Topological properties with WannierTools

Tutorial III: Weyl semimetal

Hands-on session

Hands-on based on WannierTools v2.5.1

Useful information about WannierTools:
Documentation: www.wanniertools.com
Forum: www.wanniertools.org
Open source: https://github.com/quanshengwu/wannier_tools

Tips: Open file.eps, file.pdf file in Linux, you can use command "evince". such as

```
$ evince file.eps
```

Open file.png file in Linux, you can use command "eog". such as

```
$ eog file.png
```

In this session we will study Weyl semimetal using WannierTools.

Weyl semimetal from a toy model

Weyl point is the most stable topological object since it doesn't require any symmetry protection. Let's start from a toy model Hamiltonian.

$$H = A(k_x\sigma_x + k_y\sigma_y) + [M_0 - M_1(k_x^2 + k_y^2 + k_z^2)]\sigma_z \tag{1}$$

which is a minimal model for 3D Weyl semimetal hosting only two Weyl points along z direction. The time reversal symmetry is broken. In this tutorial, we set $A = M_0 = M_1 = 1$.

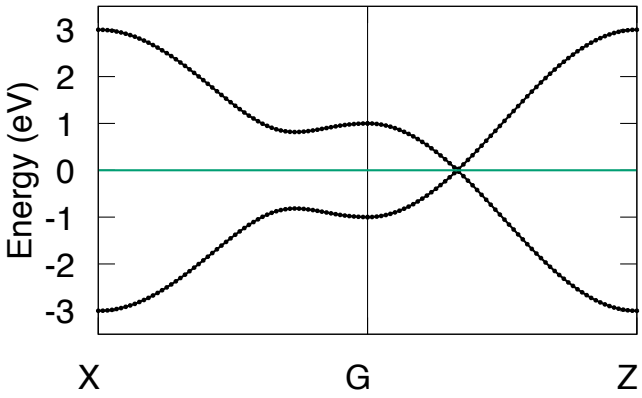
▶ Preparation First, please copy the tutorial input files of the exercise:

```
$ git clone https://github.com/wannier-developers/wannier-tutorials.git
$ cd wannier-tutorial/2020_03_Oxford/1_topological/Weyl_semimetal/3DWeyl-model/
```

▶ Calculate band structure first.

```
$ cp wt.in-bands wt.in
$ wt.x &
$ gnuplot bulkek.gnu
```

The band structure looks like this:



It's shown that there is a crossing point along G-Z direction.

▶ Find all the Weyl points

Usually, Weyl points are not along the high-symmetry line. We need to find them all through setting *F*indNodes_calc = T and setting *N*umOccupied = 1.

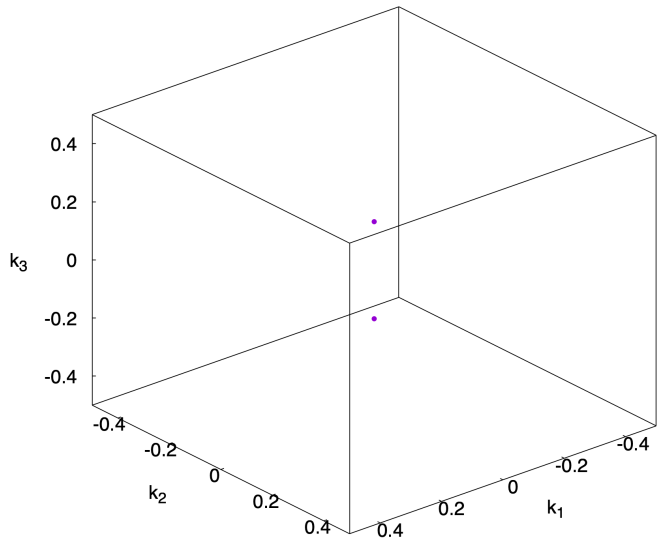
```
$ cp wt.in-findnodes wt.in
$ wt.x &
$ gnuplot Nodes.gnu
```

Weyl points are also called nodes here. The nodes data is included in the Nodes.dat file which look like

Nodes.dat

```
--
# local minimal position and the related energy gap
#        kx            ky            kz            gap           E             k1            k2            k3
    0.00000000    0.00000000    1.04719755    0.00000000   -0.00000000    0.00000000    0.00000000    0.16666667
    0.00000000   -0.00000000   -1.04719755    0.00000000   -0.00000000    0.00000000   -0.00000000   -0.16666667
```

The plot Nodes.png looks like this.

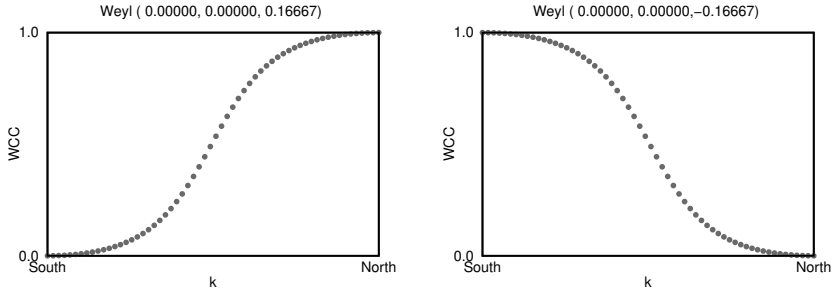


▶ Calculate chirality of Weyl points.

In order to calculate the chirality, we need give an input of the coordinates of Weyl points into the *wt.in* file. Those coordinates can be found in Nodes.dat calculated above. We prepared an input file *w*t.in-chirality.

```
$ cp wt.in-chirality wt.in
$ wt.x &
$ gnuplot wanniercenter3D_Weyl_1.gnu
$ gnuplot wanniercenter3D_Weyl_2.gnu
```

The WCCs files of the ball wrapping the individual Weyl point are called *w*anniercenter3D_Weyl_1.eps and *w*anniercenter3D_Weyl_2.eps which are shown as



which give us the chirality +1 and -1. The chirality of Weyl points can also be directly read from *W*T.out.

```
$ sed -n '/Chiralities/,/Time/p' WT.out
```

The results should look like this

```
--                                                                                                    WT.out
Chiralities
#       k1          k2          k3          kx          ky          kz       Chirality
   0.00000    0.00000    0.16667    0.00000   -0.00000    1.04720        1
   0.00000    0.00000   -0.16667    0.00000    0.00000   -1.04720       -1
```

▶ Calculate Berry curvature

The chirality of Weyl points can also been shown in the Berry curvature. Now we calculate the Berry curvature in $k_y = 0$ plane by setting **KPLANE_BULK** as

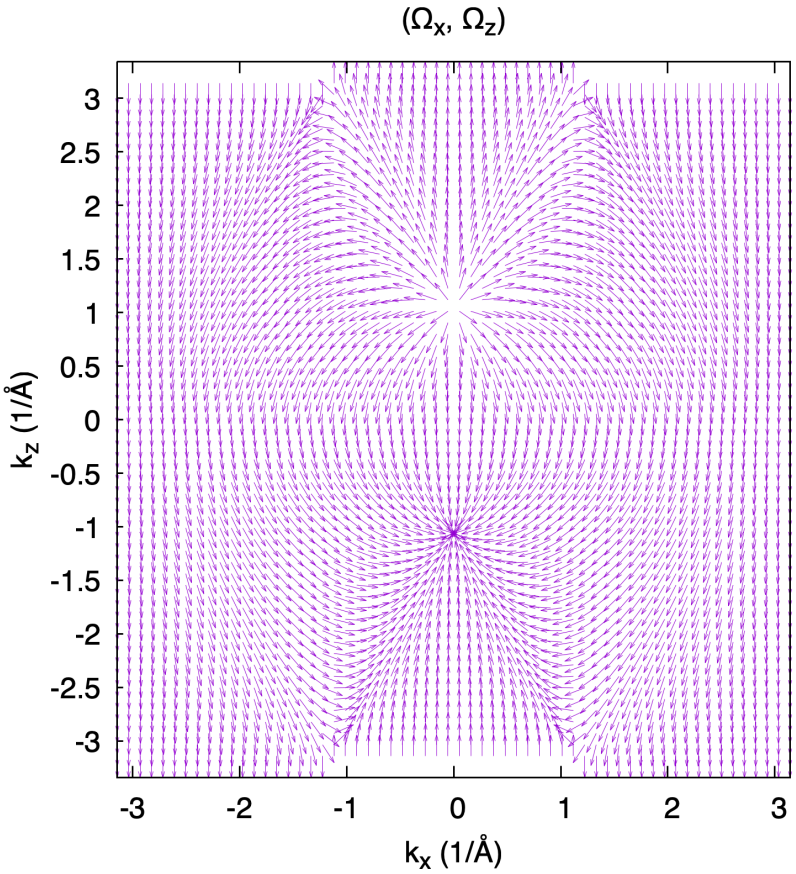
```
KPLANE_BULK
Direct
 0.00  0.00  0.00   ! Center of 3D k slice
 1.00  0.00  0.00   ! The first vector along x direction
 0.00  0.00  1.00   ! The second vector along z direction

$ cp wt.in-Berry-curvature wt.in
$ wt.x &
```

The Berry curvature is diverged at the location of Weyl points. So we normalized the Berry curvature vector $\tilde{\Omega}(\mathbf{k}) = \Omega(\mathbf{k})/|\Omega(\mathbf{k})|$. The normalized Berry curvature is written in *B*errycurvature-normalized.dat. Now we need to modify the gnuplot script *B*errycurvature-normalized.dat to plot the data. Here we prepared one script to plot it.

```
$ gnuplot Berrycurvature-normalized.gnu-tutorial
```

The generated plot *Berrycurvature-normalized.png* should look like this.



The Weyl point with positive chirality is the source of Berry curvature, while the one with negative chirality is the sink of Berry curvature.

▶ **Calculate surface state spectrum** We are going to study the side surface (010) to study the Fermi arc states since the two Weyl points are in the z axis. The **SURFACE** card is set as

```
SURFACE           ! (010) surface
 1  0  0
 0  0  1
```

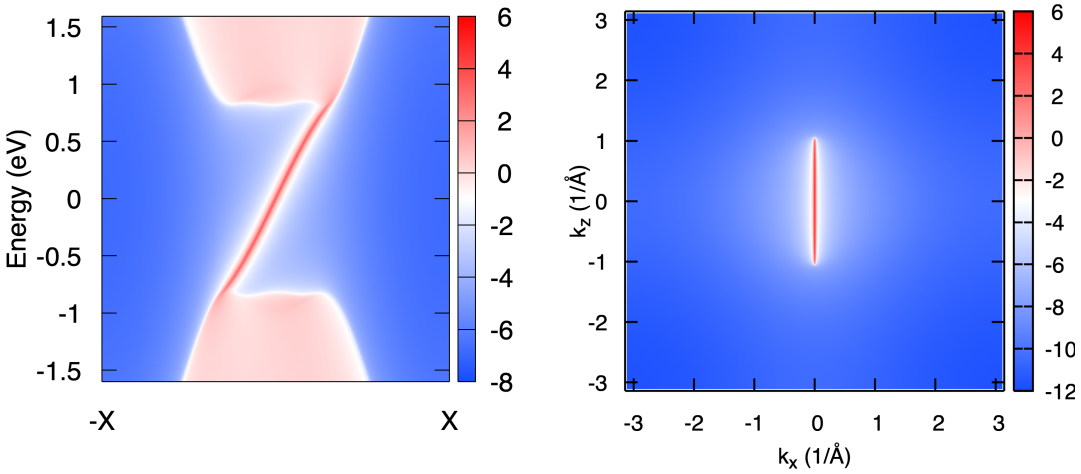
Run WannierTools:

```
$ cp wt.in-surfacestates wt.in
$ wt.x &
```

The surface state spectrum in $k-E$ and $k-k$ mode are generated with

```
$ gnuplot surfdos_l.gnu
$ gnuplot arc_l.gnu
```

The generated file *s*urfdos_l.png and *a*rc_l.png should look like this.



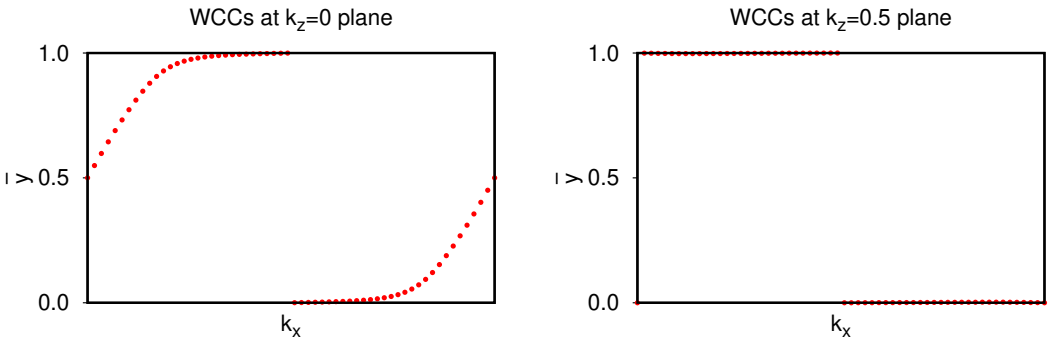
The surface state spectrum in $k-E$ mode shows that it is chiral which means that the velocity is always either positive or negative. The surface state spectrum in $k-k$ mode shows that the Fermi surface of the surface state is not close. It originate from one Weyl point and terminated at another Weyl point. This kind of Fermi surface is known as Fermi arc states.

▶ **Use WCCs to understand Fermi arc**

The position of the Weyl points is at (0, 0, $\pm k_{z0}$). The Fermi arc only shows up between two Weyl points. This can be understand from the WCCs. Let's study WCCs both $k_z = 0$ and $k_z = 0.5$ plane.

```
$ cp wt.in-wcc-kz0 wt.in
$ wt.x &
$ gnuplot wcc.gnu
$ cp wcc.eps wcc-kz0.eps
$ cp wt.in-wcc-kz0.5 wt.in
$ wt.x &
$ gnuplot wcc.gnu
$ cp wcc.eps wcc-kz0.5.eps
```

Two plots are shown as



The WCCs plots show that, the Chern number at $k_z = 0$ plane is 1 which will give us chiral mode, while the Chern number at $k_z = 0.5$ plane is 0 which will not give us topological protected surface states. That's the topological origin of the Fermi arc.

▶ **Exercises**

1. Calculate WCCs at different k_z plane by setting the original point of **KPLANE_BULK**. such as $k_z = 0.1, 0.2, 0.3$.

2. Study the surface states at (100) surface by setting **SURFACE** card as

```
SURFACE            ! (100) surface
 0  1  0
 0  0  1
```

3. Study the surface states at (001) surface by setting **SURFACE** card as

```
SURFACE            ! (001) surface
 1  0  0
 0  1  0
```

Weyl semimetal TaAs

Now we can start to study a real Weyl semimetal. TaAs is a famous Weyl semimetal both confirmed from theoretical calculation and ARPES measurement. Due to the system is quite large, in this tutorial we will only show you how to find all Weyl points.

▶ **Preparation** First, please copy the tutorial input files of the exercise:

```
$ git clone https://github.com/wannier-developers/wannier-tutorials.git
$ cd  wannier-tutorial/2020_03_Oxford/1_topological/Weyl_semimetal/TaAs/
$ tar xzvf wannier90_hr.dat.tar.gz
```

▶ Run WannierTools to find all Weyl points

```
$ cp wt.in-findnodes wt.in
$ mpirun -np 2 wt.x &
```

After the calculation, you will find all 24 nodes in file *N*odes.dat shown as:

```
--                                                                                  Nodes.dat
#        kx            ky            kz           gap            E             k1            k2            k3
   -0.93249310   -0.01476294    0.00000000    0.00000000    0.23118145   -0.51008822   -0.00807556    0.25908189
    0.93249310    0.01476294    0.00000000    0.00000000    0.23118145    0.51008822    0.00807556   -0.25908189
   -0.93249310    0.01476294    0.00000000    0.00000000    0.23118145   -0.51008822    0.00807556    0.25100633
    0.93249310   -0.01476294    0.00000000    0.00000000    0.23118145    0.51008822   -0.00807556   -0.25100633
   -0.51671209    0.03600882   -0.31744043    0.00000000    0.23782847   -0.28264954    0.01969738   -0.16296738
   -0.51671209   -0.03600882   -0.31744043    0.00000000    0.23782847   -0.28264955   -0.01969738   -0.14327000
   -0.51671209    0.03600882    0.31744043    0.00000000    0.23782847   -0.28264954    0.01969738    0.42591955
   -0.03600882    0.51671209   -0.31744043    0.00000000    0.23782847   -0.01969738    0.28264954   -0.42591955
   -0.03600881    0.51671209    0.31744043    0.00000000    0.23782847   -0.01969738    0.28264954    0.16296738
    0.03600882    0.51671209    0.31744043    0.00000000    0.23782847    0.01969738    0.28264954    0.14327000
   -0.03600882   -0.51671209    0.31744043    0.00000000    0.23782847   -0.01969738   -0.28264955    0.44561693
   -0.51671209   -0.03600881    0.31744043    0.00000000    0.23782847   -0.28264955   -0.01969738    0.44561693
   -0.01476294   -0.93249310   -0.00000000    0.00000000    0.23118145   -0.00807556   -0.51008822    0.25908189
    0.01476294    0.93249310   -0.00000000    0.00000000    0.23118145    0.00807556    0.51008822   -0.25908189
    0.01476294   -0.93249310    0.00000000    0.00000000    0.23118145    0.00807556   -0.51008822    0.25100633
    0.03600882   -0.51671209   -0.31744043    0.00000000    0.23782847    0.01969738   -0.28264955   -0.16296738
    0.03600882   -0.51671209    0.31744043    0.00000000    0.23782847    0.01969738   -0.28264955    0.42591955
   -0.03600882   -0.51671209   -0.31744043    0.00000000    0.23782847   -0.01969738   -0.28264954   -0.14327000
    0.03600882    0.51671209   -0.31744043    0.00000000    0.23782847    0.01969738    0.28264955   -0.44561693
    0.51671209   -0.03600882   -0.31744043    0.00000000    0.23782847    0.28264955   -0.01969738   -0.42591955
    0.51671209    0.03600882   -0.31744043    0.00000000    0.23782847    0.28264954    0.01969738   -0.44561693
    0.51671209   -0.03600882    0.31744043    0.00000000    0.23782847    0.28264955   -0.01969738    0.16296738
    0.51671209    0.03600882    0.31744043    0.00000000    0.23782847    0.28264955    0.01969738    0.14327000
   -0.01476294    0.93249310   -0.00000000    0.00000000    0.23118145   -0.00807556    0.51008822   -0.25100633
```

▶ Exercises

1. Calculate chirality of all Weyl points.
2. Try to study the surface state of (001) surface defined by **SURFACE** card

```
SURFACE            ! (100) surface
 1  0  0
 0  1  0
```

This could be time consuming. An example of the input file *w*t.in-surfacestates is prepared. You can modified the parameters to get high-resolution surface state plot.