors of Fe can be found in the input file Fe.in in the “a” folder (or you can derive them easily yourself from the picture). You can also do a convergence test for the cutoff energy, but 80 Ry should be sufficient.

In any case, examine the input file. You will see another difference compared to the calculations from the previous exercises: Iron is ferromagnetic, which we should consider in our calculations as well. To do this, we will first tell Quantum Espresso to do away with the usual assumption that the system is not spin polarized. This is done by setting `nspin=2`, which tells Quantum Espresso to internally use two sets of k-points: one set for spin-up electrons and another set for spin-down electrons². At each of these k-points, each band can only be occupied by a single electron.



However, if we run the calculations like this, the system will remain in a non-magnetic state. For this reason, we have to give an initial magnetization of the atoms to break the symmetry between spin-up and -down electrons. This initialization is done with the keyword `starting_magnetization`, where the one 1 in brackets stands for the first (and in this case only) element in the system. If we had more elements, we could give different initial polarizations for each element, but not for the individual atoms (at least not without some trick). In the input file, each iron atom in the whole system has an initial spin-up polarization of 1, i.e. we initialize a ferromagnetic state. During the SCF procedure, this spin-polarization is allowed to relax or change, such that we might end up with a different spin-polarization or maybe even with a nonmagnetic state after the electron density is converged.

You can see this happening when in the output files of your convergence test calculations: at each SCF step, QE reports the total (i.e. the difference between spin-up and spin-down electrons) and absolute magnetization after the estimated SCF accuracy. You will see that indeed, the spin-polarization of the iron atoms changes from 1.0 at the beginning to about 2.4 mag/cell at the end.

Notice also that after the SCF procedure is finished, Quantum Espresso now reports two sets of bands, one for spin-up and one for spin-down electrons. There are now more spin-up bands occupied than spin-down bands, as you can see from comparing the reported eigenvalues with the Fermi energy given at the end of the file.

In this case, the calculations used the PBEsol functional, but the results should be similar for LDA or PBE.

- (b) With the convergence test done, use the converged parameters to perform a geometry optimization for a spin-polarized system until the interatomic forces are smaller than 0.0004 Ry/a0 and the cell pressure smaller than 0.05. Also, you can now help QE a bit by initializing the atomic spin-polarization to something close to the final value that the SCF calculations in (a) arrived at.

² This is at least the case for Quantum Espresso: Other codes might use one set of k-points but two sets of bands instead.

- (c) Based on the geometry obtained in (b), we will now calculate the electronic band structure along a high-symmetry path in the Brillouin zone. As in Exercise 1, the path chosen visits several high-symmetry points and can be changed, shortened or extended to your liking. Set the number of included bands for the non-selfconsistent calculation such that you include all bands that are in a range of 10 eV above the Fermi energy.

After the calculation is finished, plot the calculated band structure. Here, we can choose the spin component to be plotted by the keyword `spin_component`, as shown in the supplied two input file for `bands.x`. Using these input files, you will obtain a separate band structure for each spin component. Compare the band structures in a window of 10 eV around the Fermi energy. Here as well, it is useful to shift the entire bandstructure such that the location of the Fermi energy is at 0 eV.

- (d) In the next step, we will calculate the density-of-states of iron, using a non-selfconsistent calculation of the electronic structure on a regular 25x25x25 k-point grid. Again, perform an SCF calculation with the converged parameters and geometry from (a) and (b) and do the `nscf` calculation on top. Include the same amount of bands as in (c).

After the calculation finished, use the `dos.x` code to calculate the density-of-states from the calculated band energies. The corresponding input file `dos.in` is included in the .zip file. Apart from the obligatory definition of `prefix` and `outdir`, we have two additional keywords: `bz_sum` defines how `dos.x` is supposed to calculate the density-of-states. We here opt for using a superpositions of gaussian peaks centered at each of the calculated eigenvalues. The Full width at half maximum of each of these peaks is defined by the keyword `degauss` and is set to 0.01 Ry here (You can also try a smaller one).

Running the post-processing calculation using the supplied input file will generate a file `Fe.dos`, which contains the calculated DOS and can be plotted with your favourite plotting program. In fact, it contains three different sets of data: one DOS for each of the spin-directions and the integrated DOS *intdos(E)*, which gives the number of electrons occupying the bands up to energy *E*. Like the electronic band structure, the calculated DOS is not shifted and the Fermi energy in the plot is at the value given in the header of `Fe.dos`. The plot becomes particularly insightful, when reversing the sign of the spin-down DOS and plotting both DOS together: it is then clearly visible that the spin-down DOS is essentially the spin-up DOS, but shifted to higher energies.

- (e) Now, we will calculate the DOS using the tetrahedron method. The input files look very similar to those from (d), but the lines defining the smearing are different, see the reference input file. We will use the optimized tetrahedron method here.

Repeat the steps from (e) for a 15x15x15 grid. The `dos.x` calculation can take a while in this case, it thus makes sense to restrict the energy range that is plotted to the most relevant range and use a version of `dos.x` that works on several CPUs in parallel in this case (see `dos.in` and `j.submit.sh_dospp`). If you are ready to wait for the calculation to finish, you can also do a calculation with a 25x25x25 grid.

- (f) Based on the DOS from (e) or (f), we can also extract and plot the Fermi surface of iron, i.e. the surface in reciprocal space that separates occupied and unoccupied states (or the positions in k-space, where the Fermi energy cuts through the bands).

The extraction is rather simple to do: copy the supplied input file `fs.in` to the folder, where you did the calculations for (d) or (e). Then run the `fs.x` code with `fs.in` as the input file. This will write a file `Fe.bsf`.

Under Linux, you can plot this file easily using the XCrysDen program.

Under Windows, my recommendation is using FermiSurfer, which you can download from <https://fermisurfer.osdn.jp/>. Using this program is simple enough: Download the program, right-click the generated `.bsf` files (potentially download them first from the HPC) and choose "Open with...", select `Fermisurfer.exe` (you probably have to find it manually) and open the file. This will open a window showing the plotted Fermi surface with a number of adjustment settings.

Exercise 3: Electronic structure of monolayer Al_2S_2

Here, we will have a look at the electronic properties of Al_2S_2 , a novel two-dimensional semiconductor material with a hexagonal structure and four atoms per unit cell.

- (a) Use the geometry supplied in the input file `Al2S2.in` to optimize the geometry on the PBEsol level of XC approximation.
- (b) Based on the obtained geometry, calculate then the electronic bandstructure of monolayer Al_2S_2 . As it is a 2D material, the Brillouin zone is (nearly) two-dimensional as well, and only need to consider high-symmetry paths within the plane spanned by the reciprocal lattice vectors that are parallel to the Al_2S_2 plane. Choose a path that connects the high-symmetry points Γ [(0.0 0.0 0.0)], K [(1/3 1/3 0.0)] and M [(0.5 0.0 0.0)]. Include 10 unoccupied bands for the bandstructure calculation so that we get a good look at the conduction bands as well.

Plot the obtained bandstructure using an appropriate input script for `bands.x`.

What is the value of the smallest (the fundamental) band gap within the PBEsol approximation? Where in the Brillouin zone are the valence band maximum and the conduction band minimum, respectively, located? Is Al_2S_2 a direct or an indirect semiconductor?

- (c) We will now try to understand, what kind of atomic orbitals contribute to the valence and conduction band edges. To make life easier, let us do that by studying the projected density-of-states for the atomic orbitals of Al and S. An alternative way would do a fat bands plot, but it is more complicated to convert the data to something visualizable in this case.

In this sense, do a nscf calculation with a dense grid of 30x30x1 k-points. Include 10 unoccupied bands for this calculation as well. Do not use the tetrahedron method here, as the program that does the projections to atomic orbitals, `projwfc.x`, does not work with it (at the moment).

After the calculation is done, examine the input file `projwfc.in`. Again, the input file is simple enough: it contains the obligatory prefix and `outdir`, so that `projwfc.x` can find the Quantum Espresso output data. It also contains a keyword that controls the peak width for the peak broadening procedure that `projwfc.x` uses. Additional keywords can be used to, for example, tell `projwfc.x` to write the output with k-resolution, i.e. output files for each k-point included in the calculation as opposed to simply integrating them to form PDOS datasets.

Run `projwfc.x` with the input file.

After a short time, it will write a number of output files, one file for projection of the bands onto the s-orbitals and p-orbitals of each of the four atoms in the unit cell. Apart from species and orbitals resolution, this also gives a kind of spatial resolution because we can assign a DOS contribution to each of the atoms in the system.

Now, sum the PDOS obtained for the s-orbitals of the two Al atoms and of the two S atoms, respectively. Do the same for the p-orbitals, obtaining four summed PDOS datasets.

You can do this summation either in a suitable software, such as Excel or Origin. Or you can use another postprocessing program from QE to do it, the `sumpdos.x` utility.

The command is

```
$PATH_TO_SUMPDOS.X/sumpdos.x file1 file2 file3 |tee outfile.dat
```

This will sum the files `file1`, `file2`, and `file3` and write the sum into the file `outfile.dat`. Of course, you can also sum more or less files than three. A little caveat: `sumpdos.x` has problems with the `#`, `(` and `)` characters in the filenames written by `projwfc.x`. You thus have to remove these characters from the names of all files that you want to sum. `Projwfc.x` also calculates, the total DOS; you can find it in the file `Al2S2.pdos_tot`.

Finally, plot the four PDOS datasets together with the total DOS. What states contribute to the valence and conduction band edges, respectively?

