Wannier19 - a new python code for ab initio computation of topological properties

Martin S. McFly[1] and Stepan S. Tsirkin[2]

1 Institute of Future Technology, Hill Valley, CA, USA
[2] Department of Physics, University of Zürich, Switzerland

Universität Zürich[UZH]

Abstract

Wannier90[1] is a well-known code for constructing maximally-localized wannier functions (MLWFs [2]). The powtw90.x executable of Wannier90 employs MLWFs for evaluation and interpolation of quantities like band energies and velocities as well as Berry connections, Berry curvature and orbital magnetic moments. This enables evaluate quantities, which require k-space integration over a fine grid of points in the Brillouin zone, like anomalous Hall conductivity (AHC) [3], orbital magnetization[4] and optical responses[5].
In this contribution I present a new project "Wannier19" aimed to implement some of the functionality of postw90.x in python language in a more efficient way. While the Fortran90 language (which gave part of its name to Wannier90) has been used for decades as a tool for fast numerical calculations in materials science, it requires to produce more lines of a code, compared to more modern languages like Python. In turn, the object-oriented Python allows to write a more laconic code, which saves time of the coder. This saved time can be invested into implementing new quantities to calculate, and realization of more efficient algorithms. Further goals will be to implement more features, like other optical responses and chiral anomaly, as well as other methods, not implemented in Wannier90, e.g. tetrahedron method for BZ integration and account of symmetries.
The project is on an early development stage, and by now it mainly performs calculations of AHC. Giving the same result as Wannier90, the new Wannier19 is much faster, thus allowing for finer grids in k-space, as well in the Fermi level position. This may be highly useful in particular for evaluation of the Nernst effect in Weyl semimetals. The code is hosted on github [6] and all interested are kindly invited for collaboration.

- A. A. Mostofi, J. R. Yates, G. Pizzi, Y.-S. Lee, I. Souza, D. Vanderbilt and N. Marzari, Comput. Phys. Commun. 185, 2309 (2014)
- Nicola Marzari, Arash A. Mostofi, Jonathan R. Yates, Ivo Souza, and David Vanderbilt, Rev. Mod. Phys. 84, 1419 (2012)
- Xinjie Wang, Jonathan R. Yates, Ivo Souza, and David Vanderbilt, Phys. Rev. B 74, 195118 (2006)
- M. G. Lopez, David Vanderbilt, T. Thonhauser, and Ivo Souza, Phys. Rev. B 85, 014435 (2012)
- Julen Ibañez-Azpiroz, Stepan S. Tsirkin, and Ivo Souza, Phys. Rev. B 97, 245143 (2018)
- Wannier19 code : https://github.com/stepan-tsirkin/wannier19

Motivation

Wannier90 is good, because it

- is a well-established code
- known for ages
- has a broad community of developers
- has many implemented features
- ..... (many more)

Wannier90 is *not so good*, because

- some algorithms are not written in the most optimal way
- does not use FFT. slow Fourier instead
- Fortran90 is already old-fasioned
- .. and not so flexible as python.
- hence it becomes harder to develop further features.

Thus, the goal is to re-write some of postw90.x functionality in python, and make the calculations faster. Currently working on AHC as an example, and performance is already mych better, then postw90.x.

Hosted on github

https://github.com/stepan-tsirkin/wannier19

Anomalous Hall conductivity

THe AHC tensor is given by:

$$\sigma^{\mathrm{AH}}_{\alpha\beta}(0) = -\frac{e^2}{\hbar}\int \frac{d\mathbf{k}}{(2\pi)^3}\sum_n f_{n\mathbf{k}}\Omega_{n,\alpha\beta}(\mathbf{k}) \qquad (1)$$

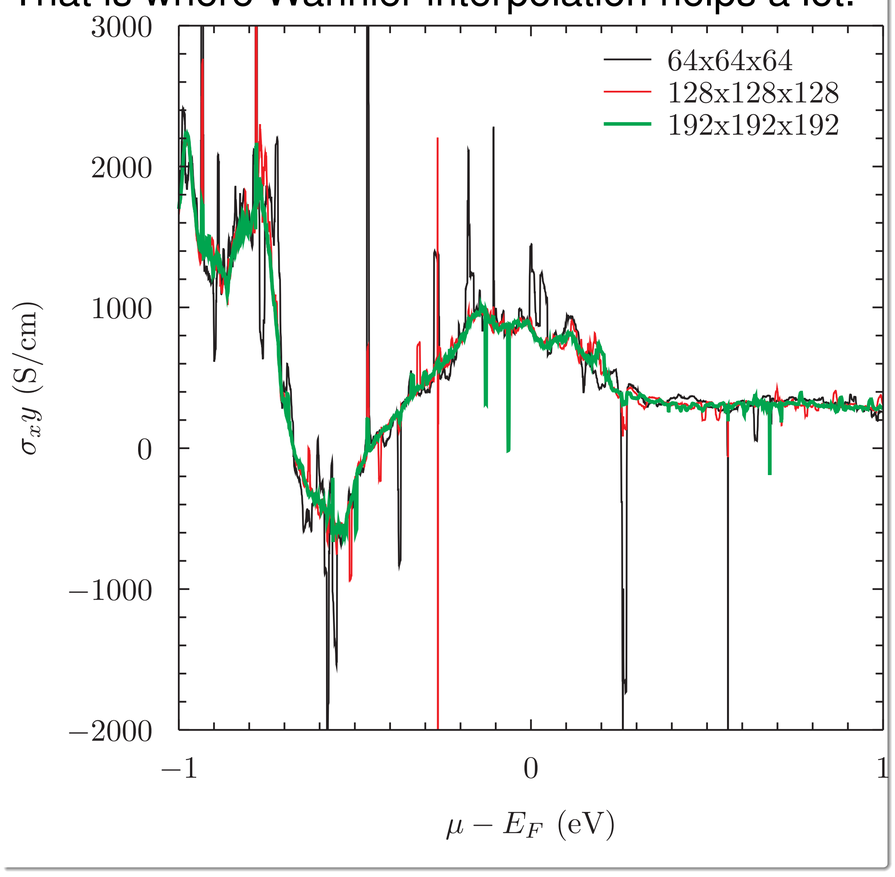
in terms of Berry curvature:

$$\Omega_{n,\alpha\beta}(\mathbf{k}) = -2\mathrm{Im}\langle\nabla_{k_\alpha}u_{n\mathbf{k}}|\nabla_{k_\beta}u_{n\mathbf{k}}\rangle. \qquad (2)$$

where $|u_{n\mathbf{k}}\rangle = e^{-i\mathbf{k}\cdot\mathbf{r}}|\psi_{n\mathbf{k}}\rangle$ is the cell-periodic Bloch function.
to evaluate it one needs:

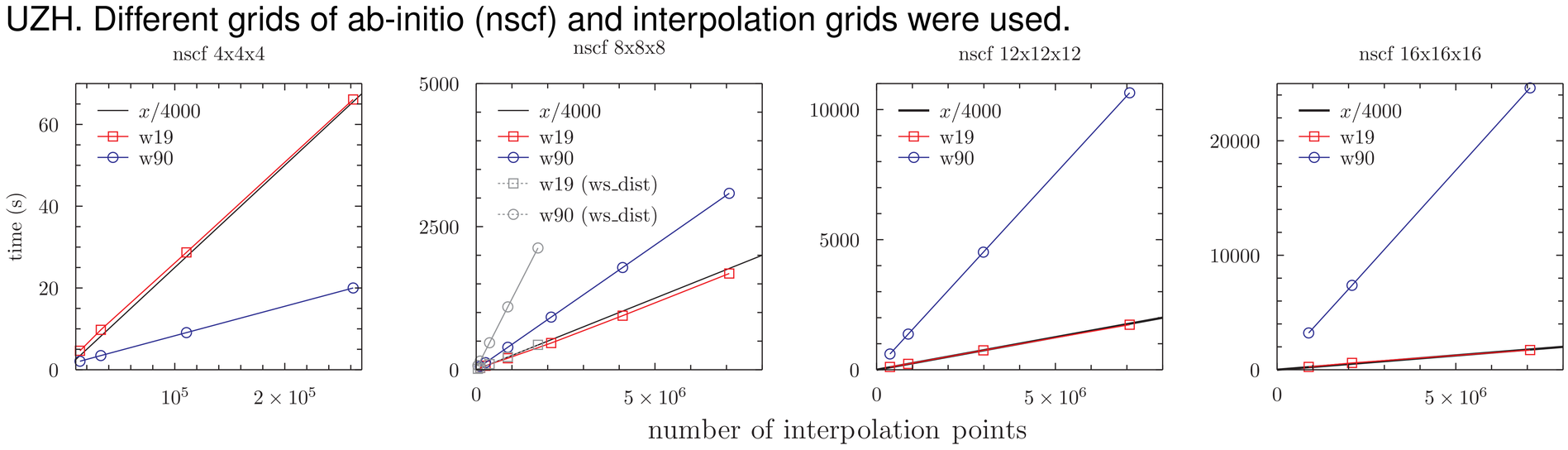
- a smooth gauge
- dense grid of **k**-points

That is where Wannier interpolation helps a lot.



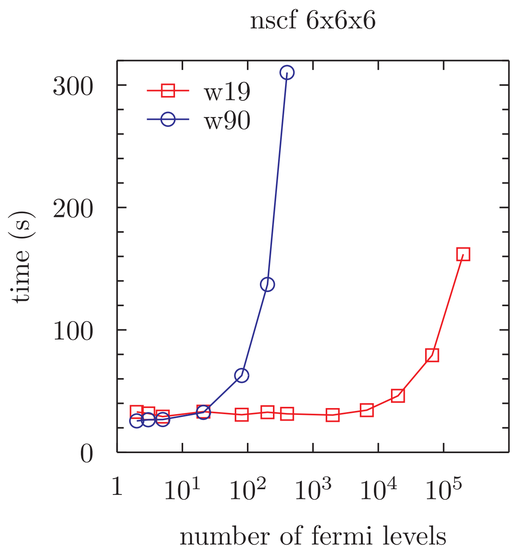
Computational time

Calculations were performed with wannier19 and wannier90 on identical 40-core nodes of a cluster at UZH. Different grids of ab-initio (nscf) and interpolation grids were used.



One can see that while for w19 the computational time does not depend on the ab-initio grid and the use of parameter "use_ws_distance". While for w90 they increase time a lot.

Also scanning multiple fermi levels has almost no cost in w19 up to 10^4 points:



Advantages achieved so far:

- computations of AHC are faster
- use_ws_distance has no cost
- multiple Fermi levels have no cost
- increased ab-inito grid has no cost for the interpolation

This also could be done within Wannier90. Anybody whants to do it?