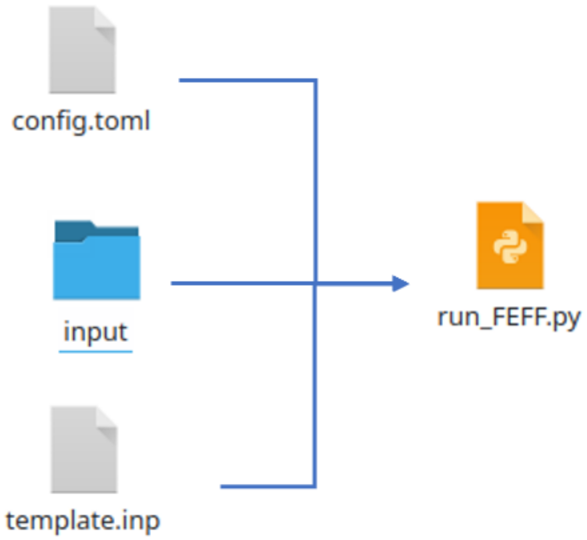
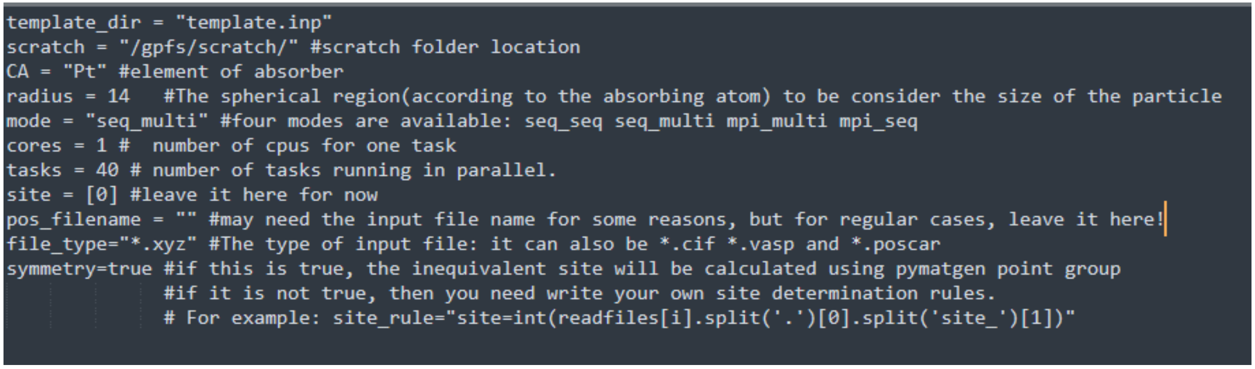
DOCUMENT of Multi-task FEFF calculation

1. The required files



Config.toml contains the configurations of this calculation.



```
template_dir = "template.inp"
scratch = "/gpfs/scratch/" #scratch folder location
CA = "Pt" #element of absorber
radius = 14   #The spherical region(according to the absorbing atom) to be consider the size of the particle
mode = "seq_multi" #four modes are available: seq_seq seq_multi mpi_multi mpi_seq
cores = 1 #  number of cpus for one task
tasks = 40 # number of tasks running in parallel.
site = [0] #leave it here for now
pos_filename = "" #may need the input file name for some reasons, but for regular cases, leave it here!
file_type="*.xyz" #The type of input file: it can also be *.cif *.vasp and *.poscar
symmetry=true #if this is true, the inequivalent site will be calculated using pymatgen point group
              #if it is not true, then you need write your own site determination rules.
              # For example: site_rule="site=int(readfiles[i].split('.')[0].split('site_')[1])"
```

Template_dir: the location of feff.inp template
Scratch: the location of scratch folder
CA: absorbing atom
Radius: the size of particle for calculating spectrum. For non-periodic samples(nanoparticles), the value requires to be large enough to cover all atoms of the particle or the distance where is enough to calculate stable spectrum.
Mode: the calculation mode, there are 4 different modes: mpi_seq,mpi_multi,seq_seq, and seq_multi. The **_seq** suffix means the program will calculate spectrum for different site or sample one by one. _multi means the program will calculate the spectrum in a batch, the size of batch is defined by tasks.
Cores: the number of cpus for mpirun
Tasks: number of tasks run simultaneously.
Site: sites we are trying to calculate (still underdeveloped)

File_type: the input file type, can be *.xyz,*.POSCAR, *.cif
Symmetry: True means the calculation will find inequivalent sites by point group (only work for molecule—xyz file)

2. How to run

2.1 use module

cd module directory

module use FEFF10

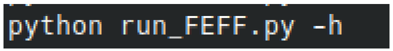
module load 10

Then, you can run feff by using **feffmpi** 8 #8 cores

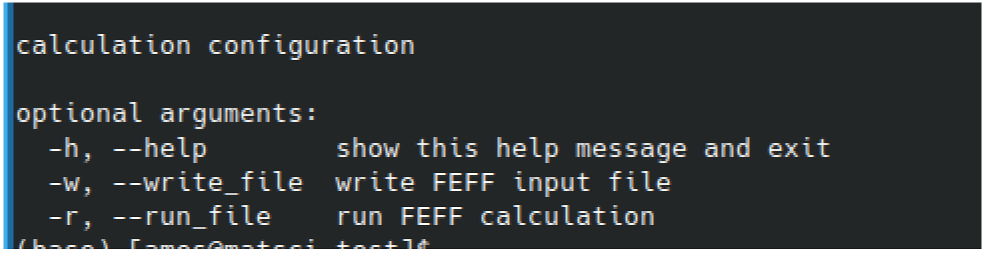
Or: **feff** #seq run

2.2 write input files

Go to `your folder`



```
python run_FEFF.py -h
```



```
calculation configuration

optional arguments:
  -h, --help        show this help message and exit
  -w, --write_file  write FEFF input file
  -r, --run_file    run FEFF calculation
```

The run_FEFF.py python script has two arguments: -w and -r

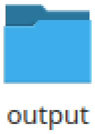
If run `python run_FEFF.py -w,` the program will write feff input files(if symmetry=True, it will write input files for inequivalent sites) automatically to a new folder called FEFF_inp.

The writing process is in parallel version, if you want to write 50000 more files, it helps!

```
100%|████████████████████████████████████████| 1000/1000 [04:46<00:00,  3.49it/s]
100%|███████████████████████████████████████| 53744/53744 [27:05<00:00, 33.07it/s]
```

2.3 run FEFF

To run FEFF correctly, first needs to check template.inp, and then check config.toml.
Then run it `python  run_FEFF.py -r`
The results will be stored in json file separately in output folder.

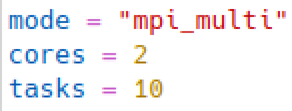


2.4 Toolbox

There is a toolbox folder inside the package. You can run by:
bash run_tool.sh
This will give you the average spectrum for each particle and store them in a csv file.
Note: change the modules in run_tool.sh to fit your system.

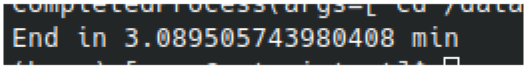
3. A small test on 55-atom Ni particle

For multi-task computing, I used 2 cores for mpi, and the maximum number of tasks can be submitted is 10, but we only have 5 tasks in this case, so it will submit all tasks simutanously.



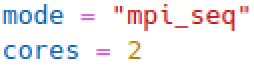
```
mode = "mpi_multi"
cores = 2
tasks = 10
```

In this case, it takes



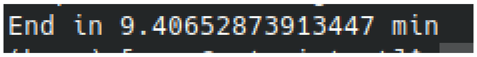
```
CompletedProcess(args=['cd /data
End in 3.089505743980408 min
```

For single-task computing(one by one)
The value of tasks is not matter.



```
mode = "mpi_seq"
cores = 2
```

It takes 9 minutes for all 5 tasks finished.



```
End in 9.40652873913447 min
```