User's Guide for KPROJ

Mingxing Chen

School of Physics and Electronics, Hunan Normal University,
Changsha, Hunan 410081, China
email: mxchen@hunnu.edu.cn

Contents

1 Introduction

This manual describes using KPROJ to obtain unfolded electronic bands by the k-projection method for doped materials and interfaces modeled by supercells within the framework of first-principles method. The method has been given in Ref. [1] (Phys. Rev. B **98**, 245421 (2018)). This paper should be cited if you publish results obtained using KPROJ. More applications can be found in Refs. [2–9] Essentially, it starts off with projections of the wavefunctions for a supercell onto the k-points of the 1×1 BZ of the primitive cell. Then, it calculates the weights of the projected wavefunctions and plot the band structure with the weights. A layer-projection scheme is used for unfolding the electronic bands of interfaces, which significantly accelerates the calculations.

2 Installation

The distribution includes a makefile for the intel fortran compiler. Once the program has been successfully compiled, an executable named **kproj** will be found. Moreover, there is a program name as **futils** (You can get it by typing "make futils" on the terminal.) for preparing files for plotting based on the output **bs_projected.dat**. If there are more than one *bs_projected.dat* files named like *bs_projected.dat_1*, *bs_projected.dat_2*, ... the program **mergefile** ("make mergefile" by the terminal command) can help you merge all the *bs_projected.dat_i* into *bs_projected.dat*. Usually, we do this for large systems. Please read section 7 when you have a large system. Directory **utils** includes **kpts_path** for generating the k-points (in reciprocal lattice vectors) along various high symmetry lines of the BZ and **ksplit** for splitting a KPOINTS file containing a list of k-points into a few KPOINTS files like *KPOINTS_1*, *KPOINTS_2*, ... **kpts_path** and **ksplit** should be compiled before using. You can compile them using ifort like "ifort -CB -o kpts_path kpts_path.f".

3 Files Used by KPROJ

KPROJ uses a couple of input and output files:

INKPROJ	in	input parameters
WAVECAR	in	the wavefunctions from vasp calculations
OUTKPROJ	out	general output, e.g., input data
bs_projected.dat	out	weights from k-projection

4 INKPROJ

The input parameters are read by a user-friendly control like VASP. A sample of INPROJ for layer-projection along Z axis for a heterostructure:

```
LZLAYER = .TRUE.
zlay1 = 0.5
zlay2 = 0.62
MAT_P2S=  5 0 0 \
                0 5 0 \
                0 0 1
LSORBIT = .TRUE.
```

LXLAYER (*logical*): If a layer-projection along the 1^{st} axis will be done. *Default:* **LXLAYER = .FALSE.** When **LXLAYER = .TRUE.**, **xlay1** and **xlay2** should be provided with suitable values.

LYLAYER (*logical*): If a layer-projection along the 2^{nd} axis will be done. *Default:* **LYLAYER = .FALSE.** When **LYLAYER = .TRUE.**, **ylay1** and **ylay2** should be provided with suitable values.

LZLAYER (*logical*): If a layer-projection along the 3^{rd} axis will be done. *Default:* **LZLAYER = .FALSE.** When **LZLAYER = .TRUE.**, **zlay1** and **zlay2** should be provided with suitable values.

When **LXLAYER**, **LYLAYER**, and **LZLAYER** are all .FALSE., the unfolding will be done for the whole space. This is useful for bulk calculations.

LSORBIT (*logical*): If the WAVECAR was obtained from a spin-orbit coupling (noncollinear) calculation. *Default:* **LSORBIT = .FALSE.**

MAT_P2S (*integers*): The transform matrix from the primitive cell to the supercell. For instance, the above sample of INKPROJ is for a 5 × 5 supercell. The **MAT_P2S** can be given either by

```
MAT_P2S= 5 0 0 \
         0 5 0 \
         0 0 1
```

or

```
MAT_P2S= 5 0 0   0 5 0   0 0 1
```

For a $\sqrt{3} \times \sqrt{3}$ of graphene, it may be a matrix like

```
MAT_P2S= 1 -1 0 \
         1 2 0 \
         0 0 1
```

5 Running KPROJ

Wavefunctions at the k-points in the BZ of the primitive cell have to exist under the working directory. Therefore, the k-points stored in KPOINTS should be those for the primitive cell. That is, the high symmetry points of the primitive cell should be represented by the reciprocal lattice vectors of the supercell in KPOINTS. For instance, if you want to do unfolding for bands along M-Γ-K for a 5 × 5 supercell of graphene, the KPOINTS file should be

```
Graphene M-G-K
101
line
reciprocal
0.00000000000000 2.50000000000000 0.00000000000000 1
0.00000000000000 0.00000000000000 0.00000000000000 1

0.00000000000000 0.00000000000000 0.00000000000000 1
1.66666666666667 1.66666666666667 0.00000000000000 1
```

The transform can be easily done by $\mathbf{K_p} = \mathbf{K_s}M^T$, where M is the transform matrix given by **MAT_P2S**, M^T the transpose of M, and $\mathbf{K_p}$ the coordinate of the k-point in the BZ of the primitive cell represented by the reciprocal lattice vectors of the supercell.

In the case $\sqrt{3} \times \sqrt{3}$ graphene, the KPOINTS file for band calculations along M-Γ-K is suggested to be

```
Graphene M-G-K
101
line
reciprocal
-0.50000000000000 1.00000000000000 0.00000000000000 1
0.00000000000000 0.00000000000000 0.00000000000000 1

0.00000000000000 0.00000000000000 0.00000000000000 1
0.00000000000000 1.00000000000000 0.00000000000000 1
```

Where the transform matrix was supposed to be the one give above.

6 Plotting

A file named *bs_projected.dat* will be obtained after a KPROJ calculation, which will be used for plotting. Prior to the plotting, a few things have to be done.

First, replacing the value for the Fermi level in *bs_projected.dat* by the one from the **SCF** calculation. Please always double check it!

Second, running the script named like **run_futils** for **futils** (You can also run futils directly). **run_futils** controls the parameters such as the energy window (defined by **emin** and **emax**) for plotting. **de** defines the energy mesh for within the window. While **dele** controls smearing of the bands (Generally I use the doubled value of **de** for it). Please check **run_futils** if the path to program **futils** is given correctly.

After the running of **futils**, files for plotting will be ready. Output files may include *plot_bands_1.dat*, *plot_bands_1.general*, *plotbnds-up.gnu* and so on. For non-magnetic and spin-orbit coupling calculations there is only one file for each type of the above files (*plotbnds.gnu* will be output instead of *plotbnds-up.gnu* in these cases).

There are three ways of plotting. One is to use gnuplot by a command like "gnuplot plotbnds.gnu". Appearance of axis labels, tick labels can be modified by editing **plotbnds.gnu**. Another one is to use xmgrace, for which file(s) with suffix agr can be found under the working directory.

The third one is to use opendx, which is slightly more complicated than the other two.

7 Tips for dealing with large systems

When your system is large, the WAVECAR may be extremely huge for a large number of k-points. Splitting the band calculation into several calculations is a good way to avoid memory problems. Generally I keep the size of WAVECARs at about 10 GBs. I estimate the size of WAVECAR by NPW × NBANDS × NKPTS × 8 in bytes (For magnetic calculations it has to be doubled). NPW, NBANDS and NKPTS can be obtained by "grep planewaves: OUTCAR" and "grep NBANDS OUTCAR" under the directory for SCF calculations. By the above estimate I can choose a proper number of k-points for each sub-calculation. Following are what I do for calculations of large systems.

First, generate the k-points using **kpts_path** for the selected high symmetry line, which will have the generated k-points stored in kpts.dat.

Then, rename kpts.dat as KPOINTS and split the KPOINTS into several subfiles using **ksplit**.

Do the standard band calculation using VASP for all KPOINTS_i and have the wavefunctions output.

Proceed to **KPROJ** calculation for all and obtain bs_projected.dat.

Copy all **bs_projected.dat** to a directory for plotting and rename them like bs_projected.dat_1, bs_projected.dat_2, ...

Merge all the bs_projected.dat_i files into one using **mergefile**.

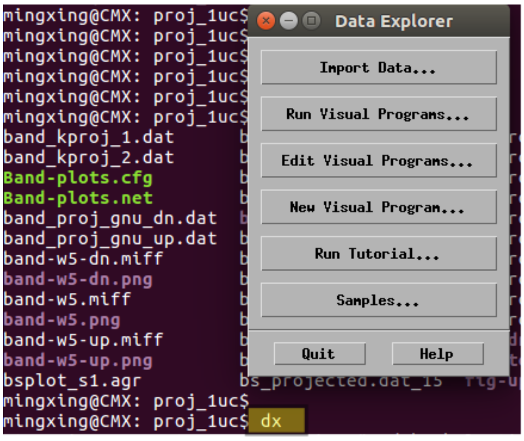
Replace the Fermi level in the merged bs_projected.dat by the correct value.

Run **run_futils** and plot using opendx or gnuplot.

8 Plotting using opendx

Two files will be used under the working directory. One is **Band-plots.cfg** and the other one is **Band-plots.net**, which can be found under the directory of **utils**. There are a few procedures for this type of plotting:

First, open a linux terminal and run opendx by the command dx.



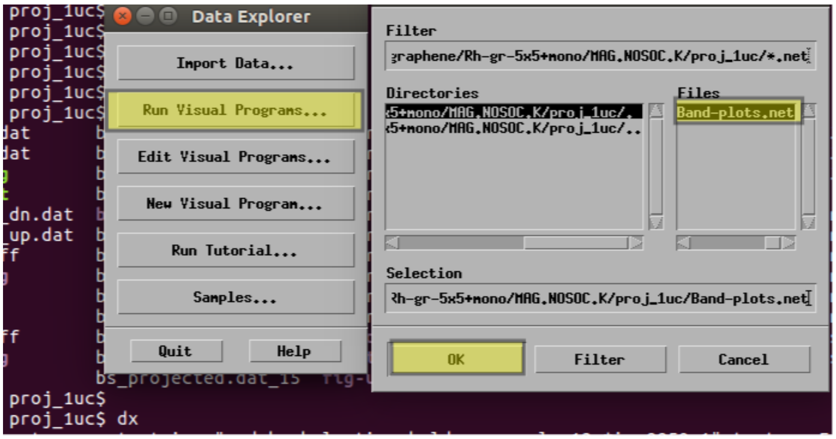
```
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
band_kproj_1.dat       b
band_kproj_2.dat       b
Band-plots.cfg         b
Band-plots.net         b
band_proj_gnu_dn.dat   b
band_proj_gnu_up.dat   b
band-w5-dn.miff        b
band-w5-dn.png         b
band-w5.miff           b
band-w5.png            b
band-w5-up.miff        b
band-w5-up.png         b
bsplot_s1.agr          bs_projected.dat_15   fig-u
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$ dx
```

Figure 1: Run opendx by the command dx.

Then click "Run visual Programs ..." on the Data Explorer of opendx. Select **Band-plots.net** and click OK (Ensure that you have copied **Band-plots.net** and **Band-plots.cfg** to the working directory.). After that a gridded 2D plot will be shown.

Go to the "control panel" and click "reset image", which will adjust the size of the plot.

Click "reset contour lines" and delet all the contour line values except for the last one.



```
proj_1uc$
proj_1uc$
proj_1uc$
proj_1uc$
proj_1uc$
proj_1uc$
dat
dat
dn.dat
up.dat
bs_projected.dat_15
proj_1uc$
proj_1uc$ dx
```

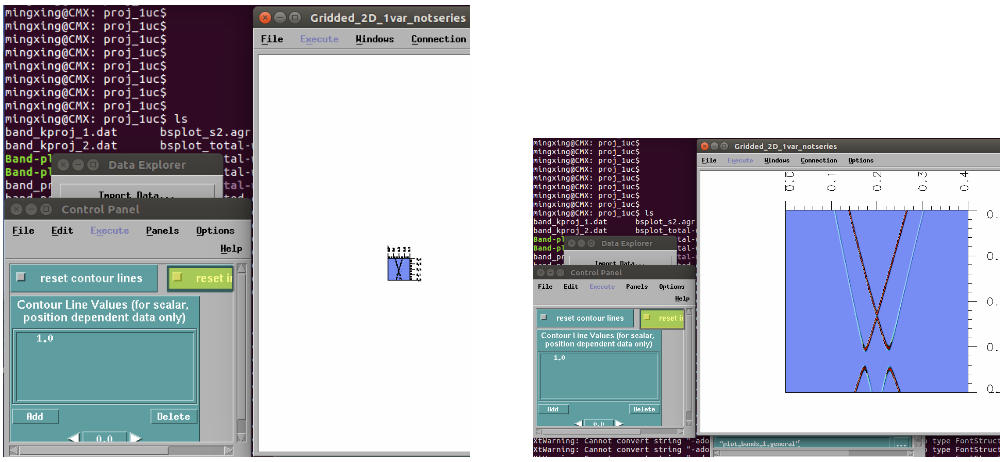
Filter: graphene/Rh-gr-5x5+mono/MAG.NOSOC.K/proj_1uc/*.net

Directories: 5+mono/MAG.NOSOC.K/proj_1uc/.
 5+mono/MAG.NOSOC.K/proj_1uc/..

Files: Band-plots.net

Selection: Rh-gr-5x5+mono/MAG.NOSOC.K/proj_1uc/Band-plots.net

Figure 2: Run visual Programs on the Data Explorer of opendx.



```
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$ ls
band_kproj_1.dat        bsplot_s2.agr
band_kproj_2.dat        bsplot_total-
Band-pl                 tal-
Band-pl                 tal-
band_pr                 tal-
XtWarning: Cannot convert string "-ado        type FontStruct
XtWarning: Cannot convert string "-ado        type FontStruct
```

"plot_bands_1.general"

Figure 3: Click "reset image" to adjust the size of the plot.

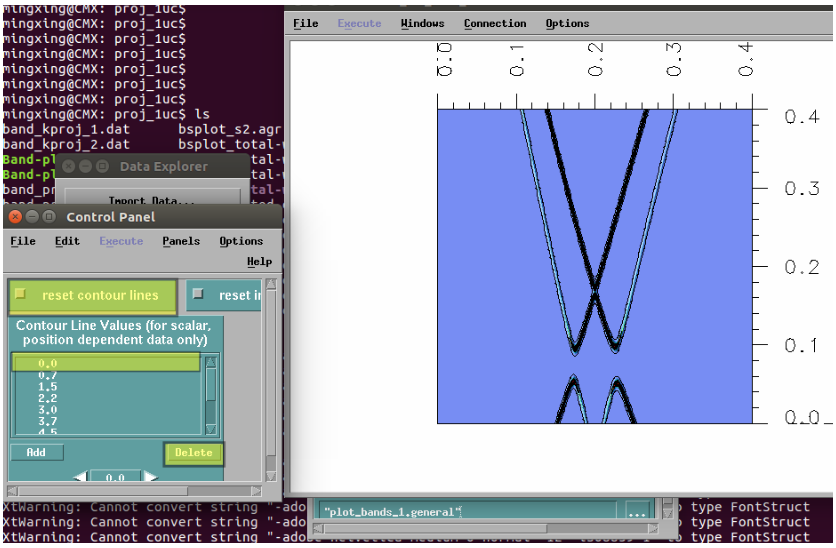
On the "Gridded_2D_1var_notseries" panel go to "file" and save image to a miff file. Click "reset contour lines" and delet all the contour line values except for the last one.

Likewise, the plot for the other spin can be done in the same way. One just needs to replace the string "plot_bands_1.general" by "plot_bands_1.general" on the control panel and click Enter.

Then go to a linux terminal and convert the miff to any type of image by running like "convert file.miff file.png".

Finally, you can further polish the plot using Inkscape or any other photoshop software.

A nice plot requires a large number of k-points for the band calculation, a dense energy mesh (**de**) and a suitable smearing factor (**dele**). In particular,



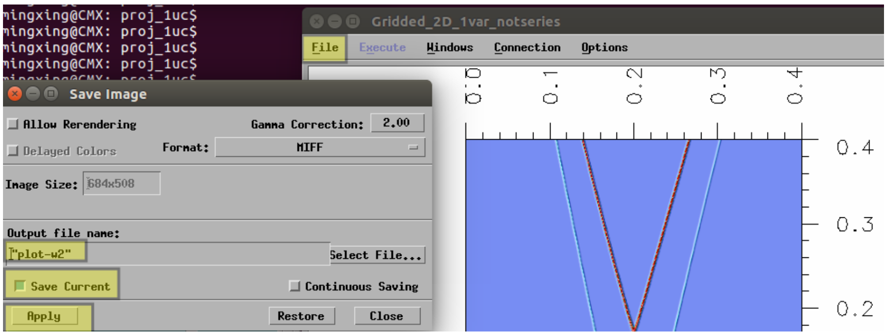
```
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$ ls
band_kproj_1.dat       bsplot_s2.agr
band_kproj_2.dat       bsplot_total-
Band-pl                tal-
Band-pl                tal-
band_pr                tal-
XtWarning: Cannot convert string "-ado
XtWarning: Cannot convert string "-ado
XtWarning: Cannot convert string "-ado
```

Contour Line Values (for scalar, position dependent data only)

0.0
0.7
1.5
2.2
3.0
3.7
4.5

"plot_bands_1.general"

Figure 4: Click ”reset contour lines” and delet the undesired contour lines.



```
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
mingxing@CMX: proj_1uc$
```

Save Image

□ Allow Rerendering Gamma Correction: 2.00

□ Delayed Colors Format: MIFF

Image Size: 684x508

Output file name: "plot-w2"

☑ Save Current □ Continuous Saving

Figure 5: Save the image.

when you are trying to make a plot for a very small energy window, the number of k-points has to be increased correspondingly.

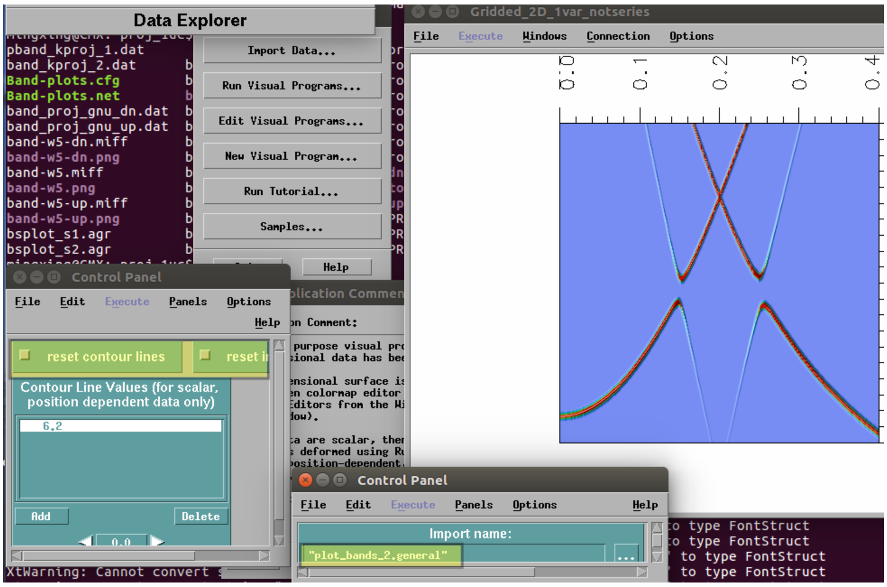
References

[1] M. Chen and M. Weinert, Phys. Rev. B **98**, 245421 (2018).

[2] M. X. Chen and M. Weinert, Nano Letters **14**, 5189 (2014).

[3] M. X. Chen and M. Weinert, Phys. Rev. B **94**, 035433 (2016).

[4] M. X. Chen, Z. Zhong, and M. Weinert, Phys. Rev. B **94**, 075409 (2016).



```
pband_kproj_1.dat
band_kproj_2.dat
Band-plots.cfg
Band-plots.net
band_proj_gnu_dn.dat
band_proj_gnu_up.dat
band-w5-dn.miff
band-w5-dn.png
band-w5.miff
band-w5.png
band-w5-up.miff
band-w5-up.png
bsplot_s1.agr
bsplot_s2.agr
XtWarning: Cannot convert
```

```
o type FontStruct
o type FontStruct
' to type FontStruct
' to type FontStruct
```

Figure 6: Plot for spin down.

[5] M. X. Chen, Z. Ge, Y. Y. Li, D. F. Agterberg, L. Li, and M. Weinert, Phys. Rev. B **94**, 245139 (2016).

[6] M. X. Chen, W. Chen, Z. Zhang, and M. Weinert, Phys. Rev. B **96**, 245111 (2017).

[7] S. Rajput, M. X. Chen, Y. Liu, Y. Y. Li, M. Weinert, and L. Li, Nature Communications **4**, 2752 (2013).

[8] Y. Y. Li, M. X. Chen, M. Weinert, and L. Li, Nature Communications **5**, 4311 (2014).

[9] W. Zhang, M. Chen, J. Dai, X. Wang, Z. Zhong, S. W. Cheong, and W. Wu, Nano Letters **18**, 2677 (2018).