MAX School on Advanced Materials and Molecular Modelling
with QUANTUM ESPRESSO

# QE-2021: Hands-on session – Day-7

## (Magnetism in transition metals )

Pietro Delugas, Alessandro Stroppa, Sukanya Ghosh,
Nandhakumar Velankanni, Subrahmanian Sappati, Lei Qiao

# QE-2021: Hands-on session – Day-7

Topics of Day-7 hands-on session:

1. Comparison of magnetic and non-magnetic solution for Fe (bcc) (`exercise1.ferromagnetic_vs_nonmagnetic/`)

2. Dependence of exchange splitting on the lattice constant. (`exercise2.volume_dependence`)

3. Comparison of *3d* occupations in transition metals (Fe, Co, Ni) (`exercise3`)

4. Plotting bands of magnetic material collinear and non collinear case. (`exercise4.bands`)

# How to run calculations remotely on the “hpc” HPC cluster

Several utility commands have been implemented specially for the QE-2021 school to aid at submitting jobs to HPC cluster(s). These are:

- **remote_mpirun** – this is like `mpirun`, but it automatically submits the calculation to a queuing system on the “hpc” HPC system. For example, a `pw.x` calculation can be submitted as:

  - `remote_mpirun pw.x -in pw.file.in`

  where `pw.file.in` is the name of the `pw.x` input file. **BEWARE:** stdin/stdout redirection does not work for `remote_mpirun`, hence you must use `-in` (or `-inp`) option (i.e., do note use “`<`” redirection operator). You do not need to specify the number of processors, because the default is set to `-np 20`.

  A different number of processors, say 8, can be requested as:

  - `NPROC=8 remote_mpirun pw.x -in pw.file.in`

- **remote_pwtk** – this automatically submits the PWTK script to queuing system on the “hpc” HPC system. Example:

  - `remote_pwtk script.pwtk`

  where `script.pwtk` is the name of the PWTK script.

- **remote_sbatch** – automatically submits the Unix-shell script to queuing system on the “hpc” HPC system. Examplee:

  - `remote_sbatch script.sh`

  where `script.sh` is the name of the Unix-shell script. When you run in the HPC cluster use increase the number of pools change the line:

  ```
  mpirun -np $NPROC $PW  < fe.scf.$latt.in -nk 2   >  fe.scf.$latt.out
  ```

  in

  ```
  mpirun -np $NPROC $PW  < fe.scf.$latt.in -nk 10  >  fe.scf.$latt.out
  ```

- **remote_squeue** – this checks the status of the job in the queuing system on the “hpc” HPC system. Example:

  - `remote_squeue`

  ```
       JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
    26098354       all submit.n    qe001  R       0:27      1 wn150
  ```

  Where status (ST) R stands for running and PD for pending.

- **hpc** – this makes `ssh` to “hpc” HPC login node, such that the user will be located in the same directory as used locally

- **rsync_to_hpc** – copies specified files to the “hpc” cluster to the same directory as is currently used locally. Example:

  - `rsync_to_hpc '*.in'`

This will copy all `*.in` files from local directory to the same directory on the “hpc” cluster, whereas:

  - `rsync_to_hpc .`

  will copy all the files in the current directory to the same directory on the “hpc” cluster.

- **rsync_from_hpc** – download the specified file from the “hpc” cluster from the same directory as is currently used locally. Example:

  - `rsync_from_hpc '*.out'`

  This will copy all `*.out` files from the “hpc” cluster to the local host (virtual machine).

# Exercise 1: Task 1

- Compare the magnetic and the non-magnetic solution for Fe (bcc)

- **Find the optimized lattice constant for non-magnetic Fe (bcc)**:

  1. go to `exercise1.ferromagnetic_vs_nonmagnetic/non_magnetic` :
  2. Use the scripts `job.sh` or `job.pwtk` to scan the lattice constants from $4.6\ \mathrm{Bohr}$ to $5.9\ \mathrm{Bohr}$.
  3. use `ev.x` to fit the data collected in `energies.dat` with the Murnaghan EOS and find the optimized lattice constant.

- **Find the optimized lattice constant for magnetic Fe (bcc)**

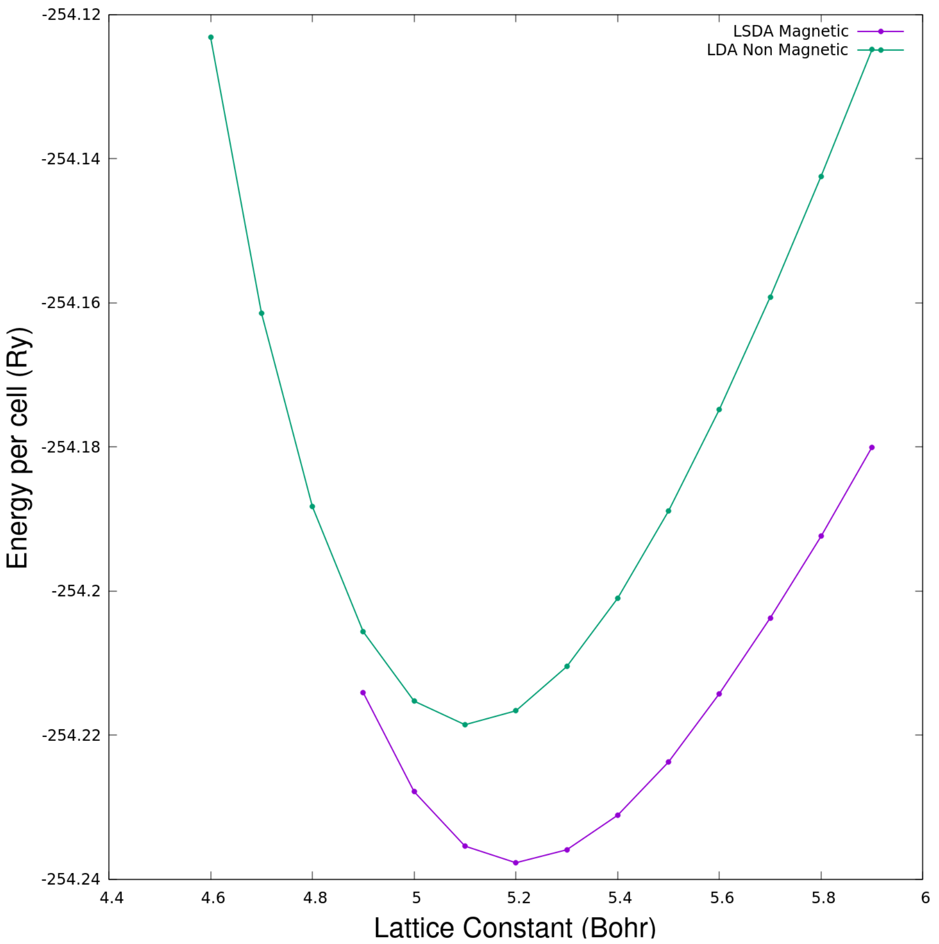
  1. go to `/Day-7/exercise1.ferromagnetic_vs_nonmagnetic/magnetic`
  2. repeat steps 2 and 3 as for the non-magnetic case.

- **Plot and compare energies and lattice constants**

  - go to `/Day-7/exercise1.ferromagnetic_vs_nonmagnetic/`
  - use the gnuplot script `compare.gp` to plot your results.

# Compare energies and lattice constants.

- The magnetic solution has a lower total energy per cell.
- The magnetic solution has a larger lattice constant than the non-magnetic.
- LSDA underestimate lattice constant $\sim 5.2$ Bohr vs. $\sim 5.4$ Bohr of the experiment.



# Remarks on input

How to select the magnetic solution:

```
&system
  ibrav=3,
  celldm(1)=$latt,
  nat= 1,
  ntyp=1,
  nspin=2,
  starting_magnetization(1)=0.3,
  ecutwfc=70.0, ecutrho=850.0,
  occupations='smearing',
  smearing='marzari-vanderbilt',
  degauss=0.02
/
```

- `nspin=2` Specify the spin polarized calculation.
- To allow a magnetic solution we have also break the symmetry between the *up* and *down* channel:
  - `starting_magnetizion(1)=0.3` Builds the starting potential assuming an initial magnetization of Fe of $0.3 \times 16$.

# Remarks on input

## Magnetic system in the non-collinear case

```
&system
   ibrav=3
   celldm(1)=5.4,
   nat=1,ntyp=1,
   noncolin=.true.,
   starting_magnetization=0.3,
   angle1(1)=0.0,
   angle2(1)=0.0,
   ecutwfc=70.0,
   ecutrho=850.0,
   occupations='smearing',
   smearing='marzari-vanderbilt',
   degauss=0.02
/
```

- non-collinear case is selected specifying `noncolin=.true.`
- `starting_magnetization` must be specified for at least one species otherwise the program will perfom a non-magnetic calculation.
- in the non-collinear case magnetization is a vector, it is possible to specify the direction of the starting magnetization for each atomic type:
  - `angle1` is the angle with the $z$ axis (in degrees);
  - `angle2` is the azimuthal angle (in degrees)

# How the magnetization is reported in output

- In the spin-polarized case the total magnetization is reported at each SCF step and at the end of the calculation:

```
total magnetization        =    2.22 Bohr mag/cell
absolute magnetization     =    2.28 Bohr mag/cell
```

- In the magnetic non-collinear case:

```
total magnetization        =    1.38     0.00     1.38 Bohr mag/cell
absolute magnetization     =    1.99 Bohr mag/cell
```

- An estimate of the magnetization at each ionic site is also printed at the end of the calculation

```
Magnetic moment per site  (integrated on atomic sphere of radius R)
atom   1 (R=0.357)  charge= 14.4267  magn=   2.2287
```

- In the magnetic non-collinear case:

```
atom number    1 relative position :    0.0000   0.0000   0.0000
charge :    14.332507  (integrated on a sphere of radius 0.357)
magnetization :          1.374808   -0.000000    1.374808
magnetization/charge:    0.095922   -0.000000    0.095922
polar coord.: r, theta, phi [deg] :     1.944273   45.000000   -0.000000
```

# Input of dos.x and projwfc.x

```
    &dos
      prefix='fe',
      bz_sum='tetrahedra_opt'
      Emin=5.0
      Emax=25.0
      DeltaE=0.05
      fildos='fe.dos'
    /
```

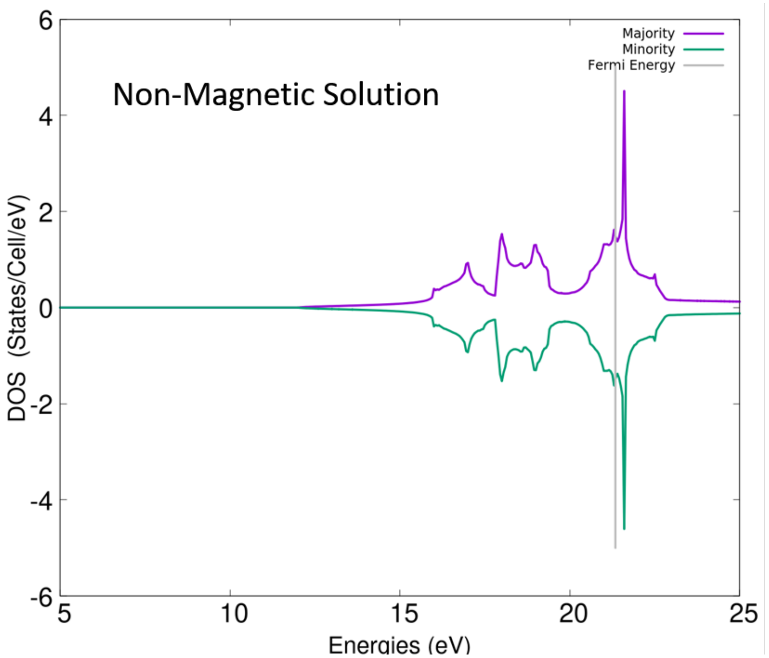
- bz_sum selects the integration method.
- Emin,Emax, and DeltaE define the range and definition of the DOS;
- to select the range first use default values, then select the right window

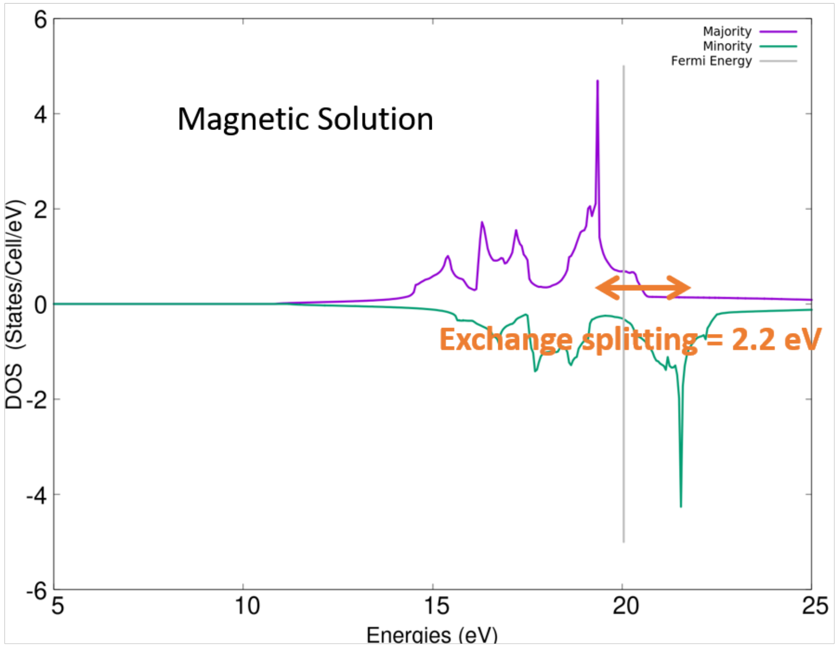
```
 &projwfc
    !outdir='./tempdir/',
    prefix='fe'
    Emin=5.0, Emax=25.0, DeltaE=0.05
 /
```

# Exercise 1: Task 2

Using the optimized lattice constants obtained in task 1 we analize the electronic structures of the two solutions.

1. Make the SCF and non-SCF computations for boths solutions.
2. Compute the DOS for both solutions
3. Compute the pDOS for the magnetic solution





# Exercise 2: Magnetization vs. volume

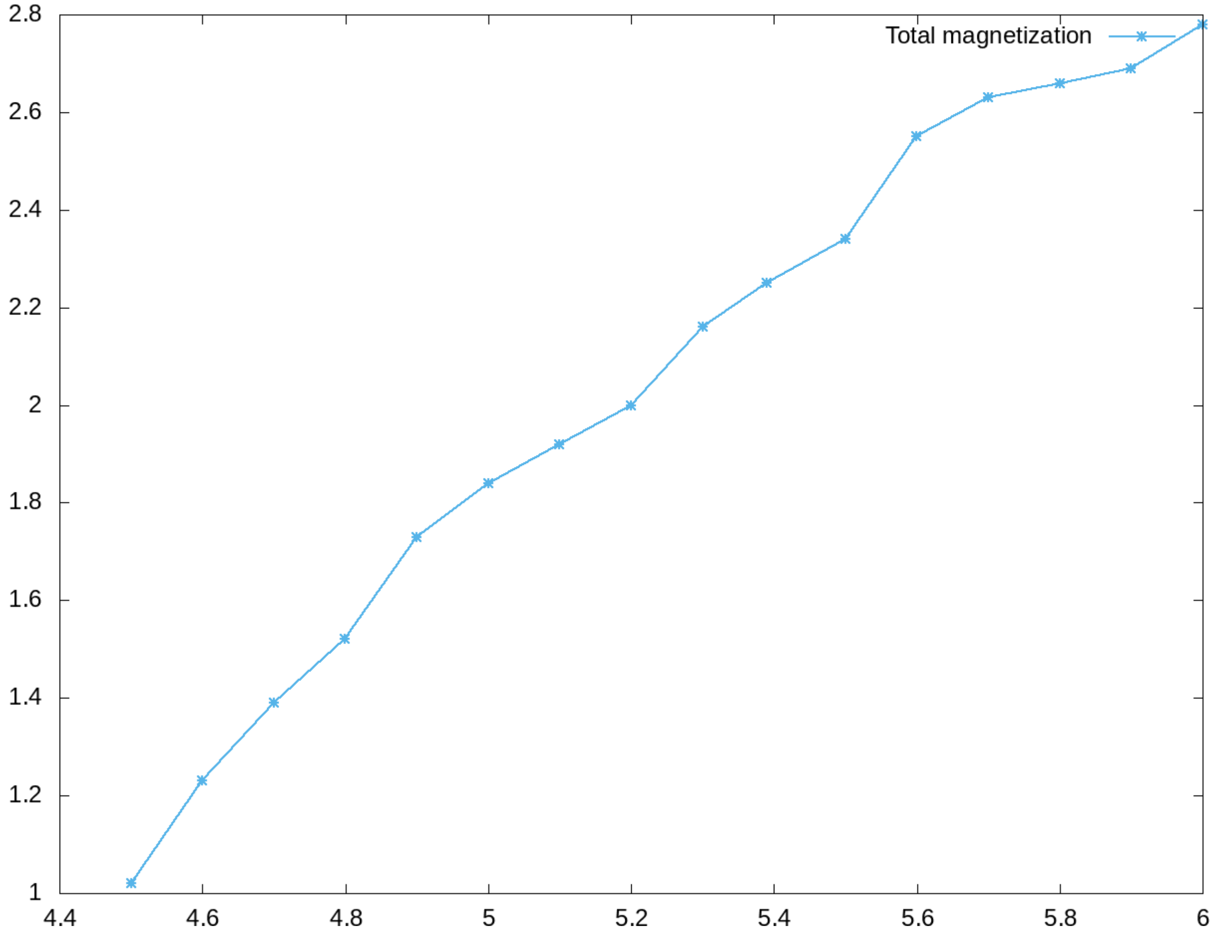
- The aim of this exercise is to see how the magnetization and the exchange splitting behaves when the lattice constant decreases

- Here we use the PBE functional that gives a more realistic lattice constant than LDA.

- The exercise proceeds along the following steps:

1. Run a script for determining the optimized lattice constant
2. Use `ev.x` to estimate the optimized lattice constant fitting the energies to the Murnaghan EOS.
3. Use the data in the file `magnetization.dat` to see how the magnetization changes when we change the lattice constant.
4. **Assignment** Compute the electronic structure for 3 different values of lattice constant and estimate the the exchange splitting as we did in exercise 1.

# Magnetization vs. alat



# Exercise 3. Comparing magnetization and exchange splitting in transition metals

In this exercise we compare the magnetic properties of three different transition metals Fe(bcc), Co(hcp), and Ni(fcc)

- With the usual procedure we compute the electronic structure for the 3 transition metals.
- use `projwf.x` to analize the contribution to the magnetization coming from the different orbitals.

```
Atom #   1: total charge =  17.0232, s =  2.4486, p =  6.1080, d =  8.4666,
spin up      =   8.8345, s =  1.2198,
spin up      =   8.8345, p =  3.0211, pz=  1.0070, px=  1.0070, py=  1.0070,
spin up      =   8.8345, d =  4.5935, dz2=  0.9308, dxz=  0.9106, dyz=  0.9106,
                                      dx2-y2=  0.9308, dxy=  0.9106,
spin down    =   8.1887, s =  1.2288,
spin down    =   8.1887, p =  3.0869, pz=  1.0290, px=  1.0290, py=  1.0290,
spin down    =   8.1887, d =  3.8731, dz2=  0.8427, dxz=  0.7292, dyz=  0.7292,
                                        dx2-y2=  0.8427, dxy=  0.7292,
  polarization =   0.6457, s = -0.0089, p = -0.0658, d =  0.7204,
Spilling Parameter:   0.0543
```

# Format of projwfc.x output

- The total DOS and the sum of projected DOS are written to file <filpdos>.pdos_tot.
  - The format for the collinear, spin-unpolarized case and the non-collinear, spin-orbit case is:

```
E DOS(E) PDOS(E)
...
```

  - The format for the collinear, spin-polarized case is:

```
E DOSup(E) DOSdw(E)  PDOSup(E) PDOSdw(E)
...
```

  - The format for the non-collinear, non spin-orbit case is:

```
E DOS(E) PDOSup(E) PDOSdw(E)
...
```

# Format of the projection files

In the collinear case and the non-collinear, non spin-orbit case projected DOS are written to file `<filpdos>.pdos_atm#N(X)_wfc#M(l)`, where N = atom number , X = atom symbol, M = wfc number, l=s,p,d,f (one file per atomic wavefunction found in the pseudopotential file)

- The format for the collinear, spin-unpolarized case is:

```
      E LDOS(E) PDOS_1(E) ... PDOS_2l+1(E)
      ...
```

  where

  – LDOS = $\sum_{m=1,2l+1} PDOS\_m(E)$
  – PDOS_m(E) = projected DOS on atomic wfc with component m

- The format for the collinear, spin-polarized case and the non-collinear, non spin-orbit case is as above with two components for both LDOS(E) and PDOS_m(E)

# Orbital order

- for l=1:

  1. $p_z$ (m=0)
  2. $p_x$ (real combination of m=+/-1 with cosine)
  3. $p_y$ (real combination of m=+/-1 with sine)

- for l=2:

  1. $d_{z^2}$ (m=0)
  2. $d_{zx}$ (real combination of m=+/-1 with cosine)
  3. $d_{zy}$ (real combination of m=+/-1 with sine)
  4. $d_{x^2-y^2}$ (real combination of m=+/-2 with cosine)
  5. $d_{xy}$ (real combination of m=+/-2 with sine)

# Exercise 4: plotting bands

- this exercise provides an example on how to plot the bands of a magnetic system.

- Spin-polarized collinear case:
  - • `cd Ni_collinear`
  - SCF calculation for Ni (bcc): • `pw.x < ni.scf.in > ni.scf.out`
  - non-SCF calculation:
    - ∗ `calculation='bands'`
    - ∗ set the list of K-points

```
K_POINTS crystal_b
6
  0.000  0.000  0.000  20  !gamma
  0.500  0.500  0.500  10  !L
  0.500  0.250  0.750  10  !W
  0.500  0.000  0.500  10  !X
  0.000  0.000  0.000  20  !gamma
  0.375  0.375  0.750  1   !K
```

· we specify the high-symmetry points and number of points of each segment

· `crystal_b` specify the format for K-points

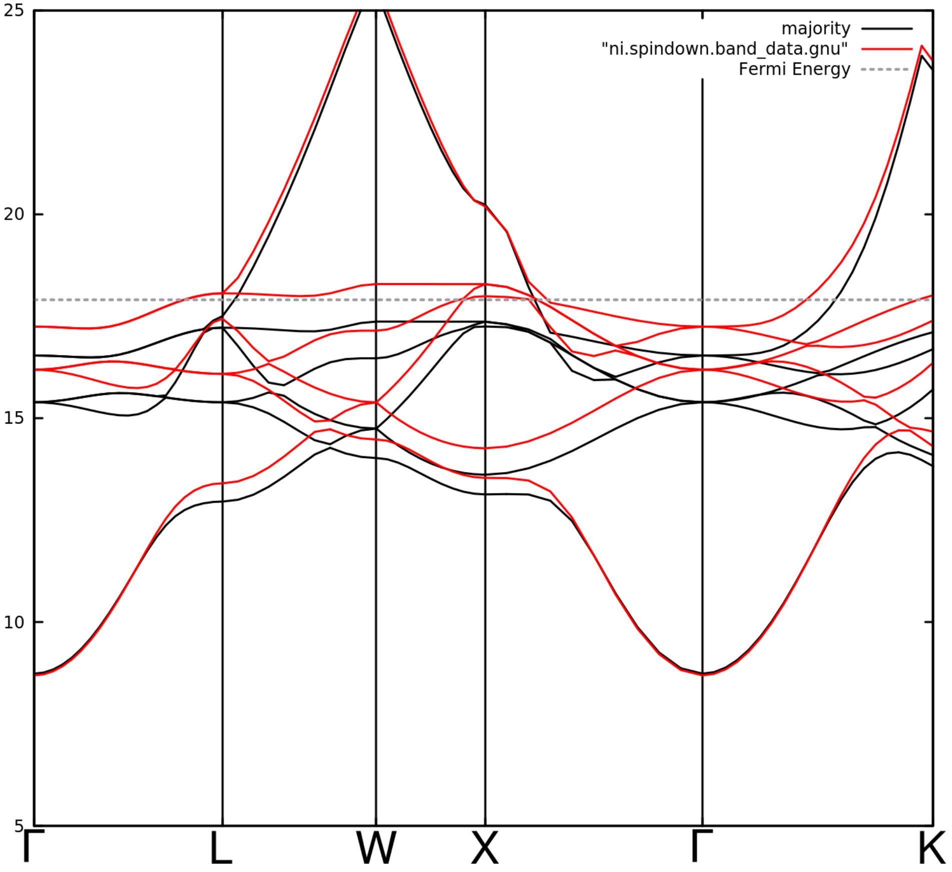
- use bands.x for preparing the band plot and analize symmetry, first for up component then for down component

```
&BANDS

  outdir=’./tempdir/’,
  prefix=’ni’,
  filband=’ni.spinup.band_data’,
  spin_component = 1,
/
```

  - ∗ spin_componet selects the spin component
  - ∗ filband selects the prefix where the bands are saved

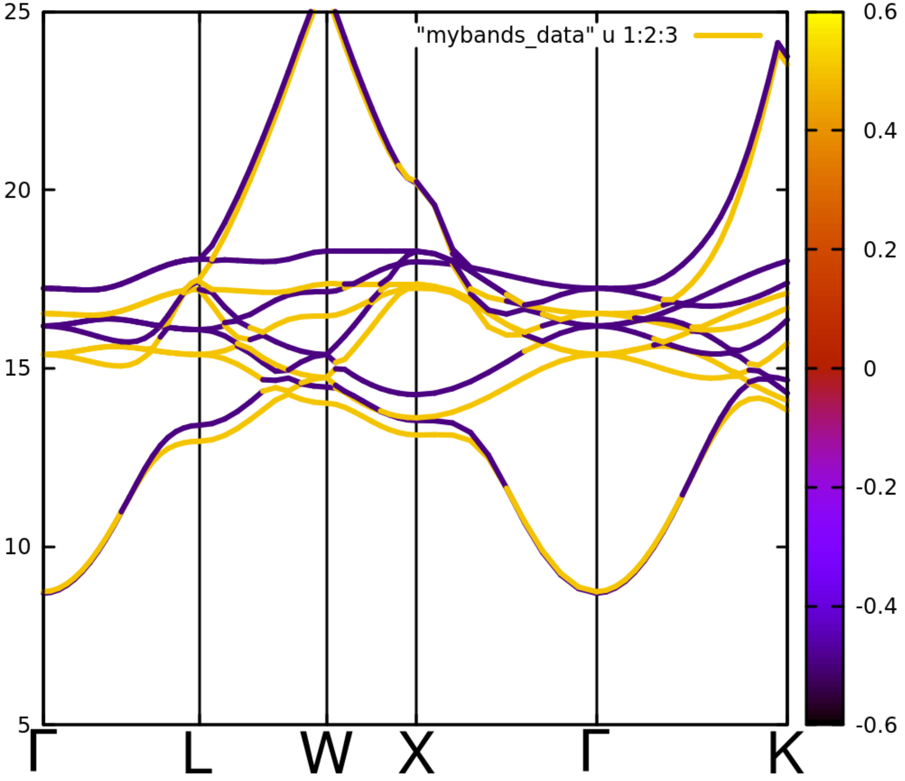
- the output of bands.x provides the length of the k-points path and the position of the high-symmetry points
- use the gnuplot script to plot the bands


25
20
15
10
5
majority
"ni.spindown.band_data.gnu"
Fermi Energy
Γ
L
W
X
Γ
K

# Non-collinear case

- Plot the Ni bands in the noncollinear case:
  - – • `cd Ni_noncollinear`
  - run the collinear scf calculation for Ni in this directory:
    - • `pw.x < ni.scf.in > ni.scf.out`
  - run the non-collinear nscf calculation for the bands
    - ∗ `spin=2` has been replaced with `noncolinear=.true.`
      - • `pw.x < ni.bands.in > ni.bands.out`
  - run ‘bands.x‘ for the noncollinear case:
  - `spin_component` has been removed and we add `lsigma(3)=.true.` that instructs the program to compute the expectation value for the z component of the spin operator for each eigenfunction and save all values in the file `ni.noncolin.data.3`. All values in this case are ceeither 1/2 or -1/2 as expected.
  - the program `plot_noncolin_bands.f90` reads this values and writes them together with the band structure in the file `my_bands.data`.
    - ∗ compile the program:
      - • `gfortran -o mino.x plot_noncolin_bands.f90`

- ∗ copy `ni.noncolin.data.3` to `ni.noncolin.data.s`
- ∗ run the program
  - `./mino.x ni.noncolin.data`
- ∗ use gnuplot and the script `bands_noncollin.gp` to plot the bands in this case.
- ∗ start gnuplot and type the command:
  - `gnuplot> load "bands_noncollin.gp"`



THE END