Longxun inc., Beijing

User guide

use pwmat to calculate phonons

PWmat2Phonopy 0.2.1

Author: Peng Chen

email: peng.chen.iphy@gmail.com

November 16, 2017

Abstract

PWmat2Phonopy provides an interface to phonopy, which can help PWmat user perform phonon calculations. You might find it most useful for tasks you already have a relaxed atom.config. The best thing of PWmat2Phonopy is that, with almost only one click, the phonon spectrum will be calculated for you.

1 Install

- Firstly, download PWmat2Phonopy-x.x.x.tar.gz from github (https://github.com/PaulChern/PWmat2Phonopy/tree/master/dist)

- Secondly, unzip the PWmat2Phonopy-x.x.x.tar.gz wherever you prefer. tar zxvf PWmat2Phonopy-x.x.x.tar.gz

- Thirdly, include /where/you/put/PWmat2Phonopy-x.x.x in you PYTHONPATH and /where/you/put/PWmat2Phonopy/bin in your PATH. vi .bashrc

```
# added by Paul Chern for the phonopy interface to pwmat
export PATH=/home/pwmat/PaulChern/Applications/PWmat2Phonopy-0.1.0/bin:$PATH
export PYTHONPATH=/home/pwmat/PaulChern/Applications/PWmat2Phonopy-0.1.0:$PYTHONPATH
```

- All settled!

2 Usage

2.1 quick start

Just type in PWmatRunPhonopy.py in the directory where atom.config, etot.input, and PP are strored, and everything will be on the way:

- these files are needed:

```
[pwmat@node0 Si]$ ls
Si.SG15.NCPP.PBE.UPF  atom.config  etot.input
```

- run PWmatRunPhonopy.py

```
[pwmat@node0 Si]$ PWmatRunPhonopy.py
```

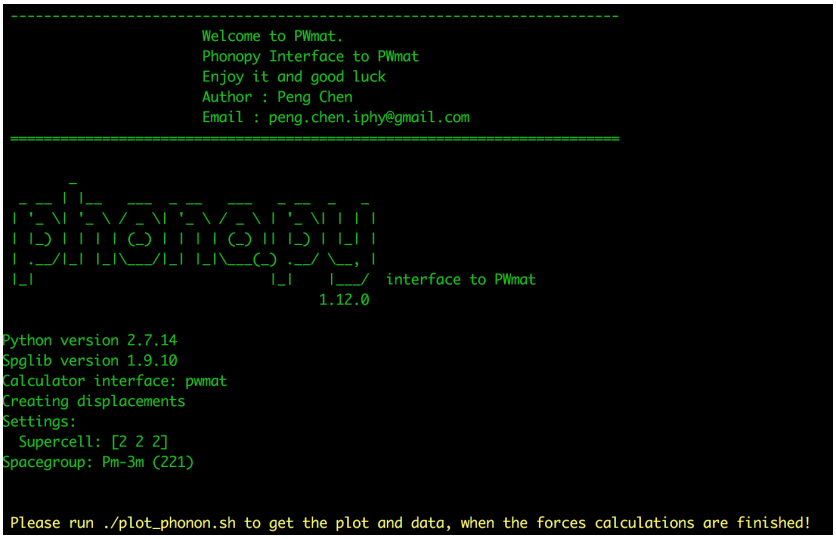
- follows:

```
[pwmat@node0 CaTiO3_ref]$ PWmatRunPhonopy.py

 ------------------------------------------------------------------------
                        Welcome to PWmat.
                        Phonopy Interface to PWmat
                        Enjoy it and good luck
                        Author : Peng Chen
                        Email : peng.chen.iphy@gmail.com
 ========================================================================

current workding directory is: /home/pwmat/PaulChern/Applications/PWmat2Phonopy/examples/CaTiO3_ref
phonon will be calculate in: /home/pwmat/PaulChern/Applications/PWmat2Phonopy/examples/CaTiO3_ref/phonon

Please type enter to edit the  pwmat2phonopy.in  file
```

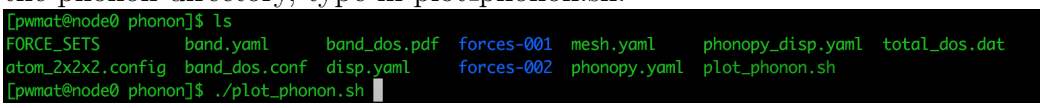
- type enter, input file 'pwmat2phonopy.in' will be open for you to modify:

```
NODES = 1    4                                  # node1 node2 for pwmat parallel configuration
WALL_TIME = 1000:00:00                          # wall time for the queue system (torque): hours:minutes:seconds
MP_N123 = 5 5 5  0 0 0                         # k-mesh for the supercell calculations; it is 10 10 10  0 0 0 for the unitcell calculation
DIM = 2 2 2                                     # supercell dimension which will be used to make displacements
PRIMITIVE_AXIS = 1.0 0.0 0.0  0.0 1.0 0.0  0.0 0.0 1.0    # the primitive cell for building the dynamical matrix
BAND =    0.0000000000    0.0000000000    0.0000000000      0.5000000000    0.0000000000    0.0000000000      0.5000000000    0.0000000000    0.5000000000      0.0000000000
00    0.0000000000    0.0000000000      0.5000000000    0.5000000000    0.5000000000      0.5000000000    0.0000000000    0.0000000000                              # speci
al q points in Brillioun zone
BAND_LABELS = \Gamma  X  M \Gamma R X                       # labels of the special q points
BAND_POINTS = 101                               # number of q-point along the high symmetry path
FC_SYMMETRY = 1
FREQUENCY_CONVERSION_FACTOR = THz               # unit of the frequency:"THz", "cm^-1", "eV", "meV"
DOS = TRUE                                      # switch to the DOS calculation
MP = 41 41 41                                   # q-mesh for the DOS calculation
FPITCH = 0.1                                    # frequency interval for DOS calculation
SIGMA = 0.1                                     # smearing width for DOS calculation
~
~
~
```

- save and exist (vim commands):



- then, PWmat jobs will be submitted; When the jobs are finished, in the phonon directory, type in plot_phonon.sh:



- finally, band_dos.pdf total_dos.dat band.yam will be produced

2.2 for experienced users

The PWmatRunPhonopy.py actually call PWmat2Phonopy, a PWmat version of phonopy with an interface to PWmat and the same functions as in phonopy. Other than using the PWmatRunPhonopy.py directly, you can acturally use PWmat2Phonopy too. The usage is essentially the same as using phonopy, the only different is that you need to include –pwmat in the command line, otherwise the PWmat2Phonopy will work for vasp. If you use –vasp, –wien2k, –abinit, and et. al., PWmat2Phonopy will work exactly the same as the phonopy does. So, if you want to learn all the functions and how to use, you can consult phonopy website or user guide.

the command line options include:
–amplitude (DISPLACEMENT_DISTANCE)

–anime (ANIME)
–band (BAND)
–band-connection (BAND_CONNECTION = .TRUE.)
–band-format (BAND_FORMAT)
–band-labels (BAND_LABELS)
–band-points (BAND_POINTS)
–cutoff-freq (CUTOFF_FREQUENCY)
-c, –cell (CELL_FILENAME)
-d (CREATE_DISPLACEMENTS = .TRUE.
–dim (DIM)
–dos (DOS = .TRUE.)
–eigvecs, –eigenvectors (EIGENVECTORS = .TRUE.)
–factor (FREQUENCY_CONVERSION_FACTOR)
–fits-debye-model (DEBYE_MODEL = .TRUE.)
–fmax (FMAX)
–fmin (FMIN)
–fpitch (FPITCH)
–gc, –gamma_center (GAMMA_CENTER)
–gv, –group_velocity (GROUP_VELOCITY = .TRUE.)
–gv-delta-q (GV_DELTA_Q)
–hdf5 (HDF5 = .TRUE.)
–irreps (IRREPS)
–lcg, –little_cogroup (LITTLE_COGROUP)
–modulation (MODULATION)
–moment (MOMENT = .TRUE.)
–moment_order (MOMENT_ORDER)
–mesh-format (MESH_FORMAT)
–mp, –mesh (MP or MESH)
–nac (NAC = .TRUE.)
–nosym (SYMMETRY = .FALSE.)
–nomeshsym (MESH_SYMMETRY = .FALSE.)
–nowritemesh (WRITE_MESH = .FALSE.)
–pa, –primitive-axis (PRIMITIVE_AXIS)
–pd, –projection-direction (PROJECTION_DIRECTION)
–pdos (PDOS)
–pr, –pretend-real (PRETEND_REAL = .TRUE.)
–q-direction (Q_DIRECTION)
–qpoints (QPOINTS)

–qpoints-format (QPOINTS_FORMAT)
–readfc (READ_FORCE_CONSTANTS = .TRUE.)
–readfc-format (READFC_FORMAT)
–read-qpoints (QPOINTS = .TRUE.)
–show-irreps (SHOW_IRREPS)
–sigma (SIGMA)
-t (TPROP)
–td (TDISP)
–tdm (TDISPMAT)
–tdm-cif (TDISPMAT_CIF)
–thm, –tetrahedron-method (TETRAHEDRON)
–tmin (TMIN)
–tmax (TMAX)
–tstep (TSTEP)
–writedm (WRITEDM = .TRUE.)
–writefc (WRITE_FORCE_CONSTANTS = .TRUE.)
–writefc-format (WRITEFC_FORMAT)
–xyz-projection (XYZ_PROJECTION = .TRUE.)

When both of command-line option and setting tag for the same purpose are set simultaneously, the command-line options overide the setting tags.

...

Input cell
-c or –cell
Unit cell crystal structure file is specified with this tag.

PWmat2Phonopy –pwmat –cell=atom.config band.conf
Without specifying this tag, default file name is searched in current directory. The default file names for the calculators are as follows:

PWmat | atom.config
VASP | POSCAR
Wien2k | case.struct
Abinit | unitcell.in
Pwscf | unitcell.in
Elk | elk.in
CRYSTAL| crystal.o

...
Create FORCE_SETS
-f or –forces
VASP interface

FORCE_SETS file is created from disp.yaml, which is an output file when creating supercells with displacements, and OUT.FORCE's, which are the PWmat output files. disp.yaml in the current directory is automatically read. The order of displacements written in disp.yaml file has to correpond to that of OUT.FORCE files .

PWmat2phonopy –pwmat -f forces-001/OUT.FORCE forces-002/OUT.FORCE ...

...
–fz
–fz option is used to subtract residual forces frown the forces calculated for the supercells with displacements. Here the residual forces mean that the forces calculated for the perfect supercell for which the number of atoms has to be the same as those for the supercells with displacements. If the forces are accurately calculated by calculators, the residual forces should be canceled when plus-minus displacements are employed (see PM), that is the default option in phonopy. Therefore –fz option is expected to be useful when PM = .FALSE. is set in the phonopy setting file.

The usage of this option is almost the same as that of -f option except that one more argument is inserted at the front. Mind that –fz is exclusively used with -f option. The example for the VASP interface is shown below:

PWmat2Phonopy –fz ./OUT.FORCE force-001/OUT.FORCE ...

where ./OUT.FORCE assumes the output file for the perfect supercell containing residual forces.
...
Graph plotting
-p
Result is plotted.

PWmat2Phonopy –pwmat -p
-p -s
Result is plotted (saved) to PDF file.

PWmat2Phonopy –pwmat -p -s
Log level
-v or –verbose
More detailed log are shown

-q or –quiet
No log is shown.
...
Crystal symmetry
–tolerance
The specified value is used as allowed tolerance to find symmetry of crystal structure. The default value is 1e-5.

PWmat2Phonopy –pwmat –tolerance=1e-3
–symmetry
Using this option, various crystal symmetry information is just printed out and phonopy stops without going to phonon analysis.

PWmat2Phonopy –pwmat –symmetry
This tag can be used together with the –cell (-c), –abinit, –pwscf, –elk, –wien2k, –crystal or –primitive_axis option.

3 What is next

In the next version, a auto-q path and –fz (eliminate the residual forces) will be automatically included.