Selected Column of the Density Matrix in wannier90

wannier90 2022 School — ICTP, Trieste

Hands-on tutorial

Valerio Vitale

May 18, 2022

Introduction

In this tutorial you will learn how to generate MLWFs starting from SCDM initial guesses, without the need of specifying initial projections. You will be using gallium arsenide as an example to learn how to wannierise isolated bands and entangled bands with SCDM, and lead as an example for automated SCDM via projectabilities. We will be using QUANTUM-ESPRESSO to generate Bloch states, band structures etc., `wannier90`to obtain MLWFs and interpolated band structures and `pw2wannier90.x` to interface DFT data from QUANTUM-ESPRESSO and `wannier90`. PART I and PART II are based on the 2020 `wannier90` 3.0 Workshop tutorial, given by Anil Damle, which in turn is based on `Example 27` of the `wannier90` repository, whereas PART III is based on the `Excerise 4` of the 2021 `wannier90` tutorial for the EPW School, given by Giovanni Pizzi and Junfeng Qiao. All tutorials from this School and previous Schools/Workshops can be found on the wannier-developers GitHub repository: https://github.com/wannier-developers/wannier-tutorials

✦	General info
☞	Main Task
▶	sub task
✍	Optional task
`text`	Text to be added to input file
text	Button in `Xmgrace`

Table 1: List of symbols and their meaning

Preliminaries

By now you should have already cloned the repository with all the tutorials. If you have not done it yet, simply type the following in your terminal:

```
$> git clone https://github.com/wannier-developers/wannier-tutorials.git
```

and navigate through the folders to today's tutorial folder

```
$> cd wannier-tutorials/2022_05_Trieste/DAY3_PM_1_SCDM/
```

PART I – SCDM for isolated bands: Top 4 valence bands of gallium arsenide

Outline: Obtain MLWFs Wannier functions out of top four isolated valence bands of gallium arsenide, starting from initial guesses obtained with the SCDM method.

Directories: `Isolated/`
`pseudo/`

Description of input files

- DFT Input Files (common to both TASK I and TASK II)

  `GaAs.scf.in` PWSCF *input file for DFT ground state calculation*

  `GaAs.bands.in` PWSCF *input file to obtain Bloch states on a specific path in k-space*

  `GaAs.nscf.in` PWSCF *input file to obtain Bloch states on a uniform* $4\times4\times4$ *k-point grid*

  `GaAs.bandsx.in` *input file for* `bands.x`

- Input Files to generate Wannier functions

  `GaAs.pw2wan.in` *input file for* `pw2wannier90` *interface*

  `GaAs.win` *input file for* `wannier90`

- Input Files for plotting

  `GaAs_band.gnu` *gnuplot file to plot* `wannier90` *interpolated bands.*

  `DFT_band.dat` *DFT bands in a format suitable for plotting (optional)*

☞ Run DFT calculations with QUANTUM-ESPRESSO

- cd into `Isolated/`
- Run PWSCF to obtain the ground state density of GaAs
  ```
  $> mpirun -np 2 /media/ictpuser/AiiDA/bin/pw.x -in GaAs.scf.in | tee scf.out
  ```

- Run PWSCF to obtain the Bloch states on a specific path
  `$> mpirun -np 2 /media/ictpuser/AiiDA/bin/pw.x -in GaAs.bands.in | tee bands.out`
- Run `bands.x` to generate a dat file for plotting
  `$> /media/ictpuser/AiiDA/bin/bands.x -in GaAs.bandsx.in | tee bandsx.out`
- Run PWSCF to obtain the Bloch states on a uniform k-point grid
  `$> mpirun -np 2 /media/ictpuser/AiiDA/bin/pw.x -in GaAs.nscf.in | tee nscf.out`

☞ Compute initial guesses with SCDM method

- Inspect the `GaAs.pw2wan.in` input file. You will need to add the SCDM keywords for isolated bands, i.e.

```
scdm_proj = .true.
scdm_entanglement = 'isolated'
```

  For isolated bands there is no need to select the extra two parameters μ and σ if these bands are the only bands in the energy window we are interested in, and we want to wannierise all of them, i.e. the number of Wannier functions is equal to the number of bands.

- Inspect the `wannier90` input file `GaAs.win`. You will need to add the keyword for the automated projections, i.e.

```
auto_projection = .true.
```

  Note there is no projections block and we simply set `auto_projections = .true.` Since the top four valence bands are isolated in gallium arsenide, to run the SCDM with, we simply need to exclude the bands from the core states and the conduction bands from our set of bands, i.e.

```
exclude_bands= 1-5, 10-13
```

  Run `wannier90` to generate a list of the required overlaps, which will be written into the `GaAs.nnkp` file (no need to run this in parallel).
  `$> /media/ictpuser/AiiDA/bin/wannier90.x -pp GaAs`
- Run `pw2wannier90` to compute the overlap between Bloch states and the initial projections $A_{mn}^{\mathbf{k}}$ from the SCDM method, that will be written in the `GaAs.mmn` and `GaAs.amn` files, respectively.
  `$> mpirun -np 2 /media/ictpuser/AiiDA/bin/pw2wannier90.x < GaAs.pw2wan.in | tee pw2wan.out`
- Run `wannier90` to compute the MLWFs from SCDM initial guesses.
  `$> mpirun -np 2 /media/ictpuser/AiiDA/bin/wannier90.x GaAs | tee GaAs.wout`
- Inspect the `GaAs.wout` output file and check that the calculation has converged. Take note of the final total spread and individual WFs spreads.

- Plot the DFT band-structure and the band-structure from Wannier interpolation. This can be done using `gnuplot` as
  ```
  $> gnuplot
  gnuplot> load "GaAs_band.gnu"
  gnuplot> replot "bands.dat.gnu" u ($1*1.1117):2 w l lt -1, "GaAs_band.dat" lt 1
  ```
  N.B. Because of different internal untis in `bands.x` and `wannier90` the two bands output files, e.g. `bands.dat.gnu` from `bands.x` and `GaAs_band.dat` from `wannier90` are not directly comparable, but you need to rescale the x-axis, that's why there's a factor of 1.1117.

- Look at the MLWFs with VESTA or XCrysden (a good isovalue is 0.5), e.g.
  ```
  $> xcrysden --xsf GaAs_0000*.xsf
  ```

- To explore the SCDM implementation try varying the number of iterations you let `wannier90` run for and compare the spread along with the visual character of the Wannier functions.

- ✍ Try using random projections and 4 sp_3 orbitals as an initial guesses, what differences do you see?

PART II – SCDM for entangled bands: Top 4 valence and 3 low-lying conduction bands of GaAs

Outline: Compute MLFWs for the top 4 valence bands and 3 low-lying conduction bands, using the SCDM method to generate the initial guesses.

Directories: `Entangled/`
`pseudo/`

You can repeat steps 2-5 from TASK I, or you can simply copy the DFT output files from the `Isolated` folder.

☞ Compute initial guesses with SCDM method

- Inspect the `GaAs.pw2wan.in` input file. For entangled bands one needs to define a filtering function, e.g. erfc(μ, σ) or Gaussian(μ, σ), and select the two parameters μ and σ that define the function, as discussed in the lecture. In particular for you will need to add the following SCDM keywords

  ```
  scdm_proj = .true.
  scdm_entanglement = 'erfc'
  scdm_mu = 11
  scdm_sigma = 4
  ```

  Can you explain why these parameters on the basis of your understanding of the SCDM method?

- Inspect the wannier90 input file GaAs.win. Now we have set num_wann = 7 and num_bands = 8 (13 - 5). Again, you will need to add the keyword for the automated projections, without any projection block, i.e.

  ```
  auto_projection = .true.
  ```

  This time we want to retain the conduction bands and only exclude the bands from the core states

  ```
  exclude_bands= 1-5
  ```

- Run wannier90 to generate a list of the required overlaps, which will be written into the GaAs.nnkp file (no need to run this in parallel).
  ```
  $> /media/ictpuser/AiiDA/bin/wannier90.x -pp GaAs
  ```

- Run pw2wannier90 to compute the overlap between Bloch states $M^{\mathbf{k}}_{mn}$ and the initial projections $A^{\mathbf{k}}_{mn}$ from the SCDM method, that will be written in the GaAs.mmn and GaAs.amn files, respectively.
  ```
  $> mpirun -np 2 /media/ictpuser/AiiDA/bin/pw2wannier90.x < GaAs.pw2wan.in | tee pw2wan.out
  ```

- Run wannier90 to compute the MLWFs from SCDM initial guesses.
  ```
  $> mpirun -np 2 /media/ictpuser/AiiDA/bin/wannier90.x GaAs
  ```

- Inspect the GaAs.wout output file and make sure that both the disentanglement step and the spread minimization step have converged. Take note of the final total spread and individual WFs spreads.

- Plot the DFT band structure and the band structure from Wannier interpolation. This can be done in, e.g., gnuplot
  ```
  $> gnuplot
  gnuplot> load "GaAs_band.gnu"
  gnuplot> replot "bands.dat.gnu" u ($1*1.1117):2 w l lt -1, "GaAs_band.dat" lt 1
  ```

- Look at the MLWFs with VESTA or XCrysden, e.g.
  ```
  $> xcrysden --xsf GaAs_0000*.xsf
  ```

- To explore the SCDM implementation try varying the number of iterations you let wannier90 run for (both the disentanglement and the localization) and compare the spread along with the visual character of the Wannier functions.

- ✍ Try using random projections, what differences do you see? Try to come up with a good set of seven initial projections.

PART III – Automated MLWFs via SCDM + projectabilities

Outline: Compute the Wannier interpolated band structure of lead, using the automated method to generate the initial guesses, i.e. SCDM + data from projectabilities. There are

three free parameters in the SCDM method, namely μ, σ and the number of bands N_b to wannierise. The first two are obtained by a fitting a complementary error function to the projectabilities, whereas the number of bands is given by the number of pseudo-atomic orbitals (PAOs) in the pseudopotentials, 9 in this case.

Directories: `Projectabilities/`
`pseudo/`

- Run PWSCF to obtain the ground state density of lead
  `$> mpirun -np 2 /media/ictpuser/AiiDA/bin/pw.x -in Pb.scf.in | tee scf.out`
- Run PWSCF to obtain the Bloch states on a specific path
  `$> mpirun -np 2 /media/ictpuser/AiiDA/bin/pw.x -in Pb.bands.in | tee bands.out`
- Run `bands.x to` generate a dat file for plotting
  `$> /media/ictpuser/AiiDA/bin/bands.x -in Pb.bandsx.in | tee bandsx.out`
- Run PWSCF to obtain the Bloch states on a uniform k-point grid
  `$> mpirun -np 4 /media/ictpuser/AiiDA/bin/pw.x -in Pb.nscf.in | tee nscf.out`
  N.B. This will take slightly longer than the gallium arsenide case, since we are using an $8 \times 8 \times 8$ k-grid instead of a $4 \times 4 \times 4$ one and the system is metallic (it will take few minutes). This is also the reason why we are running it using 4 MPI processes. Don't worry about the warning `c_bands: 1 eigenvalues not converged`

☞ Automatically find $\mu_{\rm SCDM}$ and $\sigma_{\rm fit}$ by using the protocol defined in Ref. 3

- Run `projwfc` to compute the projectabilities of the Bloch states onto the Bloch sums obtained from the PAOs in the pseudopotential
  `$> /media/ictpuser/AiiDA/bin/projwfc.x -in Pb.proj.in > proj.out`
- Run `generate_weights` to extract the projectabilitites from `proj.out` in a format suitable to be read by `Xmgrace` or `gnuplot`
  `$> ./generate_weights.sh`
- Plot the projectabilities and fit the data with the complementary error function

Xmgrace mini tutorial

We are going to use `Xmgrace` to plot the projectabilities and perform the fitting. Open `Xmgrace` by simply typing
`$> xmgrace`
To import the `p_vs_e.dat` file, click on **Data** from the top bar and then **Import -> ASCII...**. At this point a new window **Grace: Read sets** should pop up. Select `p_vs_e.dat` in the **Files** section, click **Ok** at the bottom and close the window. You should now be able to see a quite noisy function that is bounded between

1 and 0. You can modify the appearence of the plot by clicking on **Plot** in the top bar and then **Set appearance...**. In the **Main** section of the pop-up window change the symbol type from **None** to **Circle**. Change the line type from **straight** to **none**, since the lines added by default by `Xmgrace` are not meaningful. For the fitting, go to **Data -> Transformations -> Non-linear curve fitting**. In this window, select the source from the **Set box** and in the **Formula** box insert the following

```
y = 0.5 * erfc( ( x - A0 ) / A1 )
```

Select **2** as number of parameters, give 15 as initial condition for `A0` and 1 for `A1`. Click **Apply**. A new window should pop up with the stats of the fitting. In particular you should find a **Correlation coefficient** of 0.9778 and a value of 17.7651 for `A0` and 1.67056 for `A1`. These are the value of μ_{fit} and σ_{fit} we are going to use for the SCDM method. In particular, $\mu_{\text{SCDM}} = \mu_{\text{fit}} - 3\sigma_{\text{fit}} = 12.75$ eV and $\sigma_{\text{SCDM}} = \sigma_{\text{fit}} = 1.67$ eV. The motivation for this specific choice of σ_{SCDM} and σ_{SCDM} may be found in Ref. [3], where the authors also show validation of this approach on a dataset of 200 materials. You should now see the fitting function, as well as the projectabilities in the graph.

☞ Compute initial guesses with SCDM method and parameters from projectabilities

- Open the `Pb.pw2wan.in` input file and add the following SCDM keywords

```
scdm_proj = .true.
scdm_entanglement = 'erfc'
scdm_mu = 12.75
scdm_sigma = 1.67
```

- Inspect the `wannier90` input file `Pb.win`. Now we have set `num_wann = 9` (number of PAOs from pseudopotentials) and `num_bands = 13`. Again, you will need to add the keyword for the automated projections, without any projection block, i.e.

```
auto_projection = .true.
```

  This time we want to retain all the bands.

- Run `wannier90` to generate a list of the required overlaps, which will be written into the `Pb.nnkp` file (no need to run this in parallel).

```
$> /media/ictpuser/AiiDA/bin/wannier90.x -pp Pb
```

- Run `pw2wannier90` to compute the overlap between Bloch states $M^{\mathbf{k}}_{mn}$ and the initial projections $A^{\mathbf{k}}_{mn}$ from the SCDM method, that will be written in the `Pb.mmn` and `Pb.amn` files, respectively.

```
$> mpirun -np 2 /media/ictpuser/AiiDA/bin/pw2wannier90.x < Pb.pw2wan.in | tee pw2wan.out
```

- Run `wannier90` to compute the MLWFs from SCDM initial guesses.
  ```
  $> mpirun -np 2 /media/ictpuser/AiiDA/bin/wannier90.x Pb
  ```
- Inspect the `Pb.wout` output file and check that the calculation has converged. Take note of the final total spread and individual WFs spreads.
- Plot the DFT band structure and the band structure from Wannier interpolation. This can be done in, e.g., gnuplot
  ```
  $> gnuplot
  gnuplot> load "Pb_band.gnu"
  gnuplot> replot "bands.dat.gnu" u ($1*1.26916):2 w l lt -1, "Pb_band.dat" lt 1
  ```
- Check how the resulting interpolated band structure changes when performing a disentanglement step

References

For the theoretical background of the SCDM method see:
[1] A. Damle, L. Lin, and L. Ying, Compressed representation of kohn–sham orbitals via selected columns of the density matrix, Journal of Chemical Theory and Computation **11**, 1463–1469 (2015),
as well as
[2] A. Damle and L. Lin, Disentanglement via entanglement: A unified method for wannier localization, Multiscale Modeling & Simulation **16**, 1392–1410 (2018).

For how to combine the SCDM method with `wannier90` and automation of initial projections see:
[3] V. Vitale, G. Pizzi, A. Marrazzo, J. Yates, N. Marzari, A. A. Mostofi, Automated high-throughput wannierisation, npj Computational Materials **6** (1), 1-18 (2020).
as well as the data on the Materials Cloud archive
[4] V. Vitale, G. Pizzi, A. Marrazzo, J. R. Yates, N. Marzari, A. A. Mostofi, Automated high-throughput Wannierisation, Materials Cloud Archive (2019), doi: 10.24435/materialscloud:2019.0044/v2