Journal of Physics: Condensed Matter

PAPER • **OPEN ACCESS**

Wannier90 as a community code: new features and applications

To cite this article: Giovanni Pizzi *et al* 2020 *J. Phys.: Condens. Matter* **32** 165902

View the article online for updates and enhancements.

Recent citations

- Orbital torque: Torque generation by orbital current injection
Dongwook Go and Hyun-Woo Lee

- WIEN2k: An APW+lo program for calculating the properties of solids
Peter Blaha *et al*

- Giant enhancement of solid solubility in monolayer alloys by selective orbital coupling
Shiqiao Du *et al*

IOP ebooks™

Bringing you innovative digital publishing with leading voices to create your essential collection of books in STEM research.

Start exploring the collection - download the first chapter of every title for free.

This content was downloaded from IP address 128.178.106.62 on 24/02/2020 at 14:57

OPEN ACCESS
IOP Publishing
Journal of Physics: Condensed Matter
J. Phys.: Condens. Matter **32** (2020) 165902 (25pp)
https://doi.org/10.1088/1361-648X/ab51ff

Wannier90 as a community code: new features and applications

Giovanni Pizzi[1,29,30], Valerio Vitale[2,3,29], Ryotaro Arita[4,5], Stefan Blügel[6], Frank Freimuth[6], Guillaume Géranton[6], Marco Gibertini[1,7], Dominik Gresch[8], Charles Johnson[9], Takashi Koretsune[10,11], Julen Ibañez-Azpiroz[12], Hyungjun Lee[13,14], Jae-Mo Lihm[15], Daniel Marchand[16], Antimo Marrazzo[1], Yuriy Mokrousov[6,17], Jamal I Mustafa[18], Yoshiro Nohara[19], Yusuke Nomura[4], Lorenzo Paulatto[20], Samuel Poncé[21], Thomas Ponweiser[22], Junfeng Qiao[23], Florian Thöle[24], Stepan S Tsirkin[12,25], Małgorzata Wierzbowska[26], Nicola Marzari[1,29], David Vanderbilt[27,29], Ivo Souza[12,28,29], Arash A Mostofi[3,29] and Jonathan R Yates[21,29]

[1] Theory and Simulation of Materials (THEOS) and National Centre for Computational Design and Discovery of Novel Materials (MARVEL), École Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland
[2] Cavendish Laboratory, Department of Physics, University of Cambridge, Cambridge CB3 0HE, United Kingdom
[3] Departments of Materials and Physics, and the Thomas Young Centre for Theory and Simulation of Materials, Imperial College London, London SW7 2AZ, United Kingdom
[4] RIKEN Center for Emergent Matter Science, 2-1 Hirosawa, Wako, Saitama 351-0198, Japan
[5] Department of Applied Physics, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
[6] Peter Grünberg Institut and Institute for Advanced Simulation, Forschungszentrum Jülich and JARA, 52425 Jülich, Germany
[7] Department of Quantum Matter Physics, University of Geneva, Geneva, Switzerland
[8] ETH Zurich, Zurich, Switzerland
[9] Departments of Materials and Physics, Imperial College London, London SW7 2AZ, United Kingdom
[10] Department of Physics, Tohoku University, Sendai, Japan
[11] JST PRESTO, Kawaguchi, Saitama, Japan
[12] Centro de Física de Materiales, Universidad del País Vasco, E-20018 San Sebastián, Spain
[13] Institute of Physics, École Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland
[14] Korea Institute for Advanced Study, Hoegiro 85, Seoul 02455, Republic of Korea
[15] Department of Physics and Center for Theoretical Physics, Seoul National University, Seoul 08826, Republic of Korea
[16] Laboratory for Multiscale Mechanics Modeling (LAMMM), École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland
[17] Institute of Physics, Jonannes-Gutenberg University of Mainz, 55099 Mainz, Germany
[18] Department of Physics, University of California at Berkeley, Berkeley, CA 94720, United States of America
[19] ASMS Co., Ltd., 1-7-11 Higashi-Gotanda, Shinagawa-ku, Tokyo 141-0022, Japan
[20] Institut de Minéralogie, de Physique des Matériaux et de Cosmochimie (IMPMC), Sorbonne Université, CNRS UMR 7590, Case 115, 4 place Jussieu, 75252 Paris Cedex 05, France
[21] Department of Materials, University of Oxford, Parks Road, Oxford OX1 3PH, United Kingdom

[29] Members of the Wannier Developers Group, who are responsible for the long-term maintenance and sustainability of Wannier90.
[30] Author to whom any correspondence should be addressed.

Original content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

[22] Research Institute for Symbolic Computation (RISC), Johannes Kepler University, Altenberger Straße 69, 4040 Linz, Austria
[23] Fert Beijing Institute, School of Microelectronics, BDBC, Beihang University, Beijing, People's Republic of China
[24] Materials Theory, ETH Zürich, Wolfgang-Pauli-Strasse 27, CH-8093 Zürich, Switzerland
[25] Department of Physics, University of Zurich, Winterthurerstrasse 190, CH-8057 Zurich, Switzerland
[26] Institute of High Pressure Physics, Polish Academy of Sciences, Sokołowska street 29/37, 01-142 Warsaw, Poland
[27] Department of Physics and Astronomy, Rutgers University, Piscataway, NJ 08854-8019, United States of America
[28] Ikerbasque Foundation, E-48013 Bilbao, Spain

E-mail: giovanni.pizzi@epfl.ch

Received 23 July 2019, revised 10 October 2019
Accepted for publication 28 October 2019
Published 23 January 2020

Abstract

Wannier90 is an open-source computer program for calculating maximally-localised Wannier functions (MLWFs) from a set of Bloch states. It is interfaced to many widely used electronic-structure codes thanks to its independence from the basis sets representing these Bloch states. In the past few years the development of Wannier90 has transitioned to a community-driven model; this has resulted in a number of new developments that have been recently released in Wannier90 v3.0. In this article we describe these new functionalities, that include the implementation of new features for wannierisation and disentanglement (symmetry-adapted Wannier functions, selectively-localised Wannier functions, selected columns of the density matrix) and the ability to calculate new properties (shift currents and Berry-curvature dipole, and a new interface to many-body perturbation theory); performance improvements, including parallelisation of the core code; enhancements in functionality (support for spinor-valued Wannier functions, more accurate methods to interpolate quantities in the Brillouin zone); improved usability (improved plotting routines, integration with high-throughput automation frameworks), as well as the implementation of modern software engineering practices (unit testing, continuous integration, and automatic source-code documentation). These new features, capabilities, and code development model aim to further sustain and expand the community uptake and range of applicability, that nowadays spans complex and accurate dielectric, electronic, magnetic, optical, topological and transport properties of materials.

Keywords: Wannier functions, band structure interpolation, local orbitals, real-space methods, electronic structure, Wannier orbitals, density-functional theory

(Some figures may appear in colour only in the online journal)

1. Introduction

Wannier90 is an open-source code for generating Wannier functions (WFs), in particular maximally-localised Wannier functions (MLWFs), and using them to compute advanced materials properties with high efficiency and accuracy. Wannier90 is a paradigmatic example of interoperable software, achieved by ensuring that all the quantities required as input are entirely independent of the underlying electronic-structure code from which they are obtained. Most of the major and widely used electronic-structure codes have an interface to Wannier90, including Quantum ESPRESSO [1], ABINIT [2], VASP [3–5], Siesta [6], Wien2k [7], Fleur [8], Octopus [9] and ELK [10]. As a consequence, once a property is implemented within Wannier90, it can be immediately available to users of all codes that interface to it.

Over the last few years, Wannier90 has undergone a transition from a code developed by a small group of developers to a community code with a much wider developer base. This has been achieved in two principal ways: (i) hosting the source code and associated development efforts on a public GitHub repository [11]; and (ii) building a community of Wannier90 developers and facilitating personal interactions between individuals through community workshops, the most recent in 2016. In response, the code has grown significantly, gaining many novel features contributed by this community, as well as numerous fixes.

In this paper, we describe the most important novel contributions to the Wannier90 code, as embodied in its 3.0 release. The paper is structured as follows: in section 2 we first summarise the background theory for the computation of MLWFs (additional details can be found in [12]), and introduce the

notation that will be used throughout the paper. In section 3 we describe the novel features of WANNIER90 that are related to the core wannierisation and disentanglement algorithms; these include symmetry-adapted WFs, selective localisation of WFs, and parallelisation using the message-passing interface (MPI). In section 4 we describe new functionality enhancements, including the ability to handle spinor-valued WFs and calculations with non-collinear spin that use ultrasoft pseudopotentials (within QUANTUM ESPRESSO); improved interpolation of the *k*-space Hamiltonian; a more flexible approach for handling and using initial projections; and the ability to plot WFs in Gaussian cube format on WF-centred grids with non-orthogonal translation vectors. In section 5 we describe new functionalities associated with using MLWFs for computing advanced electronic-structure properties, including the calculation of shift currents, gyrotropic effects and spin Hall conductivities, as well as parallelisation improvements and the interpolation of bands originating from calculations performed with many-body perturbation theory (GW). In section 6 we describe the selected-columns-of-the-density-matrix (SCDM) method, which enables computation of WFs without the need for explicitly defining initial projections. In section 7 we describe new post-processing tools and codes, and the integration of WANNIER90 with high-throughput automation and workflow management tools (specifically, the AiiDA materials' informatics infrastructure [13]). In section 8 we describe the modern software engineering practices now adopted in WANNIER90, that have made it possible to improve the development lifecycle and transform WANNIER90 into a community-driven code. Finally, our conclusions and outlook are presented in section 9.

2. Background

WFs form a possible basis set for the electronic states of materials. As we are going to describe in the following, WFs are not unique and they can be optimised to obtain MLWFs. These are particularly useful in a number of electronic-structure applications. For instance, they enable efficient interpolation of operator matrix elements on dense grids in the Brillouin Zone (BZ), which is a key step to compute many materials properties. The interpolation is obtained starting from the value of these matrix elements and other properties of the wavefunctions (described below) computed on a coarser grid, usually with an accurate but slower *ab initio* code. MLWFs play in materials a role analogous to molecular orbitals in molecules and some typical MLWFs, e.g., in the case of those corresponding to the valence bands of GaAs, are discussed in section 3.3.

Formally, MLWFs can be introduced as follows in the independent-particle approximation. The electronic structure of a periodic system is conventionally represented in terms of one-electron Bloch states $\psi_{n\mathbf{k}}(\mathbf{r})$, which are labelled by a band index n and a crystal momentum $\mathbf{k}$ inside the first BZ, and which satisfy Bloch's theorem:

$$\psi_{n\mathbf{k}}(\mathbf{r}) = u_{n\mathbf{k}}(\mathbf{r})\mathrm{e}^{\mathrm{i}\mathbf{k}\cdot\mathbf{r}}, \tag{1}$$

where $u_{n\mathbf{k}}(\mathbf{r}) = u_{n\mathbf{k}}(\mathbf{r}+\mathbf{R})$ is a periodic function with the same periodicity of the single-particle Hamiltonian, and $\mathbf{R}$ is a Bravais lattice vector. For the moment we ignore the spin degrees of freedom and work with spinless wave functions; spinor wave functions will be treated in section 4.1. Such a formalism is also commonly applied, via the supercell approximation, to non-periodic systems, typically used to treat point, line and planar defects in crystals, surfaces, amorphous solids, liquids and molecules.

2.1. Isolated bands

A group of bands is said to be *isolated* if it is separated by energy gaps from all the other lower and higher bands throughout the BZ (this isolated group of bands may still show arbitrary crossing degeneracies and hybridisations within itself). For a set of J such bands, the electronic states can be equivalently represented by a set of J WFs per cell, that are related to the Bloch states via two unitary transformations (one continuous, one discrete) [14]:

$$|w_{n\mathbf{R}}\rangle = V\int_{\mathrm{BZ}}\frac{\mathrm{d}\mathbf{k}}{(2\pi)^3}\mathrm{e}^{-\mathrm{i}\mathbf{k}\cdot\mathbf{R}}\sum_{m=1}^{J}|\psi_{m\mathbf{k}}\rangle U_{mn\mathbf{k}}, \tag{2}$$

where $w_{n\mathbf{R}}(\mathbf{r}) = w_{n\mathbf{0}}(\mathbf{r}-\mathbf{R})$ is a periodic (but not necessarily localised) WF labelled by the quantum number $\mathbf{R}$ (the conjugate variable of the quasi-momentum $\mathbf{k}$ in the Bloch representation), V is the cell volume and $U_{\mathbf{k}}$ are unitary matrices that mix Bloch states at a given $\mathbf{k}$ and represent the gauge freedom that exists in the definition of the Bloch states and that is inherited by the WFs.

MLWFs are obtained by choosing $U_{\mathbf{k}}$ matrices that minimise the sum of the quadratic spreads of the WFs about their centres for a reference $\mathbf{R}$ (say, $\mathbf{R}=\mathbf{0}$). This sum is given by the spread functional

$$\Omega = \sum_{n=1}^{J}\left[\langle w_{n\mathbf{0}}|\mathbf{r}\cdot\mathbf{r}|w_{n\mathbf{0}}\rangle - |\langle w_{n\mathbf{0}}|\mathbf{r}|w_{n\mathbf{0}}\rangle|^2\right]. \tag{3}$$

Ω may be decomposed into two positive-definite parts [15],

$$\Omega = \Omega_{\mathrm{I}} + \widetilde{\Omega}, \tag{4}$$

where

$$\Omega_{\mathrm{I}} = \sum_{n}\left[\langle w_{n\mathbf{0}}|\mathbf{r}\cdot\mathbf{r}|w_{n\mathbf{0}}\rangle - \sum_{m\mathbf{R}}|\langle w_{m\mathbf{R}}|\mathbf{r}|w_{n\mathbf{0}}\rangle|^2\right] \tag{5}$$

is gauge invariant (i.e. invariant under the action of any unitary $U_{\mathbf{k}}$ on the Bloch states), and

$$\widetilde{\Omega} = \sum_{n}\sum_{m\mathbf{R}\neq n\mathbf{0}}|\langle w_{m\mathbf{R}}|\mathbf{r}|w_{n\mathbf{0}}\rangle|^2 \tag{6}$$

is gauge dependent. Therefore, the "wannierisation" of an isolated manifold of bands, i.e. the transformation of Bloch states into MLWFs, amounts to minimising the gauge-dependent part $\widetilde{\Omega}$ of the spread functional.

Crucially, the matrix elements of the position operator between WFs can be expressed in reciprocal space. Under the assumption that the BZ is sampled on a uniform

Monkhorst–Pack mesh of k-points composed of N points ($V\int_{\mathrm{BZ}}\frac{\mathbf{dk}}{(2\pi)^3}\to\frac{1}{N}\sum_{\mathbf{k}}$), the gauge-independent and gauge-dependent parts of the spread may be expressed, respectively, as [15]

$$\Omega_{\mathrm{I}}=\frac{1}{N}\sum_{\mathbf{k},\mathbf{b}}w_b\left[J-\sum_{mn}\left|M_{mn}^{(\mathbf{k},\mathbf{b})}\right|^2\right] \tag{7}$$

and

$$\begin{aligned}\widetilde{\Omega}=&\frac{1}{N}\sum_{\mathbf{k},\mathbf{b}}w_b\sum_{m\neq n}\left|M_{mn}^{(\mathbf{k},\mathbf{b})}\right|^2\\&+\frac{1}{N}\sum_{\mathbf{k},\mathbf{b}}w_b\sum_{n}(-\operatorname{Im}\ln M_{nn}^{(\mathbf{k},\mathbf{b})}-\mathbf{b}\cdot\bar{\mathbf{r}}_n)^2,\end{aligned} \tag{8}$$

where $\mathbf{b}$ are the vectors connecting a k-point to its neighbours, w_b are weights associated with the finite-difference representation of $\nabla_{\mathbf{k}}$ for a given geometry, the matrix of overlaps $M^{(\mathbf{k},\mathbf{b})}$ is defined by

$$M_{mn}^{(\mathbf{k},\mathbf{b})}=\langle u_{m\mathbf{k}}|u_{n,\mathbf{k}+\mathbf{b}}\rangle, \tag{9}$$

and the centres of the WFs are given by

$$\bar{\mathbf{r}}_n\equiv\langle w_{n\mathbf{0}}|\mathbf{r}|w_{n\mathbf{0}}\rangle=-\frac{1}{N}\sum_{\mathbf{k},\mathbf{b}}w_b\mathbf{b}\operatorname{Im}\ln M_{nn}^{(\mathbf{k},\mathbf{b})}. \tag{10}$$

Minimisation of the spread functional is achieved by considering infinitesimal gauge transformations $U_{mn\mathbf{k}}=\delta_{mn}+\mathrm{d}W_{mn\mathbf{k}}$, where $\mathrm{d}W$ is anti-Hermitian ($\mathrm{d}W^\dagger=-\mathrm{d}W$). The gradient of the spread functional with respect to such variations is given by

$$\mathcal{G}_{\mathbf{k}}\equiv\frac{\mathrm{d}\Omega}{\mathrm{d}W_{mn\mathbf{k}}}=4\sum_{\mathbf{b}}w_b\left(\mathcal{A}[R_{mn}^{(\mathbf{k},\mathbf{b})}]-\mathcal{S}[T_{mn}^{(\mathbf{k},\mathbf{b})}]\right), \tag{11}$$

where $\mathcal{A}$ and $\mathcal{S}$ are the super-operators $\mathcal{A}[B]=(B-B^\dagger)/2$ and $\mathcal{S}[B]=(B+B^\dagger)/2\mathrm{i}$, respectively, and

$$R_{mn}^{(\mathbf{k},\mathbf{b})}=M_{mn}^{(\mathbf{k},\mathbf{b})}M_{nn}^{(\mathbf{k},\mathbf{b})*}, \tag{12}$$

$$T_{mn}^{(\mathbf{k},\mathbf{b})}=\frac{M_{mn}^{(\mathbf{k},\mathbf{b})}}{M_{nn}^{(\mathbf{k},\mathbf{b})}}q_n^{(\mathbf{k},\mathbf{b})}, \tag{13}$$

$$q_n^{(\mathbf{k},\mathbf{b})}=\operatorname{Im}\ln M_{nn}^{(\mathbf{k},\mathbf{b})}+\mathbf{b}\cdot\bar{\mathbf{r}}_n. \tag{14}$$

For the full derivation of equation (11) we refer to [15]. This gradient is then used to generate a search direction $\mathcal{D}_{\mathbf{k}}$ for an iterative steepest-descent or conjugate-gradient minimisation of the spread [16]: at each iteration the unitary matrices are updated according to

$$U_{\mathbf{k}}\to U_{\mathbf{k}}\exp[\alpha\mathcal{D}_{\mathbf{k}}], \tag{15}$$

where α is a coefficient that can either be set to a fixed value or determined at each iteration via a simple polynomial line-search, and the matrix exponential is computed in the diagonal representation of $\mathcal{D}_{\mathbf{k}}$ and then transformed back in the original representation. Once the unitary matrices have been updated, the updated set of $M^{(\mathbf{k},\mathbf{b})}$ matrices is calculated according to

$$M^{(\mathbf{k},\mathbf{b})}=U_{\mathbf{k}}^\dagger M^{(0)(\mathbf{k},\mathbf{b})}U_{\mathbf{k}+\mathbf{b}}, \tag{16}$$

where

$$M_{mn}^{(0)(\mathbf{k},\mathbf{b})}=\langle u_{m\mathbf{k}}^{(0)}|u_{n,\mathbf{k}+\mathbf{b}}^{(0)}\rangle \tag{17}$$

is the set of initial $M^{(\mathbf{k},\mathbf{b})}$ matrices, computed once and for all, at the start of the calculation, from the original set of reference Bloch orbitals $|u_{n\mathbf{k}}^{(0)}\rangle$.

2.2. Entangled bands

It is often the case that the bands of interest are not separated from other bands in the Brillouin zone by energy gaps and overlap and hybridise with other bands that extend beyond the energy range of interest. In such cases, we refer to the bands as being *entangled*.

The difficulty in constructing MLWFs for entangled bands arises from the fact that, within a given energy window, the number of bands $\mathcal{J}_{\mathbf{k}}$ at each k-point $\mathbf{k}$ in the BZ is not a constant and is, in general, different from the target number J of WFs: $\mathcal{J}_{\mathbf{k}}\geqslant J$. Even making the energy window k-dependent would see discontinuous inclusion and exclusion of bands as the BZ is traversed. The treatment of entangled bands requires thus a more complex approach that is typically a two-step process. In the first step, a J-dimensional manifold of Bloch states is selected at each k-point, chosen to be as smooth as possible as a function of $\mathbf{k}$. In the second step, the gauge freedom associated with the selected manifold is used to obtain MLWFs, just as described in section 2.1 for the case of an isolated set of bands.

Focusing on the first step, an orthonormal basis for the J-dimensional subspace $\mathcal{S}_{\mathbf{k}}$ at each $\mathbf{k}$ can be obtained by performing a semi-unitary transformation on the $\mathcal{J}_{\mathbf{k}}$ states at $\mathbf{k}$,

$$|\widetilde{\psi}_{n\mathbf{k}}\rangle=\sum_{m=1}^{\mathcal{J}_{\mathbf{k}}}|\psi_{m\mathbf{k}}\rangle V_{mn\mathbf{k}}, \tag{18}$$

where $V_{\mathbf{k}}$ is a rectangular matrix of dimension $\mathcal{J}_{\mathbf{k}}\times J$ that is semi-unitary in the sense that $V_{\mathbf{k}}^\dagger V_{\mathbf{k}}=\mathbf{1}$.

To select the smoothest possible manifold, a measure of the intrinsic smoothness of the chosen subspace is needed. It turns out that such a measure is given precisely by the gauge-invariant part Ω_{I} of the spread functional for isolated bands [17]. Indeed, equation (7) can be expressed as

$$\Omega_{\mathrm{I}}=\frac{1}{N}\sum_{\mathbf{k},\mathbf{b}}w_b\mathrm{Tr}[P_{\mathbf{k}}Q_{\mathbf{k}+\mathbf{b}}], \tag{19}$$

where $P_{\mathbf{k}}=\sum_{n=1}^{J}|\widetilde{u}_{n\mathbf{k}}\rangle\langle\widetilde{u}_{n\mathbf{k}}|$ is the projection operator onto $\mathcal{S}_{\mathbf{k}}$, $Q_{\mathbf{k}}=\mathbf{1}-P_{\mathbf{k}}$ is its Hilbert-space complement, and 'Tr' represents the trace over the entire Hilbert space. $\mathrm{Tr}[P_{\mathbf{k}}Q_{\mathbf{k}+\mathbf{b}}]$ measures the mismatch between the subspaces $\mathcal{S}_{\mathbf{k}}$ and $\mathcal{S}_{\mathbf{k}+\mathbf{b}}$, vanishing if they overlap identically. Hence Ω_{I} measures the average mismatch of the local subspace $\mathcal{S}_{\mathbf{k}}$ across the BZ, so that an optimally-smooth subspace can be selected by minimising Ω_{I}. Doing this with orthonormality constraints on the Bloch-like states is equivalent to solving self-consistently the set of coupled eigenvalue equations [17]

$$\left[\sum_{\mathbf{b}}w_bP_{\mathbf{k}+\mathbf{b}}\right]|\widetilde{u}_{n\mathbf{k}}\rangle=\lambda_{n\mathbf{k}}|\widetilde{u}_{n\mathbf{k}}\rangle. \tag{20}$$

The solution can be achieved via an iterative procedure, whereby at the *i*th iteration the algorithm traverses the entire set of k-points, selecting at each one the J-dimensional subspace $\mathcal{S}_{\mathbf{k}}^{(i)}$ that has the smallest mismatch with the subspaces $\mathcal{S}_{\mathbf{k}+\mathbf{b}}^{(i-1)}$ at the neighbouring k-points obtained in the previous iteration. This amounts to solving

$$\left[\sum_{\mathbf{b}} w_b P_{\mathbf{k}+\mathbf{b}}^{(i-1)}\right] |\widetilde{u}_{n\mathbf{k}}^{(i)}\rangle = \lambda_{n\mathbf{k}}^{(i)} |\widetilde{u}_{n\mathbf{k}}^{(i)}\rangle, \tag{21}$$

and selecting the J eigenvectors with the largest eigenvalues [17]. Self-consistency is reached when $\mathcal{S}_{\mathbf{k}}^{(i)} = \mathcal{S}_{\mathbf{k}}^{(i-1)}$ (to within a user-defined threshold) at all the k-points. To make the algorithm more robust, the projector appearing on the left-hand-side of equation (21) is replaced with $[P_{\mathbf{k}+\mathbf{b}}^{(i)}]_{\text{in}}$, given by

$$[P_{\mathbf{k}+\mathbf{b}}^{(i)}]_{\text{in}} = \beta P_{\mathbf{k}+\mathbf{b}}^{(i-1)} + (1-\beta)[P_{\mathbf{k}+\mathbf{b}}^{(i-1)}]_{\text{in}}, \tag{22}$$

which is a linear mixture of the projector that was used as input for the previous iteration and the projector defined by the output of the previous iteration. The parameter $0 < \beta \leqslant 1$ determines the degree of mixing, and is typically set to $\beta = 0.5$; setting $\beta = 1$ reverts precisely to equation (21), while smaller and smaller values of β make convergence smoother (and thus more robust) but also slower.

In practice, equation (21) is solved by diagonalising the Hermitian operator appearing on the left-hand-side in the basis of the original $\mathcal{J}_{\mathbf{k}}$ Bloch states:

$$Z_{mn\mathbf{k}}^{(i)} = \langle u_{m\mathbf{k}}^{(0)}| \sum_{\mathbf{b}} w_b [P_{\mathbf{k}+\mathbf{b}}^{(i)}]_{\text{in}} |u_{n\mathbf{k}}^{(0)}\rangle. \tag{23}$$

Once the optimal subspace has been selected, the wannierisation procedure described in section 2.1 is carried out to minimise the gauge-dependent part $\widetilde{\Omega}$ of the spread functional within that optimal subspace.

2.3. Initial projections

In principle, the overlap matrix elements $M_{mn}^{(\mathbf{k},\mathbf{b})}$ are the only quantities required to compute and minimise the spread functional, and generate MLWFs for either isolated or entangled bands. In practice, this is generally true when dealing with an isolated set of bands, but in the case of entangled bands a good initial guess for the subspaces $\mathcal{S}_{\mathbf{k}}$ alleviates problems associated with falling into local minima of Ω_{I}, and/or obtaining MLWFs that cannot be chosen to be real-valued (when no spin-orbit coupling is included). Even in the case of an isolated set of bands, a good initial guess for the WFs, whilst not usually critical, often results in faster convergence of the spread to the global minimum. (It is important to note that both for isolated and for entangled bands multiple solutions to the wannierisation or disentanglement can exist, as discussed later.)

A simple and effective procedure for selecting an initial gauge (in the case of isolated bands) or an initial subspace and initial gauge (in the case of entangled bands) is to project a set of J trial orbitals $g_n(\mathbf{r})$ localised in real space onto the space spanned by the set of original Bloch states at each $\mathbf{k}$:

$$|\phi_{n\mathbf{k}}\rangle = \sum_{m=1}^{J \text{ or } \mathcal{J}_{\mathbf{k}}} |\psi_{m\mathbf{k}}\rangle\langle\psi_{m\mathbf{k}}|g_n\rangle, \tag{24}$$

where the sum runs up to either J or $\mathcal{J}_{\mathbf{k}}$, depending on whether the bands are isolated or entangled, respectively, and the inner product $A_{mn\mathbf{k}} = \langle\psi_{m\mathbf{k}}|g_n\rangle$ is over the Born–von Karman supercell. (In practice, the fact that the g_n are localised greatly simplifies this calculation.) The matrices $A_{\mathbf{k}}$ are square $(J \times J)$ or rectangular $(\mathcal{J}_{\mathbf{k}} \times J)$ in the case of isolated or entangled bands, respectively. The resulting orbitals are then orthonormalised via a Löwdin transformation [18]:

$$|\widetilde{\psi}_{n\mathbf{k}}\rangle = \sum_{m=1}^{J} |\phi_{m\mathbf{k}}\rangle S_{mn\mathbf{k}}^{-\frac{1}{2}} \tag{25}$$

$$= \sum_{m=1}^{J \text{ or } \mathcal{J}_{\mathbf{k}}} |\psi_{m\mathbf{k}}\rangle (A_{\mathbf{k}} S_{\mathbf{k}}^{-\frac{1}{2}})_{mn}, \tag{26}$$

where $S_{mn\mathbf{k}} = \langle\phi_{m\mathbf{k}}|\phi_{n\mathbf{k}}\rangle = (A_{\mathbf{k}}^{\dagger} A_{\mathbf{k}})_{mn}$, and $A_{\mathbf{k}} S_{\mathbf{k}}^{-\frac{1}{2}}$ is a unitary matrix in the case of isolated bands and semi-unitary in the case of entangled bands. In the case of entangled bands, once an optimally-smooth subspace has been obtained as described in section 2.2, the same trial orbitals $g_n(\mathbf{r})$ can be used to initialise the wannierisation procedure of section 2.1. In practice, the matrices $A_{\mathbf{k}}$ are computed once and for all at the start of the calculation, together with the overlap matrices $M^{(\mathbf{k},\mathbf{b})}$. These two operations need to be performed within the context of the electronic-structure code and basis set adopted; afterwards, all the operations of WANNIER90 rely only on $A_{\mathbf{k}}$ and $M^{(\mathbf{k},\mathbf{b})}$ and not on the specific representation of $\psi_{m\mathbf{k}}$ (e.g. plane waves, linearised augmented plane waves, localised basis sets, real-space grids, ...).

3. New features for wannierisation and disentanglement

In this section we provide an overview of the new features associated with the core wannierisation and disentanglement algorithms in WANNIER90, namely the ability to generate WFs of specific symmetry; selectively localise a subset of the WFs and/or constrain their centres to specific sites; and perform wannierisation and disentanglement more efficiently through parallelisation.

3.1. Symmetry-adapted Wannier functions

In periodic systems, atoms are usually found at sites $\mathbf{q}$ whose site-symmetry group G_q is a subgroup of the full point group F of the crystal [19] (the symmetry operations in the group G_q are those that leave $\mathbf{q}$ fixed). The set of points $\{\mathbf{q}_a\}$ that are symmetry-equivalent sites to $\mathbf{q}$ is called an *orbit* [20]. These are all the points in the unit cell that can be generated from $\mathbf{q}$ by applying the symmetry operations in the full space group G that do not leave $\mathbf{q}$ fixed. If $\mathbf{q}_a$ is a high-symmetry site then its Wyckoff position has a single orbit [20]; for low-symmetry sites different orbits correspond to the same

Wyckoff position. The number of points in the orbit(s) is the multiplicity n_{q_a} of the Wyckoff position. MLWFs, however, are not bound to reside on such high-symmetry sites, and they do not necessarily possess the site symmetries of the crystal [17, 21, 22]. When using MLWFs as a local orbital basis set in methods such as first-principles tight binding, DFT + U and DFT plus dynamical-mean-field theory (DMFT), which deal with beyond-DFT correlations in a local subspace such as that spanned by d orbitals (e.g. for systems containing transition metals atoms) or f orbitals (e.g. for systems containing rare-earth or actinide series atoms), it is often desirable to ensure that the WFs basis possesses the local site symmetries.

Sakuma [21] has shown that such *symmetry-adapted Wannier functions* (SAWFs) can be constructed by introducing additional constraints on the unitary matrices $U_{\mathbf{k}}$ of equation (2) during the minimisation of the spread. SAWFs, therefore, can be fully integrated within the original maximal-localisation procedure. The SAWF approach gives the user a certain degree of control over the symmetry and centres of the Wannier functions at the expense of some localisation since the final total spread of the resulting SAWFs can only be equal to, or most often larger than, that of the corresponding MLWFs with no constraints (note that in principle some SAWFs can have a smaller individual spread than any MLWFs).

For a given point $\mathbf{q}_a$ in the home unit cell $\mathbf{R} = \mathbf{0}$, the SAWFs centred at that point are denoted by

$$\{w_{ia}^{(\varrho)}(\mathbf{r}) \equiv w_i^{(\varrho)}(\mathbf{r} - \mathbf{q}_a), \quad i = 1, \ldots, n_\varrho\}, \tag{27}$$

where ϱ is the character of the irreducible representation (*irrep*) of the corresponding site-symmetry group G_a with dimension n_ϱ. For instance, in a simple fcc crystal such as copper (Cu), the site-symmetry group associated with the Cu site is O_h; one of its *irreps* [20] is e.g. 3-dimensional T_{2g} and, assuming the Cu atom is located at the origin $\mathbf{r} = \mathbf{0}$ of the unit cell, three associated SAWFs are denoted $w_{10}^{T_{2g}}(\mathbf{r})$, $w_{20}^{T_{2g}}(\mathbf{r})$ and $w_{30}^{T_{2g}}(\mathbf{r})$.

To find these SAWFs, one needs to specify appropriate unitary transformations $U_{mia\mathbf{k}}^{(\varrho)}$ of the Bloch states, defined by

$$\begin{aligned} w_{ia}^{(\varrho)}(\mathbf{r} - \mathbf{R}) &= \frac{1}{N} \sum_{\mathbf{k}} \mathrm{e}^{-\mathrm{i}\mathbf{k}\cdot\mathbf{R}} \sum_{m=1}^{J} \psi_{m\mathbf{k}}(\mathbf{r}) U_{mia\mathbf{k}}^{(\varrho)} \\ &= \frac{1}{N} \sum_{\mathbf{k}} \mathrm{e}^{-\mathrm{i}\mathbf{k}\cdot\mathbf{R}} \psi_{ia\mathbf{k}}^{(\varrho)}(\mathbf{r}), \end{aligned} \tag{28}$$

where $\{\psi_{ia\mathbf{k}}^{(\varrho)}(\mathbf{r})\}$ are basis functions of the *irrep* ϱ and are formed from linear combinations of the J eigenstates $\{\psi_{n\mathbf{k}}(\mathbf{r})\}$ of the Hamiltonian H. Since H is invariant under the full space group G, the representation of a given symmetry operation $g = (\mathcal{R}|\mathbf{t}) \in G$ (where $\mathcal{R}$ and $\mathbf{t}$ are the rotation and fractional-translation parts of the symmetry operation, respectively) in the basis $\{\psi_{n\mathbf{k}}(\mathbf{r})\}$ must be a $J \times J$ unitary matrix [19] $\widetilde{d}_{\mathbf{k}}(g)$, i.e. $\widetilde{d}_{\mathbf{k}}(g)$ represents how the J Bloch states are transformed by the symmetry operation g:

$$g\,\psi_{n\mathbf{k}}(\mathbf{r}) = \sum_{m=1}^{J} \psi_{m\mathcal{R}\mathbf{k}}(\mathbf{r}) \widetilde{d}_{mn\mathbf{k}}(g), \quad g \in G, \tag{29}$$

where the matrix elements $\widetilde{d}_{\mathbf{k}}(g)$ are given by

$$\widetilde{d}_{mn\mathbf{k}}(g) = \int \mathrm{d}\mathbf{r}\, \psi_{m\mathcal{R}\mathbf{k}}^*(\mathbf{r}) \psi_{n\mathbf{k}}\left(g^{-1}\mathbf{r}\right). \tag{30}$$

On the other hand, the Bloch functions $\{\psi_{ia\mathbf{k}}^{(\varrho)}(\mathbf{r})\}$, defined in equation (28), transform under the action of $g \in G$ as

$$g\psi_{ia\mathbf{k}}^{(\varrho)}(\mathbf{r}) = \sum_{i'a'\varrho'} \psi_{i'a'\mathcal{R}\mathbf{k}}^{(\varrho')}(\mathbf{r}) D_{i'a',ia\mathbf{k}}^{(\varrho',\varrho)}(g), \tag{31}$$

where $D_{\mathbf{k}}(g)$ is the matrix representation of the symmetry operation g in the basis of $\{\psi_{ia\mathbf{k}}^{(\varrho)}(\mathbf{r})\}$; the reader is referred to [19, 21] for details.

From equations (28), (29) and (31), it can be shown [21] that, for a symmetry operation $g_{\mathbf{k}}$ that leaves a given $\mathbf{k}$ unchanged, the following relationship holds:

$$U_{\mathbf{k}} D_{\mathbf{k}}(g_{\mathbf{k}}) = \widetilde{d}_{\mathbf{k}}(g_{\mathbf{k}}) U_{\mathbf{k}}, \quad g_{\mathbf{k}} \in G_{\mathbf{k}} \tag{32}$$

and, to obtain SAWFs, the initial unitary matrix $U_{\mathbf{k}}$ ($\mathbf{k} \in$ IBZ) must satisfy this constraint. This can be achieved iteratively, starting with the initial projection onto localised orbitals as described in section 2.3, and with knowledge of $\widetilde{d}_{\mathbf{k}}(g)$ (equation (29)) and $D_{\mathbf{k}}(g)$ (equation (31)), as discussed in detail in [21]. The matrices $\widetilde{d}_{\mathbf{k}}(g)$, which are independent of the underlying basis-set used to represent the Bloch states and are computed only once at the start of the calculation, can be calculated directly from the Bloch states via equation (30). The matrices $D_{\mathbf{k}}(g)$ are calculated by specifying the centre $\mathbf{q}_a$ and the desired symmetry of the Wannier functions (e.g. s, p, d etc) and, for each symmetry operation g_a in the site-symmetry group G_a, calculating the matrix representation of the rotational part.

For an isolated set of bands, the minimisation of $\widetilde{\Omega}$ with the constraints defined in equation (32) requires the gradient $\mathcal{G}_{\mathbf{k}}^{\mathrm{sym}}$ of the total spread Ω with respect to a symmetry-adapted gauge variation, which is then used to generate a search direction $\mathcal{D}_{\mathbf{k}}^{\mathrm{sym}}$. The symmetry-adapted gradient is given by

$$\mathcal{G}_{\mathbf{k}}^{\mathrm{sym}} = \frac{1}{n_{\mathbf{k}}} \sum_{g=(\mathcal{R}|\mathbf{t})\in G} D_{\mathbf{k}}(g) \mathcal{G}_{\mathcal{R}\mathbf{k}} D_{\mathbf{k}}^{\dagger}(g), \tag{33}$$

where $\mathcal{G}_{\mathbf{k}}$ is the original gradient given in equation (11), and $n_{\mathbf{k}}$ is the number of symmetry operations in G that leave $\mathbf{k}$ fixed. It is worth noting that there is no guarantee that equation (32) can be satisfied for any *irrep*, for example, when one is considering a target energy window with a limited number of Bloch states whose symmetry might not be compatible with the *irrep*.

In the case of entangled bands, a similar two-step approach is taken as in the case of MLWFs (section 2.2): first Ω_{I} is minimised by selecting an optimal subspace of Bloch states that are required to transform according to equation (31), followed by minimisation of $\widetilde{\Omega}$ with respect to gauge variations that respect the site symmetries within this subspace, as described for the case of isolated bands above, but with the difference that the constraint of equation (32) is modified to

$$U_{\mathbf{k}} D_{\mathbf{k}}(g_{\mathbf{k}}) = D_{\mathbf{k}}(g_{\mathbf{k}}) U_{\mathbf{k}}, \quad g_{\mathbf{k}} \in G_{\mathbf{k}}, \tag{34}$$

since the states of the optimal subspace transform according to equation (31), rather than equation (29).

An implementation of the SAWF algorithm for both isolated and entangled bands can be found in PW2WANNIER90, the interface code between QUANTUM ESPRESSO and WANNIER90. A typical calculation consists of the following steps: (a) define the symmetry operations of the site-symmetry group. These are either calculated by `pw2wannier90.x`, if the site-symmetry group is equivalent to the full space group of the crystal, or they can be provided in the `.sym` file (e.g. if the site-symmetry group contains fewer symmetry operations than the full space group). (b) Specify the site location and orbital symmetry of the SAWFs. These are defined in the projection block of the WANNIER90 input file `.win` file. (c) Run a preprocessing WANNIER90 calculation to write this information into an intermediate file (with extension `.nnkp`) which is then read by `pw2wannier90.x`. (d) Run `pw2wannier90.x` to calculate the **D** matrix in equation (31). `pw2wannier90.x` computes also the $\tilde{\mathbf{d}}$ matrix in equation (30) from the Kohn–Sham states of the DFT calculation. (e) These matrices are then written to a `.dmn` file which is read by WANNIER90 at the start of the optimisation.

3.2. Selectively-localised Wannier functions and constrained Wannier centres

Wang *et al* have proposed an alternative method [23] to the symmetry-adapted Wannier functions described in section 3.1. Their method permits the selective localisation of a *subset* of the Wannier functions, which may optionally be constrained to have specified centres. Whilst this method does not enforce or guarantee symmetry constraints, it has been observed in the cases that have been studied [23] that Wannier functions whose centres are constrained to a specific site typically possess the corresponding site symmetries.

For an isolated set of J bands, selective localisation of a subset of $J' \leqslant J$ Wannier functions is accomplished by minimising the total spread Ω with respect to only $J' \times J'$ degrees of freedom in the unitary matrix $U_{\mathbf{k}}$. The spread functional to minimise is then given by

$$\Omega' = \sum_{n=1}^{J' \leqslant J} \left[\langle w_{n\mathbf{0}} | r^2 | w_{n\mathbf{0}} \rangle - |\langle w_{n\mathbf{0}} | \mathbf{r} | w_{n\mathbf{0}} \rangle|^2 \right], \tag{35}$$

which reduces to the original spread functional Ω of equation (3) for $J' = J$. When $J' < J$, it is no longer possible to cast the functional Ω' as a sum of a gauge-independent term Ω_{I} and gauge-dependent one $\widetilde{\Omega}$, as done in equation (4) for Ω. Nevertheless, the minimisation can be carried out with methods very similar to those described in section 2. In fact, for $J' < J$, Ω' can be written as the sum of two gauge-dependent terms, $\Omega' = \Omega_{\mathrm{IOD}} + \Omega_{\mathrm{D}}$, where Ω_{IOD} is formally given by the sum of Ω_{I} and the off-diagonal term $(m \neq n), \quad m, n \leqslant J' < J$ of $\widetilde{\Omega}$, and Ω_{D} by the diagonal term $(m = n)$ of $\widetilde{\Omega}$. If one adopts the usual discrete representation on a uniform Monkhorst–Pack grid of k-points, Ω_{IOD} and Ω_{D} are given by [23]

$$\Omega_{\mathrm{IOD}} = \frac{1}{N} \sum_{\mathbf{k},\mathbf{b}} w_b \left[J' - \sum_{n}^{J' < J} \left| M_{nn}^{(\mathbf{k},\mathbf{b})} \right|^2 \right] \tag{36}$$

and

$$\Omega_{\mathrm{D}} = \frac{1}{N} \sum_{n=1}^{J' < J} \sum_{\mathbf{b},\mathbf{k}} w_b \left(\operatorname{Im} \ln M_{nn}^{(\mathbf{k},\mathbf{b})} + \mathbf{b} \cdot \bar{\mathbf{r}}_n \right)^2. \tag{37}$$

With this new spread functional, we can mimic the procedure used to obtain a set of MLWFs, and derive the gradient $\mathcal{G}'_{\mathbf{k}}$ of Ω' which gives the search direction to be used in the minimisation. The matrix elements of $\mathcal{G}'_{\mathbf{k}}$ read

$$\mathcal{G}'_{mn\mathbf{k}} = \begin{cases} \mathcal{G}_{mn\mathbf{k}} & m \leqslant J', n \leqslant J', \\ -2 \sum_{\mathbf{b}} w_b \left[R_{nm}^{(\mathbf{k},\mathbf{b})*} - \mathrm{i} T_{nm}^{(\mathbf{k},\mathbf{b})*} \right] & m \leqslant J', J' < n \leqslant J, \\ 2 \sum_{\mathbf{b}} w_b \left[R_{mn}^{(\mathbf{k},\mathbf{b})} + \mathrm{i} T_{mn}^{(\mathbf{k},\mathbf{b})} \right] & J' < m \leqslant J, n \leqslant J', \\ 0 & J' < m \leqslant J, J' < n \leqslant J, \end{cases} \tag{38}$$

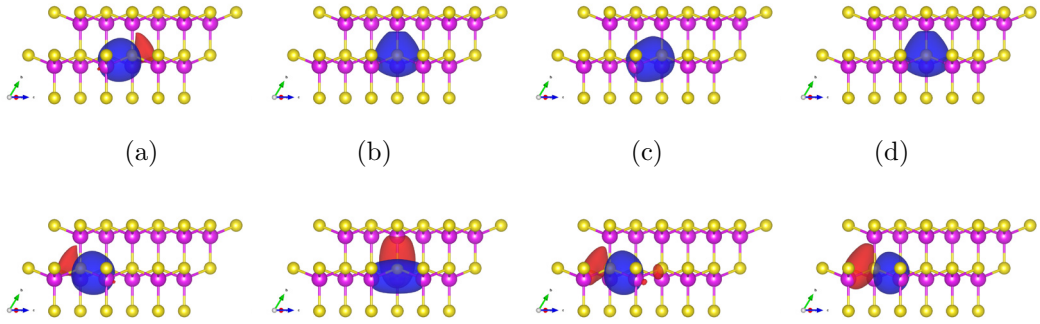
where $\mathcal{G}_{mn\mathbf{k}}$ are the matrix elements of the original gradient in equation (11) (see also [15]), and $R_{mn}^{(\mathbf{k},\mathbf{b})}$ and $T_{mn}^{(\mathbf{k},\mathbf{b})}$ are given by equations (12) and (13), respectively. As a result of the minimisation, we obtain a set of J' maximally-localised Wannier functions, known as *selectively-localised Wannier functions* (SLWFs), whose spreads are in general smaller than the corresponding MLWFs. Naturally, the remaining $J - J'$ functions will be more delocalised than their MLWF counterparts, as they are not optimised, and the overall sum of spreads will be larger (or in the best case scenario equal).

The centres of the SLWFs may be constrained by adding a quadratic penalty function to the spread functional Ω', defining a new functional given by

$$\Omega'_{\lambda} = \sum_{n=1}^{J' < J} [\langle w_{n\mathbf{0}} | r^2 | w_{n\mathbf{0}} \rangle - |\langle w_{n\mathbf{0}} | \mathbf{r} | w_{n\mathbf{0}} \rangle|^2 + \lambda (\bar{\mathbf{r}}_n - \mathbf{x}_n)^2], \tag{39}$$

where λ is a Lagrange multiplier and $\mathbf{x}_n$ is the desired centre for the nth WF. The procedure outlined above for minimising Ω' can be also adapted to deal with Ω'_{λ} (see [23] for details), and minimising Ω'_{λ} results in selectively-localised Wannier functions subject to the constraint of fixed centres (SLWF + C). As noted above, it is observed that WFs derived using the SLWF + C approach naturally possess site symmetries, and their individual spreads are usually smaller than the corresponding spreads of MLWFs, although the total spread, combination of the J' selectively optimised WFs and the $J - J'$ unoptimised functions, is larger than the total spread of the MLWFs (see, for instance last column of the table in figure 1).

In the case of entangled bands, the SLWF(+C) method implicitly assumes that a subspace selection has been performed, i.e. that a smooth J-dimensional manifold exists. Since for the Ω' and Ω'_{λ} functionals it is not possible to define an Ω_I that measures the intrinsic smoothness of the underlying manifold, the additional constraints in equations (35) and (39) can only be imposed during the wannierisation step. This means that SLWF(+C) can be seamlessly coupled with the



(a) (b) (c) (d)

(e) (f) (g) (h)

Method		$\bar{\mathbf{r}}$ [Å]	$\langle r^2\rangle - \bar{r}^2$ [Å^2]		$\bar{\mathbf{r}}$ [Å]	$\langle r^2\rangle - \bar{r}^2$ [Å^2]	Ω [Å^2]
MLWF	(a)	(–0.857,0.857,0.857)	1.78	(e)	(0.857,–0.857,0.857)	1.78	7.12
SAWF	(b)	(–1.413,1.413,1.413)	1.64	(f)	(–1.413,1.413,1.413)	2.83	10.14
SLWF	(c)	(–0.89, 0.89, 0.89)	1.42	(g)	(0.89,–0.89,0.92)	2.14	9.8
SLWF+C	(d)	(–1.413,1.413,1.413)	1.63	(h)	(1.23,–1.23,1.08)	2.72	7.87

Figure 1. Top (figure): comparison of two representative Wannier functions resulting from different minimisation schemes for gallium arsenide (larger pink spheres: Ga cation atoms, smaller yellow spheres: As anions): (a), (e) MLWF; (b), (f) SAWF; (c), (g) SLWF; (d), (h) SLWF + C. For MLWF, SLWF and SLWF + C, four *s*-type orbitals centred at the midpoints of the four Ga–As bonds are used as the initial guess. In the case of SLWF and SLWF + C, we optimise the first WF (and also constrain its centre to sit at $(\frac{1}{4}, \frac{1}{4}, \frac{1}{4})$, i.e. on the As atom, for SLWF + C), while all the other WFs are left unoptimised. For SAWF, one *s*-type and three *p*-type orbitals centred on the As atom are used as initial guess. Specifically, the first row shows one MLWF (a), one SAWF with *s* character centred on As (b), one WF obtained with the selective localisation scheme (c) and one WF obtained obtained with the selective localisation scheme with additional constraints on its centre (d). The second row shows one of the other three WFs for all four methods. In particular: (e) MLWF, (f) SAWF with *p* character, (g) unoptimised SAWF and (f) unoptimised SAWF + C. For all plots we choose an isosurface level of ± 0.5 Å$^{-\frac{3}{2}}$ (blue for + values and red for − values) using the Vesta visualisation program [28]. Bottom (table): Cartesian coordinates of the centres $\bar{\mathbf{r}}$ and minimised individual spreads $\langle r^2\rangle - \bar{r}^2$ for the two representative Wannier functions of each of the four different minimisation schemes and initial guesses described above. We also report the total spread Ω of all four valence WFs for each method.

disentanglement procedure, with no further additions to the original procedure of section 2.2.

3.3. SAWF and SLWF + C in GaAs

As an example of the capabilities of the SAWF and SLWF + C approaches, we show how to construct atom-centred WFs that possess the local site symmetries in gallium arsenide (GaAs). In particular, we discuss how to obtain one WF from the four valence bands of GaAs that is centred on the As atom and that transforms like the identity under the symmetry operations in T_d, the site-symmetry group of the As site (for completeness, we also show one MLWF and one SLWF without constraints). Since we only deal with the four valence bands of GaAs—an isolated manifold—no prior subspace selection is required for the wannierisation. All calculations were carried out with the plane-wave DFT code Quantum ESPRESSO [1], employing PAW pseudopotentials [24, 25] from the pslibrary (v1.0) [26]. For the exchange-correlation functional we use the Perdew–Burke–Ernzerhof approximation [27]. The energy cut-off for the plane-waves basis is set to 35.0 Ry, and a $4 \times 4 \times 4$ uniform grid is used to sample the Brillouin zone. The lattice parameter is set to the experimental value (5.65 Å). The overlap matrices $M_{mn}^{(\mathbf{k},\mathbf{b})}$ in equation (9), the projection matrices $A_{mn\mathbf{k}}$ in equation (26) and both $\widetilde{d}_{\mathbf{k}}(g)$ in equation (30) and $D_{\mathbf{k}}(g)$ in equation (31) have been computed with the `pw2wannier90.x` interface.

GaAs is a III–V semiconductor that crystallises in the fcc cubic structure, with a two-atom basis: the Ga cation and the As anion (space group $F\bar{4}3m$); in our example the Ga atom is placed at the origin of the unit cell, whose Wyckoff letter is *a* and site-symmetry group is $\bar{4}3m$, also known as T_d. The As atom is placed at $(\frac{1}{4}, \frac{1}{4}, \frac{1}{4})$, whose Wyckoff letter is *c* and site-symmetry group is also T_d.

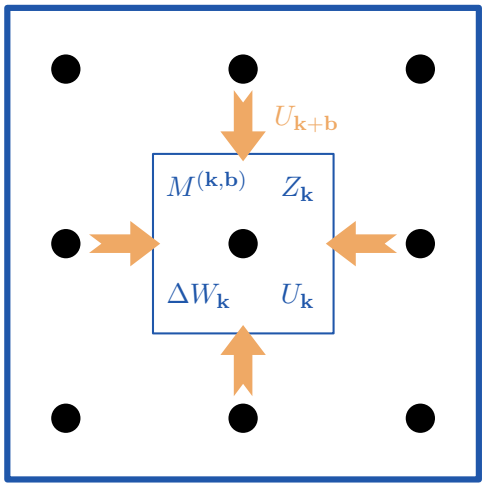


Figure 2. Illustration of the parallelisation scheme for a 3×3 mesh of k-points (black dots) and one MPI process per k-point. The calculation of the $M^{(\mathbf{k},\mathbf{b})}$, $Z_{\mathbf{k}}$, $\Delta W_{\mathbf{k}}$ and $U_{\mathbf{k}}$ matrices are distributed over processes by k-point. The $U_{\mathbf{k}+\mathbf{b}}$ matrices for the neighbouring k-points are sent from process to process (orange arrows) for the calculation of the $M^{(\mathbf{k},\mathbf{b})}$ and $Z_{\mathbf{k}}$ matrices.

Marzari and Vanderbilt [15] have shown that the MLWFs for the 4-dimensional valence manifold are centred on the four As–Ga bonds, have sp^3 character and can be found by specifying four s-like orbitals on each covalent bond as initial guess (two representatives are shown in figures 1(a) and (e)). These bond-centred functions correspond to the irreducible representation A_1 of the site-symmetry group C_{3v} of the Wyckoff position e. Hence, the MLWFs can also be obtained with the SAWF approach by specifying the centres and the shapes of the initial projections, e.g. four s-like orbitals centred on the four As–Ga bonds, and the symmetry operations in the point group C_{3v}.

Using the SAWF method we can enforce the WFs to have the local site symmetries. In particular, since T_d has 5 *irreps* of dimension 1, 1, 2, 3 and 3 respectively, one can form an $1 + 3$–dimensional representation for the four SAWFs. Thus, a set of initial projections compatible with the symmetries of the valence bands is: one s-like orbital (1-dimensional *irrep* whose character is A_1) and three p-like orbitals (3-dimensional *irrep* whose character is T_2) centred on As. Figure 1(b) shows the SAWF which corresponds to the A_1 representation and transforms like the identity under T_d and figure 1(f) one of the three SAWF corresponding to the 3-dimensional *irrep* with p character.

The same SAWF corresponding to the A_1 representation can be obtained with the SLWF + C method by selectively localising one function $J' = 1$ $(J = 4)$ and constraining its centre to sit on the As site $(\frac{1}{4}, \frac{1}{4}, \frac{1}{4})$. In the case of GaAs the SLWF + C method turns out to be very robust, to the point that four s-like orbitals randomly centred in the unit cell can be used as initial guess without affecting the result of the optimised function. Figure 1(c) shows the resulting function using the SLWF method without constraints, while figure 1(d) shows the result using SLWF + C, which is identical to the SAWF in figure 1(b). Finally, in the second row of figures 1 (e), (f), (g) and (h) one of the other three Wannier function is shown for all four minimisation scheme. In the case of SAWF, figure 1(f), this WF is centred on the As atom, has a larger spread than the corresponding MLWF (figure 1(e)) and shows a p-like character as expected. However, in the case of SLWF and SLWF + C (figures 1(g) and (h)), these WFs are not optimised and therefore they show a larger spread than the corresponding MLWF and are somewhat less symmetric (see table in figure 1).

It is worth to note that for this particular system, it is possible to achieve the result of a s-like and three p-like WFs also with the maximal localisation procedure if one carefully selects the initial projections, i.e. one s-like and three p-like orbitals on the As atom. The resulting WFs will possess the local site symmetries but will not correspond to the global minimum of the spread functional Ω. More precisely, they will correspond to a saddle point of Ω (unstable against small perturbations of the initial projections).

3.4. Parallelisation

In WANNIER90 v3.0 we have implemented an efficient parallelisation scheme for the calculation of MLWFs using the message passing interface (MPI).

3.4.1. Calculation of the spread and distribution of large matrices. The time-consuming part in the evaluation of the spread Ω is updating the $M^{(\mathbf{k},\mathbf{b})}$ matrices according to equation (16), since this requires computing overlap matrix elements between all pairs of bands, and between all k-points $\mathbf{k}$ and their neighbours $\mathbf{k} + \mathbf{b}$. Therefore, an efficient speed up for the evaluation of the spread can be achieved by distributing over several processes the calculation of the $M^{(\mathbf{k},\mathbf{b})}$ matrices for different k-points. In order to compute the $M^{(\mathbf{k},\mathbf{b})}$ according to equation (16), the $U_{\mathbf{k}+\mathbf{b}}$ matrices are sent from process to process prior to the calculation of the overlap matrices. We stress the fact that the $U_{\mathbf{k}+\mathbf{b}}$ matrices are the only large arrays that have to be shared between processes, which limits the time spent in communication. The relatively large $M^{(\mathbf{k},\mathbf{b})}$ matrices are not sent between processes for the evaluation of equations (7) and (8). Instead, it is enough to collect the contributions to the spread from the different k-points, i.e. a set of scalars, and then sum them up for evaluation of the total spread. This parallelisation scheme is illustrated in figure 2 for a 3×3 mesh of k-points with 9 MPI processes.

Moreover, we emphasise that our parallelisation scheme relies on the evaluation of relevant matrices over k-points on each process (or core, since the only parallelisation scheme currently implemented is MPI and typically each process is assigned to a different CPU core). For systems with large number of k-points and bands, it is also desirable to distribute these matrices across the available processes to reduce the memory requirements. For example, in the case of isolated bands, instead of storing all the $M^{(\mathbf{k},\mathbf{b})}$ matrices on all cores (requiring an allocation per core of dimension $J \times J \times N \times N_b$, where N_b is the number neighbours of each of the N k-points of the mesh) we distribute the matrices across the N_c cores. In

particular, only the root process stores the full matrices (for I/O purposes) while all other processes just store the $M^{(\mathbf{k},\mathbf{b})}$ matrices for the k-points associated with the given process. In such a way, the memory requirement per core (for the $M^{(\mathbf{k},\mathbf{b})}$ matrices) decreases by a factor of approximately N_c.

3.4.2. Minimisation of the spread. The minimisation of the spread functional is based on an iterative steepest-descent or conjugate-gradient algorithm. In each iteration, the unitary matrices $U_\mathbf{k}$ are updated according to $U_\mathbf{k} = U_\mathbf{k} \exp(\Delta W_\mathbf{k})$ [15], where $\Delta W_\mathbf{k} = \alpha \mathcal{D}_\mathbf{k}$, see equation (15). Updating the $U_\mathbf{k}$ matrices according to this equation is by far the most time-consuming part in the iterative minimisation algorithm, as it requires a diagonalisation of the $\Delta W_\mathbf{k}$ matrices. A significant speed-up can be obtained, however, by distributing the diagonalisation of the different $\Delta W_\mathbf{k}$ matrices over several processes, and performing the calculations fully in parallel. The evaluation of $\Delta W_\mathbf{k}$ essentially requires the calculation of the overlap matrices $M^{(\mathbf{k},\mathbf{b})}$, as discussed above.

3.4.3. Disentanglement. The disentanglement procedure is concerned with finding the optimal subspace $\mathcal{S}_\mathbf{k}$. As the functional Ω_I measures the global subspace dispersion across the Brillouin zone, at first sight it is not obvious that the task of minimising the spread Ω_I can be parallelised with respect to the k-points. In the iterative algorithm of equation (21), the systematic reduction of the spread functional at the ith iteration is achieved by minimising the spillage of the subspace $\mathcal{S}_\mathbf{k}^{(i)}$ over the neighbouring subspaces from the previous iteration $\mathcal{S}_{\mathbf{k}+\mathbf{b}}^{(i-1)}$. This problem reduces to the diagonalisation of N independent matrices (N is the total number of k-points of the mesh), where an efficient speed-up of the disentanglement procedure can be achieved by distributing the diagonalisation of the $Z_\mathbf{k}^{(i)}$ matrices of equation (23) over several processes, which can be done fully in parallel. Since the construction of $Z_\mathbf{k}^{(i)}$ only requires the knowledge of the $U_{\mathbf{k}+\mathbf{b}}^{(i-1)}$ matrices, these must be communicated between processes, as shown in figure 2. This results in a similar time spent in communication for the disentanglement part of the code as for the wannierisation part.

3.4.4. Performance. We have tested the performance of this parallelisation scheme for the calculation of the MLWFs in a FePt(5)/Pt(18) thin film. Computational details were given in [29]. The benchmarks have been performed on the JURECA supercomputer of the Jülich Supercomputing Center. We have extracted an optimal subspace of dimension $J = 414$ from a set of 580 Bloch states per k-point. The upper limit of the inner window was set to 5 eV above the Fermi energy, and 414 MLWFs were constructed by minimising the spread Ω. The performance benchmark was based on the average wall-clock time for a single iteration of the minimisation procedure (several thousand iterations are usually needed for convergence). We first analyse the weak scaling of our implementation, i.e. how the computation time varies with the number of cores N_c for a fixed number of k-points per process. We show in figure 3(a) the time per iteration for the disentanglement and wannierisation parts of the minimisation, always using one k-point per process. As we vary the number of k-points N from 4 to 144, the computation time increases only by a factor of 1.3 and 1.8 for disentanglement and wannierisation, respectively. We then demonstrate the strong scaling of our parallelisation scheme in figure 3(b), i.e. how the computation time varies with the number of cores N_c for a fixed number $N = 64$ of k-points. When varying the number of cores from 4 to 64, we observe a decrease of the computation time per iteration by a factor of 12.6 and 9.5 for disentanglement and wannierisation, respectively. The deviation from ideal scaling is mostly explained by the time spent in inter-core communication of the $U_{\mathbf{k}+\mathbf{b}}$ matrices.

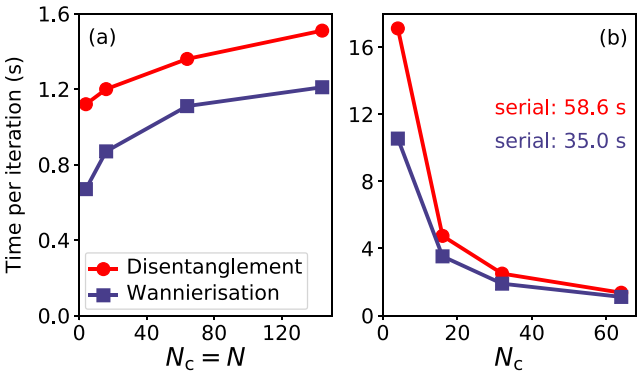


Figure 3. Plots of the time per single minimisation iteration as a function of the number of cores N_c. (a) Weak scaling of the implementation, where the number of k-points per process is fixed to one, i.e. $N_c = N$. The time only increases by a factor 1.3 (1.8) for the disentanglement (wannierisation) parts of the code, when going from $N_c = 4$ to $N_c = 144$. (b) Strong scaling of the algorithm for a fixed number of k-points $N = 64$. The time per iteration with one single CPU (serial) is reported in the figure.

4. Enhancements in functionality

In this section we describe a number of enhancements to the functionality of the core WANNIER90 code, namely: the ability to compute and visualise spinor-valued WFs, including developments to the interface with the QUANTUM ESPRESSO package to cover also the case of non-collinear spin calculations performed with ultrasoft pseudopotentials (previously not implemented); an improvement to the method for interpolating the k-space Hamiltonian; the ability to select a subset from a larger set of projections of localised trial orbitals onto the Bloch states for initialising the WFs; and new functionality for plotting WFs in Gaussian cube format on WF-centred grids with non-orthogonal translation vectors.

4.1. Spinor-valued Wannier functions with ultrasoft and projector-augmented-wave pseudopotentials

The calculation of the overlap matrix in equation (17) within the ultrasoft-pseudopotential formalism proceeds via the inclusion of so-called augmentation functions [30],

$$M_{mn}^{(\mathbf{k},\mathbf{b})} = \langle u_{m\mathbf{k}} | u_{n,\mathbf{k}+\mathbf{b}} \rangle + \sum_{Iij} Q_{ij}^{I}(\mathbf{b}) \langle \psi_{m\mathbf{k}}^{\mathrm{ps}} | B_{Iij}^{(\mathbf{k},\mathbf{b})} | \psi_{n,\mathbf{k}+\mathbf{b}}^{\mathrm{ps}} \rangle, \quad (40)$$

where $|\psi_{m\mathbf{k}}^{\mathrm{ps}}\rangle$ is the pseudo-wavefunction,

$$Q_{ij}^{I}(\mathbf{b}) = \int \mathrm{d}\mathbf{r}\; Q_{ij}^{I}(\mathbf{r}) \mathrm{e}^{-\mathrm{i}\mathbf{b}\cdot\mathbf{r}} \quad (41)$$

is the Fourier transform of the augmentation charge, and $B_{Iij}^{(\mathbf{k},\mathbf{b})} = |\beta_{Ii}^{\mathbf{k}}\rangle\langle\beta_{Ij}^{\mathbf{k}+\mathbf{b}}|$, where $|\beta_{Ii}^{\mathbf{k}}\rangle$ denotes the ith projector of the pseudopotential on the Ith atom in the unit cell. We refer to appendix B of [30] for detailed expressions.

When spin–orbit coupling is included, the Bloch functions become two-component spinors $(\psi_{n\mathbf{k}}^{\uparrow}(\mathbf{r}), \psi_{n\mathbf{k}}^{\downarrow}(\mathbf{r}))^{\mathrm{T}}$, where $\psi_{n\mathbf{k}}^{\sigma}(\mathbf{r})$ is the spin-up (for $\sigma = \uparrow$) or spin-down (for $\sigma = \downarrow$) component with respect to the chosen spin quantisation axis. Accordingly, $Q_{ij}^{I}(\mathbf{b})$ becomes $Q_{ij}^{I\sigma\sigma'}(\mathbf{b})$ (see equation (18) in [31]) and equation (40) becomes

$$M_{mn}^{(\mathbf{k},\mathbf{b})} = \langle u_{m\mathbf{k}} | u_{n,\mathbf{k}+\mathbf{b}} \rangle + \sum_{Iij\sigma\sigma'} Q_{ij}^{I\sigma\sigma'}(\mathbf{b}) \langle \psi_{m\mathbf{k}}^{\mathrm{ps},\sigma} | B_{Iij}^{(\mathbf{k},\mathbf{b})} | \psi_{n,\mathbf{k}+\mathbf{b}}^{\mathrm{ps},\sigma'} \rangle. \quad (42)$$

The above expressions, together with the corresponding ones for the matrix elements of the spin operator, have been implemented in the `pw2wannier90.x` interface between QUANTUM ESPRESSO and WANNIER90.

The plotting routines of WANNIER90 have also been adapted to work with the complex-valued spinor WFs obtained from calculations with spin–orbit coupling. It then becomes necessary to decide how to represent graphically the information contained in the two spinor components.

One option is to only plot the norm $|w_{n\mathbf{k}}(\mathbf{r})| = \sqrt{|w_{n\mathbf{k}}^{\uparrow}(\mathbf{r})|^2 + |w_{n\mathbf{k}}^{\downarrow}(\mathbf{r})|^2}$ of spinor WFs (where the up- and down-spin components of the spinor WF $w^{\uparrow,\downarrow}$ are obtained as in equation (2) by replacing ψ with $\psi^{\uparrow,\downarrow}$), which is reminiscent of the total charge density in the case of a 2×2 density matrix in non-collinear DFT. Another possibility is to plot independently the up- and down-spin components of the spinor WF. Since each of them is in general complex-valued, two options are provided in the code: (i) to plot only the magnitudes $|w_{n\mathbf{k}}^{\uparrow}(\mathbf{r})|$ and $|w_{n\mathbf{k}}^{\downarrow}(\mathbf{r})|$ of the two components; or (ii) to encode the phase information by outputting $|w_{n\mathbf{k}}^{\uparrow}(\mathbf{r})|\mathrm{sgn}(\mathrm{Re}\{w_{n\mathbf{k}}^{\uparrow}(\mathbf{r})\})$ and $|w_{n\mathbf{k}}^{\downarrow}(\mathbf{r})|\mathrm{sgn}(\mathrm{Re}\{w_{n\mathbf{k}}^{\downarrow}(\mathbf{r})\})$, where sgn is the sign function. Which of these various options is adopted by the WANNIER90 code is controlled by two input parameters, `wannier_plot_spinor_mode` and `wannier_plot_spinor_phase`.

Finally we note that, for WFs constructed from ultrasoft pseudopotentials or within the projector-augmented-wave (PAW) method, only pseudo-wavefunctions represented on the soft FFT grid are considered in plotting WFs within the present scheme, that is, the WFs are not normalised. We emphasise that this affects only plotting of the WFs in real-space and not the calculation of the MLWFs (the overlap matrices being correctly computed by the interface codes).

4.2. Improved Wannier interpolation by minimal-distance replica selection

The interpolation of band structures (and many other quantities) based on Wannier functions is an extremely powerful tool [32–34]. In many respects it resembles Fourier interpolation, which uses discrete Fourier transforms to reconstruct faithfully continuous signals from a discrete sampling, provided that the signal has a finite bandwidth and that the sampling rate is at least twice the bandwidth (the so-called Nyquist–Shannon condition).

In the context of Wannier interpolation, the 'sampled signal' is the set of matrix elements

$$H_{mn\mathbf{k}_j} = \langle \chi_{m\mathbf{k}_j} | H | \chi_{n\mathbf{k}_j} \rangle \quad (43)$$

of a lattice-periodic operator such as the Hamiltonian, defined on the same uniform grid $\{\mathbf{k}_j\}$ that was used to minimise the Wannier spread functional (see section 2.1). The states $|\chi_{n\mathbf{k}_j}\rangle$ are the Bloch sums of the WFs, related to *ab initio* Bloch eigenstates by $|\chi_{n\mathbf{k}_j}\rangle = \sum_m |\psi_{m\mathbf{k}_j}\rangle U_{mn\mathbf{k}_j}$.

To reconstruct the 'continuous signal' $H_{mn\mathbf{k}}$ at arbitrary $\mathbf{k}$, the matrix elements of equation (43) are first mapped onto real space using the discrete Fourier transform

$$\widetilde{H}_{mn\mathbf{R}} = \langle w_{m\mathbf{0}} | H | w_{n\mathbf{R}} \rangle = \frac{1}{N} \sum_{j=1}^{N} \mathrm{e}^{-\mathrm{i}\mathbf{k}_j\cdot\mathbf{R}} H_{mn\mathbf{k}_j}, \quad (44)$$

where $N = N_1 \times N_2 \times N_3$ is the grid size (which is also the number of k-points in WANNIER90). The matrices $H_{mn\mathbf{k}_j}$ are then interpolated onto an arbitrary $\mathbf{k}$ using an inverse discrete Fourier transform,

$$H_{mn\mathbf{k}} = \sum_{\mathbf{R}'} \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot\mathbf{R}'} \widetilde{H}_{mn\mathbf{R}'}, \quad (45)$$

where the sum is over N lattice vectors $\mathbf{R}'$, and the interpolated energy eigenvalues are obtained by diagonalising $H_{\mathbf{k}}$. In the limit of an infinitely dense grid of k-points the procedure is exact and the sum in equation (45) becomes an infinite series. Owing to the real-space localisation of the Wannier functions, the matrix elements $\widetilde{H}_{mn\mathbf{R}}$ become vanishingly small when the distance between the Wannier centres exceeds a critical value L (the 'bandwidth' of the Wannier Hamiltonian), so that actually only a finite number of terms contributes significantly to the sum in equation (45). This means that, even with a finite $N_1 \times N_2 \times N_3$ grid, the interpolation is still accurate provided that—by analogy with the Nyquist–Shannon condition—the 'sampling rate' N_i along each cell vector $\mathbf{a}_i$ is sufficiently large to ensure that $N_i|\mathbf{a}_i| > 2L$.

Still, the result of the interpolation crucially depends on the choice of the N lattice vectors to be summed over in equation (45). Indeed, when using a finite grid, there is a considerable freedom in choosing the set $\{\mathbf{R}'\}$ as $\widetilde{H}_{mn\mathbf{R}}$ is invariant under $\mathbf{R} \to \mathbf{R} + \mathbf{T}$ for any vector $\mathbf{T}$ of the Born–von Karman superlattice generated by $\{\mathbf{A}_i = N_i\mathbf{a}_i\}$. The phase factor in

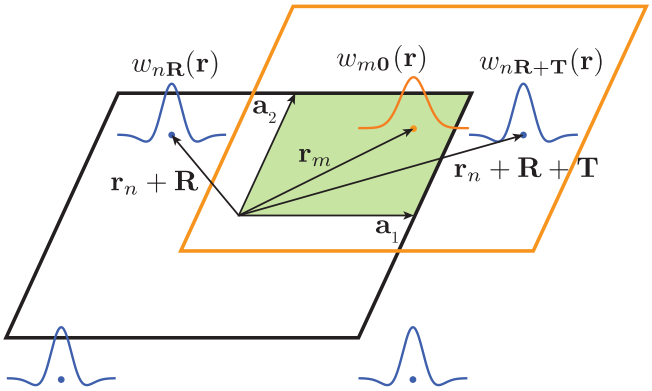


Figure 4. Owing to the periodicity of the Wannier functions over the Born–von Karman supercell (with size 2 × 2 here), the matrix element $\widetilde{H}_{mn\mathbf{R}}$ describes the interaction between the mth WF $w_{m\mathbf{0}}$ (shown in orange) with centre $\mathbf{r}_m$ inside the home unit cell $\mathbf{R} = \mathbf{0}$ (green shaded area) and the nth WF $w_{n\mathbf{R}}$ (shown in blue) centred inside the unit cell $\mathbf{R}$, or any of its supercell-periodic replicas displaced by a superlattice vector $\mathbf{T}$. When performing Wannier interpolation, we now impose a minimal-distance condition by choosing the replica $w_{n,\mathbf{R}+\mathbf{T}}$ of $w_{n\mathbf{R}}$ whose centre lies within the Wigner–Seitz supercell centred at $\mathbf{r}_m$ (thick orange line).

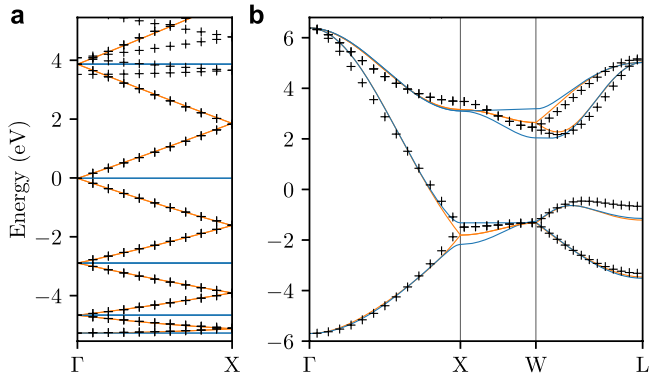


Figure 5. Comparison between the bands obtained using the earlier interpolation procedure (blue lines), those obtained using the (current) modified approach of equation (47) (orange lines), and the *ab initio* bands (black crosses). (a) Linear chain of carbon atoms, with 12 atoms per unit cell (separated by a distance of 1.3 Å along the z direction) and Γ-point sampling. 36 Wannier functions have been computed starting from projections over p_x and p_y orbitals on carbon atoms and s-orbitals midbond between them. A frozen window up to the Fermi energy (set to zero in the plot) has been considered, while the disentanglement window included all states up to ∼14 eV above the Fermi level. (b) Bulk silicon, with the BZ sampled on an unconverged 3 × 3 × 3 grid of k-points.

equation (45) is also invariant when $\mathbf{k} \in \{\mathbf{k}_j\}$, but not for arbitrary $\mathbf{k}$. Hence we need to choose, among the infinite set of 'replicas' $\mathbf{R}' = \mathbf{R} + \mathbf{T}$ of $\mathbf{R}$, which one to include in equation (45). We take the original vectors $\mathbf{R}$ to lie within the Wigner–Seitz supercell centred at the origin. If some of them fall on its boundary then their total number exceeds N and weight factors must be introduced in equation (45). For each combination of m, n and $\mathbf{R}$, the optimal choice of $\mathbf{T}$ is the one that minimises the distance

$$|\mathbf{r}_m - (\mathbf{r}_n + \mathbf{R} + \mathbf{T})| \tag{46}$$

between the two Wannier centres. With this choice, the spurious effects arising from the artificial supercell periodicity are minimised.

Earlier versions of WANNIER90 implemented a simplified procedure whereby the vectors $\mathbf{R}'$ in equation (45) were chosen to coincide with the unshifted vectors $\mathbf{R}$ that are closer to the origin than to any other point $\mathbf{T}$ on the superlattice, irrespective of the WF pair (m, n). As illustrated in figure 4, this procedure does not always lead to the shortest distance between the pair of WFs, especially when some of the N_i are small and the Wannier centres are far from the origin of the cell.

WANNIER90 now implements an improved algorithm that enforces the minimal-distance condition of equation (46), yielding a more accurate Fourier interpolation. The algorithm is the following:

(a) For each term in equation (45) pick, among all the replicas $\mathbf{R}' = \mathbf{R} + \mathbf{T}$ of $\mathbf{R}$, the one that minimises the distance between Wannier centres (equation (46)).
(b) If there are $\mathcal{N}_{mn\mathbf{R}}$ different vectors $\mathbf{T}$ for which the distance of equation (46) is minimal, then include all of them in equation (45) with a weight factor $1/\mathcal{N}_{mn\mathbf{R}}$.

An equivalent way to describe these steps is that (a) we choose $\mathbf{T}$ such that $\mathbf{r}_n + \mathbf{R} + \mathbf{T}$ falls inside the Wigner–Seitz supercell centred at $\mathbf{r}_m$ (see figure 4), and that (b) if it falls on a face, edge or vertex of the Wigner–Seitz supercell, we keep all the equivalent replicas with an appropriate weight factor. In practice the condition in step (b) is enforced within a certain tolerance, to account for the numerical imprecision in the values of the Wannier centres and in the definition of the unit cell vectors. Although step (b) is much less important than (a) for obtaining a good Fourier interpolation, it helps ensuring that the interpolated bands respect the symmetries of the system; if step (b) is skipped, small artificial band splittings may occur at high-symmetry points, lines, or planes in the BZ.

The procedure outlined above amounts to replacing equation (45) with

$$H_{mn\mathbf{k}} = \sum_{\mathbf{R}} \frac{1}{\mathcal{N}_{mn\mathbf{R}}} \sum_{j=1}^{\mathcal{N}_{mn\mathbf{R}}} \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot(\mathbf{R}+\mathbf{T}^{(j)}_{mn\mathbf{R}})} \widetilde{H}_{mn\mathbf{R}}, \tag{47}$$

where $\{\mathbf{T}^{(j)}_{mn\mathbf{R}}\}$ are the $\mathcal{N}_{mn\mathbf{R}}$ vectors $\mathbf{T}$ that minimise the distance of equation (46) for a given combination of m, n and $\mathbf{R}$; $\mathbf{R}$ lies within the Wigner–Seitz supercell centred on the origin.

The benefits of this modified interpolation scheme are most evident when considering a large unit cell sampled at the Γ point only. In this case $N = 1$ so that equation (45) with $\{\mathbf{R}'\} = \{\mathbf{R}\} = \{\mathbf{0}\}$ would reduce to $H_{mn\mathbf{k}} = \widetilde{H}_{mn\mathbf{0}}$, yielding interpolated bands that do not disperse with $\mathbf{k}$. This is nonetheless an artefact of the choice $\{\mathbf{R}'\} = \{\mathbf{0}\}$ (of earlier versions of WANNIER90) and not an intrinsic limitation of Wannier interpolation, as first demonstrated in [32] for one-dimensional systems. Indeed, equation (47), which in a sense extends [32] to any spatial dimension, becomes in this case

$$H_{mn\mathbf{k}} = \frac{\widetilde{H}_{mn\mathbf{0}}}{\mathcal{N}_{mn\mathbf{0}}} \sum_{j=1}^{\mathcal{N}_{mn\mathbf{0}}} \mathrm{e}^{\mathrm{i}\mathbf{k}\cdot\mathbf{T}^{(j)}_{mn\mathbf{0}}}, \tag{48}$$

which can produce dispersive bands. This is illustrated in figure 5(a) for the case of a one-dimensional chain of carbon atoms: the interpolated bands obtained from equation (45) with $\{\mathbf{R}'\} = \{\mathbf{R}\} = \{\mathbf{0}\}$ (earlier version of WANNIER90) are flat, while those obtained from equation (47) (new versions of WANNIER90) are in much better agreement with the dispersive *ab initio* bands up to a few eV above the Fermi energy.

Clear improvements in the interpolated bands are also obtained for bulk solids, as shown in figure 5(b) for the case of silicon. The earlier implementation breaks the two-fold degeneracy along the X − W line, with one of the two bands becoming flat. The new procedure recovers the correct degeneracies, and reproduces more closely the *ab initio* band structure (the remaining small deviations are due to the use of a coarse *k*-point mesh that does not satisfy the Nyquist–Shannon condition, and would disappear for denser *k*-grids together with the differences between the two interpolation procedures).

4.3. Selection of projections

In many cases, and particularly for entangled bands, it is necessary to have a good initial guess for the MLWFs in order to properly converge the spread to the global minimum. Determining a good initial guess often involves a trial and error approach, using different combinations of orbital types, orientations and positions. While for small systems performing many computations of the projection matrices is relatively cheap, for large systems there is a cost associated with storing and reading the wavefunctions to compute new projection matrices for each new attempt at a better initial guess. Previously, the number of projections that could be specified had to be equal to the number J of WFs to be constructed. The latest version of the code lifts this restriction, making it possible to define in the pre-processing step a larger number $J_+ > J$ of projection functions to consider as initial guesses. In this way, the computationally expensive and potentially I/O-heavy construction of the projection matrices $A_{\mathbf{k}}$ (of dimension $J \times J_+$ at each $\mathbf{k}$) is performed only once for all possible projections that a user would like to consider.

Once the $A_{\mathbf{k}}$ matrices have been obtained, one proceeds with constructing the MLWFs by simply selecting, via a new input parameter (`select_projections`) of the WANNIER90 code, which J columns to use among the J_+ that were computed by the interface code. Experimenting with different trial orbitals can thus be achieved by simply selecting a different set of projections within the WANNIER90 input file, without the need to perform the pre-processing step again.

Similarly, another use case for this new option is the construction of WFs for the same material but for different groups of bands. Typically one would have to modify the WANNIER90 input file and run the interface code multiple times, while now the interface code may compute $A_{\mathbf{k}}$ for a superset of trial orbitals just once, and then different subsets may be chosen by simple modification of a single input parameter. As a demonstration, we have adapted `example11` (silicon band structure) of the WANNIER90 distribution, that considers two band groups: (a) the valence bands only, described by four bond-centred *s* orbitals, and (b) the four valence and the four lowest-lying conduction bands together, described by atom-centred sp^3 orbitals. In the example, we specify projections onto all 12 trial orbitals, and the different cases are covered by specifying in the WANNIER90 input file which subset of projections is required.

4.4. Plotting cube files with non-orthogonal vectors

In WANNIER90 v3.0 it is possible to plot the MLWFs in real-space in Gaussian cube format, including the case of non-orthogonal cell lattice vectors. Many modern visualisation programs such as Vesta [28] are capable of handling non-orthogonal cube files and the cube file format can be read by many computational chemistry programs. WANNIER90's representation of MLWFs in cube format can be significantly more compact than using the alternative `xsf` format. With the latter, MLWFs are calculated (albeit with a coarse sampling) on a supercell of the computational cell that can be potentially large (the extent of the supercell is controlled by an input parameter `wannier_plot_supercell`). Whereas, with the cube format, each Wannier function is represented on a grid that is centred on the Wannier function itself and has a user-defined extent, which is the smallest parallelepiped (whose sides are aligned with the cell vectors) that can enclose a sphere with a user-defined radius `wannier_plot_radius`. Because MLWFs are strongly localised in real space, relatively small cut-offs are all that is required, significantly smaller than the length-scale over which the MLWFs themselves are periodic. As a result, the cube format is particularly useful when a more memory-efficient representation is needed. The cube format can be activated by setting the input parameter `wannier_plot_mode` to `cube`, and the code can handle both isolated molecular systems (treated within the supercell approximation) as well as periodic crystals by setting `wannier_plot_mode` to either `molecule` or `crystal`, respectively.

5. New post-processing features

Once the electronic bands of interest have been disentangled and wannierised to obtain well-localised WFs, the WANNIER90 software package includes a number of modules and utilities that use these WFs to calculate various electronic-structure properties. Much of this functionality exists within `postw90.x`, an MPI-parallel code that forms an integral part of the WANNIER90 package. In v2.x of WANNIER90, `postw90.x` included functionality for computing densities of states and partial densities of states, energy bands and Berry curvature along specified lines and planes in *k*-space, anomalous Hall conductivity, orbital magnetisation and optical conductivity, Boltzmann transport coefficients within the relaxation time approximation, and band energies and derivatives on a generic user-defined list of *k*-points. Some further functionality exists in a set of utilities that are provided as part of the WANNIER90 package, including a code (`w90pov.F90`) to plot WFs rendered using the Persistence of Vision Raytracer (POV-Ray) [35]

code and to compute van der Waals interactions with WFs (`w90vdw.F90`).

In addition, there are a number of external packages for computing advanced properties based on WFs and which interface to WANNIER90. These include codes to generate tight-binding models such as PYTHTB [36] and TBMODELS [37], quantum transport codes such as SISL [38], GOLLUM [39], OMEN [40] and NANOTCAD-VIDES [41], the EPW [42] code for calculating properties related to electron-phonon interactions and WANNIERTOOLS [43] for the investigation of novel topological materials.

Below we describe some of the new post-processing features of WANNIER90 that have been introduced in the latest version of the code, v3.0.

5.1. postw90.x: shift current

The photogalvanic effect (PGE) is a nonlinear optical response that consists in the generation of a direct current (DC) when light is absorbed [44–46]. It can be divided phenomenologically into linear (LPGE) and circular (CPGE) effects, which have different symmetry requirements within the acentric crystal classes. The CPGE requires elliptically-polarised light, and occurs in gyrotropic crystals (see next subsection). The LPGE occurs with linearly or unpolarised light as well; it is present in piezoelectric crystals and is given by

$$J_a(0) = 2\sigma_{abc}(0;\omega,-\omega)E_b(\omega)E_c(-\omega), \tag{49}$$

where $\mathbf{J}(0)$ is the induced DC photocurrent density, $\mathbf{E}(\omega) = \mathbf{E}^*(-\omega)$ is the amplitude of the optical electric field, and $\sigma_{abc} = \sigma_{acb} = \sigma^*_{abc}$ is a nonlinear photoconductivity tensor.

The shift current is the part of the LPGE photocurrent generated by interband light absorption [47]. Intuitively, it arises from a coordinate shift accompanying the photoexcitation of electrons from one band to another. Like the intrinsic anomalous Hall effect [48], the shift current involves off-diagonal velocity matrix elements between occupied and empty bands, depending not only on their magnitudes but also on their phases [49–52].

The shift current along direction a induced by light that is linearly polarised along b is described by the following photoconductivity tensor [52, 53]:

$$\sigma^{\text{shift}}_{abb}(0;\omega,-\omega) = -\frac{\pi|e|^3}{\hbar^2}\int_{\text{BZ}}\frac{\mathrm{d}\mathbf{k}}{(2\pi)^3}\sum_{n,m} f_{nm\mathbf{k}}\,R^{ab}_{nm\mathbf{k}} \times \left|r^b_{nm\mathbf{k}}\right|^2\delta(\omega_{mn\mathbf{k}}-\omega). \tag{50}$$

Here, $f_{nm\mathbf{k}} = f_{n\mathbf{k}} - f_{m\mathbf{k}}$ is the difference between occupation factors, $\hbar\omega_{mn\mathbf{k}} = \epsilon_{m\mathbf{k}} - \epsilon_{n\mathbf{k}}$ is the difference between energy eigenvalues of the Bloch bands, $r^b_{nm\mathbf{k}}$ is the bth Cartesian component of the interband dipole matrix (the off-diagonal part of the Berry connection matrix $\mathbf{A}_{nm\mathbf{k}} = i\langle u_{n\mathbf{k}}|\partial_{\mathbf{k}}u_{m\mathbf{k}}\rangle$), and

$$R^{ab}_{nm\mathbf{k}} = \partial_{k_a}\arg\left(r^b_{nm\mathbf{k}}\right) - A^a_{nn\mathbf{k}} + A^a_{mm\mathbf{k}} \tag{51}$$

is the *shift vector* (not to be confused with the lattice vector $\mathbf{R}$, or with the matrix $R^{(\mathbf{k},\mathbf{b})}$ defined in equation (12)). The shift vector has units of length, and it describes the real-space shift of wavepackets under photoexcitation.

The numerical evaluation of equation (51) is tricky because the individual terms therein are gauge dependent, and only their sum is unique. Different strategies were discussed in the early literature in the context of model calculations [51, 54], and more recently for *ab initio* calculations. The *ab initio* implementation of Young and Rappe [55] employed a gauge-invariant k-space discretisation of equation (51), inspired by the discretised Berry-phase formula for electric polarisation [56].

The implementation in WANNIER90 is based instead on the formulation of Sipe and co-workers [52, 57]. In this formulation, the shift (interband) contribution to the LPGE tensor in equation (49) is expressed as

$$\sigma^{\text{shift}}_{abc}(0;\omega,-\omega) = \frac{\mathrm{i}\pi|e|^3}{4\hbar^2}\int_{\text{BZ}}\frac{\mathrm{d}\mathbf{k}}{(2\pi)^3}\sum_{n,m} f_{nm\mathbf{k}} \times \left(r^b_{mn\mathbf{k}}r^{c;a}_{nm\mathbf{k}} + r^c_{mn\mathbf{k}}r^{b;a}_{nm\mathbf{k}}\right) \times \left[\delta(\omega_{mn\mathbf{k}}-\omega) + \delta(\omega_{nm\mathbf{k}}-\omega)\right], \tag{52}$$

where

$$r^{b;a}_{nm\mathbf{k}} = \partial_{k_a}r^b_{nm\mathbf{k}} - \mathrm{i}\left(A^a_{nn\mathbf{k}} - A^a_{mm\mathbf{k}}\right)r^b_{nm\mathbf{k}} \tag{53}$$

is the *generalised derivative* of the interband dipole. When $b = c$, equation (52) becomes equivalent to equation (50) [52].

The generalised derivative $r^{b;a}_{nm\mathbf{k}}$ is a well-behaved (covariant) quantity under gauge transformation but—as in the case of the gauge-invariant shift vector—this is not the case for the individual terms in equation (53), leading to numerical instabilities. To circumvent this problem, Sipe and co-workers used $\mathbf{k}\cdot\mathbf{p}$ perturbation theory to recast equation (53) as a summation over intermediate virtual states where the individual terms are gauge covariant [52, 57]. That strategy has been successfully employed to evaluate the shift-current spectrum from first principles [58, 59].

As it is well known, similar 'sum-over-states' expressions can be written for other quantities involving $\mathbf{k}$ derivatives, such as the inverse effective-mass tensor and the Berry curvature. When evaluating those expressions, a sufficient number of virtual states should be included to achieve convergence. Alternatively, one can work with a basis spanning a finite number of bands, such as a tight-binding or Wannier basis, and carefully reformulate $\mathbf{k}\cdot\mathbf{p}$ perturbation theory within that incomplete basis to avoid truncation errors. This reformulation was carried out in [60] for the inverse effective-mass tensor, and in [33] for the Berry curvature; the formalism of [33] is at the core of the `berry.F90` module of `postw90`, where Berry curvatures and related quantities are computed by Wannier interpolation. The same interpolation strategy was used in [61] and [62] to evaluate equation (52), and the approach of [62] is now implemented in the `berry.F90` module.

5.2. postw90.x: gyrotropic module

In the previous subsection we considered the shift current, an effect that occurs in piezoelectric crystals. Here we turn

to a host of effects that occur in a different group of acentric crystals: those belonging to the *gyrotropic* crystal classes, which include the chiral, polar, and optically-active crystal classes [44].

To motivate the gyrotropic effects considered below, let us start from the more familiar magneto-optical effects. To review, the spontaneous magnetisation of ferromagnets endows their conductivity tensor $\sigma_{ab}(\omega)$ with an antisymmetric part. In the DC limit this antisymmetric conductivity describes the anomalous Hall effect (AHE), and at finite frequencies it describes magneto-optical effects such as Faraday rotation in transmission and magnetic circular dichroism in absorption. In paramagnets, those effects appear under applied magnetic fields.

As first pointed out in [63, 64], an antisymmetric conductivity can be induced in certain nonmagnetic (semi)conductors by purely electrical means: by passing a current through the sample. Symmetry arguments indicate that this is allowed in the gyrotropic crystal classes, and the first experimental demonstration consisted in the measurement of a current-induced change in the rotatory power of p-doped trigonal tellurium [65, 66]. When linearly polarised light of frequency ω propagates along the trigonal $\hat{\mathbf{z}}$ axis of tellurium in the presence of a current density $\mathbf{j} = j_z\hat{\mathbf{z}}$, the change in rotatory power is proportional to $\widetilde{D}_{zz}(\omega)\, j_z$, where

$$\widetilde{D}_{ab}(\omega) = \int_{\mathrm{BZ}} \frac{\mathrm{d}\mathbf{k}}{(2\pi)^3} \sum_n f_0(\epsilon_{n\mathbf{k}})\partial_{k_a}\widetilde{\Omega}^b_{n\mathbf{k}}(\omega). \tag{54}$$

In this expression f_0 is the equilibrium occupation factor, and

$$\widetilde{\mathbf{\Omega}}_{n\mathbf{k}}(\omega) = -\sum_m \frac{\omega^2_{mn\mathbf{k}}}{\omega^2_{mn\mathbf{k}} - \omega^2}\mathrm{Im}[\mathbf{A}_{nm\mathbf{k}} \times \mathbf{A}_{mn\mathbf{k}}], \tag{55}$$

where $\mathbf{A}_{nm\mathbf{k}}$ is the Berry connection matrix introduced in section 5.1. At zero frequency, $\widetilde{\mathbf{\Omega}}_{n\mathbf{k}}(\omega)$ reduces to the Berry curvature $\mathbf{\Omega}_{n\mathbf{k}} = \boldsymbol{\nabla}_{\mathbf{k}} \times \mathbf{A}_{nn\mathbf{k}}$.

The DC or transport limit of the above *current-induced Faraday effect* is the current-induced AHE, or *nonlinear AHE* [67–71]. Like the linear (spontaneous) AHE in ferromagnetic metals, the nonlinear (current-induced) AHE in gyrotropic conductors has an intrinsic contribution associated with the Berry curvature. It is given by $j_a \propto \tau\varepsilon_{adc}D_{bd}E_bE_c$, where $\mathbf{E}$ is the electric field, τ is the relaxation time of the conduction electrons, ε_{abc} is the alternating tensor, and $D_{ab} = \widetilde{D}_{ab}(\omega = 0)$ is the 'Berry-curvature dipole' [67]. After performing an integration by parts in equation (54), the quantities D_{ab} and $\widetilde{D}_{ab}(\omega)$ can be easily evaluated with the help of the `berry.F90` module.

Along with nonlinear magneto-optical and anomalous Hall effects, the flow of electrical current in a gyrotropic conducting medium also generates a net magnetisation. This *kinetic magnetoelectric effect* was originally proposed for bulk chiral conductors [64, 72], and later for two-dimensional (2D) inversion layers with an out-of-plane polar axis [73, 74], where it has been studied intensively [75]. The kinetic magnetoelectric effect in 2D—also known as the *Edelstein effect*—is a purely spin effect, whereas in bulk crystals an orbital contribution is also present [72]. The orbital kinetic magnetoelectric effect is given by $M_a \propto \tau K_{ba}E_b$, where the tensor K_{ab} is obtained from D_{ab} by replacing the Berry curvature with the intrinsic magnetic moment of the Bloch states, [76–78] a quantity that is also provided by the `berry.F90` module [79].

Another phenomenon characteristic of gyrotropic crystals is the *circular photogalvanic effect* (CPGE) that was mentioned briefly in section 5.1. This nonlinear optical effect consists in the generation of a photocurrent that reverses sign with the helicity of light [44–46, 64, 80], and it occurs when light is absorbed via interband or intraband scattering processes. The intraband contribution to the CPGE can be expressed in terms of the Berry curvature dipole D_{ab} as $j_a \propto \frac{\omega\tau^2 D_{ab}}{1+\omega^2\tau^2}\mathrm{Im}\,[\mathbf{E}(\omega) \times \mathbf{E}^*(\omega)]_b$ [67, 81, 82].

The above effects are being very actively investigated in connection with novel materials ranging from topological semimetals [68, 83, 84] to monolayer and bilayer transition-metal dichalcogenides [69–71]. The sensitivity of both the Berry curvature and the intrinsic orbital moment to the details of the electronic structure, together with the need to sample them on a dense mesh of k points, calls for the development of accurate and efficient *ab initio* methodologies, and the Wannier interpolation technique is ideally suited for this task.

The Wannier interpolation methodology for gyrotropic effects was presented in [78], where it was applied to p-doped trigonal tellurium, and the resulting computer code has been incorporated in `postw90` as the `gyrotropic.F90` module. The reader is referred to [78] for more details such as the prefactors in the expressions above, as well as the formulas for natural optical activity, which is also evaluated in the same module.

5.3. postw90.x: spin Hall conductivity

The spin Hall effect (SHE) is a phenomenon in which a spin current is generated by applying an electric field. The current is often transverse to the field (Hall-like), but this is not always the case [85]. The SHE is characterised by the spin Hall conductivity (SHC) tensor $\sigma^{\mathrm{spin},c}_{ab}$ as follows:

$$J^{\mathrm{spin},c}_a(\omega) = \sigma^{\mathrm{spin},c}_{ab}(\omega)E_b(\omega), \tag{56}$$

where $J^{\mathrm{spin},c}_a$ is the spin-current density along direction a with its spin pointing along c, and E_b is the external electric field of frequency ω applied along b. In non-magnetic materials the equal number of up- and down-spin electrons forces the AHE to vanish, resulting in a pure spin current.

Like the AHC, the SHC contains both intrinsic and extrinsic contributions [86]. The intrinsic contribution to the SHC can be calculated from the following Kubo formula, [87]

$$\sigma^{\mathrm{spin},c}_{ab}(\omega) = -\frac{e^2}{\hbar}\frac{1}{VN}\sum_{\mathbf{k}}\sum_n f_{n\mathbf{k}}\Omega^{\mathrm{spin},c}_{n\mathbf{k},ab}(\omega), \tag{57a}$$

$$\Omega^{\mathrm{spin},c}_{n\mathbf{k},ab}(\omega) = \hbar^2\sum_{m\neq n}\frac{-2\mathrm{Im}[\langle\psi_{n\mathbf{k}}|\frac{2}{\hbar}j^{\mathrm{spin},c}_a|\psi_{m\mathbf{k}}\rangle\langle\psi_{m\mathbf{k}}|v_b|\psi_{n\mathbf{k}}\rangle]}{(\epsilon_{n\mathbf{k}} - \epsilon_{m\mathbf{k}})^2 - (\hbar\omega + \mathrm{i}\eta)^2}, \tag{57b}$$

where s_c, v_a and $j^{\mathrm{spin},c}_a = \frac{1}{2}\{s_c, v_a\}$ are the spin, velocity and spin current operators, respectively; V is the cell volume, and N is the total number of k-points used to sample the BZ. Equations (57) are very similar to the Kubo formula for the

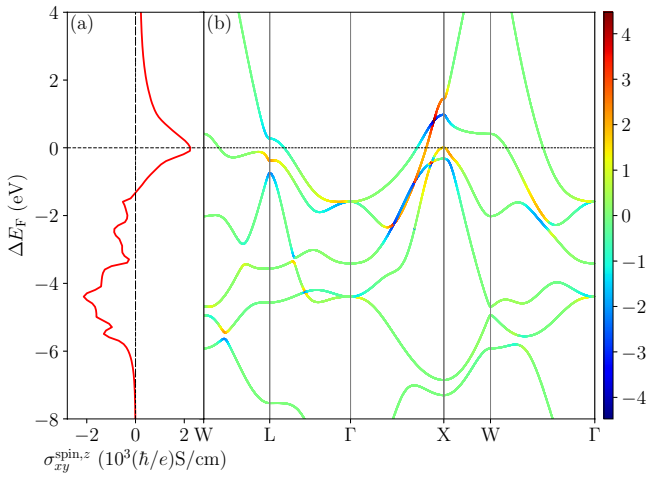


Figure 6. (a) Intrinsic spin Hall conductivity $\sigma^{spin,z}_{xy}$ of fcc Pt, plotted as a function of the shift in Fermi energy relative to its self-consistent value. (b) Band structure of fcc Pt, colour-coded by a dimensionless function $r(\Omega^{spin,z}_{n\mathbf{k},xy})$ of the spin Berry curvature (equation (57*b*)). The function $r(x)$ is equal to $x/10$ when $|x| < 10$, and to $\log_{10}(|x|)\mathrm{sgn}(x)$ when $|x| \geqslant 10$.

AHC, except for the replacement of a velocity matrix element by a spin-current matrix element. As mentioned in the previous two subsections, Wannier-interpolation techniques are very efficient at calculating such quantities.

A Wannier-interpolation method scheme for evaluating the intrinsic SHC was developed in [87] (see also [88] for a related but independent work). The required quantities from the underlying *ab initio* calculation are the spin matrix elements $S^{(0)}_{mn\mathbf{k},a} = \langle\psi^{(0)}_{m\mathbf{k}}|s_a|\psi^{(0)}_{n\mathbf{k}}\rangle$, the Hamiltonian matrix elements $H^{(0)}_{mn\mathbf{k}} = \langle\psi^{(0)}_{m\mathbf{k}}|H|\psi^{(0)}_{n\mathbf{k}}\rangle = \epsilon^{(0)}_{m\mathbf{k}}\delta_{mn}$, and the overlap matrix elements of equation (17). Since the calculation of all these quantities has been previously implemented in `pw2wannier90.x` (the interface code between PWSCF and WANNIER90), this advantageous interpolation scheme can be readily used while keeping to a minimum the interaction between the *ab initio* code and WANNIER90.

The application of the method to fcc Pt is illustrated in figure 6. Panel (a) shows the calculated SHC as a function of the Fermi-level position, and panel (b) depicts the 'spin Berry curvature' of equation (57*b*) that gives the contribution from each band state to the SHC. The aforementioned functionalities have been incorporated in the `berry.F90`, `kpath.F90` and `kslice.F90` modules of `postw90.x`.

5.4. postw90.x: parallelisation improvements

The original implementation of the `berry.F90` module in `postw90.x` (for computing Berry-phase properties such as orbital magnetisation and anomalous Hall conductivity [79]), introduced in WANNIER90 v2.0, was written with code readability in mind and had not been optimised for computational speed. In WANNIER90 v3.0, all parts of the `berry.F90` module have been parallelised while keeping the code readable; moreover, its scalability has been improved, accelerating its performance by several orders of magnitude [89].

Table 1. Wall-time for some of the runs performed with the Berry module, before (WANNIER90 v2.0) and after (WANNIER90 v3.0) the optimisations, for the test system described in the main text. N_c indicates the number of cores used in the calculation.

Mode	*k*-grid	N_c	Time (s)
version 3.0			
Berry 3D	$30 \times 30 \times 30$	24	6903
	$30 \times 30 \times 30$	48	3527
	$30 \times 30 \times 30$	480	441
	$100 \times 100 \times 100$	480	13 041
	$100 \times 100 \times 100$	7680	957
Berry 2D	100×100	24	1389
Berry 1D	10 000	24	12 639
version 2.0			
Berry 3D	$30 \times 30 \times 30$	24	56 497
	$30 \times 30 \times 30$	48	40 279

To illustrate the improvements in performance, we present calculations on a 128-atom supercell of GaAs interstitially doped with Mn. We emphasise that here we are not interested in the results of the calculation but simply on its performance testing, and that the choice of the system does not affect the scaling results that we report. We use a lattice constant of the elementary cell of 5.65 Å. We use norm-conserving relativistic pseudopotentials with the PBE exchange-correlation functional. The energy cut-off for the plane waves is set to 40 Ry, and the Brillouin-zone sampling of the supercell is $3 \times 3 \times 3$. We use a Gaussian metallic smearing with a broadening of 0.015 Ry. For the non-self-consistent step of the calculation, 600 bands are computed and used to construct 517 Wannier functions. The initial projections are chosen as a set of sp^3 orbitals centred on each Ga and As atom, and a set of *d* orbitals on Mn. The calculations were performed on the Prometheus supercomputer of PL-GRID (in Poland). The code was compiled with the Intel ifort compiler (v15.0.2), using the OpenMPI libraries (v1.8.4) and BLAS/LAPACK routines from Intel MKL (v11.3.1).

The Berry-phase calculations can be performed in three distinct ways: (i) 3D quantities in *k*-space (routine `berry_main`), (ii) the same quantities resolved on 2D planes (routine `kslice.F90`), and (iii) 1D paths (routine `kpath.F90`) in the Brillouin zone. In the benchmarks, we will refer to these three cases as 'Berry 3D', 'Berry 2D', and 'Berry 1D', respectively.

The first optimisation target was the function `utility_rotate` in the module `utility.F90`, which calculates a matrix product of the form $B = R^\dagger AR$ using Fortran's built-in `matmul` function. The new routine `utility_rotate_new` uses instead BLAS and performs about 5.7 times better than the original one, giving a total speedup for `berry_main` of about 55%.

A second performance-critical section of code was identified in the routine `get_imfgh_k_list`, which took more than 50% of the total run-time of `berry_main`. This

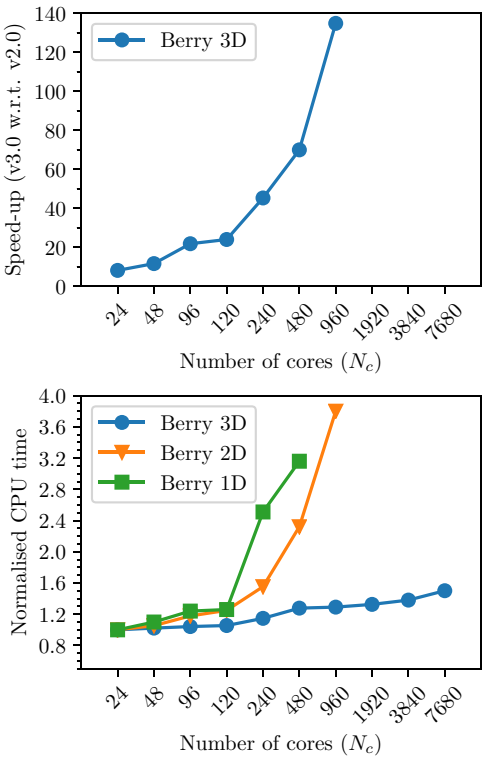


Figure 7. (Top) Speedup of the new WANNIER90 v3.0 with respect to v2.0, for a run of the berry module (mode 'Berry 3D') on the test system described in the text, demonstrating the improvements implemented in the new version of the code. (Bottom) Total CPU time (defined as total walltime times number of CPUs) for the three cases 'Berry 3D', 'Berry 2D' and 'Berry 1D' (whose meaning is described in the main text), normalised with respect to the same case run with $N_{\text{cpu}} = 24$, for the WANNIER90 v3.0 code. The 'Berry 1D' and 'Berry 2D' tests scan a 1D or 2D grid of points in the BZ, respectively; for these tests, the total number of grid points is 10 000, therefore they can scale only up to a few hundreds of cores, above which the communication cost overweights the advantage coming from parallelisation. Instead, we emphasise that calculations with $N_{\text{cpu}} \geqslant 480$ for 'Berry 3D' were run on a denser grid ($100 \times 100 \times 100$ rather than $30 \times 30 \times 30$) and values have been rescaled using the time measured for both grids at $N_{\text{cpu}} = 480$ to show the scalability of the code on thousands of CPUs.

routine computes three quantities: $F_{\alpha\beta}$, $G_{\alpha\beta}$ and $H_{\alpha\beta}$, which are defined in equations (51), (66) and (56) of [79]. By some algebraic transformations, it was possible to reduce 25 calls to `matmul`, carried out in the innermost runtime-critical loop, to only 5 calls. After replacement of `matmul` with the corresponding function from BLAS, the speed up of this routine exceeds a factor of 11, and the total time spent in `berry_main` is 2.5 times shorter (including the speed-up from the first optimisation).

In the third step, a bottleneck was eliminated in the initialisation phase, where `mpi_bcast` was waiting more than two minutes for the master rank to broadcast the parameters. The majority of this time was spent in loops computing matrix products of the form $S = (V_1)^\dagger S_0 V_2$. Again, we replaced this with two calls to the BLAS `gemm` routine. This resulted in a speed-up of a factor of 610 for the calculation of this matrix product in our test case, and the total initialisation time dropped to less than 15 seconds. In total, the `berry_main` routine runs about 5 times faster than it did originally.

Finally, the routines `kslice.F90` and `kpath.F90` were parallelised. The scalability results of `berry_main`, `kslice.F90` and `kpath.F90` are presented in figure 7, and a comparison with the scalability of the previous version of BERRY_MAIN is also given. Absolute times for some of the calculations are reported in table 1.

5.5. GW bands interpolation

While density-functional theory (DFT) is the method of choice for most applications in materials modelling, it is well known that DFT is not meant to provide spectral properties such as band structures, band gaps and optical spectra. Green's function formulation of many-body perturbation theory (MBPT) [90] overcomes this limitation, and allows the excitation spectrum to be obtained from the knowledge of the Green's function. Within MBPT the interacting electronic Green's function $G(\mathbf{r}, \mathbf{r}', \omega)$ may be expressed in terms of the non-interacting Green's function $G^0(\mathbf{r}, \mathbf{r}', \omega)$ and the so-called self-energy $\Sigma(\mathbf{r}, \mathbf{r}', \omega)$, where several accurate approximations for Σ have been developed and implemented into first-principles codes [91]. While maximally-localised Wannier functions for self-consistent GW quasiparticles have been discussed in [92], here we focus on the protocol to perform bands interpolation at the one-shot G_0W_0 level. For solids, the G_0W_0 approximation has proven to be an excellent compromise between accuracy and computational cost and it has become the most popular MBPT technique in computational materials science [93]. In the standard one-shot G_0W_0 approach, Σ is written in terms of the Kohn–Sham (KS) Green's function and the RPA dielectric matrix, both obtained from the knowledge of DFT-KS orbitals and eigenenergies. Quasi-particle (QP) energies are obtained from:

$$\epsilon^{\text{QP}}_{n\mathbf{k}} = \epsilon_{n\mathbf{k}} + Z_{n\mathbf{k}} \langle \psi_{n\mathbf{k}} | \Sigma(\epsilon_{n\mathbf{k}}) - V_{\text{xc}} | \psi_{n\mathbf{k}} \rangle, \qquad (58)$$

where $\psi_{n\mathbf{k}}$ and $\epsilon_{n\mathbf{k}}$ are the KS orbitals and eigenenergies, $Z_{n\mathbf{k}}$ is the so-called renormalisation factor and V_{xc} is the DFT exchange-correlation potential. In addition, in the standard G_0W_0 approximation the QP orbitals are approximated by the KS orbitals. At variance with DFT, QP corrections for a given k-point require knowledge of the KS orbitals and eigenenergies at all points $(\mathbf{k} + \mathbf{q})$ in reciprocal space. In practice, codes such as Yambo [94] compute QP corrections on a regular grid and rely on interpolation schemes to obtain the full band structure along high-symmetry lines. WANNIER90 supports the use of G_0W_0 QP corrections through the general interface `gw2wannier90.py` distributed with WANNIER90, while dedicated tools for QUANTUM ESPRESSO and Yambo allow for an efficient use of symmetries. Thanks to the software interface, QP corrections can be computed in the irreducible BZ (IBZ) and later unfolded to the full BZ to comply with WANNIER90 requirements. In addition, the interface facilitates the use of a denser k-point grid to converge the self-energy and of a coarser grid to obtain MLWFs, as long as the two grids are commensurate. This is particularly efficient in the case of two-dimensional materials, where the k-point convergence of the self-energy is typically very slow while Wannier interpolation is already accurate with much coarser k-point grids. Finally,

the interface takes care of correcting and possibly reordering in energy both the KS eigenvalues and the corresponding input matrices (like $M_{mn}^{(\mathbf{k},\mathbf{b})}$, $A_{mn\mathbf{k}}$). After reading these eigenvalues and matrices, WANNIER90 can proceed as usual and all functionalities are available (band-structure interpolation and beyond) at the level of G_0W_0 calculations.

6. Automatic Wannier functions: the SCDM method

An alternative method for generating localised Wannier functions, known as the selected columns of the density matrix (SCDM) algorithm, has been proposed by Damle, Lin and Ying [95, 96]. At its core the scheme exploits the information stored in the real-space representation of the single-particle density matrix, a gauge-invariant quantity. Localisation of the resulting functions is a direct consequence of the well-known nearsightedness principle [97, 98] of electronic structure in extended systems with a gapped Hamiltonian, i.e. insulators and semiconductors. In these cases, the density matrix is exponentially localised along the off-diagonal direction in its real-space representation $\rho(\mathbf{r},\mathbf{r}')$ and it is generally accepted that Wannier functions with an exponential decay also exist; numerical studies have confirmed this claim for a number of materials, and there exist formal proofs for multiband time-reversal-invariant insulators [99–101]. Since the SCDM method does not minimise a given gauge-dependent localisation measure via a minimisation procedure, it is free from any issue regarding the dependence on initial conditions, i.e. it does not require a good initial guess of localised orbitals. It also avoids other problems associated with a minimisation procedure, such as getting stuck in local minima. More generally, the localised Wannier functions provided by the SCDM method can be used as starting points for the MLWF minimisation procedure, by using them to generate the $A_{\mathbf{k}}$ projection matrices needed by WANNIER90.

For extended insulating systems, the density matrix is given by

$$\rho = \sum_{\mathbf{k}} P_{\mathbf{k}} = \sum_{n=1}^{J} \sum_{\mathbf{k}} |\psi_{n\mathbf{k}}\rangle\langle\psi_{n\mathbf{k}}|. \tag{59}$$

As shown in section 2, the $P_{\mathbf{k}}$ are the spectral projectors associated with the crystal Hamiltonian operator $H_{\mathbf{k}}$ onto the valence space $\mathcal{S}_{\mathbf{k}}$, hence their rank is J. Moreover, they are analytic functions of $\mathbf{k}$ and also manifestly gauge invariant [102, 103]. As mentioned above, the nearsightedness principle [98] guarantees that the columns of the kernels $P_{\mathbf{k}}(\mathbf{r},\mathbf{r}') = \langle\mathbf{r}|P_{\mathbf{k}}|\mathbf{r}'\rangle$ are localised along the off-diagonal direction and therefore they may be used to construct a localised basis. If we consider a discretisation of the J Bloch states at each $\mathbf{k}$ on a real-space grid of N_g points, we can arrange the wavefunctions into the columns of a unitary $N_g \times J$ k-dependent matrix $\Psi_{\mathbf{k}}$

$$\Psi_{\mathbf{k}} = \begin{pmatrix} \psi_{1\mathbf{k}}(\mathbf{r}_1) & \dots & \psi_{J\mathbf{k}}(\mathbf{r}_1) \\ \vdots & \ddots & \vdots \\ \psi_{1\mathbf{k}}(\mathbf{r}_{N_g}) & \dots & \psi_{J\mathbf{k}}(\mathbf{r}_{N_g}) \end{pmatrix}, \tag{60}$$

such that $P_{\mathbf{k},ij} = \left(\Psi_{\mathbf{k}}\Psi_{\mathbf{k}}^{\dagger}\right)_{ij}$ is a $N_g \times N_g$ matrix. In this representation, it is straightforward to see that the columns of $P_{\mathbf{k}}(\mathbf{r}_i,\mathbf{r}_j)$ are projections of extremely localised functions (i.e. Dirac-delta functions localised on the grid points) onto the valence eigenspace. As a result, selecting any linearly-independent subset of J of them will yield a localised basis for the span of $P(\mathbf{r},\mathbf{r}')$. However, randomly selecting J columns does not guarantee that a well-conditioned basis will be obtained. For instance, there could be too much overlap between the selected columns. Conceptually, the most well conditioned columns may be found via a QR factorisation with column pivoting (QRCP) applied to $P(\mathbf{r},\mathbf{r}')$, in the form $P\Pi = QR$, with Π being a matrix permuting the columns of P, Q a unitary matrix and R an upper-triangular matrix (not to be confused with the lattice vector $\mathbf{R}$, or with the matrix $R^{(\mathbf{k},\mathbf{b})}$ defined in equation (12), or with the shift vector of equation (51)), and where Π is chosen so that $|R_{11}| \geqslant |R_{22}| \geqslant \cdots \geqslant |R_{nn}|$. Then the J columns forming a localised basis set are chosen to be the first J of the matrix with permuted columns $P\Pi$.

The SCDM-k [96] method suggests that it is sufficient to apply the QRCP factorisation at $\mathbf{k} = \mathbf{0}$ (Γ point) only, and use the same selection of columns at all k-points. However, this is still often impractical since P_{Γ} is prohibitively expensive to construct and store in memory. Therefore an alternative procedure is proposed, for which the columns can be computed via the QRCP of the (smaller) matrix $\Psi_{\Gamma}^{\dagger}$ instead:

$$\Psi_{\Gamma}^{\dagger}\Pi = Q'R', \tag{61}$$

i.e. the same Π matrix is obtained by computing a QRCP on $\Psi^{\dagger}$ only. Once the set of columns has been obtained, we need to impose the orthonormality constraint on the chosen columns without destroying their locality in real space. This can be achieved by a Löwdin orthogonalisation, similarly to equation (26). In particular, the selection of columns of Ψ_{Γ} can be used to select the columns of all $\Psi_{\mathbf{k}}$, which in turn define the $A_{mn\mathbf{k}}$ matrices needed as input by WANNIER90 to start the MLWF minimisation procedure, by defining $A_{mn\mathbf{k}} = \psi_{m\mathbf{k}}^{*}(\mathbf{r}_{\Pi(n)})$, where the $\Pi(n)$ is the index of the nth column of P after permutation with Π. In fact, we can write the nth column of P after permutation, $P_{\mathbf{k}}(\mathbf{r},\mathbf{r}_{\Pi(n)})$, as

$$P_{\mathbf{k}}(\mathbf{r},\mathbf{r}_{\Pi(n)}) = \sum_{m=1}^{J} \psi_{m\mathbf{k}}(\mathbf{r})\psi_{m\mathbf{k}}^{*}(\mathbf{r}_{\Pi(n)}) \tag{62}$$

$$= \phi_{n\mathbf{k}}(\mathbf{r}) \equiv \sum_{m=1}^{J} \psi_{m\mathbf{k}}(\mathbf{r})A_{mn\mathbf{k}}. \tag{63}$$

The unitary matrix $U_{\mathbf{k}}$ sought for is then constructed via Löwdin orthogonalisation

$$U_{\mathbf{k}} = A_{\mathbf{k}}(A_{\mathbf{k}}^{\dagger}A_{\mathbf{k}})^{-\frac{1}{2}} = A_{\mathbf{k}}S_{\mathbf{k}}^{-\frac{1}{2}}. \tag{64}$$

We can also extend the SCDM-k method to the case where the Bloch states are represented as two-component spinor wavefunctions $\psi_{n\mathbf{k}}^{\alpha}(\mathbf{r})$, e.g. when including spin–orbit interaction

in the Hamiltonian. Here, $\alpha = \uparrow, \downarrow$ is the spinor index. In this case, we include the spin index as well as the position index to perform QRCP. First, we define the $2N_g \times J$ matrix $\Psi_{\mathbf{k}}$

$$\Psi_{\mathbf{k}} = \begin{pmatrix} \psi^{\uparrow}_{1\mathbf{k}}(\mathbf{r}_1) & \dots & \psi^{\uparrow}_{J\mathbf{k}}(\mathbf{r}_1) \\ \psi^{\downarrow}_{1\mathbf{k}}(\mathbf{r}_1) & \dots & \psi^{\downarrow}_{J\mathbf{k}}(\mathbf{r}_1) \\ \vdots & \ddots & \vdots \\ \psi^{\uparrow}_{1\mathbf{k}}(\mathbf{r}_{N_g}) & \dots & \psi^{\uparrow}_{J\mathbf{k}}(\mathbf{r}_{N_g}) \\ \psi^{\downarrow}_{1\mathbf{k}}(\mathbf{r}_{N_g}) & \dots & \psi^{\downarrow}_{J\mathbf{k}}(\mathbf{r}_{N_g}) \end{pmatrix}. \tag{65}$$

Next, as in the spinless case, the QRCP of $\Psi^{\dagger}_{\Gamma}$ is computed, and the first J columns of the Π matrix are selected. Now, $\Pi(n)$, the index of the nth column of P after permutation with Π, determines both the position index $\mathbf{r}_{\Pi(n)}$ and the spin index $\alpha_{\Pi(n)}$. We define $A_{mn\mathbf{k}} = [\psi^{\alpha_{\Pi(n)}}_{m\mathbf{k}}(\mathbf{r}_{\Pi(n)})]^*$ and perform Löwdin orthogonalisation to obtain the unitary matrix $U_{\mathbf{k}}$.

In the case of entangled bands, we need to introduce a so-called quasi-density matrix defined as

$$P_{\mathbf{k}} = \sum_n |\psi_{n\mathbf{k}}\rangle f(\epsilon_{n\mathbf{k}}) \langle \psi_{n\mathbf{k}}|, \tag{66}$$

where $f(\epsilon_{n\mathbf{k}}) \in [0, 1]$ is a generalisation of the Fermi–Dirac probability for the occupied states. Also in this case we only use the information at Γ to generate the permutation matrix. Depending on what kind of entangled manifold one is interested in, $f(\epsilon)$ can be modelled with various functional forms. In particular, the authors of [96] suggest the following three forms:

1. Isolated manifold, e.g. the valence bands of an insulator or a semiconductor: $f(\epsilon)$ is a step function, with the step inside the energy gap $\Delta\epsilon_{\mathrm{g}} = \epsilon_{\mathrm{c}} - \epsilon_{\mathrm{v}}$, where $\epsilon_{\mathrm{c(v)}}$ represents the minimum (maximum) of the conduction (valence) band:
$$f(\epsilon) = \theta(\epsilon_{\mathrm{v}} + \Delta\epsilon_{\mathrm{g}}/2 - \epsilon). \tag{67}$$
Both $\Delta\epsilon_{\mathrm{g}}$ and ϵ_{v} are not free parameters, as they may be obtained directly from the *ab initio* calculation.
2. Entangled manifold (case I), e.g. the valence bands and low-lying conduction bands in a semiconductor: $f(\epsilon)$ is a complementary error function:
$$f(\epsilon) = \frac{1}{2}\mathrm{erfc}\left(\frac{\epsilon - \mu}{\sigma}\right), \tag{68}$$
where μ is used to shift the mid-value of the complementary error function, so that states with energy equal to μ have a weight of $f(\mu) = 1/2$. The parameter σ is used to gauge the 'broadness' of the distribution function.
3. Entangled manifold (case II), e.g. the d bands in a transition metal: $f(\epsilon)$ is a Gaussian function
$$f(\epsilon) = \exp\left(-\frac{(\epsilon - \mu)^2}{\sigma^2}\right). \tag{69}$$

The procedure then follows as in the previous case, by computing a QRCP factorisation on the quasi-density matrix. It is worth to note that in the case of an entangled manifold, the SCDM method requires the selection of two real numbers: μ and σ, as well as the number of Wannier functions to disentangle J. These parameters play a crucial role in the selection of the columns of the density matrix. While the selection of these parameters requires some care, as a rule of thumb (e.g. in entangled case I) σ is of the order 2–5 eV (which is the energy range of a typical bandwidth), while μ can often be set around the Fermi energy (but the exact value depends on various factors, including the number J of bands chosen and the specific properties of the bands of interest). It is worth to mention that since the SCDM-k method is employed as an alternative way of specifying a set of initial projections and hence to compute the $A_{\mathbf{k}}$ matrices in equation (26), the disentanglement procedure can be used in exactly the same way as described in section 2.2. However, in the case of entangled bands the column selection is done on a quasi-density matrix, which implicitly defines a working subspace larger than the target subspace of dimension J. We find that for well-known systems SCDM-k is typically already capable of selecting a smooth manifold and no further subspace selection is needed.

This method is now implemented as part of the `pw2wannier90.x` interface code to QUANTUM ESPRESSO. We have decided to implement the algorithm in the interface code(s) rather than in WANNIER90 itself, because the SCDM method requires knowledge of the wavefunctions $\psi_{n\mathbf{k}}$, which are only available in the *ab initio* code.

In WANNIER90 only a single new input parameter `auto_projections` is required. This disables the check on the number of projections specified in the input file (as we rely on SCDM to provide us with the initial guesses) and adds a new entry to the `<seedname>.nnkp` file (which is read by `pw2wannier90.x` in order to compute the quantities required by WANNIER90) that specifies the number of Wannier functions required. The remaining control parameters for the SCDM method are specified in the input file for the `pw2wannier90.x` code, including whether to use the SCDM method, the functional form of the $f(\epsilon)$ function in equation (66) and, optionally, the values of μ and σ in the definition of $f(\epsilon)$.

7. Automation and workflows: *AiiDA*– WANNIER90 plugin

AiiDA [13] (Automated Interactive Infrastructure and Database for Computational Science) is an informatics infrastructure that helps researchers in managing, automating, storing and sharing their computations and results. AiiDA automatically tracks the entire provenance of every calculation to ensure full reproducibility, which is also stored in a tailored database for efficient querying of previous results. Moreover, it provides a workflow engine, allowing researchers to implement high-level workflows to automate sequences of tedious or complex calculation steps. AiiDA supports simulation codes via a plugin interface, and over 30 different plugins are available to date [104].

Among these, the *AiiDA* – WANNIER90 plugin provides support for the WANNIER90 code. Users interact with the code

(to submit calculations and retrieve the results) via the high-level python interface provided by AiiDA rather than directly creating the WANNIER90 input files. AiiDA will then handle automatically the various steps involved in submitting calculations to a cluster computer, retrieving and storing the results, and parsing them into a database. Furthermore, using the AiiDA workflow system users can chain pre-processing and post-processing calculations automatically (e.g. the preliminary electronic structure calculation with an *ab initio* code). These scientific workflows, moreover, can encode in a reproducible form the scientific knowledge of expert computational researchers in the field on how to run the simulations, choose the numerical parameters and recover from potential errors. In turn, their availability reduces the training time of new researchers, eliminates sources of error and enables large-scale high-throughput simulations.

The *AiiDA* – WANNIER90 plugin expects that each calculation takes a few well-defined input parameters. Among the most important ones, a WANNIER90 calculation run via AiiDA requires that the following input nodes are provided: an input crystal `structure`, a node of `parameters` with a dictionary of input flags for WANNIER90, a node with the list of `kpoints`, a node representing the atomic `projections`, and a `local_input_folder` or `remote_input_folder` node containing the necessary input files (`.amn`, `.mmn`, `.nnkp`, `.eig`, `.dmn`) for the WANNIER90 calculation as generated by an *ab initio* code.

All of these parameters, with the exception of projections, are generic to AiiDA to facilitate their reuse with different simulation codes. More detailed information on all inputs can be found in the *AiiDA* – WANNIER90 package documentation [105].

After the WANNIER90 execution is completed, the *AiiDA* – WANNIER90 plugin provides parsers that are able to detect whether the convergence was successful and retrieve key parameters including the centres of the Wannier functions and their spread, as well as the different components of the spread (Ω_{I}, Ω_{D}, Ω_{OD} and Ω), and (if computed) the maximum imaginary/real ratio of the Wannier functions and the interpolated band structure.

The whole simulation is stored in the form of a graph, representing explicitly the provenance of the data generated including all inputs and outputs of the codes used in the workflow. An example of a provenance graph, automatically generated by AiiDA when running a QUANTUM ESPRESSO calculation followed by a WANNIER90 calculation, is shown in figure 8.

To demonstrate the usefulness of this approach, we refer to [106] that reports the implementation and verification results of a fully-automated workflow (implemented within AiiDA, using the *AiiDA* – WANNIER90 plugin described in this section) to compute Wannier functions of any material without any user input (besides its crystal structure). In addition, a virtual machine containing the codes (AiiDA with its plugins, QUANTUM ESPRESSO and WANNIER90 including the SCDM implementation described in section 6, and the automation workflows) is distributed. This virtual machine allows any researcher to reproduce the results of the paper and, even more, to perform simulations on new materials using the same protocol, without the need of installing and configuring all codes.

We emphasise that the availability of a platform to run WANNIER90 in a fully-automated high-throughput way via the *AiiDA* – WANNIER90 plugin has already proved to be beneficial for the WANNIER90 code itself. Indeed, it has pushed the development of additional features or improvements now part of WANNIER90 v3.0, including additional output files to facilitate output parsing and improvements in some of the algorithms and their default parameters to increase robustness.

8. Modern software engineering practices

In this section, we describe a number of modern software engineering practices that are now part of the development cycle of the WANNIER90 code. In particular, WANNIER90 includes a number of tests that are run at every commit via a continuous integration approach, as well as nightly in a dedicated test farm. Version control is handled using git and the code is hosted on the GitHub platform [107]. We follow the fork and pull-request model, in which users can duplicate (fork) the project into their own private repository, make their own changes, and make a pull request (i.e. request that their changes be incorporated back into the main repository). When a pull request is made, a series of tests are automatically performed: the test suite is run both in serial and parallel using the Travis continuous integration platform [108], and code coverage is checked using CODECOV [109]. If these tests are successful then the changes are reviewed by members of the WANNIER90 developers group and, if the code meets the published coding guidelines, it can be merged into the development branch.

In addition, while interaction with end users happens via a mailing-list forum, discussion among developers is now tracked using GitHub issues. This facilitates the maintenance of independent conversation threads for each different code issue, new feature proposal or bug. These can easily reference code lines as well as be referenced in code commit messages. Moreover, for every new bug report a new issue is opened, and pull requests that close the issue clearly refer to it. This approach facilitates tracking back the reasoning behind the changes in case a similar problem resurfaces.

In the remainder of this section we describe more in detail some of these modern software engineering practices.

8.1. Code documentation (FORD)

The initial release of WANNIER90 came with extensive documentation in the form of a *User Guide* describing the methodology, input flags to the program and format of the input and output files. This document was aimed at the end users running the software. Documentation of the code itself was done via standard code comments. In order to foster not only a community of users but also of code *contributors* to

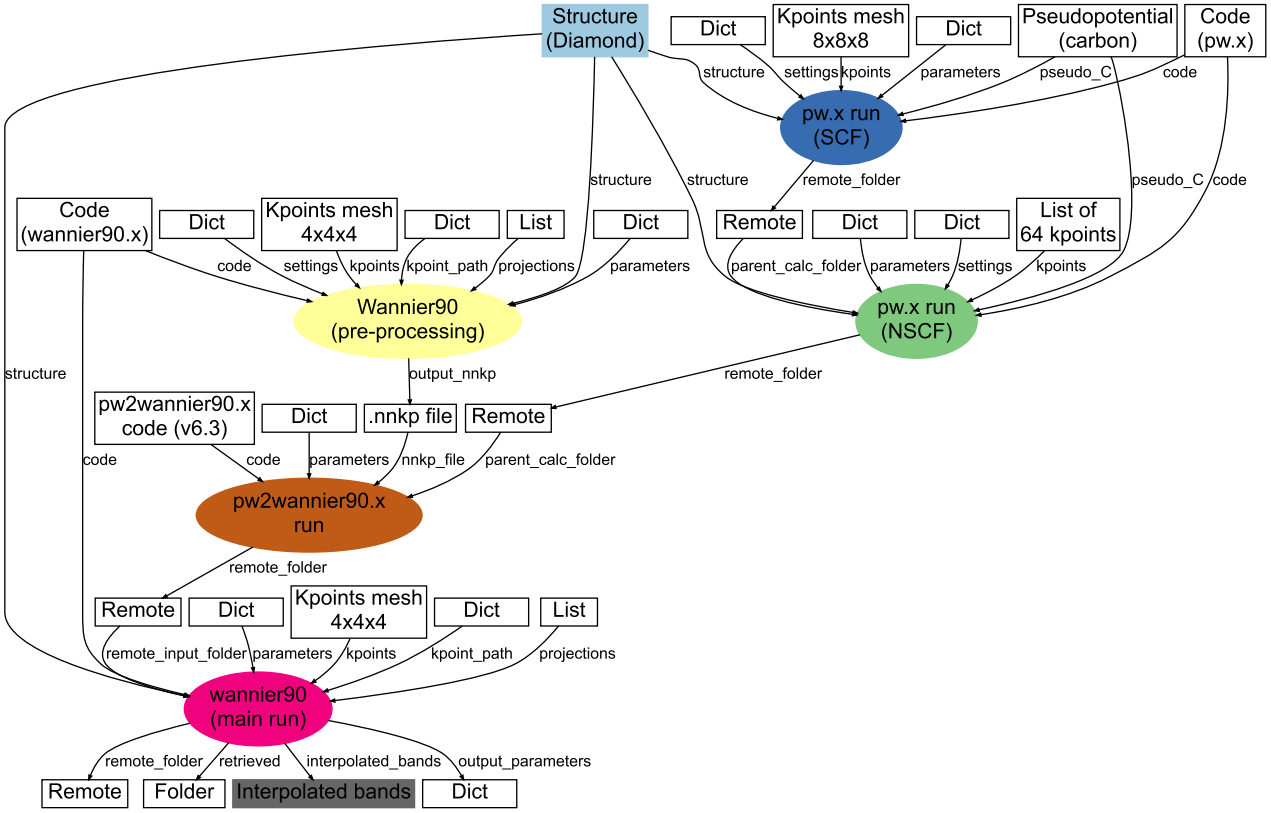


Figure 8. The provenance graph automatically generated by AiiDA when running a WANNIER90 calculation for a diamond crystal using QUANTUM ESPRESSO as the DFT code. Rectangles represent executions of calculations, ellipses represent data nodes, and diamonds are code executables. Graph edges connect calculations to their inputs and outputs. In particular, the following calculations are visible: Quantum ESPRESSO pw.x SCF (dark blue) and NSCF (green), QUANTUM ESPRESSO `pw2wannier90.x` (brown), and WANNIER90 pre-processing (yellow) and minimisation run (purple). The initial diamond structure (light blue) and the final interpolated band structure (dark grey) are also highlighted.

WANNIER90, we have now created an additional documentation of the internal structure of the code. This makes the code more approachable, particularly for new contributors. To create this code documentation in a fully automated fashion, we use the FORD (FORtran Documenter) [110] documentation generator. We have chosen this over other existing documentation solutions because of FORD's specific support for Fortran. This tool parses the Fortran source, and generates a hyperlinked (HTML) index of source files, modules, procedures, types and programs defined in the code. Furthermore, it constructs graphs showing the dependencies between different modules and subroutines. Additional information can be provided in the form of special in-code comments (marked with double exclamation marks) describing in more detail variables, modules or subroutines. By tightly coupling the code to its documentation using in-code comments, the documentation maintenance efforts are greatly reduced, decreasing the risk of having outdated documentation. The compiled version of the documentation for the most recent code version is made available on the Wannier90 website [111].

8.2. Testing infrastructure and continuous integration

With the recent opening to the community of the WANNIER90 development, it has become crucial to create a non-regression test suite to ensure that new developments do not break existing functionalities of the code. Its availability facilitates the maintenance of the code and ensures its long-term stability.

The WANNIER90 test suite relies on a modified version of James Spencer's python `testcode.py` [112]. This provides the functionality to run tests and compare selected quantities parsed from the output files against benchmarked values.

At present, the WANNIER90 test suite includes over 50 tests which are run both in serial and parallel and cover over 60% of the source code (with many modules exceeding 80% coverage). The code coverage is measured with the CODECOV software [109]. Developers are now required to add tests when adding new features to the code to ensure that their additions work as expected. This also ensures that future changes to the code will never break that functionality. Two different test approaches are implemented, serving different purposes.

First, the WANNIER90 repository is now linked with the Travis continuous integration platform [108] to prevent introducing errors and bugs into the main code branch. Upon any commit to the GitHub repository, the test suite is run both in serial and in parallel. Any test failure is reported back to the GitHub webpage. Additionally, for tests run against pull requests, any failed test results in the pull request being blocked and not permitted to merge. Contributors will first need to change their code to fix the problems highlighted in the tests; pull requests are able to be merged only after all tests pass successfully.

Second, nightly automatic tests are run on a BUILDBOT test farm. The test farm compiles and runs the code with a combination of compilers and libraries (current compilers include GFortran v6.4.0 and v7.3.0, Intel Fortran Compiler v17 and v18, and PGI compiler v18.05; current MPI libraries include Open MPI v1.10.7 and v3.1.3, Intel MPI v17 and MVAPICH v2.3b). This ensures that the code runs correctly on various high-performance computer (HPC) architectures. More information on the test farm can be found on the WANNIER90 GitHub wiki website [113].

In addition to these tests, we have implemented git pre-commit hooks to help keep the same code style in all source files. The current pre-commit hooks run Patrick Seewald's Fortran source code formatter FPRETTIFY [114] to remove trailing whitespaces at the end of a line and to enforce a consistent indentation style. These precommit hooks, besides validating the code, can reformat it automatically. Developers may simply run the formatter code to convert the source to a valid format. If a developer installs the pre-commit hooks, these will be run automatically before every commit. Even if this is not the case, these tests are also run on Travis; therefore, a pull request that does not conform to the standard code style cannot be merged before the style is fixed.

8.3. Command-line interface and dry-run

The command-line interface of the code has been improved. Just running `wannier90.x` without parameters shows a short explanation of the available command line options. In addition, a `-v` flag has been added to print the version of the code, as well as a new `-d` dry-run mode, that just parses the input file to perform all needed checks of the inputs without running the actual calculation. The latter functionality is particularly useful to be used in input validators for WANNIER90 or to pre-calculate quantities computed by the code at the beginning of the simulation (such as nearest-neighbour shells, *b*-vectors or expected memory usage) and use this information to validate the run or optimise it (e.g. to decide the parallelisation strategy within automated AiiDA workflows).

8.4. Library mode

WANNIER90 also comes with a library mode, where the core code functionality can be compiled into a library that can then be linked by external programs. This library mode is used as the default interaction protocol by some interface codes. The library mode provides only support for a subset of the full functionality, in particular at the moment it only supports serial execution. We have now added and improved support for the use of excluded bands also within the library mode. Moreover, beside supporting the generation of a statically-linked library, we now also support the generation of dynamically-linked versions. Finally, we have added a minimal test code, run together with all other tests in the test suite, that serves both to verify that the library functionality works as expected, and as an example of the interface of the library mode.

9. Conclusions and outlook

WANNIER90 v2.0 was released in October 2013 with a small update for v2.1 in January 2017. The results and developments of the past years, presented in this work, were released in WANNIER90 v3.0 in February 2019. Thanks to the transition of WANNIER90 to a community code, WANNIER90 includes now a large number of new functionalities and improvements that make it very robust, efficient and rich with features. These include the implementation of new methods for the calculation of WFs and for the generation of the initial projections; parallelisation and optimisations; interfaces with new codes, methods and infrastructures; new user functionality; improved documentation; and various bug fixes. The effect of enlarging the community of developers is not only visible in the large number of contributions to the code, but also in the modern software engineering practices that we have put in place, that help improve the robustness and reliability of the code and facilitate its maintenance by the core WANNIER90 developers group and its long-term sustainability.

The next major improvement that we are planning is the implementation of a more robust and general library mode. The features that we envision are: (1) the possibility to call the code from C or Fortran codes without the need to store files but by passing all variables from memory; (2) a more general library interface that is easily extensible in the future when new functionality is added; and (3) the possibility to run WANNIER90 from a parallel MPI code, both by running each instance in parallel and by allowing massively-parallel codes to call, in parallel, various instances of WANNIER90 on various structures or with different parameters. This improvement will demand a significant restructuring of most of the codebase and requires a good design of the new interface. Currently we are drafting the new library interface, by collecting feedback and use cases from the various contributors and users of the code, to ensure that the new library mode can be beneficial to all different possible use cases.

Acknowledgments

We acknowledge code contributions by Daniel Aberg (W90POV code), Lampros Andrinopoulos (W90VDW code), Pablo Aguado Puente (`gyrotropic.F90` module), Raffaello Bianco (*k*-slice plotting), Marco Buongiorno Nardelli (`dosqc` v1.0 subroutines upon which some of `transport.F90` is based), Stefano de Gironcoli (`pw2wannier90.x` interface to QUANTUM ESPRESSO), Pablo Garcia Fernandez (matrix elements of the position operator), Nicholas Hine (W90VDW code), Young-Su Lee (specialised Γ-point routines and transport), Antoine Levitt (preconditioning), Graham Lopez (extension of `pw2wannier90.x` to add terms needed for orbital magnetisation), Radu Miron (constrained centres), Nicolas Poilvert (transport routines), Michel Posternak (original plotting routines), Rei Sakuma (symmetry-adapted Wannier functions), Gabriele Sclauzero (disentanglement in spheres in *k*-space), Matthew Shelley (transport routines), Christian Stieger (routine to print the *U* matrices), David Strubbe (various bug fixes and improvements), Timo Thonhauser

(extension of `pw2wannier90.x` to add terms needed for orbital magnetisation), as well as the participants of the first WANNIER90 developers meeting in San Sebastián (Spain) in 2016 for useful discussions (Daniel Fritsch, Victor Garcia Suarez, Pablo Garcia Fernandez, Jan-Philipp Hanke, Ji Hoon Ryoo, Jürg Hutter, Javier Junquera, Liang Liang, Michael Obermeyer, Gianluca Prandini, Christian Stieger, Paolo Umari). The WDG acknowledges financial support from the NCCR MARVEL of the Swiss National Science Foundation, the European Union's Centre of Excellence E-CAM (Grant No. 676531), and the Thomas Young Centre for Theory and Simulation of Materials (Grant No. TYC-101).

ORCID iDs

Giovanni Pizzi https://orcid.org/0000-0002-3583-4377
Valerio Vitale https://orcid.org/0000-0003-2714-8585
Ryotaro Arita https://orcid.org/0000-0001-5725-072X
Stefan Blügel https://orcid.org/0000-0001-9987-4733
Frank Freimuth https://orcid.org/0000-0001-6193-5991
Guillaume Géranton https://orcid.org/0000-0003-2474-8411
Marco Gibertini https://orcid.org/0000-0003-3980-5319
Dominik Gresch https://orcid.org/0000-0002-0150-0424
Charles Johnson https://orcid.org/0000-0002-8384-7623
Takashi Koretsune https://orcid.org/0000-0002-5334-0792
Julen Ibañez-Azpiroz https://orcid.org/0000-0003-0922-0926
Hyungjun Lee https://orcid.org/0000-0002-5629-682X
Jae-Mo Lihm https://orcid.org/0000-0003-0900-0405
Daniel Marchand https://orcid.org/0000-0002-0944-5175
Antimo Marrazzo https://orcid.org/0000-0003-2053-9962
Yuriy Mokrousov https://orcid.org/0000-0003-1072-2421
Jamal I Mustafa https://orcid.org/0000-0002-0950-0728
Yoshiro Nohara https://orcid.org/0000-0003-4790-2784
Yusuke Nomura https://orcid.org/0000-0003-4956-562X
Lorenzo Paulatto https://orcid.org/0000-0001-6343-0006
Samuel Poncé https://orcid.org/0000-0003-1159-8389
Thomas Ponweiser https://orcid.org/0000-0001-7934-783X
Junfeng Qiao https://orcid.org/0000-0003-3728-0109
Florian Thöle https://orcid.org/0000-0001-6006-9876
Stepan S Tsirkin https://orcid.org/0000-0001-5064-0485
Małgorzata Wierzbowska https://orcid.org/0000-0002-4826-7076
Nicola Marzari https://orcid.org/0000-0002-9764-0199
David Vanderbilt https://orcid.org/0000-0002-2465-9091
Ivo Souza https://orcid.org/0000-0001-9901-5058
Arash A Mostofi https://orcid.org/0000-0002-6883-8278
Jonathan R Yates https://orcid.org/0000-0002-1896-0101

References

[1] Giannozzi P *et al* 2017 Advanced capabilities for materials modelling with Quantum ESPRESSO *J. Phys.: Condens. Matter* **29** 465901
[2] Gonze X *et al* 2009 ABINIT: first-principles approach to material and nanosystem properties *Comput. Phys. Commun.* **180** 2582
[3] Kresse G and Hafner J 1993 *Ab initio* molecular dynamics for liquid metals *Phys. Rev.* B **47** 558
[4] Kresse G and Furthmüller J 1996 Efficiency of *ab initio* total energy calculations for metals and semiconductors using a plane-wave basis set *Comput. Mater. Sci.* **6** 15
[5] Kresse G and Furthmüller J 1996 Efficient iterative schemes for *ab initio* total-energy calculations using a plane-wave basis set *Phys. Rev.* B **54** 11169
[6] Soler J M, Artacho E, Gale J D, García A, Junquera J, Ordejón P and Sánchez-Portal D 2002 The SIESTA method for *ab initio* order-N materials simulation *J. Phys.: Condens. Matt.* **14** 2745
[7] Blaha P, Schwarz K, Madsen G, Kvasnicka D and Luitz J 2001 *WIEN2k: an Augmented Plane Wave plus Local Orbitals Program for Calculating Crystal Properties* (Wien: Technical University of Vienna) www.wien2k.at/reg_user/textbooks/usersguide.pdf
[8] Blügel S and Bihlmayer G 2006 Full-potential linearized augmented planewave method *Computational Nanoscience: Do It Yourself!* vol 31, ed J Grotendorst *et al* (Jülich: John von Neumann Institute for Computing) pp 85–129
[9] Andrade X *et al* 2015 Real-space grids and the Octopus code as tools for the development of new simulation approaches for electronic systems *Phys. Chem. Chem. Phys.* **17** 31371
[10] The Elk code 2019 http://elk.sourceforge.net
[11] WANNIER90 official GitHub repository 2019 https://github.com/wannier-developers/wannier90
[12] Marzari N, Mostofi A A, Yates J R, Souza I and Vanderbilt D 2012 Maximally localized Wannier functions: Theory and applications *Rev. Mod. Phys.* **84** 1419
[13] Pizzi G, Cepellotti A, Sabatini R, Marzari N and Kozinsky B 2016 AiiDA: automated interactive infrastructure and database for computational science *Comput. Mater. Sci.* **111** 218
[14] Wannier G H 1937 The structure of electronic excitation levels in insulating crystals *Phys. Rev.* **52** 191
[15] Marzari N and Vanderbilt D 1997 Maximally localized generalized Wannier functions for composite energy bands *Phys. Rev.* B **56** 12847
[16] Mostofi A A, Yates J R, Lee Y-S, Souza I, Vanderbilt D and Marzari N 2008 Wannier90: a tool for obtaining maximally-localised Wannier functions *Comput. Phys. Commun.* **178** 685
[17] Souza I, Marzari N and Vanderbilt D 2001 Maximally localized Wannier functions for entangled energy bands *Phys. Rev.* B **65** 035109
[18] Löwdin P 1950 On the non-orthogonality problem connected with the use of atomic wave functions in the theory of molecules and crystals *J. Chem. Phys.* **18** 365
[19] Evarestov R and Smirnov V 2012 *Site Symmetry in Crystals: Theory and Applications* (*Springer Series in Solid-State Sciences*) (Berlin: Springer)
[20] Hahn T 2006 *International Tables for Crystallography, Vol. A. Space-group Symmetry* (New York: Wiley)
[21] Sakuma R 2013 Symmetry-adapted Wannier functions in the maximal localization procedure *Phys. Rev.* B **87** 235109
[22] Thygesen K S, Hansen L B and Jacobsen K W 2005 Partly occupied Wannier functions: Construction and applications *Phys. Rev.* B **72** 125119
[23] Wang R, Lazar E A, Park H, Millis A J and Marianetti C A 2014 Selectively localized Wannier functions *Phys. Rev.* B **90** 165125
[24] Blöchl P E 1994 Projector augmented-wave method *Phys. Rev.* B **50** 17953
[25] Kresse G and Joubert D 1999 From ultrasoft pseudopotentials to the projector augmented-wave method *Phys. Rev.* B **59** 1758
[26] Corso A D 2014 Pseudopotentials periodic table: from H to Pu *Comput. Mater. Sci.* **95** 337

[27] Perdew J P, Burke K and Ernzerhof M 1996 Generalized gradient approximation made simple *Phys. Rev. Lett.* **77** 3865

[28] Momma K and Izumi F 2011 VEST A 3 for three-dimensional visualization of crystal, volumetric and morphology data *J. Appl. Crystallogr.* **44** 1272

[29] Géranton G, Freimuth F, Blügel S and Mokrousov Y 2015 Spin–orbit torques in $L1_0-$FePt/Pt thin films driven by electrical and thermal currents *Phys. Rev.* B **91** 014417

[30] Ferretti A, Calzolari A, Bonferroni B and Di Felice R 2007 Maximally localized Wannier functions constructed from projector-augmented waves or ultrasoft pseudopotentials *J. Phys.: Condens. Matt.* **19** 036215

[31] Dal Corso A and Conte A M 2005 Spin–orbit coupling with ultrasoft pseudopotentials: application to Au and Pt *Phys. Rev.* B **71** 115106

[32] Lee Y-S, Nardelli M B and Marzari N 2005 Band structure and quantum conductance of nanostructures from maximally localized Wannier functions: the case of functionalized carbon nanotubes *Phys. Rev. Lett.* **95** 076804

[33] Wang X, Yates J R, Souza I and Vanderbilt D 2006 *Ab initio* calculation of the anomalous Hall conductivity by Wannier interpolation *Phys. Rev.* B **74** 195118

[34] Yates J R, Wang X, Vanderbilt D and Souza I 2007 Spectral and Fermi surface properties from Wannier interpolation *Phys. Rev.* B **75** 195121

[35] Persistence of Vision Pty. Ltd. 2004 Persistence of Vision (TM) Raytracer www.povray.org/

[36] Coh S and Vanderbilt D Python tight binding (PythTB) code www.physics.rutgers.edu/pythtb/index.html

[37] Gresch D, Wu Q, Winkler G W, Häuselmann R, Troyer M and Soluyanov A A 2018 Automated construction of symmetrized Wannier-like tight-binding models from *ab initio* calculations *Phys. Rev. Mat.* **2** 103805

[38] Papior N 2018 sisl: v0.9.5 https://doi.org/10.5281/zenodo.597181

[39] Ferrer J *et al* 2014 GOLLUM: a next-generation simulation tool for electron, thermal and spin transport *New J. Phys.* **16** 093029

[40] Luisier M, Schenk A, Fichtner W and Klimeck G 2006 Atomistic simulation of nanowires in the $sp^3d^5s^*$ tight-binding formalism: From boundary conditions to strain calculations *Phys. Rev.* B **74** 205323

[41] NanoTCAD ViDES 2019 http://vides.nanotcad.com

[42] Poncé S, Margine E, Verdi C and Giustino F 2016 EPW: electron–phonon coupling, transport and superconducting properties using maximally localized Wannier functions *Comput. Phys. Commun.* **209** 116

[43] Wu Q, Zhang S, Song H-F, Troyer M and Soluyanov A A 2018 WannierTools: an open-source software package for novel topological materials *Comput. Phys. Commun.* **224** 405

[44] Belinicher V I and Sturman B I 1980 The photogalvanic effect in media lacking a center of symmetry *Sov. Phys.—Usp.* **23** 199

[45] Sturman B I and Fridkin V M 1992 *The Photovoltaic and Photorefractive Effects in Noncentrosymmetric Materials* (Philadelphia, PA: Gordon and Breach)

[46] Ivchenko E L and Pikus G E 1997 *Superlattices and Other Heterostructures* (Berlin: Springer)

[47] Tan L Z, Zheng F, Young S M, Wang F, Liu S and Rappe A M 2016 Shift current bulk photovoltaic effect in polar materials—hybrid and oxide perovskites and beyond *npj Comput. Mater.* **2** 16026

[48] Nagaosa N, Sinova J, Onoda S, MacDonald A H and Ong N P 2010 Anomalous Hall effect *Rev. Mod. Phys.* **82** 1539

[49] von Baltz R and Kraut W 1981 Theory of the bulk photovoltaic effect in pure crystals *Phys. Rev.* B **23** 5590

[50] Belinicher V I, Ivchenko E L and Sturman B I 1982 Kinetic theory of the displacement photovoltaic effect in piezoelectrics *Sov. Phys.—JETP* **56** 359

[51] Kristoffel N, von Baltz R and Hornung D 1982 On the intrinsic bulk photovoltaic effect: Performing the sum over intermediate states *Z. Phys.* B **47** 293

[52] Sipe J E and Shkrebtii A I 2000 Second-order optical response in semiconductors *Phys. Rev.* B **61** 5337

[53] Fregoso B M, Morimoto T and Moore J E 2017 Quantitative relationship between polarization differences and the zone-averaged shift photocurrent *Phys. Rev.* B **96** 075421

[54] Presting H and Von Baltz R 1982 Bulk photovoltaic effect in a ferroelectric crystal: a model calculation *Phys. Status Solidi* b **112** 559

[55] Young S M and Rappe A M 2012 First principles calculation of the shift current photovoltaic effect in ferroelectrics *Phys. Rev. Lett.* **109** 116601

[56] King-Smith R D and Vanderbilt D 1993 Theory of polarization of crystalline solids *Phys. Rev.* B **47** 1651

[57] Aversa C and Sipe J E 1995 Nonlinear optical susceptibilities of semiconductors: results with a length-gauge analysis *Phys. Rev.* B **52** 14636

[58] Nastos F and Sipe J E 2006 Optical rectification and shift currents in GaAs and GaP response: Below and above the band gap *Phys. Rev.* B **74** 035201

[59] Rangel T, Fregoso B M, Mendoza B S, Morimoto T, Moore J E and Neaton J B 2017 Large bulk photovoltaic effect and spontaneous polarization of single-layer monochalcogenides *Phys. Rev. Lett.* **119** 067402

[60] Graf M and Vogl P 1995 Electromagnetic fields and dielectric response in empirical tight-binding theory *Phys. Rev.* B **51** 4940

[61] Wang C, Liu X, Kang L, Gu B-L, Xu Y and Duan W 2017 First-principles calculation of nonlinear optical responses by Wannier interpolation *Phys. Rev.* B **96** 115147

[62] Ibañez-Azpiroz J, Tsirkin S S and Souza I 2018 *Ab initio* calculation of the shift photocurrent by Wannier interpolation *Phys. Rev.* B **97** 245143

[63] Baranova N B, Boddanov Y V and Zel'dovich B Y 1977 Electrical analog of the Faraday effect and other new optical effects in liquids *Opt. Commun.* **22** 243

[64] Ivchenko E L and Pikus G E 1978 New photogalvanic effect in gyrotropic crystals *JETP Lett.* **27** 604

[65] Vorob'ev E L, Ivchenko E L, Pikus G E, Farbshtein I I, Shalygin V A and Shturbin A V 1979 Optical activity in tellurium induced by a current *JETP Lett.* **29** 441

[66] Shalygin V A, Sofronov A N, Vorob'ev E L and Farbshtein I I 2012 Current-induced spin polarization of holes in tellurium *Phys. Solid State* **54** 2362

[67] Sodemann I and Fu L 2015 Quantum nonlinear Hall effect induced by Berry curvature dipole in time-reversal invariant materials *Phys. Rev. Lett.* **115** 216806

[68] Zhang Y, Sun Y and Yan B 2018 Berry curvature dipole in Weyl semimetal materials: an *ab initio* study *Phys. Rev.* B **97** 041101

[69] Zhang Y, van den Brink J, Felser C and Yan B 2018 Electrically tuneable nonlinear anomalous Hall effect in two-dimensional transition-metal dichalcogenides WTe_2 and $MoTe_2$ *2D Mater.* **5** 044001

[70] You J-S, Fang S, Xu S-Y, Kaxiras E and Low T 2018 Berry curvature dipole current in the transition metal dichalcogenides family *Phys. Rev.* B **98** 121109

[71] Ma Q *et al* 2019 Observation of the nonlinear Hall effect under time-reversal-symmetric conditions *Nature* **565** 337

[72] Levitov L S, Nazarov Y V and Eliashberg G M 1985 Magnetoelectric effects in conductors with mirror isomer symmetry *Sov. Phys.—JETP* **61** 133

[73] Edelstein V M 1990 Spin polarization of conduction electrons induced by electric current in two-dimensional asymmetric electron systems *Solid State Commun.* **73** 233

[74] Aronov A G and Lyanda-Geller Y 1989 Nuclear electric resonance and orientation of carrier spins by an electric field *JETP Lett.* **50** 431

[75] Ganichev S D, Trushin M and Schliemann J 2012 Spin polarization by current *Handbook of Spin Transport and Magnetism* ed E Y Tsymbal and I Zutic (New York: CRC Press) p 487

[76] Yoda T, Yokoyama T and Murakami S 2015 Current-induced orbital and spin magnetizations in crystals with helical structure *Sci. Rep.* **5** 12024

[77] Zhong S, Moore J E and Souza I 2016 Gyrotropic magnetic effect and the magnetic moment on the Fermi surface *Phys. Rev. Lett.* **116** 077201

[78] Tsirkin S S, Puente P A and Souza I 2018 Gyrotropic effects in trigonal tellurium studied from first principles *Phys. Rev.* B **97** 035158

[79] Lopez M G, Vanderbilt D, Thonhauser T and Souza I 2012 Wannier-based calculation of the orbital magnetization in crystals *Phys. Rev.* B **85** 014435

[80] Asnin V M, Bakun A A, Danishevskii A M, Ivchenko E L, Pikus G E and Rogachev A A 1978 Observation of a photo-emf that depends on the sign of the circular polarization of the light *JETP Lett.* **28** 74

[81] Deyo E, Golub L E, Ivchenko E L and Spivak B 2009 Semiclassical theory of the photogalvanic effect in non-centrosymmetric systems (arXiv:0904.1917)

[82] Moore J E and Orenstein J 2010 Confinement-induced Berry phase and helicity-dependent photocurrents *Phys. Rev. Lett.* **105** 026805

[83] de Juan F, Grushin A G, Morimoto T and Moore J E 2017 Quantized circular photogalvanic effect in Weyl semimetals *Nat. Commun.* **8** 15995

[84] Flicker F, de Juan F, Bradlyn B, Morimoto T, Vergniory M G and Grushin A G 2018 Chiral optical response of multifold fermions *Phys. Rev.* B **98** 155145

[85] Wimmer S, Seemann M, Chadova K, Ködderitzsch D and Ebert H 2015 Spin–orbit-induced longitudinal spin-polarized currents in nonmagnetic solids *Phys. Rev.* B **92** 041101

[86] Sinova J, Valenzuela S O, Wunderlich J, Back C H and Jungwirth T 2015 Spin Hall effects *Rev. Mod. Phys.* **87** 1213

[87] Qiao J, Zhou J, Yuan Z and Zhao W 2018 Calculation of intrinsic spin Hall conductivity by Wannier interpolation *Phys. Rev.* B **98** 214402

[88] Ryoo J H, Park C-H and Souza I 2019 Computation of intrinsic spin Hall conductivities from first principles using maximally localized Wannier functions *Phys. Rev.* B **99** 235113

[89] Ponweiser T and Wierzbowska M 2014 Optimizing the post-Wannier Berry-phase code for optical and anomalous Hall conductivities and orbital magnetization www.prace-ri.eu/IMG/pdf/WP204.pdf

[90] Fetter A L and Walecka J D 1971 *Quantum Theory of Many-Particle Systems* (Boston McGraw-Hill)

[91] Martin R M, Reining L and Ceperley D M 2016 *Interacting Electrons: Theory and Computational Approaches* (Cambridge: Cambridge University Press)

[92] Hamann D R and Vanderbilt D 2009 Maximally localized Wannier functions for GW quasiparticles *Phys. Rev.* B **79** 045109

[93] Reining L 2018 The GW approximation: content, successes and limitations *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **8** e1344

[94] Marini A, Hogan C, Grüning M and Varsano D 2009 Yambo: an *ab initio* tool for excited state calculations *Comput. Phys. Commun.* **180** 1392

[95] Damle A, Lin L and Ying L 2015 Compressed representation of Kohn–Sham orbitals via selected columns of the density matrix *J. Chem. Theor. Comput.* **11** 1463

[96] Damle A and Lin L 2018 Disentanglement via entanglement: a unified method for Wannier localization *Multiscale Model. Sim.* **16** 1392

[97] Kohn W 1973 Construction of Wannier functions and applications to energy bands *Phys. Rev.* B **7** 4388

[98] Prodan E and Kohn W 2005 Nearsightedness of electronic matter *Proc. Natl Acad. Sci.* **102** 11635

[99] Brouder C, Panati G, Calandra M, Mourougane C and Marzari N 2007 Exponential localization of Wannier functions in insulators *Phys. Rev. Lett.* **98** 046402

[100] He L and Vanderbilt D 2001 Exponential decay properties of Wannier functions and related quantities *Phys. Rev. Lett.* **86** 5341

[101] Fiorenza D, Monaco D and Panati G 2016 Construction of real-valued localized composite Wannier functions for insulators *Ann. Henri Poincaré* **17** 63

[102] Nenciu G 1991 Dynamics of band electrons in electric and magnetic fields: rigorous justification of the effective Hamiltonians *Rev. Mod. Phys.* **63** 91

[103] Panati G and Pisante A 2013 Bloch bundles, Marzari–Vanderbilt functional and maximally localized Wannier functions *Commun. Math. Phys.* **322** 835

[104] AiiDA plugin registry https://aiidateam.github.io/aiida-registry

[105] Documentation of the AiiDA-Wannier90 plugin https://aiida-wannier90.readthedocs.io

[106] Vitale V, Pizzi G, Marrazzo A, Yates J R, Marzari N and Mostofi A A 2019 Automated high-throughput Wannierisation (arXiv:1909.00433)

[107] GitHub www.github.com

[108] Travis-CI www.travis-ci.org

[109] codecov.io https://codecov.io

[110] MacMackin C 2018 FORD https://doi.org/10.5281/zenodo.1422473

[111] Wannier90 developers documentation generated with FORD 2019 www.wannier.org/ford/

[112] Spencer J Testcode https://github.com/jsspencer/testcode

[113] Wannier90 GitHub wiki page https://github.com/wannier-developers/wannier90/wiki

[114] Fprettify autoformatter for moder Fortran code https://github.com/pseewald/fprettify