Tutorial for the **bands4vasp** post-processing package

David Dirnberger

July 30, 2025

Contents

1 Introduction

In this tutorial we show how to use *bands4vasp* to get the effective band structure, Fermi vectors, the 2D- and 3D-Fermisurface. This informations are derived from the VASP output files of unfolding calculations[8] or spd- and site projection.[10] To demonstrate the functionality of *bands4vasp* we choose the Ru doped pnictide $BaFe_2As_2$ with a Ru concentration of 25% $\Rightarrow BaFe_{1.5}Ru_{0.5}As_2$. For more information about this material see the literature.[3], [5]–[7] All the functions and parameters of *bands4vasp* are described in detail in the bands4vasp_documentation.

2 Bands and Fermi vectors

To get the band structure of the system, there has to be at least one PRJCAR, PROCAR or PROCAR.prim file. In our example we choose the path $\Gamma - M - X - \Gamma - Z - R - A - Z$[3], [6] of symmetry points in the first Brillouin zone. To reproduce this example we made the VASP files (b4vasp_EBS_example.tar.gz) available at our GitHub page. First we go to the directory of the calculation data and copy the default INPAR file to the current directory with the following commands:

```
$ cd [folder-path]
$ b4vasp --inpar
```

In our example we set "PATHPOINTS = /G M X /G Z R A Z" to show the letters of the symmetry points at the x-axis. The data are organized in seperate directories for every single path (1unfolding, 2unfolding, ..., 7unfolding). To pass this structure and do a band structure analysis enter the following command:

```
$ b4vasp %unfolding
```

The band structure is visualized in the EBSbloch plot. To check the completeness of the calculated Fermi vectors we take a look at the EBSbloch.processed plot in the directory "./bands4vasp_img/".

(a) *complete band structure* (b) *band structure of the first path*

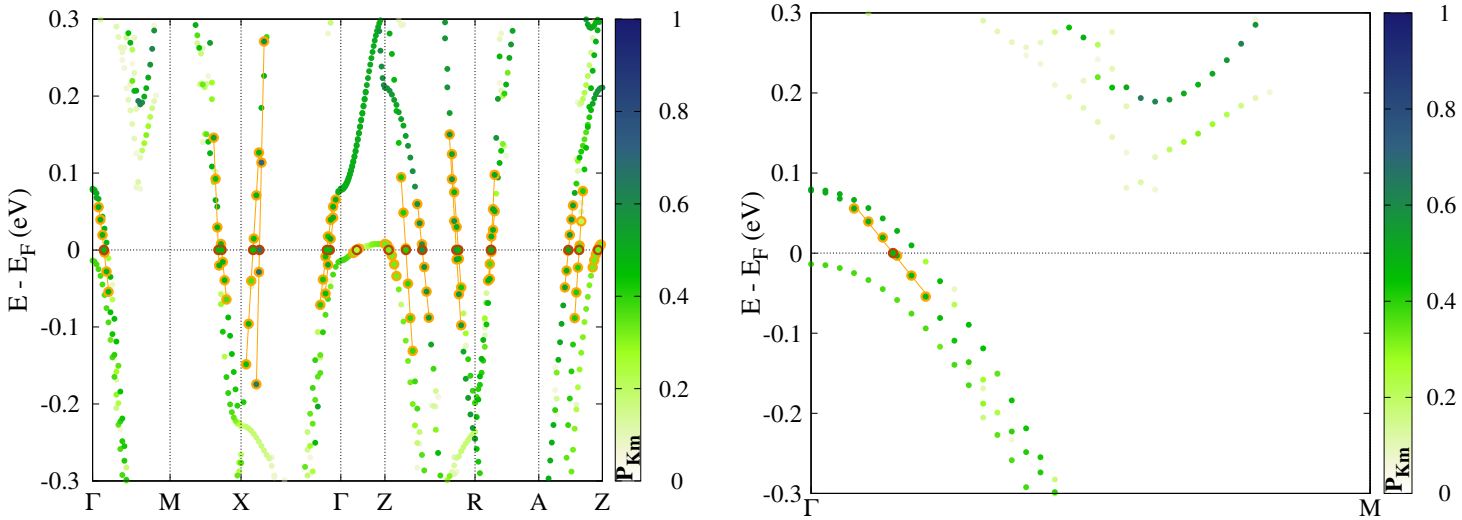

Figure 1: EBSbloch.processed plots

We can see in figure 1a) that there is one Fermi root missing in the first path. To get a closer look at the first path we pass the "1unfolding" folder.

```
$ b4vasp 1unfolding
```

In figure 1b) we can see that the Bloch character associated with the two eigenenstates of the second band crossing Fermi differs about 0.5. To consider this Bloch character difference we set "DBLOCH=0.5" in the INPAR file and

redo the calculation.
HINT: If you have a huge data set you can use the following command for recalculations:

```
$ b4vasp -rs
```

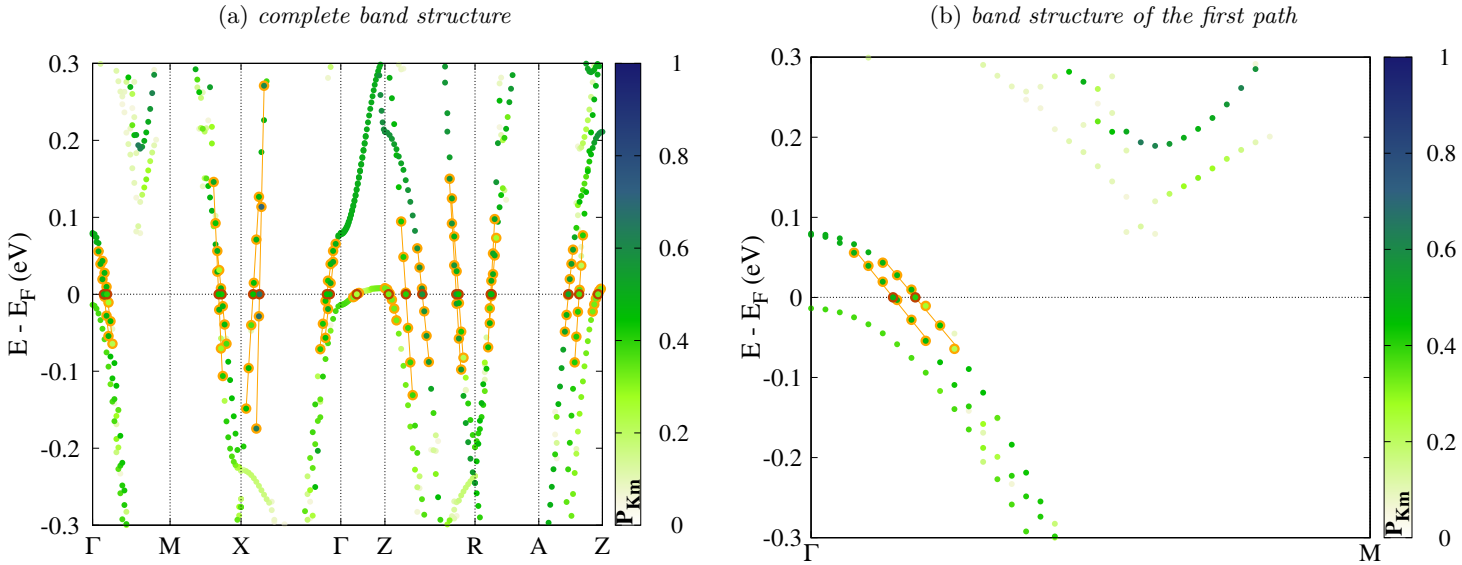

Figure 2: EBSbloch.processed plots

Now all Fermi roots are found correctly as we can see in figure 2. The data of the calculated Fermi vectors is written in *FERMIROOTS.dat*.

HINT: If more Fermi roots are missing or the band structure is more complex, one should use all general control parameters which regulate the root-finding-algorithm (EDELTA2, EDIF, EGAP, BLOCH_TRESHOLD, DBLOCH, ODISTINCT, GRADIENTD, NPOINTS, BNDDIFF, ...) and as well the orbital images, if available. For more information see the *bands4vasp_documentation*.

3 Fermi surface

Fermi surfaces can only be derived from VASP data, which must originate from line calculations. These lines can be arranged arbitrarily but must lie within the same plane. In our example, we performed 50 radial centered line calculations around the center and 60 line calculations around a corner in the Γ-plane of the Brillouin zone. To set up such a radial sampling, it is recommended to use the `b4vasp --pre-circle` command. This data set is organized into folders for each individual line calculation (1radial, 2radial, ..., 110radial). To reproduce this example, we made the VASP files available in the archive b4vasp_Fermisurface_example.tar.gz, which can be downloaded from our bands4vasp GitHub page.
Caution: Due to online storage limitations, only the PRJCAR files of this calculation were uploaded. This means that any results involving orbital character — which require the PROCAR and PROCAR.prim files — can only be reproduced if the example is redone from scratch with the appropriate LORBIT setting enabled in the VASP calculation.

Note: If you want to reproduce this example from scratch, you need to prepare a folder that contains all required input files for the unfolding calculation. These typically include INCAR, POSCAR, POSCAR.prim, POTCAR, CHGCAR, WAVECAR, and KPOINTS, although the exact set may vary depending on the type of calculation. Since the KPOINTS file defines the path and density of the k-point sampling, it needs to be prepared using the special #makepath flag followed by the desired sampling vectors (see the documentation at bands4vasp documentation).

For our example, the following KPOINTS file was used for the center:

```
k-gen
50
p
line
cart
#makepath
0 0 0
0.05000000 0.00000000 0.00000000
0.00000000 0.05000000 0.00000000
```

And the following one for the corners:

```
k-gen
50
p
line
cart
#makepath
0.12480477243662 0.1248047724366 0
0.07000000 0.17351550 0.00000000
0.17351550 0.07000000 0.00000000
```

For a full overview of the entire process — from system relaxation and self-consistent calculations to the unfolding step — we recommend reading the Unfolding Wiki, especially the example involving "B-doped cubic diamond Si": Unfolding patch documentation.

Once the folder is fully prepared with all required files and the correct KPOINTS sampling setup, the desired sampling can be initialized via bands4vasp.
First we copy the default INPAR file to the current directory with the following command:

```
$ b4vasp --inpar
```

The huge data set results in a high density of points in the images. To have a clear view at the important area around the Fermi level, we decrease the energy interval of interest EDELTA1 and EDELTA2 to 0.2. To have an idea how the Fermi surface should look like we also activate the spectral function SPECFUN = .TRUE. and for reasons of comparison we deactivate the interpolation of the spectral function "SLIMSPEC = .TRUE." and lower the smearing "SIGMA = 0.02".

```
$ b4vasp --fermi %radial
```

(a) *calculated Fermi surface* (b) *spectral function of the Fermi surface*

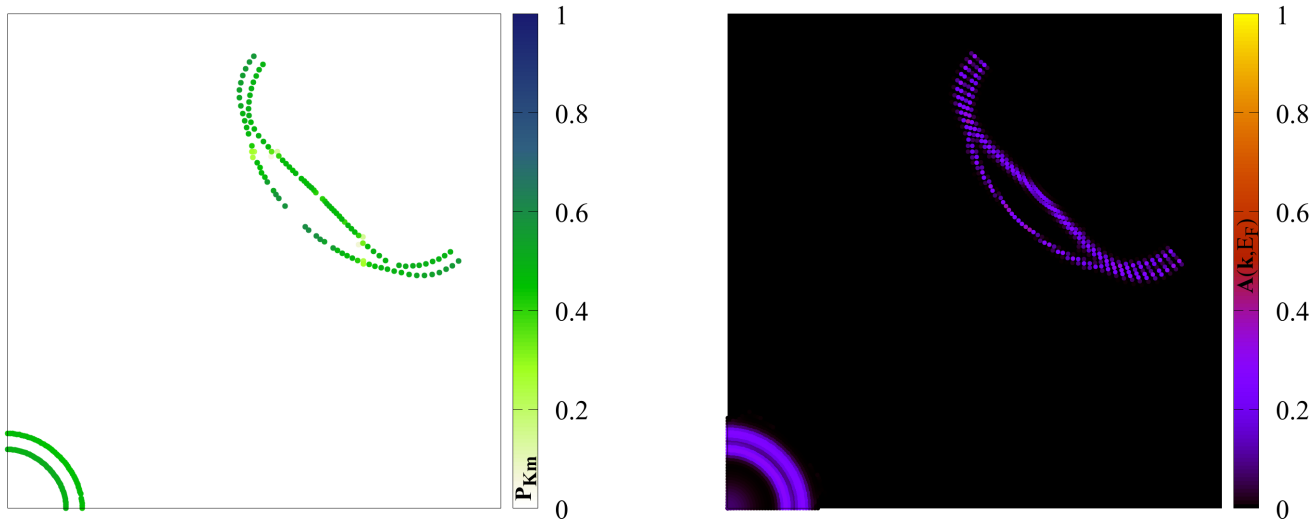

Figure 3: Comparison of calculated Fermi surface and Fermi surface spectral function

First we take a look at the Fermisurface_bloch and Fermisurface_bloch.spec images (figure 3) and see that the 'middle part' of the corner is partly missing. To get to know which parameter needs to be modified we take a look at the effective band structure with respect to the Bloch- and orbital character only for a path were the Fermi root is missing. Therefore we increase the point size in INPAR "PSFAC = 1.7", set the background color to gray "BACKCOLOUR = gray" and start the calculation.

```
$ b4vasp 80radial
```

(a) *EBSbloch* (b) *EBSbloch.processed*

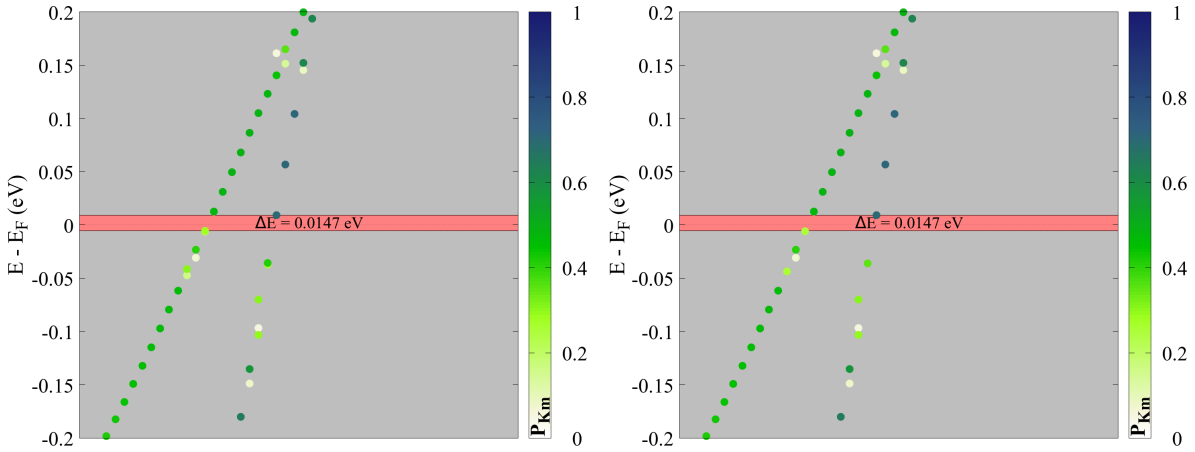

Figure 4: Comparison of the raw band structure with the processed one

Figure 4 reveals a Bloch character difference of about 0.5, so we set this value in the INPAR file 'DBLOCH = 0.5'. There are some eigenstates with a Bloch character near zero, which distort the averaging and thus the calculated Fermi roots. We increase the threshold for the Bloch character to avoid this "BLOCH_CHARACTER=0.15". The eigenstates in the right effective band in figure 4 are not well aligned. The parameter GRADIENTD is taking account of this issue "GRADIENTD = 0.23", but it is related with the EGAP parameter, so we set "EGAP = 0.075", which means that the maximum energy difference of two neighbouring states is 0.075 (for more information see the documantation). Because we have quite a lot of states in a small energy interval, we can set the number of involving eigenstates to 4 per side "NPOINTS = 4". The EBSorbit_ALL image shows all orbitals associated with the eigenstates of the system (figure 5). This image shows that there is a maximum orbital difference of about 0.4, so we also set this in INPAR "ODISTINCT = 0.4". Now we redo the calculation with the new parameters.

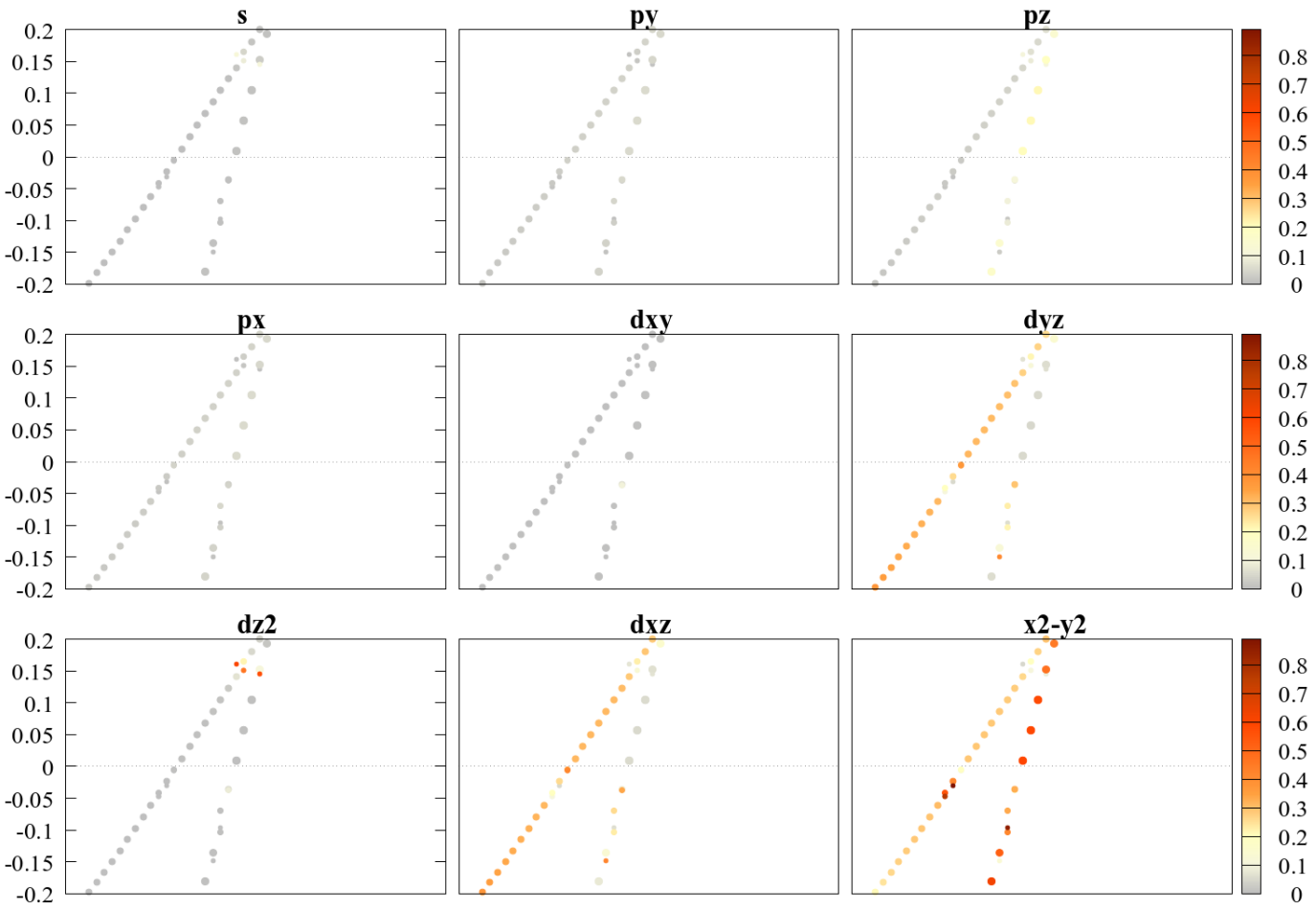

Figure 5: Effective band structure with color highlighted orbital character

```
$ b4vasp -rs
```

Figure 6a) shows an accurate detection of the two bands and as well the correct position of the calculated Fermi roots. Now we want to know if the adapted parameters fit for all bands and let therefore run b4vasp in the fermi mode over all data.

```
$ b4vasp --fermi %radial
```

Figure 6b) shows a well aligned Fermi surface with a simular structure as the spectral Fermi surface. This means we are almost done. At least we set the background back to white, take advantage of the symmetry of our system with "SYMPOINT1 = 0.0 0.0 0.0" and redo the calculation.

(a) *EBSbloch.processed of the single path*

(b) *Fermisurface_bloch*

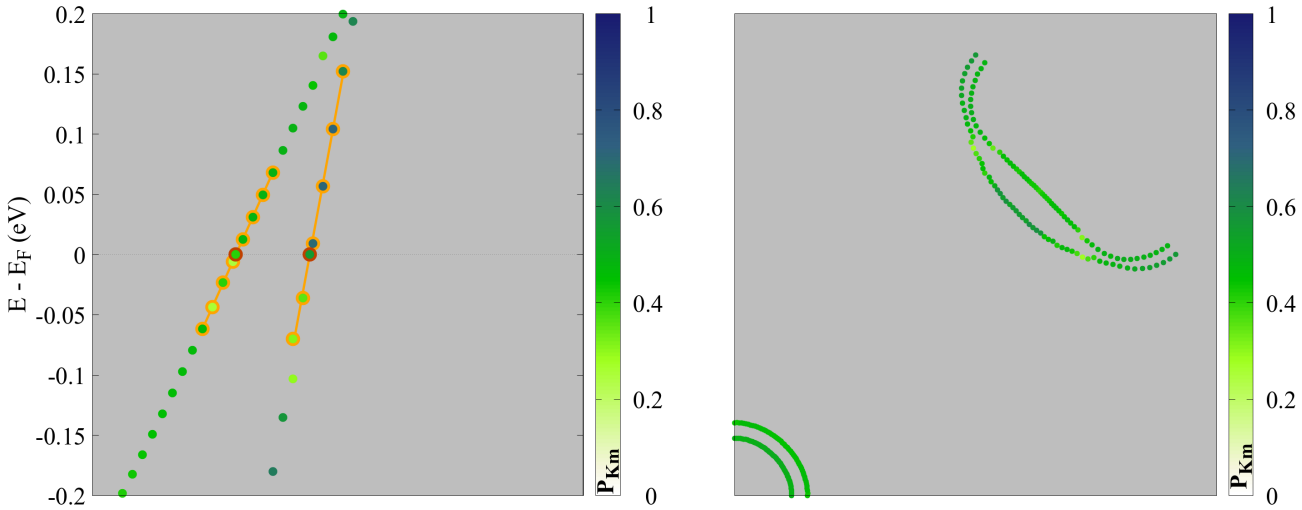

Figure 6: Second run, after changing the parameters

```
$ b4vasp -rs --fermi
```

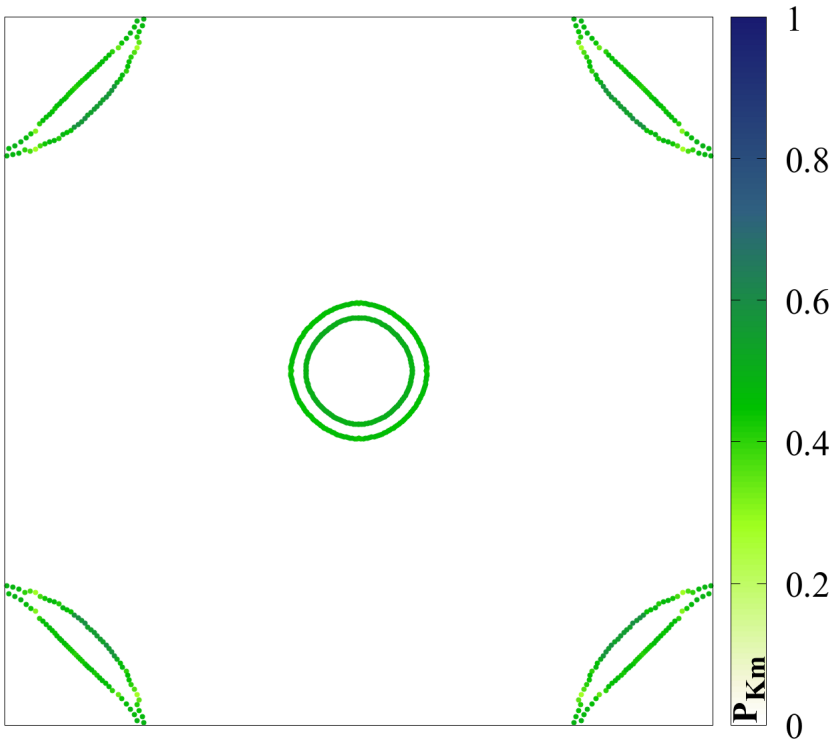

Figure 7: The final calculated Fermi surface

4 3D-Fermi surface

A 3D-Fermi surface is build of several 2D-Fermi surfaces. In our example we considered 9 equidistant surfaces in the first Brillouin zone. We also took advantage of the symmetry of our system and calculated the Fermi vectors only for the first quadrant as we did in section 3. After we have all calculated Fermi vectors (figure 8 a), we need to seperate the different surface areas. In our case they are describing deformed cylinders, that is the reason why we did a coordinate transformation to cylindrical coordinates. We seperated them by there radius, the Bloch and the orbital character. Once we have these seperated surfaces (figure 8 b), we use the hight and the angle for a bilinear interpolation of all other values (radius, Bloch- and orbital character) (figure 8 c). Now we are able to tranform the interpolated data back into cartesian coordinates and obtain the 3D-Fermi surface (figure 8).

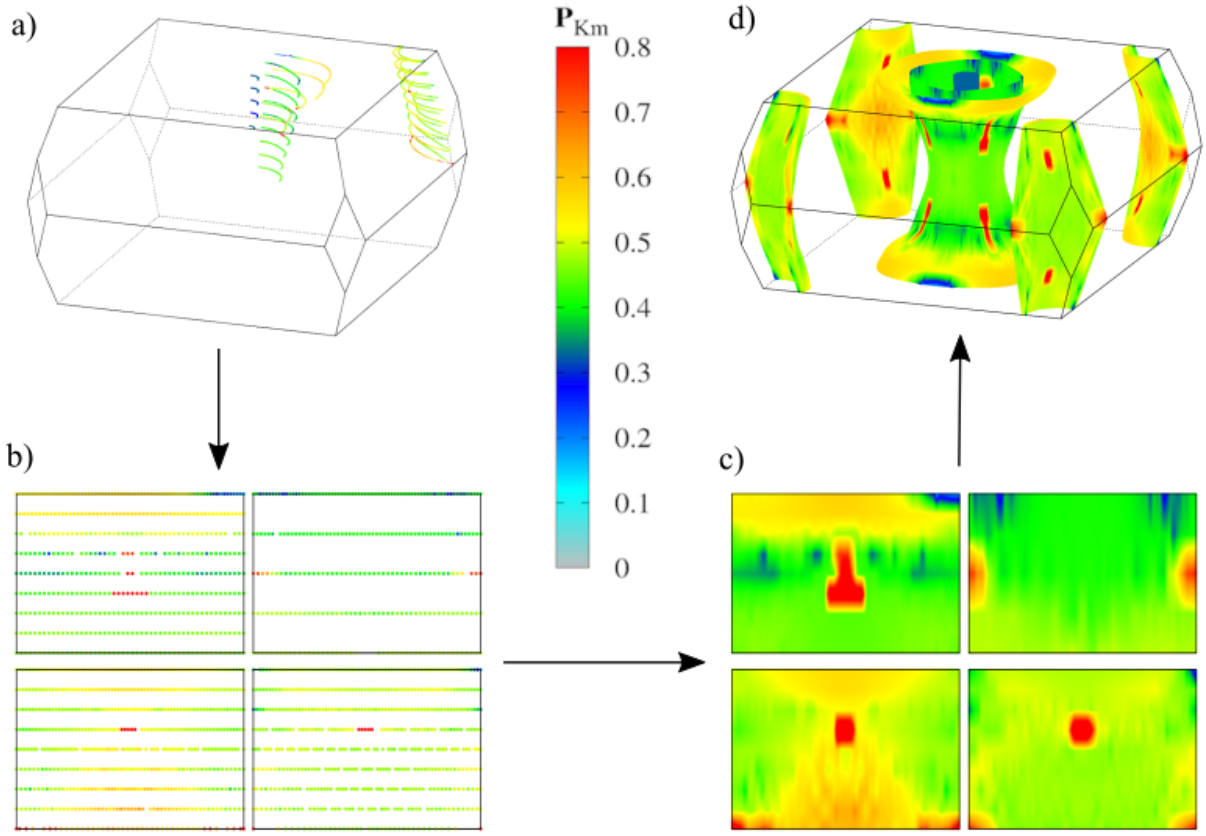

Figure 8: Method of deriving the 3d-Fermi surface

References

[1] *bands4vasp GitHub page*
https://github.com/QuantumMaterialsModelling/bands4vasp

[2] *Unfolding Patch for VASP GitHub page*
https://github.com/QuantumMaterialsModelling/UnfoldingPatch4vasp

[3] David Dirnberger; Georg Kresse; Cesare Franchini; and Michele Reticcioli. *Electronic State Unfolding for Plane Waves: Energy Bands, Fermi Surfaces, and Spectral Functions.* The journal of physical chemistry C.

[4] David Dirnberger. *Electronic state unfolding for plane waves, an automatized computational implementation* https://utheses.univie.ac.at/detail/69219#

[5] Reticcioli, M.; Profeta, G.; Franchini, C.; Continenza, A. *Effective band structure of Ru-doped BaFe2As2.* J. Phys.: Conf. Ser. 2016, 689, 012027.

[6] Reticcioli, M.; Profeta, G.; Franchini, C.; Continenza, A. *Ru doping in iron-based pnictides: The "unfolded" dominant role of structural effects for superconductivity.* Phys. Rev. B: Condens. Matter, Mater. Phys. 2017, 95, 214510.

[7] Wang, L.; Berlijn, T.; Wang, Y.; Lin, C. H.; Hirschfeld, P. J.; Ku, W. *Effects of disordered Ru substitution in BaFe2As2: Possible realization of superdiffusion in real materials.* Phys. Rev. Lett. 2013, 110, 037001.

[8] *VASP wiki for the unfolding method*
https://www.vasp.at/wiki/index.php/LKPROJ

[9] *VASP wiki for the KPOINTS file*
https://www.vasp.at/wiki/index.php/KPOINTS

[10] *VASP wiki for the spd- and site projection*
https://www.vasp.at/wiki/index.php/LORBIT