

Introduction to **dmftproj**

V. Vildosola, L. Pourovskii, C. Martins, M. Aichhorn

August 27, 2012

Contents

1	Introduction to dmftproj	5
1.1	General description of the program dmftproj	5
1.1.1	Scheme of Wien2k+DMFT(CTQMC)	5
1.1.2	Structure of the program	6
1.2	Expressions of LAPW and APW+lo basis sets	6
1.2.1	LAPW basis	6
1.2.2	APW+lo basis	8
1.2.3	Summary	9
1.3	Calculation of the projectors	10
1.3.1	Correlated orbitals and “Wannier-projectors”	10
1.3.2	Introduction of the “ Θ -projectors”	11
1.3.3	Projectors in cubic or other basis	12
1.4	Taking into account the spin-orbit coupling	12
1.4.1	Consequences on the DMFT equations	13
1.4.2	Consequences on the construction of the Wannier projectors	13
1.4.3	Consequences on the construction of the Θ -projectors	14
1.5	Symmetry operations and their representations	14
1.5.1	Symmetry operations \mathcal{T} and local rotations R_{loc}	14
1.5.2	Description of the symmetry operations in case.dmftsym	15
1.5.3	Computation of the rotation matrices	16
1.5.4	Putting the Wannier and Θ -projectors in the global frame	17
1.5.5	Upfolding of the self-energy	17
1.6	Local quantities and sum over the irreducible Brillouin zone	17
1.6.1	Space groups and time-reversal operator	17
1.6.2	Performing the Brillouin zone integration	18
1.6.3	Application 1: calculation of the local density	20
1.6.4	Application 2: calculation of the local Green function	21
2	How to install and run the program	23
2.1	How to install dmftproj	23
2.2	Execution of the program dmftproj	23
2.2.1	In order to perform a DMFT calculation	23
2.2.2	In order to calculate a momentum-resolved spectral function	24
2.3	Description of the master input file case.indmftpr	24
2.4	Description of the output files	26
2.4.1	Main output file case.outdmftpr	26
2.4.2	Warning messages	30
2.4.3	Error messages	30
2.4.4	Output files created by dmftproj to perform a DMFT calculation	31

2.4.5	Output files created by <code>dmftproj</code> to calculate momentum-resolved spectral functions	31
3	Implementation of self-consistency in Wien2k+DMFT	33
3.1	LDA+DMFT charge density self-consistency in LAPW/APW+lo	33
3.1.1	Charge density and total energy in the LAPW/APW+lo method in DFT-LDA	33
3.1.2	Charge density and total energy in LAPW/APW+lo within LDA+DMFT . . .	35
3.2	Implementation within the Wien2k code	36
3.2.1	Summary of changes in the <i>lapw2</i> program	37
3.2.2	Changes in Wien2k shell scripts	37
3.3	Running self-consistent LDA+DMFT calculation with Wien2k	38

Chapter 1

Introduction to dmftproj

1.1 General description of the program dmftproj

SRC_dmftproj is a package to append to the Wien2k code which build the projectors to the localized orbitals onto which DMFT will be applied. In the following sections, we give a description of the construction of these projectors and other quantities that are needed for a Wien2k+DMFT calculation.

1.1.1 Scheme of Wien2k+DMFT(CTQMC)

The LDA+DMFT set of equations in the Kohn-Sham basis set, $|\psi_{k\nu}^\sigma\rangle$, are basically as follows.

The lattice Green function,

$$[G^\sigma(\mathbf{k}, i\omega_n)]_{\nu\nu'}^{-1} = (i\omega_n + \mu - \varepsilon_{k\nu}^\sigma)\delta_{\nu\nu'} - \Sigma_{\nu\nu'}^\sigma(\mathbf{k}, i\omega_n) \quad (1.1)$$

where $\varepsilon_{k\nu}^\sigma$'s are the eigenvalues of the LDA hamiltonian (obtained from lapw1).

The initial $\hat{G}^\sigma(\mathbf{k}, i\omega_n)$ is diagonal since the self-energy $\Sigma_{\nu\nu'} = 0$.

The local Green function is obtain by projecting the lattice Green function to the set of correlated orbitals m 's of the correlated atom α and summing over the full Brillouin zone as follows:

$$[\mathbf{G}_{loc}^\alpha(i\omega_n)]_{mm'}^\sigma = \sum_{\mathbf{k}, \nu\nu'} P_{lm, \nu}^{\alpha, \sigma}(\mathbf{k}) G_{\nu\nu'}^\sigma(\mathbf{k}, i\omega_n) [P_{lm', \nu'}^{\alpha, \sigma}(\mathbf{k})]^* \quad (1.2)$$

where

$$P_{lm, \nu}^{\alpha, \sigma}(\mathbf{k}) = \langle \chi_{lm}^{\alpha, \sigma} | \psi_{k\nu}^\sigma \rangle$$

are the matrix elements of the projectors defined as $\hat{P}_{lm}^{\alpha, \sigma} = \sum |\chi_{lm}^{\alpha, \sigma}\rangle \langle \chi_{lm}^{\alpha, \sigma}|$ that are going to be calculated in the SRC_dmftproj package. $|\chi_{lm}^{\alpha, \sigma}\rangle$ conform a set of Wannier-like functions to which we will refer about in Sec. 1.3.1.

The self-consistent condition of DMFT relates the *local* G of Eq. 1.2 with the one that comes out of the effective impurity problem, $\mathbf{G}_{loc}^{\alpha, \sigma}(i\omega_n) = \mathbf{G}_{imp}^{\alpha, \sigma}(i\omega_n)$.

The lattice self-energy $\Sigma_{\nu\nu'}(\mathbf{k}, i\omega_n)$ in Eq. 1.1 is the unfolded impurity self-energy defined as:

$$[\Delta \Sigma_{imp}^\alpha(i\omega_n)]_{mm'} = [\Sigma_{imp}(i\omega_n)]_{mm'} - [\Sigma_{dc}]_{mm'}$$

where $[\Sigma_{imp}(i\omega_n)]_{mm'}$ is calculated by a CT-QMC impurity solver and $[\Sigma_{dc}]_{mm'}$ the double-counting correction. So that,

$$\Sigma_{\nu\nu'}^\sigma(\mathbf{k}, i\omega_n) = \sum_{\alpha, mm'} [P_{lm, \nu}^{\alpha, \sigma}(\mathbf{k})]^* [\Delta \Sigma_{imp}^\alpha(i\omega_n)]_{mm'}^\sigma P_{lm', \nu'}^{\alpha, \sigma}(\mathbf{k}) \quad (1.3)$$

1.1.2 Structure of the program

Figure 1.1 displays a general overview of the structure of the program `dmftproj`. More details on the action of each main subroutine are available in the following sections.

1.2 Expressions of LAPW and APW+lo basis sets

In Wien2k, the eigenfunctions of the Kohn-Sham Hamiltonian $\psi_{\mathbf{k}\nu}^\sigma(\mathbf{r})$ are expanded as follows:

$$\psi_{\mathbf{k}\nu}^\sigma(\mathbf{r}) = \sum_{|\mathbf{K}| \leq K_{max}} c_{\mathbf{K}\nu}^\sigma(\mathbf{k}) \phi_{\mathbf{K}}^{\mathbf{k}\sigma}(\mathbf{r}) \quad (1.4)$$

with ν the band index, σ the spin degree of freedom and \mathbf{K} the reciprocal lattice vectors. The basis functions $\phi_{\mathbf{K}}^{\mathbf{k}\sigma}(\mathbf{r})$ are of course the APW or LAPW functions. For numerical approaches, the total number of basis functions considered is of course limited, hence a maximal value K_{max} .

1.2.1 LAPW basis

In the case of LAPW's basis functions, we have:

$$\phi_{\mathbf{K}}^{\mathbf{k}\sigma}(\mathbf{r}) = \begin{cases} \frac{1}{\sqrt{\Omega}} e^{i(\mathbf{k}+\mathbf{K})\cdot\mathbf{r}} & \text{if } \mathbf{r} \in I \\ \sum_l \sum_{m=-l}^{+l} \left[A_{lm}^{\alpha, \mathbf{k}+\mathbf{K}} u_l^{\alpha, \sigma}(r^\alpha, E_{1l}^\alpha) + B_{lm}^{\alpha, \mathbf{k}+\mathbf{K}} \dot{u}_l^{\alpha, \sigma}(r^\alpha, E_{1l}^\alpha) \right] Y_m^l(\hat{\mathbf{r}}^\alpha) & \text{if } \mathbf{r} \in S_{MT}^\alpha (\alpha = 1, \dots, N_{at}) \end{cases} \quad (1.5)$$

- \mathbf{K} are the reciprocal lattice vectors, Ω is the unit cell volume, N_{at} is the total number of atoms in the cell and I denotes the interstitial region.
- $\mathbf{r}^\alpha = \mathcal{T}_\alpha(\mathbf{r}) = \mathcal{R}_\alpha(\mathbf{r} - \mathbf{r}_\alpha)$ is the position inside the muffin-tin sphere S_{MT}^α , given with respect to the center of the sphere \mathbf{r}_α . \mathcal{T}_α is the transformation from the global coordinates of the crystal to the local coordinates associated to the atom α , which is composed of the rotation \mathcal{R}_α and the translation by the vector \mathbf{r}_α . The length of \mathbf{r}^α is denoted r^α , and the angles θ^α and ϕ^α specifying the direction of the vector in spherical coordinates, are indicated as $\hat{\mathbf{r}}^\alpha$.
- $Y_m^l(\hat{\mathbf{r}})$ are the spherical harmonics within the standard convention of Condon-Shortley (a factor $(-1)^m$ is included in the definition).
- $u_l^{\alpha, \sigma}(r, \varepsilon)$ is the radial solution of the Schrödinger equation for the free atom α at the energy ε (the linearization energy E_{1l}^α should be close to the center of the band the basis function is aimed to reproduce). The following normalization is specified :

$$\int_0^{R_{MT}^\alpha} r^2 |u_l^{\alpha, \sigma}(r, \varepsilon)|^2 dr = 1 \quad \forall \alpha, \sigma, l, \varepsilon$$

- $\dot{u}_l^{\alpha, \sigma}(E_{1l}^\alpha)$ is the energy derivative of $u_l^{\alpha, \sigma}$ at E_{1l}^α . Because of the previous normalization condition, $u_l^{\alpha, \sigma}$ and $\dot{u}_l^{\alpha, \sigma}$ are orthogonal. (it may be shown that they are also both orthogonal to the core states)

$$\dot{u}_l^{\alpha, \sigma}(r', E_{1l}) = \left. \frac{\partial u_l^{\alpha, \sigma}(r', E)}{\partial E} \right|_{E=E_{1l}^\alpha}$$

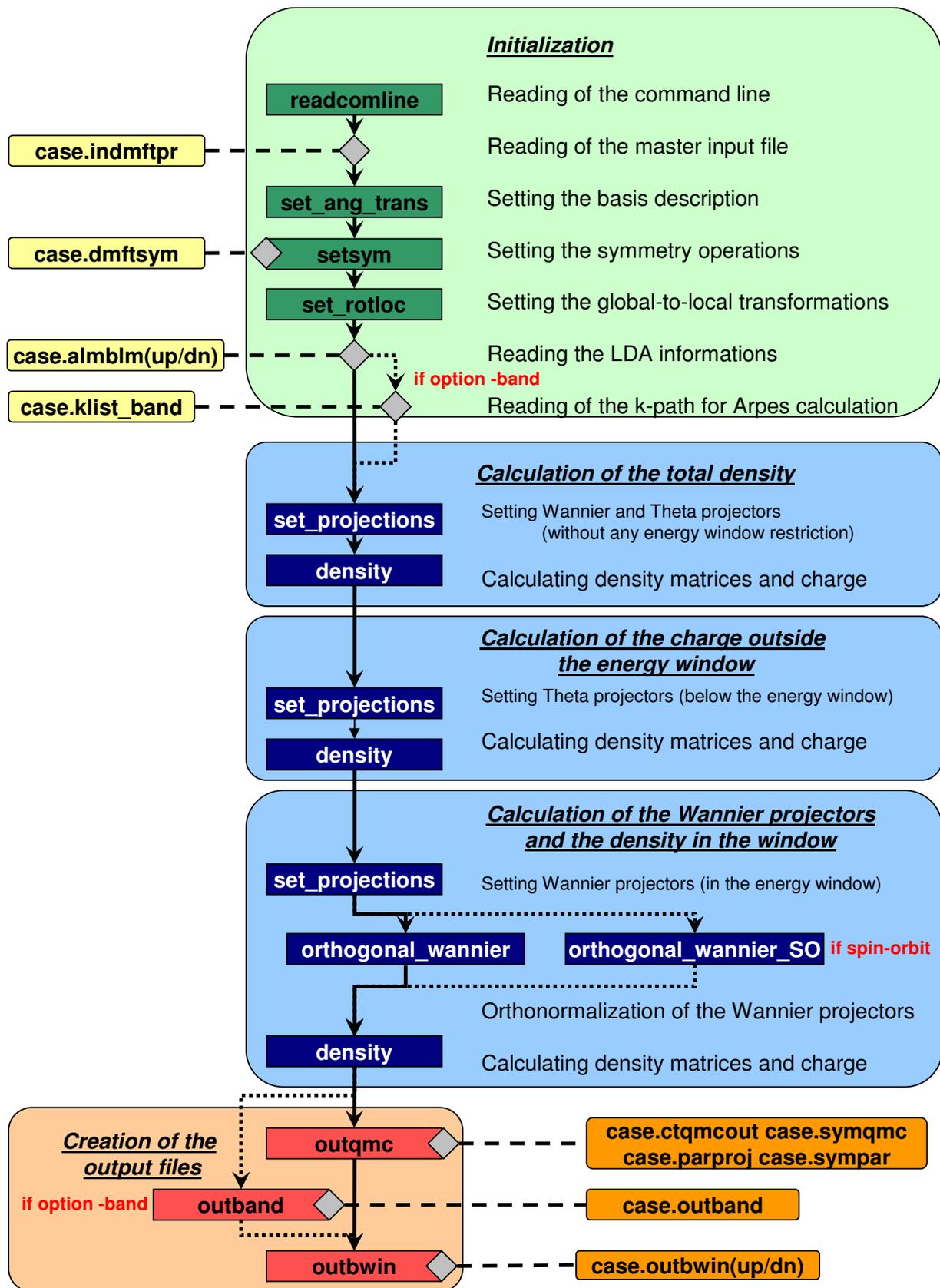


Figure 1.1: Structure of the program dmftproj

- the $A_{lm}^{\alpha, \mathbf{k}+\mathbf{K}}$ and $B_{lm}^{\alpha, \mathbf{k}+\mathbf{K}}$ are the expansion coefficients which are uniquely defined by requiring the continuity in value and slope of the basis wavefunction at the sphere boundary. They are given by:

$$A_{lm}^{\alpha, \mathbf{k}+\mathbf{K}} = \frac{4\pi}{\sqrt{\Omega}} i^l Y_l^{m*}(\mathcal{R}_\alpha(\widehat{\mathbf{k}+\mathbf{K}})) R_{MT}^{\alpha 2} a_l^\alpha(\mathbf{k}+\mathbf{K}) e^{i(\mathbf{k}+\mathbf{K})\cdot\mathbf{r}_\alpha} \quad (1.6)$$

$$B_{lm}^{\alpha, \mathbf{k}+\mathbf{K}} = \frac{4\pi}{\sqrt{\Omega}} i^l Y_l^{m*}(\mathcal{R}_\alpha(\widehat{\mathbf{k}+\mathbf{K}})) R_{MT}^{\alpha 2} b_l^\alpha(\mathbf{k}+\mathbf{K}) e^{i(\mathbf{k}+\mathbf{K})\cdot\mathbf{r}_\alpha} \quad (1.7)$$

where

$$a_l^\alpha(\mathbf{k}+\mathbf{K}) = \|\mathbf{k}+\mathbf{K}\| j_l'(\|\mathbf{k}+\mathbf{K}\|R_{MT}^\alpha) \dot{u}_l^{\alpha, \sigma}(R_{MT}^\alpha, E_{1l}^\alpha) - j_l(\|\mathbf{k}+\mathbf{K}\|R_{MT}^\alpha) [\dot{u}_l^{\alpha, \sigma}]'(R_{MT}^\alpha, E_{1l}^\alpha)$$

$$b_l^\alpha(\mathbf{k}+\mathbf{K}) = j_l(\|\mathbf{k}+\mathbf{K}\|R_{MT}^\alpha) [\dot{u}_l^{\alpha, \sigma}]'(R_{MT}^\alpha, E_{1l}^\alpha) - \|\mathbf{k}+\mathbf{K}\| j_l'(\|\mathbf{k}+\mathbf{K}\|R_\alpha) u_l^{\alpha, \sigma}(R_{MT}^\alpha, E_{1l}^\alpha).$$

We used the notation $[...] = \partial[...] / \partial r$ and denoted $j_l(x)$ the Bessel function of order l .

To account for semicore bands another type of basis functions are used, called *LOCAL ORBITALS* (*LO*).

$$\phi_{lm, \alpha}^{LO}(\mathbf{r}) = \begin{cases} 0 & \text{if } \mathbf{r} \in I \\ \left[A_{lm}^{\alpha, LO} u_l^{\alpha, \sigma}(r^\alpha, E_{1l}^\alpha) + B_{lm}^{\alpha, LO} \dot{u}_l^{\alpha, \sigma}(r^\alpha, E_{1l}^\alpha) + C_{lm}^{\alpha, LO} u_l^{\alpha, \sigma}(r^\alpha, E_{2l}^\alpha) \right] Y_m^l(\hat{\mathbf{r}}^\alpha) & \text{if } \mathbf{r} \in S_{MT}^\alpha \end{cases} \quad (1.8)$$

The semicore states are supposed to be very dispersionless bands centered close to E_{2l}^α and having negligible contribution to the interstitial.

1.2.2 APW+lo basis

In the case of APW's basis functions, we have, with the same notations as previously:

$$\phi_{\mathbf{K}}^{\mathbf{k}\sigma}(\mathbf{r}) = \begin{cases} \frac{1}{\sqrt{\Omega}} e^{i(\mathbf{k}+\mathbf{K})\cdot\mathbf{r}} & \text{if } \mathbf{r} \in I \\ \sum_l \sum_{m=-l}^{+l} A_{lm}^{\alpha, \mathbf{k}+\mathbf{K}} u_l^{\alpha, \sigma}(r^\alpha, \varepsilon) Y_m^l(\hat{\mathbf{r}}^\alpha) & \text{if } \mathbf{r} \in S_{MT}^\alpha \end{cases} \quad (1.9)$$

In this case, however, only the $A_{lm}^{\alpha, \mathbf{k}+\mathbf{K}}$ coefficients remain. They are uniquely defined by requiring the continuity in value of the basis wavefunction at the sphere boundary and are given by:

$$A_{lm}^{\alpha, \mathbf{k}+\mathbf{K}} = \frac{4\pi}{\sqrt{\Omega}} i^l Y_l^{m*}(\mathcal{R}_\alpha(\widehat{\mathbf{k}+\mathbf{K}})) \frac{j_l(\|\mathbf{k}+\mathbf{K}\|R_{MT}^\alpha)}{u_l^{\alpha, \sigma}(R_{MT}^\alpha, \varepsilon)} e^{i(\mathbf{k}+\mathbf{K})\cdot\mathbf{r}_\alpha} \quad (1.10)$$

The APW's need to be augmented with auxiliary basis functions, called *local orbitals* (*lo*), that are not matched to any plane wave in the interstitial.

$$\phi_{lm, \alpha}^{lo}(\mathbf{r}) = \begin{cases} 0 & \text{if } \mathbf{r} \in I \\ \left[A_{lm}^{\alpha, lo} u_l^{\alpha, \sigma}(r^\alpha, E_{1l}^\alpha) + B_{lm}^{\alpha, lo} \dot{u}_l^{\alpha, \sigma}(r^\alpha, E_{1l}^\alpha) \right] Y_m^l(\hat{\mathbf{r}}^\alpha) & \text{if } \mathbf{r} \in S_{MT}^\alpha \end{cases} \quad (1.11)$$

To account for semicore bands in the APW frame *the LOCAL ORBITALS* (*LO*) are built as follows:

$$\phi_{lm, \alpha}^{LO}(\mathbf{r}) = \begin{cases} 0 & \text{if } \mathbf{r} \in I \\ \left[A_{lm}^{\alpha, LO} u_l^{\alpha, \sigma}(r^\alpha, E_{1l}^\alpha) + C_{lm}^{\alpha, LO} u_l^{\alpha, \sigma}(r^\alpha, E_{2l}^\alpha) \right] Y_m^l(\hat{\mathbf{r}}^\alpha) & \text{if } \mathbf{r} \in S_{MT}^\alpha \end{cases} \quad (1.12)$$

In general the total number of basis functions will be determined by the number of plane waves considered to describe the interstitial plus the number of local orbitals LO and/or lo.

1.2.3 Summary

Whatever the choice for the basis, any eigenstate of the Kohn-Sham Hamiltonian $\psi_{\mathbf{k}\nu}^\sigma(\mathbf{r})$ can thus always be written as follows in Wien2k:

$$\psi_{\mathbf{k}\nu}^\sigma(\mathbf{r}) = \begin{cases} \frac{1}{\sqrt{\Omega}} \sum_{|\mathbf{K}| \leq K_{max}} c_{\mathbf{K}\nu}^\sigma(\mathbf{k}) e^{i(\mathbf{k}+\mathbf{K})\mathbf{r}} & \text{if } \mathbf{r} \in I \\ \sum_{l=0}^{l_{max}} \sum_{m=-l}^{+l} \left[A_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) u_l^{\alpha,\sigma}(r^\alpha, E_{1l}^\alpha) + B_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) u_l^{\alpha,\sigma}(r^\alpha, E_{1l}^\alpha) \right. \\ \left. + C_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) u_l^{\alpha,\sigma}(r^\alpha, E_{2l}^\alpha) \right] Y_m^l(\mathbf{r}^\alpha) & \text{if } \mathbf{r} \in S_{MT}^\alpha \end{cases} \quad (1.13)$$

More precisely, inside the muffin-tin spheres, the coefficients $A_{lm}^{\nu\alpha}(\mathbf{k}, \sigma)$, $B_{lm}^{\nu\alpha}(\mathbf{k}, \sigma)$ and $C_{lm}^{\nu\alpha}(\mathbf{k}, \sigma)$ are given by:

- if the atom α is described in the LAPW(+LO) representation,

$$\begin{aligned} A_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) &= \sum_{|\mathbf{K}| \leq K_{max}} c_{\mathbf{K}\nu}^\sigma(\mathbf{k}) A_{lm}^{\alpha, \mathbf{k}+\mathbf{K}} + \sum_{nLO=0}^{LO_{MAX}(\alpha, l, m)} c_{LO}^{\nu, \sigma} A_{lm}^{\alpha, LO} \\ B_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) &= \sum_{|\mathbf{K}| \leq K_{max}} c_{\mathbf{K}\nu}^\sigma(\mathbf{k}) B_{lm}^{\alpha, \mathbf{k}+\mathbf{K}} + \sum_{nLO=0}^{LO_{MAX}(\alpha, l, m)} c_{LO}^{\nu, \sigma} B_{lm}^{\alpha, LO} \\ C_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) &= \sum_{nLO=0}^{LO_{MAX}(\alpha, l, m)} c_{LO}^{\nu, \sigma} C_{lm}^{\alpha, LO} \end{aligned} \quad (1.14)$$

- if the atom α is described in the APW+lo representation,

$$\begin{aligned} A_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) &= \sum_{|\mathbf{K}| \leq K_{max}} c_{\mathbf{K}\nu}^\sigma(\mathbf{k}) A_{lm}^{\alpha, \mathbf{k}+\mathbf{K}} + \sum_{nlo=1}^{lo_{max}} c_{lo}^{\nu, \sigma} A_{lm}^{\alpha, lo} + \sum_{nLO=0}^{LO_{MAX}(\alpha, l, m)} c_{LO}^{\nu, \sigma} A_{lm}^{\alpha, LO} \\ B_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) &= \sum_{nlo=1}^{lo_{max}} c_{lo}^{\nu, \sigma} B_{lm}^{\alpha, lo} \\ C_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) &= \sum_{nLO=0}^{LO_{MAX}(\alpha, l, m)} c_{LO}^{\nu, \sigma} C_{lm}^{\alpha, LO} \end{aligned} \quad (1.15)$$

where $LO_{MAX}(\alpha, l, m)$ can be 0, if no LO are needed, or the number of necessary LO. In the current version of the program LO_{MAX} is assumed to be at most 1.

By running `lapw2` with the option `-alm` (cf. section 2.2), the coefficients are first read in the following form:

$$c_{\mathbf{K}\nu}^\sigma(\mathbf{k}) A_{lm}^{\alpha, \mathbf{k}+\mathbf{K}}, c_{lo}^{\nu, \sigma} A_{lm}^{\alpha, lo}, c_{LO}^{\nu, \sigma} A_{lm}^{\alpha, LO}, c_{lo}^{\nu, \sigma} B_{lm}^{\alpha, lo}, c_{\mathbf{K}\nu}^\sigma(\mathbf{k}) B_{lm}^{\alpha, \mathbf{k}+\mathbf{K}}, c_{LO}^{\nu, \sigma} B_{lm}^{\alpha, LO}, \text{ and } c_{LO}^{\nu, \sigma} C_{lm}^{\alpha, nLO}.$$

Then, the overlap matrix $\mathcal{O}_{lm, l'm'}^{\alpha, \sigma}$ between the local orbitals $|u_l^{\alpha, \sigma} Y_m^l\rangle$ are calculated and printed in the file `case.almb1m(up/dn)`. We remind that:

$$\begin{aligned} \langle u_l^{\alpha, \sigma}(E_{1l}^\alpha) Y_m^l | u_{l'}^{\alpha', \sigma'}(E_{1l'}^{\alpha'}) Y_{m'}^{l'} \rangle &= \delta_{\alpha\alpha'} \delta_{ll' mm'} \delta_{\sigma\sigma'} & \forall E_{1l}^\alpha, E_{1l'}^{\alpha'} \\ \langle u_l^{\alpha, \sigma}(E_{1l}^\alpha) Y_m^l | u_{l'}^{\alpha, \sigma}(E_{1l}^\alpha) Y_{m'}^{l'} \rangle &= 0 & \forall E_{1l}^\alpha \\ \langle u_l^{\alpha, \sigma}(E_{1l}^\alpha) Y_m^l | u_{l'}^{\alpha', \sigma'}(E_{2l'}^{\alpha'}) Y_{m'}^{l'} \rangle &= \mathcal{O}_{lm, l'm'}^{\alpha, \sigma} \delta_{\alpha\alpha'} \delta_{\sigma\sigma'} \neq 0 & \forall E_{1l}^\alpha, E_{2l'}^{\alpha'} \end{aligned} \quad (1.16)$$

Finally, the coefficients redefined in Eq. 1.14 or 1.15 are also printed in the file `case.almb1m(up/dn)`.

Remarks:

All the previous orbitals were defined in the unit cell. To extend their definition to the total crystal, it is enough to perform the following operation:

$$\phi_\mu(\mathbf{r}) = \frac{1}{\sqrt{\mathcal{N}}} \sum_{\mathbf{R} \in \mathcal{B}} e^{i\mathbf{k} \cdot \mathbf{R}} \phi_\mu(\mathbf{r} - \mathbf{R}) \quad \text{with} \quad \phi_\mu = \phi_{\mathbf{K}}^k, \phi_{lm,\alpha}^{lo} \text{ or } \phi_{lm,\alpha}^{LO} \quad (1.17)$$

where \mathcal{N} is the number of unit cells in the volume of the crystal and \mathbf{R} the translation vectors of the Bravais lattice. With this definition, the basis functions in the interstitial part are now $e^{i(\mathbf{k}+\mathbf{K}) \cdot \mathbf{r}}/\sqrt{V}$, with \mathbf{r} running in all the crystal volume V .

1.3 Calculation of the projectors

1.3.1 Correlated orbitals and “Wannier-projectors”

The DMFT self-consistent loop involves two different bases:

- the Bloch basis $|\psi_{\mathbf{k}\nu}^\sigma\rangle$, which is used to describe the lattice quantities,
- the Wannier-like functions $|\chi_{lm}^{\alpha,\sigma}\rangle$ which define a localized basis set, where α specifies an atom in the unit cell, (l, m) are the orbital indices and σ the spin degree of freedom.

In the following, we call “Wannier projectors” $P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k})$ the matrix elements of the projectors $\hat{P}_{lm}^{\alpha,\sigma}$ between these two bases. They have the following expression:

$$P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}) = \langle \chi_{lm}^{\alpha,\sigma} | \psi_{\mathbf{k}\nu}^\sigma \rangle \quad (1.18)$$

and are explicitly calculated in the program `dmftproj` by following the procedure introduced by Anisimov *et al.* (*PRL* **71**, 125119 (2005)).

If the orbital (l, m) of the atom α is considered as correlated, we start from the local orbitals $|\phi_{lm}^{\alpha,\sigma}\rangle$, which we choose to be the solution of the Schrödinger equation within the corresponding muffin-tin sphere S_{MT}^α at the corresponding linearization energy E_{1l}^α :

$$|\phi_{lm}^{\alpha,\sigma}\rangle = |u_l^{\alpha,\sigma}(E_{1l}^\alpha)Y_m^l\rangle \quad (1.19)$$

They can be expanded in the Bloch basis as

$$|\phi_{lm}^{\alpha,\sigma}\rangle = \sum \langle \psi_{\mathbf{k}\nu}^\sigma | \phi_{lm}^{\alpha,\sigma} \rangle | \psi_{\mathbf{k}\nu}^\sigma \rangle. \quad (1.20)$$

where the sum runs on all the Bloch states of the system, which is a complete basis.

In order to construct the Wannier-projectors, the user has to define in the input file `case.indmftpr` (cf. section 2.3) a suitable energy window $\mathcal{W} = [\varepsilon_1, \varepsilon_2]$ in which lie the Bloch states $|\psi_{\mathbf{k}\nu}^\sigma\rangle$ used to perform the projection. From these definitions, a first set of Wannier-like functions $|\tilde{\chi}_{lm}^{\alpha,\sigma}\rangle$ is given by:

$$|\tilde{\chi}_{lm}^{\alpha,\sigma}\rangle = \sum_{|\psi_{\mathbf{k}\nu}^\sigma\rangle \in \mathcal{W}} |\psi_{\mathbf{k}\nu}^\sigma\rangle \langle \psi_{\mathbf{k}\nu}^\sigma | u_l^{\alpha,\sigma}(E_{1l}^\alpha)Y_m^l \rangle \quad (1.21)$$

where the sum runs both on the band indices ν and on the \mathbf{k} -points of 1BZ. The corresponding *temporary* projectors are:

$$\tilde{P}_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}) = \langle u_l^{\alpha,\sigma}(E_{1l}^\alpha)Y_m^l | \psi_{\mathbf{k}\nu}^\sigma \rangle \quad \forall \nu \in \mathcal{W} \quad \forall \mathbf{k} \in 1\text{BZ}. \quad (1.22)$$

Since an energy window was chosen, the number of bands in the sum of (1.21) depends on \mathbf{k} and σ . In practice, we define the temporary projection matrix $\tilde{P}_{m\nu}^{\alpha,\sigma}(\mathbf{k})$ for all the bands ν such that $\nu_{min}(k, \sigma) \leq \nu \leq \nu_{max}(k, \sigma)$.

Due to the truncation in the sum over the Bloch states, the Wannier-like orbitals $|\tilde{\chi}_{lm}^{\alpha,\sigma}\rangle$ should be orthonormalized in order to give the final set of Wannier functions. One thus needs to calculate the overlap matrix:

$$\langle \tilde{\chi}_{lm}^{\alpha,\sigma} | \tilde{\chi}_{lm'}^{\alpha',\sigma'} \rangle = \sum_{\mathbf{k} \in 1\text{BZ}} O_{m,m'}^{\alpha,\alpha'}(\mathbf{k}, \sigma) \delta_{\sigma\sigma'} \quad (1.23)$$

$$\text{with } O_{m,m'}^{\alpha,\alpha'}(\mathbf{k}, \sigma) = \sum_{\nu=\nu_{min}(\mathbf{k},\sigma)}^{\nu_{max}(\mathbf{k},\sigma)} \tilde{P}_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}) \left[\tilde{P}_{lm',\nu}^{\alpha',\sigma}(\mathbf{k}) \right]^* \quad (1.24)$$

However, for practical reasons, it is easier to remain in the reciprocal space to perform this orthonormalization. Indeed, in this case, it is enough to multiply each *temporary* projector by the inverse square root of $O_{m,m'}^{\alpha,\alpha'}(\mathbf{k}, \sigma)$ and the "true" Wannier projectors $P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k})$ finally reads:

$$P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}) = \sum_{\alpha'm'} \left\{ [O(\mathbf{k}, \sigma)]^{-1/2} \right\}^{\alpha,\alpha'} \tilde{P}_{lm',\nu}^{\alpha',\sigma}(\mathbf{k}) \quad \forall \nu \in \mathcal{W} \quad \forall \mathbf{k} \in 1\text{BZ}. \quad (1.25)$$

As a result, the program `dmftproj` calculates roughly the Wannier projectors $P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k})$ in two steps:

- a first subroutine `set_projection` produces the *temporary* projectors $\tilde{P}_{lm,\nu}^{\alpha,\sigma}(\mathbf{k})$. Using the decomposition of the Bloch states on the LAPW+LO (or APW+lo) basis given by the general equation (1.13) and the relations (1.16), the *temporary* projectors can be merely calculated by:

$$\tilde{P}_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}) = A_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) + \sum_{n_{LO}=1}^{LOMAX} c_{LO}^{\nu,\sigma} C_{lm}^{\alpha,LO} \mathcal{O}_{lm,l'm'}^{\alpha,\sigma} \quad (1.26)$$

- a second subroutine `orthogonal_wannier` performs the orthonormalization described in (1.25).

1.3.2 Introduction of the “ Θ -projectors”

The `dmftproj` program also provides projectors to calculate quantities for a given atom α and a particular orbital (spin) character lm (σ) – such as the spectral functions $A_{lm}^{\sigma\alpha}(\mathbf{k}, \omega)$. A new type of projectors is thus introduced. They are called “ Θ -projectors”. Contrary to the previously introduced Wannier projectors $P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k})$, the definition of the Θ -projectors is not restricted to the correlated orbitals only.

As we have previously explained in section 1.2, inside the muffin-tin sphere S_{MT}^{α} associated to the atom α we can write $\psi_{\mathbf{k}\nu}^{\sigma}(\mathbf{r})$ as:

$$\begin{aligned} \psi_{\mathbf{k}\nu}^{\sigma}(\mathbf{r}) &= \sum_{l=0}^{l_{max}} \sum_{m=-l}^{+l} \left[A_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) u_l^{\alpha,\sigma}(r^{\alpha}, E_{1l}^{\alpha}) + B_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) \dot{u}_l^{\alpha,\sigma}(r^{\alpha}, E_{1l}^{\alpha}) + C_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) u_l^{\alpha,\sigma}(r^{\alpha}, E_{2l}^{\alpha}) \right] Y_m^l(\hat{\mathbf{r}}^{\alpha}) \\ &= \sum_{l=0}^{l_{max}} \sum_{m=-l}^{+l} \left[A_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) u_{lm,1}^{\alpha,\sigma}(\mathbf{r}^{\alpha}) + B_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) u_{lm,2}^{\alpha,\sigma}(\mathbf{r}^{\alpha}) + C_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) u_{lm,3}^{\alpha,\sigma}(\mathbf{r}^{\alpha}) \right] \end{aligned} \quad (1.27)$$

with the notation:

$$u_{lm,1}^{\alpha,\sigma}(\mathbf{r}^{\alpha}) = u_l^{\alpha,\sigma}(r^{\alpha}, E_{1l}^{\alpha}) Y_m^l(\hat{\mathbf{r}}^{\alpha}) \quad , \quad \dot{u}_{lm,2}^{\alpha,\sigma}(\mathbf{r}^{\alpha}) = \dot{u}_l^{\alpha,\sigma}(r^{\alpha}, E_{1l}^{\alpha}) Y_m^l(\hat{\mathbf{r}}^{\alpha}) \quad (1.28)$$

$$\text{and } u_{lm,3}^{\alpha,\sigma}(\mathbf{r}^{\alpha}) = u_l^{\alpha,\sigma}(r^{\alpha}, E_{2l}^{\alpha}) Y_m^l(\hat{\mathbf{r}}^{\alpha}).$$

The orbital character α, l, m, σ thus contributes in the eigenstates $\psi_{\mathbf{k}\nu}^\sigma(\mathbf{r})$ through three terms.

However, the basis $\{u_{lm,i}^{\alpha,\sigma}\}_{i=\{1,2,3\}}$ is not orthonormalized as already mentioned in section 1.2. To make the calculations easier, we introduce an orthonormal basis set $\{\phi_{lm,j}^{\alpha,\sigma}\}_{j=1,2,3}$ for each atomic orbital (l, m) . These orbitals are defined from the initial basis $\{u_{lm,i}^{\alpha,\sigma}\}_{i=\{1,2,3\}}$ as follows:¹

$$\forall i \quad u_{lm,i}^{\alpha,\sigma}(\mathbf{r}^\alpha) = \sum_{j=1}^3 c_{ij} \phi_{lm,j}^{\alpha,\sigma} \quad \text{with} \quad \mathbf{C} = \begin{pmatrix} 1 & 0 & \langle u_{lm,1}^{\alpha,\sigma} | u_{lm,2}^{\alpha,\sigma} \rangle \\ 0 & \langle u_{lm,2}^{\alpha,\sigma} | u_{lm,2}^{\alpha,\sigma} \rangle & \langle u_{lm,2}^{\alpha,\sigma} | u_{lm,3}^{\alpha,\sigma} \rangle \\ \langle u_{lm,3}^{\alpha,\sigma} | u_{lm,1}^{\alpha,\sigma} \rangle & \langle u_{lm,3}^{\alpha,\sigma} | u_{lm,2}^{\alpha,\sigma} \rangle & 1 \end{pmatrix}^{\frac{1}{2}}. \quad (1.29)$$

We can then rewrite (1.27) as:

$$\psi_{\mathbf{k}\nu}^\sigma(\mathbf{r}) = \sum_{l=0}^{l_{max}} \sum_{m=-l}^{+l} \sum_{j=1}^3 \Theta_{lm\nu j}^{\alpha,\sigma}(\mathbf{k}) \phi_{lm,j}^{\alpha,\sigma}(\mathbf{r}^\alpha). \quad (1.30)$$

The matrix elements $\Theta_{lm\nu j}^{\alpha,\sigma}(\mathbf{k})$ are the “ Θ -projectors”, which are thus defined by:

$$\Theta_{lm\nu j}^{\alpha,\sigma}(\mathbf{k}) = \langle \phi_{lm,j}^{\alpha,\sigma} | \psi_{\mathbf{k}\nu}^\sigma \rangle = A_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) c_{1j} + B_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) c_{2j} + C_{lm}^{\nu\alpha}(\mathbf{k}, \sigma) c_{3j}. \quad (1.31)$$

They are calculated in the subroutine `set_projections`.

1.3.3 Projectors in cubic or other basis

It is possible by requiring it in `case.indmftpr` to get the projectors to the correlated orbitals in cubic symmetry. For this, the program reads the unitary transformation from spherical harmonics to cubic basis from file `SRC_templates/case.cf_d_eg_t2g` (in subroutine `set_ang_trans.f`).

From the previous descriptions, it appears that the projectors $P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k})$ and $\Theta_{lm\nu j}^{\alpha,\sigma}(\mathbf{k})$ are defined in the complex spherical harmonics basis. This does not imply that the program can only calculate them in this basis.

It is possible to get the projectors in cubic symmetry or in any desired basis $|\varphi_i^{(l)}\rangle$ by requiring it in the input file (cf. section 2.3). For this, the program reads the corresponding unitary transformation $U_{i,m}^{(l)} = \langle \varphi_i^{(l)} | Y_{lm} \rangle$, which is then used to transform the projectors:

$$P_{li,\nu}^{\alpha,\sigma}(\mathbf{k}) = \sum_{lm} U_{i,m}^{(l)} P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}) \quad \text{and} \quad \Theta_{li\nu j}^{\alpha,\sigma}(\mathbf{k}) = \sum_{lm} U_{i,m}^{(l)} \Theta_{lm\nu j}^{\alpha,\sigma}(\mathbf{k}) \quad (1.32)$$

While the unitary transformations from spherical harmonics to cubic basis is read in the files `case.cf_p_cubic`, `case.cf_d_eg_t2g` and `case.cf_f_mm2` which are in the directory `SRC_templates`, any other unitary transformation requires an additional file which must be put in the running directory (cf. section 2.3). The reading of these files is performed in the subroutine `set_ang_trans`.

1.4 Taking into account the spin-orbit coupling

In Wien2k, spin-orbit corrections can be introduced via a second variational approach. Although the spin-orbit term couples the spin-up and spin-down wavefunctions, the wavefunctions $\psi_{\mathbf{k}\nu}(\mathbf{r})$ are still of the Bloch form:

$$\begin{aligned} \psi_{\mathbf{k}\nu}(\mathbf{r}) &= \sum_n z_{n\uparrow}^\nu \varphi_{\mathbf{k}n}^\uparrow(\mathbf{r}) + \sum_n z_{n\downarrow}^\nu \varphi_{\mathbf{k}n}^\downarrow(\mathbf{r}) = \psi_{\mathbf{k}\nu}^+(\mathbf{r}) + \psi_{\mathbf{k}\nu}^-(\mathbf{r}) \\ &= [u_{\mathbf{k}\nu}^+(\mathbf{r}) + u_{\mathbf{k}\nu}^-(\mathbf{r})] e^{i\mathbf{k}\cdot\mathbf{r}}. \end{aligned} \quad (1.33)$$

¹We remind that the functions $u_i^{\alpha,\sigma}(r^\alpha, E) Y_m^l(\hat{\mathbf{r}}^\alpha)$ are normalized to 1 for each value of E by definition. As a result, $\langle u_{lm,1}^{\alpha,\sigma} | u_{lm,1}^{\alpha,\sigma} \rangle = \langle u_{lm,3}^{\alpha,\sigma} | u_{lm,3}^{\alpha,\sigma} \rangle = 1$ and $\langle u_{lm,1}^{\alpha,\sigma} | u_{lm,2}^{\alpha,\sigma} \rangle = 0$.

In the present version of the program `dmftproj`, a calculation of LDA+DMFT which takes into account the spin-orbit coupling is available *only with **spin-polarized** input files*. In this case, the basis $|\psi_{\nu\mathbf{k}}^+\rangle, |\psi_{\nu\mathbf{k}}^-\rangle$ is indeed the natural output of the DFT calculations and can be easily read from the “up” and “down” files.

1.4.1 Consequences on the DMFT equations

According to equation (1.33), it is convenient to choose the basis $|\psi_{\nu\mathbf{k}}^+\rangle, |\psi_{\nu\mathbf{k}}^-\rangle$ to describe the lattice quantities in the LDA+DMFT implementation. A new set of index $i = \{+, -\}$ must thus be introduced to extend the formalism presented in section 1.1.1. The Wannier projectors can then be written as follows:

$$\begin{aligned} [P_{lm,\nu}^{\alpha,\sigma}]^i(\mathbf{k}) &= \langle \chi_{lm}^{\alpha,\sigma} | \psi_{\mathbf{k}\nu}^i \rangle \quad \forall i \in \{+, -\} \quad \text{and} \quad \forall \sigma \in \{\uparrow, \downarrow\} \\ \text{or } \mathbf{P}_{lm,\nu}^\alpha(\mathbf{k}) &= \left(\begin{array}{c|c} [P_{lm,\nu}^{\alpha,\uparrow}]^+(\mathbf{k}) & [P_{lm,\nu}^{\alpha,\downarrow}]^+(\mathbf{k}) \\ \hline [P_{lm,\nu}^{\alpha,\uparrow}]^-(\mathbf{k}) & [P_{lm,\nu}^{\alpha,\downarrow}]^-(\mathbf{k}) \end{array} \right) = \left(\begin{array}{c|c} [P_{lm,\nu}^{\alpha,\uparrow}]^+(\mathbf{k}) & 0 \\ \hline 0 & [P_{lm,\nu}^{\alpha,\downarrow}]^-(\mathbf{k}) \end{array} \right) \end{aligned} \quad (1.34)$$

where we remind that ν runs over both spin and orbital indices. In addition, the Green’s function and the self-energy of the solid are now related by:

$$\{[G(\mathbf{k}, i\omega_n)]^{-1}\}_{\nu\nu'}^{ij} = (i\omega_n + \mu - \varepsilon_{k\nu})\delta_{\nu\nu'}\delta_{ij} - \Sigma_{\nu\nu'}^{ij}(\mathbf{k}, i\omega_n) \quad \text{with} \quad i, j = \{+, -\}. \quad (1.35)$$

With these notations, the formula (1.2) which gives the expression of the local Green’s function merely can be written as:

$$[G_{loc}^\alpha(i\omega_n)]_{mm'}^{\sigma\sigma'} = \sum_{k,\nu\nu',ij} [P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k})]^i G_{\nu\nu'}^{ij}(\mathbf{k}, i\omega_n) [P_{lm',\nu'}^{\alpha,\sigma'}(\mathbf{k})]^{j*} \quad (1.36)$$

and the expression (1.3) for the lattice self-energy correction is:

$$\Sigma_{\nu\nu'}^{ij}(\mathbf{k}, i\omega_n) = \sum_{\alpha, mm', \sigma\sigma'} [P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k})]^{i*} [\Delta\Sigma_{imp}^\alpha(i\omega_n)]_{mm'}^{\sigma\sigma'} [P_{lm',\nu'}^{\alpha,\sigma'}(\mathbf{k})]^j. \quad (1.37)$$

Consequently, the equations of the DMFT loop are formally the same as in the case without spin-orbit coupling. However, the computations now involve matrices which are double in size.

1.4.2 Consequences on the construction of the Wannier projectors

The construction of the Wannier projectors is still done in two steps.

- The *temporary* Wannier projectors are first calculated – separately for the $|\psi_{\nu\mathbf{k}}^+\rangle$ and the $|\psi_{\nu\mathbf{k}}^-\rangle$ basis functions – in the subroutine `set_projections` with the following expression:

$$[\tilde{P}_{lm,\nu}^{\alpha,\sigma}]^i(\mathbf{k}) = [A_{lm}^{\nu\alpha}(\mathbf{k}, \sigma)]^i + \sum_{n_{LO}=1}^{LOMAX} c_{LO}^{\nu,\sigma} [C_{lm}^{\alpha,LO} \mathcal{O}_{lm,l'm'}^{\alpha,\sigma}]^i \quad \text{with} \quad i = \{+, -\}. \quad (1.38)$$

The coefficients $[A_{lm}^{\nu\alpha}(\mathbf{k}, \sigma)]^i$ and $[C_{lm}^{\alpha,LO} \mathcal{O}_{lm,l'm'}^{\alpha,\sigma}]^i$ are the analog of the coefficients $A_{lm}^{\nu\alpha}(\mathbf{k}, \sigma)$ and $C_{lm}^{\alpha,LO} \mathcal{O}_{lm,l'm'}^{\alpha,\sigma}$ given in (1.13) for the $|\psi_{\mathbf{k}\nu}^\sigma\rangle$. They are read directly in the output files `case.almblmup` and `case.almblmdn`, which were created when running `x lapw2 -alm -so -up/dn`.

- The orthogonalization is then performed in order to get the “true” Wannier projectors but this step cannot be done independently for the two parts of the Wannier projector, since the overlap matrix $O_{m,m'}^{\alpha,\alpha'}(\mathbf{k}, \sigma)$ is defined by:

$$\langle \tilde{\chi}_{lm}^{\alpha,\sigma} | \tilde{\chi}_{lm'}^{\alpha',\sigma'} \rangle = \sum_{\mathbf{k} \in \text{1BZ}} \left[O^{\alpha,\alpha'}(\mathbf{k}) \right]_{m,m'}^{\sigma,\sigma'} \quad (1.39)$$

$$\begin{aligned} \text{with } \left[O^{\alpha,\alpha'}(\mathbf{k}) \right]_{m,m'}^{\sigma,\sigma'} &= \sum_{\nu=\nu_{\min}(\mathbf{k})}^{\nu_{\max}(\mathbf{k})} \langle \tilde{\chi}_{lm}^{\alpha,\sigma} | \psi_{\mathbf{k}\nu} \rangle \langle \psi_{\mathbf{k}\nu} | \tilde{\chi}_{lm'}^{\alpha',\sigma'} \rangle \\ &= \sum_{\nu=\nu_{\min}(\mathbf{k})}^{\nu_{\max}(\mathbf{k})} \sum_{i,j=+,-} \left[\tilde{P}_{lm,\nu}^{\alpha,\sigma} \right]^i(\mathbf{k}) \left[\tilde{P}_{lm',\nu}^{\alpha',\sigma'} \right]^{j*}(\mathbf{k}). \end{aligned} \quad (1.40)$$

The subroutine `orthogonal_wannier.SO` is thus used, instead of `orthogonal_wannier`.

By requiring it in the input file, it is still possible to get the projectors of the correlated orbitals in cubic symmetry or in any desired basis. This option was also extended in order to define the Wannier projectors related to states which mix spin up and spin down complex spherical harmonics.

1.4.3 Consequences on the construction of the Θ -projectors

When the spin-orbit interaction is introduced, the definition of the projectors $\Theta_{lm,\nu j}^{\alpha,\sigma}(\mathbf{k})$ must be extended to “matrix” projectors $\Theta_{lm\nu j}^{\alpha,\sigma}(\mathbf{k})$, in a similar fashion as previously described for the Wannier projectors

$$\Theta_{lm\nu j}^{\alpha,\sigma}(\mathbf{k}) = \left(\begin{array}{c|c} \langle \phi_{lm,j}^{\alpha,\uparrow} | \psi_{\mathbf{k}\nu}^+ \rangle & \langle \phi_{lm,j}^{\alpha,\downarrow} | \psi_{\mathbf{k}\nu}^+ \rangle \\ \langle \phi_{lm,j}^{\alpha,\uparrow} | \psi_{\mathbf{k}\nu}^- \rangle & \langle \phi_{lm,j}^{\alpha,\downarrow} | \psi_{\mathbf{k}\nu}^- \rangle \end{array} \right) = \left(\begin{array}{c|c} \langle \phi_{lm,j}^{\alpha,\uparrow} | \psi_{\mathbf{k}\nu}^+ \rangle & 0 \\ 0 & \langle \phi_{lm,j}^{\alpha,\downarrow} | \psi_{\mathbf{k}\nu}^- \rangle \end{array} \right). \quad (1.41)$$

However, since the calculation can be performed separately for the $|\psi_{\nu\mathbf{k}}^+\rangle$ and the $|\psi_{\nu\mathbf{k}}^-\rangle$ basis functions, no changes occur in their construction (as soon as σ is replaced by i in section 1.3.2), which is still performed in the subroutine `set_projections`.

As for the Wannier projectors, it is of course possible to get the Θ -projectors in any desired basis which mixes spin up and spin down complex spherical harmonics, for a calculation including the spin-orbit coupling.

1.5 Symmetry operations and their representations

1.5.1 Symmetry operations \mathcal{T} and local rotations R_{loc}

The symmetry operations of the crystallographic space group of a compound have the general following form:

$$\forall \mathbf{r} \in \mathbb{R}^3 \quad \mathcal{T}(\mathbf{r}) = \mathcal{R}(\mathbf{r}) + \mathbf{v} \quad (1.42)$$

where \mathbf{v} is a translation vector which belongs to the Bravais lattice or is a rational fraction of such a vector and \mathcal{R} is an operator of the *crystallographic point group* of the compound. In Wien2k, the symmetry operations \mathcal{T} of the studied compound are calculated during the *initialization step* and described at the end of the `case.struct` file within the following form of a 3×4 matrix :

$$\begin{array}{cccc} r_{xx} & r_{xy} & r_{xz} & t_x \\ r_{yx} & r_{yy} & r_{yz} & t_y \\ r_{zx} & r_{zy} & r_{zz} & t_z \end{array} \iff \forall \mathbf{u} \in \mathbb{R}^3 \quad \mathcal{T}(\mathbf{u}) = \begin{pmatrix} r_{xx} & r_{xy} & r_{xz} \\ r_{yx} & r_{yy} & r_{yz} \\ r_{zx} & r_{zy} & r_{zz} \end{pmatrix} \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}. \quad (1.43)$$

The previous symmetry operations are defined in the “*global coordinate system*”. On the contrary, the local orbitals of each atom α of the unit cell are defined in a “*local coordinate system*” associated to this atom α . We insist that this *local frame* may be different for each atom α , even if they are of the same atomic species A . This *local frame* is defined in two steps:

- First, the symmetry operation \mathcal{T} which transforms the first representant of the atomic species A into the atom α is applied.
- Then, the specific local rotation R_{loc}^A associated to the atomic species A and defined in the `case.struct` file is applied.

As a result, the transformation from the global coordinate system to the local frame of an atom α is:

$$R_{loc}^\alpha = R_{loc}^A \cdot \mathcal{T}. \quad (1.44)$$

1.5.2 Description of the symmetry operations in `case.dmftsym`

The symmetry operations \mathcal{T} and the local rotation R_{loc}^A are described as matrices in real space \mathbb{R}^3 in the `case.struct` file. Each transformations S , with $S = \mathcal{T}$ or R_{loc}^A , may be proper (its determinant is 1) or improper (its determinant is -1). In the first case, S can be completely described by three Euler angles α , β and γ . In the second case, S can always be written as the product of an inversion $I = -Id$ and a proper transformation. As a result $I \cdot S$ can be completely described by three Euler angles α , β and γ .

The three Euler angles associated to a proper transformation S are calculated in the program `euler` of the `Wien2k` package. The convention used is the following:

$$S = R[\mathbf{z}](\gamma) \cdot R[\mathbf{y}](\beta) \cdot R[\mathbf{z}](\alpha) \quad (1.45)$$

where $R[\mathbf{z}](\gamma)$ stands for the rotation of angle $\gamma \in [0; 2\pi[$ around the \mathbf{z} -axis, $R[\mathbf{y}](\beta)$ is the rotation of angle $\beta \in [0; \pi]$ around the \mathbf{y} -axis and $R[\mathbf{z}](\alpha)$ the rotation of angle $\alpha \in [0; 2\pi[$ around the \mathbf{z} -axis. This corresponds in \mathbb{R}^3 to:

$$D^{\mathbb{R}^3}(S) = \begin{pmatrix} \cos \gamma \cos \beta \cos \alpha - \sin \gamma \sin \alpha & -\sin \gamma \cos \alpha - \cos \gamma \cos \beta \sin \alpha & \cos \gamma \sin \beta \\ \sin \gamma \cos \beta \cos \alpha + \cos \gamma \sin \alpha & \cos \gamma \cos \alpha - \sin \gamma \cos \beta \sin \alpha & \sin \gamma \sin \beta \\ -\sin \beta \cos \alpha & \sin \beta \sin \alpha & \cos \beta \end{pmatrix} \quad (1.46)$$

However, the program `euler` does not calculate the Euler angles, it rather gives three parameters \mathbf{a} , \mathbf{b} and \mathbf{c} which are linked to α , β and γ as follows:

- if $\beta = 0$, then $\mathbf{b} = 0$, $\mathbf{c} = 0$ and $\mathbf{a} = 2\pi - \alpha$ for $\alpha \in]0; 2\pi[$ and $\mathbf{a} = 0$ when $\alpha = 0$,
- if $\beta = \pi$, then $\mathbf{b} = 0$, $\mathbf{a} = 0$ and $\mathbf{c} = 2\pi - \gamma$ for $\gamma \in]0; 2\pi[$ and $\mathbf{c} = 0$ when $\gamma = 0$,
- if $\beta \in]0; \pi[$, then $\mathbf{b} = \beta$, $\mathbf{a} = \pi - \alpha$ for $\alpha \in [0; \pi]$ and $\mathbf{a} = 3\pi - \alpha$ for $\alpha \in]\pi; 2\pi[$ and similarly for \mathbf{c} and γ .²

When running `lapw2` with the option `-alm`, a file `case.dmftsym` is produced. This file contains all the informations about the symmetry operations and local rotations of the system, namely:

- the total number of symmetry operations \mathcal{T} of the system,
- the permutation table of the representants of all atomic species, when the symmetry operation \mathcal{T} is applied,
- a description of each symmetry operation \mathcal{T} and of each specific local rotation R_{loc}^A : its *Euler angles* \mathbf{a} , \mathbf{b} and \mathbf{c} , its determinant *iprop* and the matrices in real space \mathbb{R}^3 of \mathcal{T} and R_{loc}^A (or $I \cdot \mathcal{T}$ and $I \cdot R_{loc}^A$ if they are improper).

²These relations hold if the matrix used as an input of `euler` is directly `iz(iord)`.

Remark

Only the operator part \mathcal{R} of \mathcal{T} are considered. The translation vector \mathbf{t} of \mathcal{T} introduce only phase factor in the definition of the coefficients and can thus be neglected for calculating local quantities.

1.5.3 Computation of the rotation matrices

To apply a symmetry operation \mathcal{T} or a global-to-local transformation R_{loc}^α to a projector $P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k})$ and $\Theta_{lm\nu j}^{\alpha,\sigma}(\mathbf{k})$, one must use the corresponding transformation matrices $\mathcal{D}^{\frac{1}{2}}(S)$ in spin-space or $\mathcal{D}^l(S)$ in the space spanned by the spherical harmonics Y_m^l , with $S = \mathcal{T}$ or R_{loc}^α . These rotation matrices are calculated in the subroutines `setsym` and `set_rotloc`.

For a proper rotations, the spinor rotation $\mathcal{D}^{\frac{1}{2}}(S)$ is defined by the following matrix in the basis $\{|\uparrow\rangle, |\downarrow\rangle\}$:

$$\forall \alpha, \beta, \gamma \quad \mathcal{D}^{\frac{1}{2}}(\mathbf{a}, \mathbf{b}, \mathbf{c}) = \begin{pmatrix} \cos \frac{\mathbf{b}}{2} e^{i\frac{\mathbf{a}+\mathbf{c}}{2}} & \sin \frac{\mathbf{b}}{2} e^{-i\frac{\mathbf{a}-\mathbf{c}}{2}} \\ -\sin \frac{\mathbf{b}}{2} e^{i\frac{\mathbf{a}-\mathbf{c}}{2}} & \cos \frac{\mathbf{b}}{2} e^{-i\frac{\mathbf{a}+\mathbf{c}}{2}} \end{pmatrix}. \quad (1.47)$$

The rotation matrices $\mathcal{D}^l(S)$ are calculated by the subroutine `dmat` which creates the matrix whose coefficients are in the basis $\{|l, -l\rangle, \dots, |l, l\rangle\}$:

$$\begin{aligned} \left(\mathcal{D}^l(\mathbf{a}, \mathbf{b}, \mathbf{c})\right)_{m,n} &= e^{i(n\mathbf{a}+m\mathbf{c})} \sqrt{\frac{(l+m)!(l-m)!}{(l+n)!(l-n)!}} \sum_{k=0}^{2l} (-1)^{l-m-k} C_{m,n}^{l,k} \left(\sin \frac{\mathbf{b}}{2}\right)^{2l-m-n-2k} \left(\cos \frac{\mathbf{b}}{2}\right)^{2k+m+n} \\ \text{with } C_{m,n}^{l,k} &= \frac{(l+n)!(l-n)!}{(l-m-k)!(m+n+k)!(l-n-k)!k!} \quad \text{and } m, n \in [-l; l] \end{aligned} \quad (1.48)$$

For each $(\mathcal{D}^l(\mathbf{a}, \mathbf{b}, \mathbf{c}))_{m,n}$, the value of k considered in the sum are only those for which :

$$l - m - k \geq 0 \quad \text{and} \quad l - n - k \geq 0 \quad \text{and} \quad m + n + k \geq 0. \quad (1.49)$$

To understand better this expression, we give, as an example, the rotation matrix in the subspace $l = 1$. The basis considered is thus $\{|1, -1\rangle, |1, 0\rangle, |1, 1\rangle\}$ and we get:

$$\begin{aligned} \mathcal{D}^1(\mathbf{a}, \mathbf{b}, \mathbf{c}) &= \begin{pmatrix} (\cos \frac{\mathbf{b}}{2})^2 e^{-i(\mathbf{a}+\mathbf{c})} & -\sqrt{2} \sin \frac{\mathbf{b}}{2} \cos \frac{\mathbf{b}}{2} e^{-i\mathbf{c}} & (\sin \frac{\mathbf{b}}{2})^2 e^{i(\mathbf{a}-\mathbf{c})} \\ \sqrt{2} \sin \frac{\mathbf{b}}{2} \cos \frac{\mathbf{b}}{2} e^{-i\mathbf{a}} & (\cos \frac{\mathbf{b}}{2})^2 - (\sin \frac{\mathbf{b}}{2})^2 & -\sqrt{2} \sin \frac{\mathbf{b}}{2} \cos \frac{\mathbf{b}}{2} e^{i\mathbf{a}} \\ (\sin \frac{\mathbf{b}}{2})^2 e^{-i(\mathbf{a}-\mathbf{c})} & \sqrt{2} \sin \frac{\mathbf{b}}{2} \cos \frac{\mathbf{b}}{2} e^{i\mathbf{c}} & (\cos \frac{\mathbf{b}}{2})^2 e^{i(\mathbf{a}+\mathbf{c})} \end{pmatrix} \\ &= \begin{pmatrix} \frac{1 + \cos \mathbf{b}}{2} e^{-i(\mathbf{a}+\mathbf{c})} & -\frac{\sin \mathbf{b}}{\sqrt{2}} e^{-i\mathbf{c}} & \frac{1 - \cos \mathbf{b}}{2} e^{i(\mathbf{a}-\mathbf{c})} \\ \frac{\sin \mathbf{b}}{\sqrt{2}} e^{-i\mathbf{a}} & \cos \mathbf{b} & -\frac{\sin \mathbf{b}}{\sqrt{2}} e^{i\mathbf{a}} \\ \frac{1 - \cos \mathbf{b}}{2} e^{-i(\mathbf{a}-\mathbf{c})} & \frac{\sin \mathbf{b}}{\sqrt{2}} e^{i\mathbf{c}} & \frac{1 + \cos \mathbf{b}}{2} e^{i(\mathbf{a}+\mathbf{c})} \end{pmatrix} \end{aligned} \quad (1.50)$$

By comparing this expression with standard expressions and by taking into account the definition of the parameters \mathbf{a} , \mathbf{b} and \mathbf{c} , it appears that $\mathcal{D}^l(\mathbf{a}, \mathbf{b}, \mathbf{c})$ corresponds to $\mathcal{D}^l(R)$. In spin-space however, some phase factors need to be compensated because of factor $\frac{1}{2}$.

We have only discussed here the proper transformation. As we mentioned previously, a transformation can also be improper and can thus be written as the product of a proper rotation and a spatial inversion $I = -Id$. However, inversion I acts only on the \mathbf{r} -space and doesn't affect the spin-space. As a result, taking into account this transformation consists merely in adding a factor $(-1)^l$ in $\mathcal{D}^l(S)$ – and $\mathcal{D}^l(\mathbf{a}, \mathbf{b}, \mathbf{c})$ – for integer values of l , whereas no change in $\mathcal{D}^{\frac{1}{2}}(S)$ – and $\mathcal{D}^{\frac{1}{2}}(\mathbf{a}, \mathbf{b}, \mathbf{c})$ – occurs.

1.5.4 Putting the Wannier and Θ -projectors in the global frame

The coefficients $A_{lm}^{\nu\alpha}(\mathbf{k}, \sigma)$, $B_{lm}^{\nu\alpha}(\mathbf{k}, \sigma)$ and $C_{lm}^{\nu\alpha}(\mathbf{k}, \sigma)$ of (1.13) are defined in the *local coordinate system* associated to the atom α . However, to calculate local quantities, we have to perform a sum over the full Brillouin zone, which implies a symmetrization in the *global coordinate system* (cf. section 1.6). As a result, Wannier and Θ -projectors must be rotated to the *global coordinate system* too.

However, the transformation must be applied **without** taking into account the spinor rotation matrix $\mathcal{D}^{\frac{1}{2}}(R_{loc}^\alpha)$ and even when the calculation explicitly includes the spin-orbit coupling, to be consistent with the conventions used in `lapw2`. The expression is thus:

$$\left[P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}) \right]_{Global} = \sum_{m'} \mathcal{D}^l(R_{loc}^\alpha)_{m,m'} \left[P_{lm',\nu}^{\alpha,\sigma}(\mathbf{k}) \right]_{Local} \quad (1.51)$$

This step is done in the subroutine `rot_projectmat`.

1.5.5 Upfolding of the self-energy

The impurity problem in DMFT is solved for only one atom of a given sort in its local coordinate system. The impurity self-energy (in the *local coordinate system*) is the same among the equivalent atoms. However, in the upfolding procedure of (1.3), the impurity self-energy of each atom must be previously transformed from the local to the global frame. Finally, the upfolding procedure must thus be written:

$$\Sigma_{\nu\nu'}^\sigma(\mathbf{k}, i\omega_n) = \sum_{\alpha, mm'} \left[P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}) \right]^* \left[\sum_{nn'} \mathcal{D}^l(R_{loc}^\alpha)_{m,n} [\Delta\Sigma_{imp}^\alpha(i\omega_n)]_{nn'}^\sigma \mathcal{D}^l(R_{loc}^{\alpha^{-1}})_{n',m'} \right] P_{lm',\nu'}^{\alpha,\sigma}(\mathbf{k}) \quad (1.52)$$

Remark

If the spin-orbit coupling is taken into account, the upfolding procedure of (1.37) is also modified and becomes:

$$\Sigma_{\nu\nu'}^{ij}(\mathbf{k}, i\omega_n) = \sum_{\alpha, mm'} \sum_{\sigma\sigma'} \left[\mathbf{P}_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}) \right]^{i*} \left[\Delta\Sigma_{imp}^{\alpha Global}(i\omega_n) \right]_{mm'}^{\sigma,\sigma'} \left[\mathbf{P}_{lm',\nu'}^{\alpha,\sigma'}(\mathbf{k}) \right]^j \quad (1.53)$$

with

$$\left[\Delta\Sigma_{imp}^{\alpha Global}(i\omega_n) \right]_{mm'}^{\sigma,\sigma'} = \sum_{nn', \tau\tau'} \mathcal{D}^l(R_{loc}^\alpha)_{m,n} \mathcal{D}^{\frac{1}{2}}(R_{loc}^\alpha)_{\sigma,\tau} \left[\Delta\Sigma_{imp}^\alpha(i\omega_n) \right]_{nn'}^{\tau\tau'} \mathcal{D}^l(R_{loc}^{\alpha^{-1}})_{n',m'} \mathcal{D}^{\frac{1}{2}}(R_{loc}^{\alpha^{-1}})_{\tau',\sigma'}$$

1.6 Local quantities and sum over the irreducible Brillouin zone

1.6.1 Space groups and time-reversal operator

One introduces the “*crystallographic space group*” of a crystal as the set of all the symmetry operations \mathcal{T} which leave the system invariant after applying any of them to it. These operations have the following form:

$$\forall \mathbf{r} \in \mathbb{R}^3 \quad \mathcal{T}(\mathbf{r}) = \mathcal{R}(\mathbf{r}) + \mathbf{v} \quad (1.54)$$

where \mathbf{v} is a translation vector which belongs to the Bravais lattice or is a rational fraction of such a vector and \mathcal{R} is an operator of the *crystallographic point group*: a rotation through integral multiple

of $\frac{2\pi}{n}$ about some axis, a rotation-reflection, a rotation-inversion, a reflection or an inversion.

In addition, to describe the symmetry of magnetic crystals, one uses the *Shubnikov magnetic point groups*. In this case, the operation which reverses the magnetic moment is the *time reversal* operation Θ . If \mathcal{G} denotes one of the ordinary *crystallographic point groups*, there are three types of Shubnikov group corresponding to it:

- **type I:** the “*uncolored*” or standard point group \mathcal{G} ;
- **type II:** the “*grey*” point group defined as $\mathcal{G} + \Theta\mathcal{G}$
- **type III:** the “*black and white*” point group which is given by: $\mathcal{H} + \Theta(\mathcal{G} \setminus \mathcal{H})$, where \mathcal{H} is a halving subgroup of \mathcal{G} .

In Wien2k, a “*grey*” (type II) point group is always used when performing a *paramagnetic* computation and a “*black and white*” (type III) point group is always considered when the calculation is *spin-polarized*. The same conventions are applied when implementing the program `dmftproj`.

The time reversal operation Θ is an antilinear operator. In the basis $|\mathbf{r}, lm, \sigma\rangle$, the expression of Θ is:

$$\Theta = -i\sigma_y.K \quad \text{where} \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad (1.55)$$

and K is the conjugation operator in this same basis, or in other words:

$$\forall |\mathbf{r}, lm, \sigma\rangle \quad \forall \lambda \quad K(\lambda|\mathbf{r}, lm, \sigma\rangle) = \lambda^*(-1)^m \delta_{m,-m'} |\mathbf{r}, lm', \sigma\rangle \quad (1.56)$$

It is possible to use a matrix representation for $|\Theta$ or any antilinear operator:

$$\mathcal{D}^l(\Theta)_{mm'} = (-1)^m \delta_{m,-m'} \quad \text{and} \quad \mathcal{D}^{\frac{1}{2}}(\Theta) = -i\sigma_y$$

In this case, it is necessary to impose the following convention: *all the matrices at the right-hand side of the “antilinear” matrix should be complex conjugate*. This implies some minor changes to take into account when changing the basis or defining the Hermitian adjoint.

1.6.2 Performing the Brillouin zone integration

We consider a quantity \mathcal{O} which commutes with any symmetry operation of the system $[\mathcal{O}, \mathcal{S}] = 0$ and has the following form in the Bloch basis $|\psi_{\mathbf{k}\nu}^\sigma\rangle$:

$$\langle \psi_{\mathbf{k}\nu}^\sigma | \mathcal{O} | \psi_{\mathbf{k}'\nu'}^{\sigma'} \rangle = \mathcal{O}_{\nu\nu'}^{\sigma\sigma'}(\mathbf{k}) \delta_{\mathbf{k}\mathbf{k}'}, \quad (1.57)$$

where σ is either the spin degree of freedom (if the calculation does not include spin-orbit coupling) or $\sigma = \pm$ otherwise. This is particularly the case of the Hamiltonian H , the Green function G or the charge density ρ . We would like to compute the value of the elements $(\mathcal{O}_l^\alpha)^{\sigma\sigma'}$ in the local basis $|w_{lm}^{\alpha\sigma}\rangle$ of Wannier functions:

$$(\mathcal{O}_l^\alpha)^{\sigma\sigma'} = \langle w_{lm}^{\alpha\sigma} | \mathcal{O} | w_{lm}^{\alpha\sigma'} \rangle \quad (1.58)$$

Case without spin-orbit coupling

In this case, the spin σ is a good quantum number and using the shortcuts $(\mathcal{O}_l^{\alpha,\sigma})_{mm'} = (\mathcal{O}_l^\alpha)^{\sigma\sigma}$ and $\mathcal{O}_{\nu\nu'}^\sigma(\mathbf{k}) = \mathcal{O}_{\nu\nu'}^{\sigma\sigma}(\mathbf{k})$, we have:

$$(\mathcal{O}_l^\alpha)^{\sigma\sigma'} = (\mathcal{O}_l^{\alpha,\sigma})_{mm'} \cdot \delta_{\sigma\sigma'} = \sum_{\mathbf{k} \in 1BZ} \sum_{\nu, \nu'} P_{lm, \nu}^{\alpha, \sigma}(\mathbf{k}) \mathcal{O}_{\nu\nu'}^\sigma(\mathbf{k}) \left[P_{lm', \nu'}^{\alpha, \sigma}(\mathbf{k}) \right]^* \cdot \delta_{\sigma\sigma'} \quad (1.59)$$

A sum over the \mathbf{k} -points in the first Brillouin-zone (1BZ) is necessary. However, in order to reduce the computational time, Wien2k solves the Kohn-Sham equations for the \mathbf{k} -points in the irreducible Brillouin zone (IBZ). As a result, we calculate first the unsymmetrized quantity in IBZ $[(\mathcal{O}_l^{\alpha,\sigma})_{mm'}]_{unsym}$:

$$[(\mathcal{O}_l^{\alpha,\sigma})_{mm'}]_{unsym} = \sum_{\mathbf{k} \in IBZ} \sum_{\nu,\nu'} P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}) \mathcal{O}_{\nu\nu'}^\sigma(\mathbf{k}) \left[P_{lm',\nu'}^{\alpha,\sigma}(\mathbf{k}) \right]^* \quad (1.60)$$

- **if the compound is *paramagnetic*:** (magnetic point group $\mathcal{G} + \Theta\mathcal{G}$)

We apply a symmetrization with respect to all the operations \mathcal{S} of the crystallographic space group \mathcal{G} :

$$[(\mathcal{O}_l^{\alpha,\sigma})_{mm'}]_{\mathcal{G}} = \sum_{\mathcal{S} \in \mathcal{G}} \sum_{n,n'} \mathcal{D}^l(\mathcal{S})_{m,n} \left[(\mathcal{O}_l^{S^{-1}\alpha, S^{-1}\sigma})_{nn'} \right]_{unsym} \mathcal{D}^l(\mathcal{S}^{-1})_{n',m'} \quad (1.61)$$

and then a symmetrization with respect to the time-reversal operator Θ to get $(\mathcal{O}_l^{\alpha,\sigma})_{mm'}$:

$$(\mathcal{O}_l^{\alpha,\sigma})_{mm'} = \frac{1}{2} \left[[(\mathcal{O}_l^{\alpha,\sigma})_{mm'}]_{\mathcal{G}} + \sum_{n,n'} \mathcal{D}^l(\Theta)_{m,n} [(\mathcal{O}_l^{\alpha,\sigma})_{nn'}]_{\mathcal{G}}^* \left[\mathcal{D}^l(\Theta^{-1})_{n',m'} \right]^* \right] \quad (1.62)$$

- **if the calculation is *spin-polarized*:** (magnetic point group $\mathcal{H} + \Theta(\mathcal{G} \setminus \mathcal{H})$)

By assuming that the magnetization axis of the compound lies along the \mathbf{z} -axis, the halving subgroup \mathcal{H} of \mathcal{G} is defined as the symmetry operations which do not change the orientation of the magnetization and thus keep the \mathbf{z} -axis invariant:

$$\mathcal{S} \in \mathcal{H} \iff \mathcal{S}\sigma = \sigma \quad \text{and} \quad \mathcal{R} \in \mathcal{G} \setminus \mathcal{H} \iff \mathcal{R}\sigma = -\sigma$$

Since $\Theta\sigma = -\sigma$ too, we finally can write:

$$\begin{aligned} (\mathcal{O}_l^{\alpha,\sigma})_{mm'} &= \sum_{\mathcal{S} \in \mathcal{H}} \sum_{n,n'} \mathcal{D}^l(\mathcal{S})_{m,n} \left[(\mathcal{O}_l^{S^{-1}\alpha, S^{-1}\sigma})_{nn'} \right]_{unsym} \mathcal{D}^l(\mathcal{S}^{-1})_{n',m'} \\ &+ \sum_{\mathcal{R} \in \mathcal{G} \setminus \mathcal{H}} \sum_{n,n'} \mathcal{D}^l(\Theta\mathcal{R})_{m,n} \left[(\mathcal{O}_l^{\mathcal{R}^{-1}\alpha, \sigma})_{nn'} \right]_{unsym}^* \left[\mathcal{D}^l(\mathcal{R}^{-1}\Theta^{-1})_{n',m'} \right]^* \end{aligned} \quad (1.63)$$

Case with spin-orbit coupling

In this case, the spin indices must be explicitly treated and:

$$(\mathcal{O}_l^{\alpha})_{mm'}^{\sigma\sigma'} = \sum_{\mathbf{k} \in IBZ} \sum_{\nu,\nu'} \sum_{i,j} \left[P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}) \right]^i \mathcal{O}_{\nu\nu'}^{ij}(\mathbf{k}) \left[P_{lm',\nu'}^{\alpha,\sigma'}(\mathbf{k}) \right]^j^* \quad (1.64)$$

In a similar fashion, we calculate first the unsymmetrized quantity in IBZ $[(\mathcal{O}_l^{\alpha})_{mm'}^{\sigma\sigma'}]_{unsym}$:

$$\left[(\mathcal{O}_l^{\alpha})_{mm'}^{\sigma\sigma'} \right]_{unsym} = \sum_{\mathbf{k} \in IBZ} \sum_{\nu,\nu'} \sum_{i,j} \left[P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}) \right]^i \mathcal{O}_{\nu\nu'}^{ij}(\mathbf{k}) \left[P_{lm',\nu'}^{\alpha,\sigma'}(\mathbf{k}) \right]^j^* \quad (1.65)$$

- **if the compound is *paramagnetic*:** (magnetic point group $\mathcal{G} + \Theta\mathcal{G}$)

We apply a symmetrization with respect to all the operations \mathcal{S} of the crystallographic space group \mathcal{G} :

$$\left[(\mathcal{O}_l^\alpha)^{\sigma\sigma'} \right]_{mm'}^{\mathcal{G}} = \sum_{\mathcal{S} \in \mathcal{G}} \sum_{n,n'} \sum_{\tau,\tau'} \mathcal{D}^l(\mathcal{S})_{m,n} \mathcal{D}^{\frac{1}{2}}(\mathcal{S})_{\sigma,\tau} \left[(\mathcal{O}_l^{\mathcal{S}^{-1}\alpha})_{nn'}^{\tau\tau'} \right]_{\text{unsym}} \mathcal{D}^l(\mathcal{S}^{-1})_{n',m'} \mathcal{D}^{\frac{1}{2}}(\mathcal{S}^{-1})_{\tau',\sigma'} \quad (1.66)$$

and then a symmetrization with respect to the time-reversal operator Θ , as follows:

$$\begin{aligned} (\mathcal{O}_l^\alpha)^{\uparrow\uparrow}_{mm'} &= \frac{1}{2} \left[\left[(\mathcal{O}_l^\alpha)^{\uparrow\uparrow}_{mm'} \right]^{\mathcal{G}} + \sum_{n,n'} \mathcal{D}^l(\Theta)_{m,n} \left[(\mathcal{O}_l^\alpha)^{\downarrow\downarrow}_{nn'} \right]^{\mathcal{G}*} \left[\mathcal{D}^l(\Theta^{-1})_{n',m'} \right]^* \right] \\ (\mathcal{O}_l^\alpha)^{\downarrow\downarrow}_{mm'} &= \frac{1}{2} \left[\left[(\mathcal{O}_l^\alpha)^{\downarrow\downarrow}_{mm'} \right]^{\mathcal{G}} + \sum_{n,n'} \mathcal{D}^l(\Theta)_{m,n} \left[(\mathcal{O}_l^\alpha)^{\uparrow\uparrow}_{nn'} \right]^{\mathcal{G}*} \left[\mathcal{D}^l(\Theta^{-1})_{n',m'} \right]^* \right] \\ (\mathcal{O}_l^\alpha)^{\uparrow\downarrow}_{mm'} &= \frac{1}{2} \left[\left[(\mathcal{O}_l^\alpha)^{\uparrow\downarrow}_{mm'} \right]^{\mathcal{G}} - \sum_{n,n'} \mathcal{D}^l(\Theta)_{m,n} \left[(\mathcal{O}_l^\alpha)^{\downarrow\uparrow}_{nn'} \right]^{\mathcal{G}*} \left[\mathcal{D}^l(\Theta^{-1})_{n',m'} \right]^* \right] \end{aligned} \quad (1.67)$$

- **if the calculation is *spin-polarized*:** (magnetic point group $\mathcal{H} + \Theta(\mathcal{G} \setminus \mathcal{H})$)

With \mathcal{H} the halving subgroup of \mathcal{G} defined as the symmetry operations which do not change the orientation of the magnetization, we can write:

$$\begin{aligned} (\mathcal{O}_l^\alpha)^{\uparrow\uparrow}_{mm'} &= \sum_{\mathcal{S} \in \mathcal{G}} \sum_{n,n'} \mathcal{D}^l(\mathcal{S})_{m,n} \left[(\mathcal{O}_l^{\mathcal{S}^{-1}\alpha})_{nn'}^{\uparrow\uparrow} \right]_{\text{unsym}} \mathcal{D}^l(\mathcal{S}^{-1})_{n',m'} \\ &\quad + \sum_{\mathcal{R} \in (\mathcal{G} \setminus \mathcal{H})} \sum_{n,n'} \mathcal{D}^l(\Theta\mathcal{R})_{m,n} \left[(\mathcal{O}_l^{\mathcal{R}^{-1}\alpha})_{nn'}^{\uparrow\uparrow} \right]_{\text{unsym}}^* \left[\mathcal{D}^l(\mathcal{R}^{-1}\Theta^{-1})_{n',m'} \right]^* \\ (\mathcal{O}_l^\alpha)^{\downarrow\downarrow}_{mm'} &= \sum_{\mathcal{S} \in \mathcal{G}} \sum_{n,n'} \mathcal{D}^l(\mathcal{S})_{m,n} \left[(\mathcal{O}_l^{\mathcal{S}^{-1}\alpha})_{nn'}^{\downarrow\downarrow} \right]_{\text{unsym}} \mathcal{D}^l(\mathcal{S}^{-1})_{n',m'} \\ &\quad + \sum_{\mathcal{R} \in (\mathcal{G} \setminus \mathcal{H})} \sum_{n,n'} \mathcal{D}^l(\Theta\mathcal{R})_{m,n} \left[(\mathcal{O}_l^{\mathcal{R}^{-1}\alpha})_{nn'}^{\downarrow\downarrow} \right]_{\text{unsym}}^* \left[\mathcal{D}^l(\mathcal{R}^{-1}\Theta^{-1})_{n',m'} \right]^* \\ (\mathcal{O}_l^\alpha)^{\uparrow\downarrow}_{mm'} &= \sum_{\mathcal{S} \in \mathcal{G}} \sum_{n,n'} \mathcal{D}^l(\mathcal{S})_{m,n} \left[(\mathcal{O}_l^{\mathcal{S}^{-1}\alpha})_{nn'}^{\uparrow\downarrow} \right]_{\text{unsym}} \mathcal{D}^l(\mathcal{S}^{-1})_{n',m'} e^{-i(\alpha+\gamma)} \\ &\quad + \sum_{\mathcal{R} \in (\mathcal{G} \setminus \mathcal{H})} \sum_{n,n'} \mathcal{D}^l(\Theta\mathcal{R})_{m,n} \left[(\mathcal{O}_l^{\mathcal{R}^{-1}\alpha})_{nn'}^{\uparrow\downarrow} \right]_{\text{unsym}}^* \left[\mathcal{D}^l(\mathcal{R}^{-1}\Theta^{-1})_{n',m'} \right]^* e^{i(\alpha-\gamma)} \end{aligned} \quad (1.68)$$

where α and γ in the exponential are the Euler angles of the considered symmetry operation \mathcal{S} or \mathcal{R} .

1.6.3 Application 1: calculation of the local density

The program `dmftproj` allows to calculate the local density matrix on a given atomic sort out of the Wannier and Θ -projectors. This step is performed by the subroutine `density`, which first computes the unsymmetrized quantity:

$$\left[(\mathcal{O}_l^{\alpha,\sigma})_{mm'} \right]_{\text{unsym}} = \sum_{\mathbf{k}' \in \text{IBZ}} \sum_{\nu} P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}') \left[\omega_{\nu}^{\sigma}(\mathbf{k}') \right] \left[P_{lm',\nu}^{\alpha,\sigma}(\mathbf{k}') \right]^* \quad (1.69)$$

In this expression, \mathbf{k}' samples a tetrahedral mesh in IBZ and $\omega_\nu^\sigma(\mathbf{k}')$ represents its corresponding weight, depending on the band index (ν, σ) . More precisely, its value is smaller for the bands which cross the Fermi level and it is 0 if the band is empty.

The symmetrization step (1.61) or (1.63) is performed by the inner subroutine `symmetrize_mat`. For a paramagnetic case, the time reversal operation is applied globally (1.62) by the subroutine `add_timeinv`; for a spin-polarized calculation, it is applied to any symmetry operation of \mathcal{H} in `setsym`.

If the spin-orbit coupling is taken into account, the matrix density is calculated in density from the unsymmetrized quantity:

$$\left[(\mathcal{O}_l^\alpha)_{mm'}^{\sigma\sigma'} \right]_{unsym} = \sum_{\mathbf{k}' \in IBZ} \sum_{\nu} \sum_{i,j} \left[P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}') \right]^i \left[\omega_\nu(\mathbf{k}') \right] \left[P_{lm',\nu}^{\alpha,\sigma'}(\mathbf{k}') \right]^j{}^*. \quad (1.70)$$

The symmetrization step (1.68) is performed in `symmetrize_mat`, the time reversal operation being applied to any symmetry operation of \mathcal{H} in `setsym`.³

All the previous calculations are performed in the global system coordinates. Before displaying the results, the density matrix is put back to the local coordinate system of the corresponding atom:

$$\left[(\mathcal{O}_l^\alpha)_{mm'}^{\sigma\sigma'} \right]_{Local} = \sum_{n,n'} \sum_{\tau,\tau'} \mathcal{D}^l(R_{loc}^{\alpha-1})_{m,n} \mathcal{D}^{\frac{1}{2}}(R_{loc}^{\alpha-1})_{\sigma,\tau} \left[(\mathcal{O}_l^\alpha)_{nn'}^{\tau\tau'} \right]_{Global} \mathcal{D}^l(R_{loc}^\alpha)_{n',m'} \mathcal{D}^{\frac{1}{2}}(R_{loc}^\alpha)_{\tau',\sigma'} \quad (1.71)$$

This step is performed in the subroutine `rot_dens`.

Remark

- If the operation $R_{loc}^\alpha = R_{loc}^A \cdot \mathcal{T}$ is antiunitary (which may occurs when \mathcal{T} is in the subgroup \mathcal{H}), the last three terms in (1.71) must be complex conjugate.
- The density matrices can also be calculated out of the Θ -projectors. The formula are the same as those previously described if $P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}')$ are replaced by $\Theta_{lm,\nu j}^{\alpha,\sigma}(\mathbf{k}')$

1.6.4 Application 2: calculation of the local Green function

To estimate the local Green function out of the lattice Green function in the DMFT loop, an integration over the first Brillouin Zone is required (cf. equation (1.2)). The approach described in 1.6.2 holds also in this case. The unsymmetrized quantity has then the following form:

$$\left[(\mathcal{O}_l^{\alpha,\sigma})_{mm'} \right]_{unsym} = \sum_{\mathbf{k}' \in IBZ} \sum_{\nu\nu'} P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}') \left[G_{\nu\nu'}^\sigma(\mathbf{k}', i\omega_n) \omega(\mathbf{k}') \right] \left[P_{lm',\nu'}^{\alpha,\sigma}(\mathbf{k}') \right]^* \quad (1.72)$$

or with spin-orbit

$$\left[(\mathcal{O}_l^\alpha)_{mm'}^{\sigma\sigma'} \right]_{unsym} = \sum_{\mathbf{k}' \in IBZ} \sum_{\nu\nu'} \sum_{ij} \left[P_{lm,\nu}^{\alpha,\sigma}(\mathbf{k}') \right]^i \left[G_{\nu\nu'}^{ij}(\mathbf{k}', i\omega_n) \omega(\mathbf{k}') \right] \left[P_{lm',\nu'}^{\alpha,\sigma'}(\mathbf{k}') \right]^j{}^*$$

where $G_{\nu\nu'}^\sigma(\mathbf{k}', i\omega_n)$ is still defined by (1.1) and $G_{\nu\nu'}^{ij}(\mathbf{k}', i\omega_n)$ by (1.35). \mathbf{k}' samples the tetrahedral mesh in IBZ and $\omega(\mathbf{k}')$ is merely the geometric factor corresponding to a tetrahedron for fully occupied bands.⁴

To perform the symmetrization, all the required data are provided in the output files `case.ctqmcout` and `case.symqmc` (cf. section 2.4), which are generated by the subroutine `outputqmc`.

³We remind that the spin-orbit calculation is applied only in the spin-polarized case in the current version.

⁴Since the chemical potential will change in the DMFT cycle, the calculated $\omega_\nu^\sigma(\mathbf{k}')$ with LDA cannot be used.

Chapter 2

How to install and run the program

2.1 How to install dmftproj

Since version 11.1 of Wien2k, all necessary routines are included in the Wien2k package. Older version would require some changes in the SRC_lapw2 part, so we highly recommend an update to the latest version of Wien2k.

The dmftproj program itself is part of the TRIQS package, and automatically compiled in the installation process.

2.2 Execution of the program dmftproj

In this part, we assume that the self-consistent cycle in Wien2k was run up to convergence (with the line command `run_lapw` or `runsp_lapw [-so]`) and that the master input file `case.indmftpr` was edited in the running directory.

2.2.1 In order to perform a DMFT calculation

Before running dmftproj, the coefficients $A_{lm}^{\nu\alpha}(\mathbf{k}, \sigma)$, $B_{lm}^{\nu\alpha}(\mathbf{k}, \sigma)$ and $C_{lm}^{\nu\alpha}(\mathbf{k}, \sigma)$ of (1.13) and some other general informations on the system (symmetry operations and local rotations) must be calculated. The lapw2 program has thus to be executed in the following way:

```
x lapw2 -almd [-up/-dn -c -so]
```

The new flag “-almd” is written in the first line of `case.in2(c)` so that the input files (`case.almb1m(up/dn)` and `case.dmftsym`) for dmftproj are produced. The master input file required by dmftproj is called `case.indmftpr` and will be described in the next section. The program dmftproj should be executed as follows:

```
dmftproj [-sp] [-so]
```

The options stand for spin-polarized (-sp) calculations and for calculation which takes into account the SO interaction (-so). For technical reasons, the SO coupling is however implemented for the spin-polarized case only.

2.2.2 In order to calculate a momentum-resolved spectral function

To produce the output needed to calculate the momentum-resolved spectral function after a DMFT calculation was performed, one must first prepare the desired \mathbf{k} -point path for $A(k, \omega)$ in a `case.klist_band` file. Then, the same procedure as used in Wien2k to calculate Kohn-Sham band structure must be applied:

```
x lapw1 [-up/-dn-so] -band
x lapw2 -almd [-up/-dn-so] -band
```

It is then enough to execute the following command in the same directory:

```
dmftproj [-sp] [-so] -band
```

2.3 Description of the master input file `case.indmftpr`

The master input file `case.indmftpr` of `dmftproj` has the following structure. We give an example of such a file in the case of SrVO_3 :

a)	3	! nsort
	1 1 3	! mult(isrt)
	3	! lmax

b-1)	complex	! basis
	1 0 0 0	! lsort(1)
	0 0 0 0	! nb_irep(1)

b-2)	cubic	! basis
	1 1 2 0	! lsort(1)
	0 0 2 0	! nb_irep(1)
	01	
	0	! ifSO

b-3)	cubic	! basis
	1 1 0 0	! lsort(1)
	0 0 0 0	! nb_irep(1)

c)	-0.075 0.14	! e_bot and e_top
	0.59953	! e_Fermi

Part a)

This part contains the general description for the considered material (here Sr_2RuO_4):

nsort: the number (integer) of inequivalent atomic sort in the system;
in SrVO_3 , it is 3 because there are Sr, V and O.

mult(isrt): nsort integers equal to the multiplicity of each atomic sort in the unit cell. The order for the description must be the same as in the `case.struct` file;
in SrVO_3 , there are 1 Sr, 1 V and 3 O.

lmax: maximal l number (integer) considered in the system;
the value here is 3 because we consider only the s,p,d and f shells.

Part b)

Part b) describes the chosen “treatment” for each atomic sort. It must then be repeated `nsort` time. Moreover, the order for the description must be the same as in the `case.struct` file.

basis: name associated to the description of the local basis (for the projector). It can be:

- *complex* for complex spherical harmonics,
- *cubic* for cubic spherical harmonics.
- *fromfile* for a basis described in an added file. In this last case, the complete name of the file must be written the line just after:

```
-----  
fromfile    ! basis  
file_name  
1 1 2 0    ! lsort(1)  
-----
```

In this here-above example, the file is called “`file_name`”. The name of the file must have a size of *less than 25 characters*. This file must be in the directory where the computation is performed. It must contain the description of the basis for all the included orbital (*Here, for the p and d shells only.*). Indeed, the s-orbital does not need a basis description in this file, if it is included. No newline for the separation of the basis description is needed.

lsort(l): (`lmax+1`) integers to describe the “treatment” of each orbital:

- 0 means that the corresponding orbital is not included,
- 1 means that the orbital is included but no projector $P_{lm,\nu}^{\alpha\sigma}(\mathbf{k})$ will be calculated for it. (this corresponds to an orbital “significantly present” in the considered energy window but which do not require a DMFT calculation to treat its correlation level),
- 2 means that the orbital is included and a projector $P_{lm,\nu}^{\alpha\sigma}(\mathbf{k})$ will be calculated for it. (this typically corresponds to an orbital “significantly present” in the considered energy window and *correlated*.)

nb_irep(l): (`lmax+1`) integers to describe the number of irreducible representation (`irep`) to consider for each orbital `l`. This description is read (and used) only if the corresponding `lsort(l)=2`. only *one* particular `irep` can be “*correlated*”. If the number of `irep` is more than 2 for one orbital, the line after must describe the treatment of each `irep` (`nb_irep(l)` number in this line) with a flag (0/1) (cf. case b-2):

- 0 means that the orbital is not *correlated* ; no projector $P_{lm,\nu}^{\alpha\sigma}(\mathbf{k})$ will be calculated for it,
- 1 means that the orbital is *correlated* and a projector $P_{lm,\nu}^{\alpha\sigma}(\mathbf{k})$ will be calculated for it.

ifSO: a flag (0/1) which states if the spin-orbit coupling is taken into account for this atomic sort. This flag is necessary *only if* one of the orbital of the sort is *correlated*. That is why this line does not appear in the section b-1 and b-3.

Part c)

This part finally precises the parameters for the computation of the projectors:

e_bot and **e_top** define the size of the energy window (given in Rydberg and relatively to the Fermi level);
in SrVO₃, we choose here [-0.075;0.112] Ry.

e_Fermi is the Fermi level obtained at the end of the DFT-LDA calculation. This line is read only when one wants to calculate the momentum-resolved spectral function for the system;
in SrVO₃, the LDA-Fermi level is 0.59884 Ry.

Further remarks

The number of irreducible representation depends on the chosen local basis.

- If one considers the *complex basis*, there is only one possible, which is the complete one, or in other words `irep=0` necessarily;
- If one considers the *cubic basis*, d ($l = 2$) and f ($l = 3$) shells do have irreducible representations (in the case without spin-orbit). For instance, in the case $l = 2$, it is possible to specify whether the e_g or the t_{2g} manifolds is correlated. Then one has to put `irep=2` and write “10” (or “01”) in the next line if e_g (or t_{2g} , respectively) are correlated (cf. part b-2).
- When one uses the option `fromfile`, several `irep` may be considered even if the spin-orbit coupling is included.

The additional file which describes a basis must be written with usual Wien2k conventions. We display here the file `case.cf_p_rel` as an example:

-0.81649658	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.57735027	0.0	0.0	0.0
* 0.0	0.0	-0.57735027	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.81649658	0.0
0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
0.57735027	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.81649658	0.0	0.0	0.0
0.0	0.0	0.81649658	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.57735027	0.0	0.0
* 0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

This file describe the decomposition of the spin-orbitals with $j = 1/2$ and $j = 3/2$ in the complex basis set. The `irep` are indicated by a star (*) at the beginning of the line where the last vector of the `irep` is given. *Here, there are two irep of respective size 2 ($j = 1/2$) and 4 ($j = 3/2$).*

2.4 Description of the output files

2.4.1 Main output file `case.outdmftpr`

The file `case.outdmftpr` is the main output file. We display here as an example the one obtained for SrVO_3 when using the previously described `case.indmftpr` file.

- a) **Introduction:** the option used are reminded (`-so`, `-sp`, `-band`). *Here we used the simple line command: `dmftproj`*

```

Welcome in DMFTPROJ: PROJECTION TO LOCALIZED BASIS

This prgm will build the Wannier projectors to the
localized orbitals of an atom onto which DMFT will be applied.

You are performing a computation
without Spin-Orbit.
using Paramagnetic Wien2k input files.

```

- b) **Description of the included orbitals:** For each atomic sort, the included orbitals (with 1 or 2 in `case.indmftpr`) are reminded. At the end, the (upper and lower) limits of the energy window can be read.

```

-----
Sorts of atoms = 3
Equivalent sites per each sort: 1 1 3
-----
For the sort 1 :
The orbital l= 0 is included.
-----
For the sort 2 :
The orbital l= 0 is included.
The orbital l= 1 is included.
The orbital l= 2 is included.
-----
For the sort 3 :
The orbital l= 0 is included.
The orbital l= 1 is included.

The Eigenstates are projected in an energy window from -0.07500 Ry to 0.14000 Ry around
the Fermi level.
-----

```

c) Description of the basis representations: For each included orbital, the basis representation used is reminded. When considering the cubic basis or a basis described in an additional file, the program displays the number of irreducible representations (**irep**) and their respective dimension (**degree**) but also the transformation matrix (from the complex basis to the one considered). *In the example displayed, the description of the cubic d-orbital of atomic sort 2 was not displayed.*

```

-----
Basis representation for each sort.
-----
For the sort 1 :
The atomic sort 1 is studied in the complex basis representation.
-----
For the sort 2 :
The atomic sort 2 is studied in the cubic basis representation.

The basis for s-orbital is still 1.000000
The basis for orbital l= 1 has the following properties :
- number of ireps : 1
- degree of each ireps : 3
The transformation matrix is block diagonal in the spin-space. The up/up and down/down
blocks are the same and defined as :
  0.707107  0.000000  0.000000  0.707107  0.000000  0.000000
  0.000000  0.000000  0.000000  0.000000  1.000000  0.000000
 -0.707107  0.000000  0.000000  0.707107  0.000000  0.000000

similar description for the cubic basis for orbital l= 2
-----
For the sort 3 :
The atomic sort 3 is studied in the complex basis representation.
-----

```

d) Description of the correlated orbitals: For each atomic sort, the correlated orbitals or the correlated irep (with 2 in case.indmftpr) are reminded.

Precisions about correlated orbitals.

For the sort 1 :
No orbital is included as correlated.

For the sort 2 :
The irep 2 of orbital l= 2 is considered as correlated.

For the sort 3 :
No orbital is included as correlated.

- e) **Description of the symmetry operations and local rotations:** The symmetry operations of the system are described by giving their Euler angles and their determinant. Similarly, the Global-to-local rotations for each atomic sort are also described. These data are read from the case.dmftsym file.

Symmetry operations of the system

Number of Symmetries = 48

Properties of the symmetry operations :
alpha, beta, gamma are their Euler angles.
iprop is the value of their determinant.

SYM.OP.	alpha	beta	gamma	iprop
1	0.0	0.0	0.0	-1
2	180.0	0.0	0.0	1
3	270.0	90.0	270.0	1
⋮	⋮	⋮	⋮	⋮

Global-to-local-coordinates rotations

Properties of the symmetry operations :
alpha, beta, gamma are their Euler angles.
iprop is the value of their determinant.

SORT	alpha	beta	gamma	iprop
1	0.0	0.0	0.0	1
2	0.0	0.0	0.0	1
3	180.0	90.0	180.0	1

- f) **Reading of the coefficients Alm, Blm:** The file(s) case.almbm(up/dn) is (are) read and the Fermi level of the system is displayed.

Reading of the file srvo3.almbm

The value of the Fermi Energy is 0.59953 Ry.
All the considered energies are now given with respect to this value. (E_Fermi is now 0 Ry)

- g) **Computation of the density matrices:** The first step of dmftproj is to calculate the density matrices for all the included orbitals of the system, using the “ Θ -projectors” without any restriction on the energy window. (The results are the same as those obtained by running x lapwdm) For the correlated orbitals, the density matrices are also calculated by considering only the part $u_l^{\alpha,\sigma}(E_{1l}^\alpha)Y_m^l$ of the LAPW basis: it enables to estimate how “localized” the considered correlated orbitals is. *In the following example, we have only displayed the density matrices of the V-3d states. These states are really well-localized in the muffin-tin spher, that is why the two matrices are almost equal.*

```

-----
Computation of the Occupancies and Density Matrices up to E_Fermi

Creation of the projectors...
Evaluation of the density matrices...
-----
Density Matrices for the Correlated States :
Sort = 2 Atom = 2 and Orbital l = 2

0.414  0.000  -0.000  0.000  0.000  0.000  -0.000  0.000  0.000  0.000
-0.000  0.000  0.414  0.000  0.000  0.000  0.000  0.000  0.000  0.000
0.000  -0.000  0.000  -0.000  0.467  0.000  0.000  -0.000  -0.000  0.000
-0.000  0.000  0.000  0.000  0.000  0.000  0.467  0.000  0.000  -0.000
0.000  -0.000  0.000  -0.000  -0.000  0.000  0.000  0.000  0.467  0.000
The charge of the orbital is : 2.23058

-----
Density Matrices for all the States of the System :
Sort = 2 Atom = 2 and Orbital l = 2

0.418  0.000  -0.000  0.000  0.000  0.000  -0.000  0.000  0.000  0.000
-0.000  0.000  0.418  0.000  0.000  0.000  0.000  0.000  0.000  0.000
0.000  -0.000  0.000  -0.000  0.469  0.000  0.000  -0.000  -0.000  0.000
-0.000  0.000  0.000  0.000  0.000  0.000  0.469  0.000  0.000  -0.000
0.000  -0.000  0.000  -0.000  -0.000  0.000  0.000  0.000  0.469  0.000
The charge of the orbital is : 2.24519
-----

```

h) Computation of the density matrices below the energy window: The second step of the dmftproj is to calculate the density matrices for all the included orbitals of the system below the lower limit of the energy window. However, only the total charge below the energy window is displayed, as seen below.

```

-----
Computation of the total Charge below the lower limit of the energy window : -0.07500 Ry

Creation of the projectors...
Evaluation of the density matrices...

TOTAL CHARGE = 40.00000
-----

```

i) Computation of the Wanniers projectors: The third and last step of dmftproj is to calculate the Wannier projectors (in the desired energy window). The corresponding density matrix based on them is displayed. *For SrVO₃ and with our choice for the energy window (which contains only the t_{2g} bands), we get one electron in the window, the three t_{2g} bands being equally filled.* Finally, the output files are written.

```

-----
Computation of the Occupancies and Density Matrices in the desired energy window
[-0.07500 ; 0.14000]

Creation of the projectors...
Orthonormalization of the projectors...
Evaluation of the density matrices...
-----
Density Matrices for the Correlated States :
Sort = 2 Atom = 2 and Orbital l = 2

0.000  0.000  0.000  -0.000  0.000  -0.000  -0.000  -0.000  0.000  0.000
0.000  0.000  0.000  -0.000  0.000  0.000  0.000  0.000  -0.000  0.000
0.000  0.000  0.000  -0.000  0.333  0.000  0.000  -0.000  -0.000  0.000
-0.000  0.000  0.000  -0.000  0.000  0.000  0.333  -0.000  0.000  -0.000
0.000  -0.000  -0.000  -0.000  -0.000  -0.000  0.000  0.000  0.333  -0.000

The charge of the orbital is : 1.00053

Writing the file case.ctqmcout...
Writing the file case.symqmc...
Writing the file case.parproj...
Writing the file case.sympar...
Writing the file case.outbwin...

END OF THE PRGM
-----

```

2.4.2 Warning messages

The following Warning may appear in the file `case.outdmftpr`. It is *strongly* advised to modify some input parameters as described below, and rerun the calculation if they appear:

- **WARNING** : more than 1 irep is included as correlated. 2 irep at least are considered as correlated for an orbital (l,isrt). Choose only one of the irep or do not specify the number of irep to consider all of them as correlated.
- **WARNING**: An eigenvalue of the overlap matrix is negative. or **WARNING**: An eigenvalue of the overlap matrix is almost zero. The energy window used is too small to define the correlated subspace on which you want to make the projections. Increase the size of the energy window.

2.4.3 Error messages

In addition, `dmftproj` may interrupt with the following messages:

- The number of ireps considered for (isrt,l) is not possible. The number of ireps defined in the file `case.indmftpr` is greater than the number of possible suborbitals for the orbital (l,isrt).
- The energy window is ill-defined. The limits of the energy window are not correctly written in the file `case.indmftpr`.
- You must use Spin-Polarized input files to perform Spin-Orbit computation, with this version. The calculation was run with the option `-so` but without the option `-sp`.
- You must include at least one orbital. No orbital is included (with 1 or 2) in the file `case.indmftpr`.
- The number of ireps considered for the orbital (isrt,l) is wrong. The number of ireps defined in the file `case.indmftpr` is not consistent with the description of the cubic basis or the basis defined in the additional file.

- The current version of DMFTproj cannot be used with more than 1 LO orbital by atom. This is not the case for the orbital (isrt,l) The orbital (isrt,l) is decomposed in LAPW basis with more than 1 “LO orbital”.
- The subroutine zheev ends with info=i. In sqrt_eigenvec, a pbm occurs in zheev. The function zheev from common libraries was interrupted with “info=i” during an orthogonalization procedure.
- An eigenvalue of the overlap matrix has the value 0. The calculation can not be performed further. The projection on a Wannier state is 0. Either increase the size of the energy window or do not include the corresponding orbital for the projection.
- Command line option is not recognized. Only the options -sp, -so and -band may be used when running dmftproj.
- The basis description for (isrt,l) is not recognized. Check the structure of the file file_name An error occurs when reading the additional file file_name.
- The basis description for (isrt,l) mixes up and down states. This option can not be used in a computation without Spin-Orbit. Modify the structure of the file file_name. According to the matrix read in the file file_name, orbital states with up and dn spin are not similar.
- Characters required for the basis transformation file_name is too long. The name of the file file_name is too long.

2.4.4 Output files created by dmftproj to perform a DMFT calculation

The subroutine outputqmc.f of dmftproj produces the following four data files required to perform a DMFT calculation and some simple post-processing of the results:

- i) case.ctqmcout and case.symqmc: these files contain all the main data to perform the DMFT self-consistent loop.

More precisely, case.ctqmcout describes the number of k-points in IBZ, the total number of orbitals, and correlated orbitals, the multiplicity of each sort of atoms, the rotation matrices, the projectors $P_{lm,\nu}^{\alpha\sigma}(\mathbf{k})$ to correlated orbitals, the \mathbf{k} -point weights, the eigenvalues of the Hamiltonian in Kohn-Sham basis. case.symqmc contains the number of symmetry operations for the symmetrization, the corresponding permutation matrices, the rotations matrices $\mathcal{D}_{mm'}^l(R)$, and additional transformation if symmetry-adapted basis is required

- ii) case.parproj and case.sympar: these files contain the same informations as the previous ones but are used to recalculate partial quantities (for example the spectral density) after the DMFT calculation. Instead of describing the Wannier projectors, they give the projectors $\Theta_{lm\nu j}^{\alpha,\sigma}(\mathbf{k})$.

The subroutine outbwin.f produces the output file(s) case.outbwin(up/dn) required for the charge density recalculation by lapw2 (see chapter 3). Each file contains, for each \mathbf{k} -point the range of bands include to the window and the \mathbf{k} -point weight.

2.4.5 Output files created by dmftproj to calculate momentum-resolved spectral functions

The flag -band allows to produce the file case.outband which contains:

the projectors of the correlated orbitals, $P_{lm,\nu}^{\alpha\sigma}(\mathbf{k})$ for the new k-points, the corresponding eigenvalues, the projectors $\Theta_{lm\nu j}^{\alpha}(\mathbf{k})$, if some fatband character plotting are desired, information on the new \mathbf{k} -list (especially, the labels for special \mathbf{k} -points).

It is then possible to plot the momentum-resolved spectral function, to compare the result of the LDA+DMFT calculation with the experimental curves obtained with Angle-Resolved Photo-Emission Spectroscopy (ARPES).

Chapter 3

Implementation of self-consistency in Wien2k+DMFT

In this part of the notes we describe our new implementation of a fully self-consistent (SC) LDA+DMFT scheme in the Wien2k framework. The section 3.1 presents our approach to SC over the charge density within LDA+DMFT in LAPW/APW+lo and lists the formulas used for recalculations of the charge density and total energy from the DMFT occupancy matrix in the Bloch basis. Section 3.2 describes the actual implementation of this approach in the Wien2k code, in particular, in the `lapw2` program and Wien2k shell scripts (`run`, `runsp`, `x`). Finally, section 3.3 gives practical information on how to run SC LDA+DMFT calculations.

3.1 LDA+DMFT charge density self-consistency in LAPW/APW+lo

3.1.1 Charge density and total energy in the LAPW/APW+lo method in DFT-LDA

Generally, in a LDA-DFT band structure method the charge density is obtained by summing up all contributions from occupied KS eigenstates for a given point in real space:

$$\rho(\mathbf{r}) = \sum_{k,\nu} \langle \mathbf{r} | \Psi_{k\nu} \rangle w_{k\nu} \langle \Psi_{k\nu} | \mathbf{r} \rangle, \quad (3.1)$$

where $|\Psi_{k\nu}\rangle$ is the KS eigenvector for the band ν and a given \mathbf{k} -point, w_{nk} is the corresponding weight, which is equal to 1 for occupied bands and 0 for empty ones (w may also take a fractional value for finite temperatures or for some BZ integration schemes, as, for example, the tetrahedron method).

In any band structure method the KS eigenstates are expanded in a certain basis set $\{\phi\}$ (see eq. (1.4)), and precise formulas for actual implementation of (3.1) are related to a particular basis set.

Charge density within MT-spheres

In the LAPW framework the basis functions within MT-spheres (see eqs. 1.5, 1.8, 1.9, 1.11, 1.12) consist of linear combinations of the radial solution of the Schrödinger equation $u_l^{\alpha,\sigma}$ evaluated at a certain chosen energy (the linearization energy E^α for valence band or semi-core orbital), and the corresponding energy derivative $\dot{u}_l^{\alpha,\sigma}$ evaluated at the same energy. If one designates the set of radial functions as $x_l^{\alpha,\sigma} = \{u_l^{\alpha,\sigma}(E_{1l}^\alpha), \dot{u}_l^{\alpha,\sigma}(E_{1l}^\alpha), u_l^{\alpha,\sigma}(E_{2l}^\alpha), \dot{u}_l^{\alpha,\sigma}(E_{2l}^\alpha), \dots\}$ then those radial wave functions contribute to a given eigenvector $\Psi_{k\nu}$ with coefficients $S_{lm}^{\nu,\alpha}(\mathbf{k}, \sigma) = \{A_{lm}^{\nu,\alpha}(\mathbf{k}, \sigma), B_{lm}^{\nu,\alpha}(\mathbf{k}, \sigma), C_{lm}^{\nu,\alpha}(\mathbf{k}, \sigma), \dots\}$, where $A_{lm}^{\nu,\alpha}(\mathbf{k}, \sigma)$ is given by eq. (1.14), the coefficients $B_{lm}^{\nu,\alpha}(\mathbf{k}, \sigma)$ can be defined similarly:

$$B_{lm}^{\nu,\alpha}(\mathbf{k}, \sigma) \equiv \sum_{\mathbf{G}}^{N_{PW}} c_{\mathbf{G}}^{\nu,\sigma}(\mathbf{k}) B_{lm}^{\alpha,\mathbf{k}+\mathbf{G}} + \sum_{n_{lo}=1}^{l_{max}} c_{lo}^{\nu,\sigma} B_{lm}^{\alpha,lo} + \sum_{n_{LO}=1}^{LOMAX} c_{LO}^{\nu,\sigma} B_{lm}^{\alpha,LO}, \quad (3.2)$$

and coefficients $C_{lm}^{\nu,\alpha}(\mathbf{k}, \sigma)$ for semicore orbitals are equal to $c_{LO}^{\nu,\sigma} C_{lm}^{\alpha,nLO}$ (see eq.(1.15)).

Using the above designation the charge density (3.1) can be rewritten for a given MT-sphere α (and for a given spin) as

$$\rho_{\sigma\alpha}(\mathbf{r}) = \sum_k \sum_{l'} \sum_{ij} x_{li}^{\alpha\sigma}(r) x_{l'j}^{\alpha\sigma}(r) \sum_{mm'} Y_m^l(\hat{r}) \left(Y_{m'}^{l'}(\hat{r}) \right)^* \sum_{\nu} S_{lmi}^{\nu,\alpha}(\mathbf{k}, \sigma) \left(S_{l'm'j}^{\nu,\alpha}(\mathbf{k}, \sigma) \right)^* w_{k\nu}, \quad (3.3)$$

where the sum over the $i(j)$ indices runs over all radial wavefunctions $\{x\}$ and corresponding coefficients $\{S\}$.

To represent the angular dependence of the charge density it is expanded, within a given MT-spheres, in real spherical harmonics $Y_m^{Rl}(\hat{r})$:

$$\rho_{\sigma\alpha}(\mathbf{r}) = \sum_{lm} \rho_{\sigma\alpha}^{lm}(r) Y_m^{Rl}(\hat{r}), \quad (3.4)$$

Due to a particular crystal symmetry many $Y_m^{Rl}(\hat{r})$ contributions must be equal to zero, the list of $\{lm\}$ for which non-zero contribution is possible is given as input into the Wien2k *lapw2* program that computes the charge density. For $\rho_{\sigma\alpha}^{lm}(r) = \int d\hat{r} \rho_{\sigma\alpha}(\mathbf{r}) Y_m^{Rl}(\hat{r})$ one obtains from (3.3):

$$\rho_{\sigma\alpha}^{l_1 m_1}(r) = \sum_k \sum_{l'} \sum_{ij} x_{li}^{\alpha\sigma}(r) x_{l'j}^{\alpha\sigma}(r) \sum_{mm'} C_{lml_1 m_1}^{l' m'} \sum_{\nu} S_{lmi}^{\nu,\alpha}(\mathbf{k}, \sigma) \left(S_{l'm'j}^{\nu,\alpha}(\mathbf{k}, \sigma) \right)^* w_{k\nu}, \quad (3.5)$$

where $C_{lml_1 m_1}^{l' m'} = \int d\Omega Y_m^l(\hat{r}) \left(Y_{m'}^{l'}(\Omega) \right)^* \left(Y_{m_1}^{l_1}(\Omega) \right)$ are the corresponding Gaunt coefficients.

The kinetic energy density within MT-spheres is calculated in a similar way:

$$\epsilon_{\sigma\alpha}^{l_1 m_1}(r) = \sum_k \sum_{l'} \sum_{ij} x_{li}^{\alpha\sigma}(r) x_{l'j}^{\alpha\sigma}(r) \sum_{mm'} C_{lml_1 m_1}^{l' m'} \sum_{\nu} S_{lmi}^{\nu,\alpha}(\mathbf{k}, \sigma) \left(S_{l'm'j}^{\nu,\alpha}(\mathbf{k}, \sigma) \right)^* w_{k\nu} \epsilon_{k\nu}, \quad (3.6)$$

where $\epsilon_{k\nu}$ is the KS eigenvalue for a given \mathbf{k} -point and the band ν .

Charge density in the interstitial

In the interstitial region the LAPW basis functions are plain waves $\frac{1}{\sqrt{V}} e^{i(\mathbf{k}+\mathbf{G})\mathbf{r}}$ (see eqs. 1.5,1.9), which contributes to a given KS eigenvector $|\Psi_{k\nu}\rangle$ with the corresponding coefficients $a_G^{k\nu}$. Therefore, the equation (3.1) in the interstitial reads:

$$\begin{aligned} \rho^I(\mathbf{r}) &= \sum_k \sum_{\nu} \left[\sum_G a_G^{k\nu} e^{i\mathbf{k}\mathbf{G}\mathbf{r}} \sum_{G'} \left(a_{G'}^{k\nu} \right)^* e^{-i\mathbf{k}\mathbf{G}'\mathbf{r}} \right] w_{k\nu} \\ &= \sum_k \sum_{\nu} \left[\sum_G a_G^{k\nu} e^{i\mathbf{G}\mathbf{r}} \sum_{G'} \left(a_{G'}^{k\nu} \right)^* e^{-i\mathbf{G}'\mathbf{r}} \right] w_{k\nu}, \end{aligned} \quad (3.7)$$

where $\mathbf{k}_{\mathbf{G}} = \mathbf{k} + \mathbf{G}$.

In the actual computation of (3.7) one may transform the interstitial wavefunction $\sum_G a_G^{k\nu} e^{i\mathbf{G}\mathbf{r}}$ to a auxiliary mesh in the real space via FFT:

$$b_{\mathbf{R}}^{k\nu} = \sum_G a_G^{k\nu} e^{i\mathbf{G}\mathbf{R}}, \quad (3.8)$$

therefore getting rid off the double sum over \mathbf{G} and \mathbf{G}' in (3.7). The charge density on the axillary mesh then reads:

$$\rho^I(\mathbf{R}) = \sum_k \sum_\nu b_{\mathbf{R}}^{k\nu} (b_{\mathbf{R}}^{k\nu})^* w_{k\nu}, \quad (3.9)$$

which is then again transformed back to the reciprocal space via inverse FFT:

$$\rho^I(\mathbf{G}) = \sum_R \rho_{\mathbf{R}}^I e^{-i\mathbf{G}\mathbf{R}}. \quad (3.10)$$

The kinetic energy density on the real space mesh is calculated in the same way:

$$\epsilon^I(\mathbf{R}) = \sum_k \sum_\nu b_{\mathbf{R}}^{k\nu} (b_{\mathbf{R}}^{k\nu})^* w_{k\nu} \epsilon_{k\nu}, \quad (3.11)$$

and then also obtained in the reciprocal space via inverse FFT.

Total energy

The electron-nuclea, Hartree, exchange and correlation contributions to the total energy are calculated from the charge density $\rho(\mathbf{r})$ by the Wien2k code in a standard way, which does not need to be changed in LDA+DMFT (provided that correct LDA+DMFT charge density is supplied to the routines computing those contributions).

The kinetic energy is the only term in the DFT-LDA total energy expression that needs to be recalculated explicitly within LDA+DMFT. In DFT-LDA framework it is given by the sum of KS eigenvalues for occupied bands:

$$E_{kin} = \sum_{k\nu} w_{k\nu} \epsilon_{k\nu}. \quad (3.12)$$

3.1.2 Charge density and total energy in LAPW/APW+lo within LDA+DMFT

Within the projective technique for formation of the correlated orbitals (see Sec. 1.3.1) we use the KS states laying within a chosen energy window \mathcal{W} to form Wannier-like functions that are treated as correlated orbitals. Solving the corresponding quantum impurity problem produces the local self-energy within the correlated orbitals basis set, which is then upfolded (1.3) into the lattice self-energy $\Sigma_{\nu\nu'}(\mathbf{k}, i\omega_n)$. The lattice self-energy $\hat{\Sigma}(\mathbf{k}, i\omega_n)$ is generally non-diagonal in the subspace of the KS eigenstates $\{\nu\}$ ($\nu \in \mathcal{W}$) leading to a non-diagonal lattice Green's function (1.1) within \mathcal{W} and to the corresponding density matrix

$$N_{\nu\nu'}^k = \sum_n G_{\nu\nu'}(\mathbf{k}, i\omega_n) e^{i\omega_n 0^+} \quad (3.13)$$

being also non-diagonal.

Therefore, for the charge density contribution of the states within \mathcal{W} eq. (3.1) now reads:

$$\rho_{\mathcal{W}}(\mathbf{r}) = \sum_{k,\nu\nu'} \langle \mathbf{r} | \Psi_{k\nu} \rangle N_{\nu\nu'}^k w_k \langle \Psi_{k\nu'} | \mathbf{r} \rangle, \quad (3.14)$$

where w_k is the simple (geometrical) weight of a given \mathbf{k} -point.

The kinetic energy operator remains diagonal in the KS eignestates basis:

$$\hat{\mathcal{E}}_{kin}^k = Tr \hat{H}_{KS}^k \hat{N}^k = \sum_\nu \langle \Psi_{k\nu} | \hat{H}_{KS}^k \hat{N}^k | \Psi_{k\nu} \rangle = \sum_\nu \epsilon_{k\nu} N_{\nu\nu}^k \quad (3.15)$$

hence, the corresponding contribution to the kinetic energy density reads:

$$\epsilon_{\mathcal{W}}(\mathbf{r}) = \sum_{k,\nu} \langle \mathbf{r} | \Psi_{k\nu} \rangle N_{\nu\nu}^k w_k \epsilon_{k\nu} \langle \Psi_{k\nu} | \mathbf{r} \rangle, \quad (3.16)$$

Starting with eqs. (3.14,3.16) and reexpressing them in the LAPW/APW+lo basis set as described in Secs. 3.1.1 and 3.1.1 one obtains equations for the charge and kinetic energy density that are similar to ones in the standard DFT-LDA framework. The main modification there consists in substituting the diagonal in $\{\nu\nu'\}$ and real matrix of weights $w_{k\nu}$ with the Hermitian matrix $\hat{N}^k w_k$. In result, in the formulas for charge density the summation over ν is also substituted with the double summation over ν and ν' . Correspondingly, the equation (3.5) for the charge density within MT-spheres is rewritten as follows:

$$\rho_{\sigma\alpha}^{l_1 m_1, DMFT}(r) = \rho_{\sigma\alpha}^{bw, l_1 m_1}(r) + \sum_k \sum_{l'l'} \sum_{ij} x_{li}^{\alpha\sigma}(r) x_{l'j}^{\alpha\sigma}(r) \sum_{mm'} C_{lml_1 m_1}^{l'l'm'} \sum_{\nu\nu'}^{\mathcal{W}} S_{lmi}^{\nu,\alpha}(\mathbf{k}, \sigma) \left(S_{l'm'j}^{\nu',\alpha}(\mathbf{k}, \sigma) \right)^* N_{\nu\nu'}^k w_k \quad (3.17)$$

and the corresponding kinetic energy density reads

$$\epsilon_{\sigma\alpha}^{l_1 m_1, DMFT}(r) = \epsilon_{\sigma\alpha}^{bw, l_1 m_1}(r) + \sum_k \sum_{l'l'} \sum_{ij} x_{li}^{\alpha\sigma}(r) x_{l'j}^{\alpha\sigma}(r) \sum_{mm'} C_{lml_1 m_1}^{l'l'm'} \sum_{\nu}^{\mathcal{W}} S_{lmi}^{\nu,\alpha}(\mathbf{k}, \sigma) \left(S_{l'm'j}^{\nu,\alpha}(\mathbf{k}, \sigma) \right)^* N_{\nu\nu}^k \epsilon_{k\nu} \quad (3.18)$$

Here $\rho_{\sigma\alpha}^{bw, l_1 m_1}(r)$ and $\epsilon_{\sigma\alpha}^{bw, l_1 m_1}(r)$ are the charge and kinetic energy density contributions, respectively, due to the KS states below \mathcal{W} .

Similarly, the equations (3.9) and (3.11) for the interstitial charge and kinetic energy density on the real space mesh, respectively, should be modified as follows:

$$\rho^{I, DMFT}(\mathbf{R}) = \rho^{bw, I}(\mathbf{R}) + \sum_k \sum_{\nu\nu'} b_{\mathbf{R}}^{k\nu} (b_{\mathbf{R}}^{k\nu'})^* N_{\nu\nu'}^k w_k, \quad (3.19)$$

$$\epsilon^{I, DMFT}(\mathbf{R}) = \epsilon^{bw, I}(\mathbf{R}) + \sum_k \sum_{\nu} b_{\mathbf{R}}^{k\nu} (b_{\mathbf{R}}^{k\nu})^* N_{\nu\nu}^k \epsilon_{k\nu}, \quad (3.20)$$

where $\rho^{bw, I}(\mathbf{R})$ and $\epsilon^{bw, I}(\mathbf{R})$ are the corresponding contributions from the bands below the window.

Finally, in the expression for the total energy the kinetic energy contribution from states within \mathcal{W} now reads:

$$E_{kin}^{DMFT} = \sum_{k\nu} N_{\nu\nu}^k w_k \epsilon_{k\nu}. \quad (3.21)$$

In addition, the Hubbard energy $\langle H_U \rangle$ should be added to the total energy, and the double counting correction term E_{DC} should be subtracted from it. Hence, the total energy in LDA+DMFT reads:

$$E_{tot} = E_{kin}^{bw} + E_{kin}^{DMFT} + E_{el.-nucl.}[\rho_{DMFT}] + E_{Hartree}[\rho_{DMFT}] + E_{ex.-corr.}[\rho_{DMFT}] + \langle H_U \rangle - E_{DC}, \quad (3.22)$$

where E_{kin}^{bw} is the contribution to the kinetic energy from the bands below \mathcal{W} .

3.2 Implementation within the Wien2k code

To implement the charge density and total energy calculations described in Sec. 3.1.2 some changes have been made to the *lapw2* program and to some shell scripts of Wien2k. All changes in standard subroutines of *lapw2* and shell scripts are inserted between the comments `!QDMFT ... !END QDMFT`. In addition, a new module `QDMFT` (file `qdmft.F`) has been included to *lapw2*, it includes all new subroutines added to the *lapw2* program.

3.2.1 Summary of changes in the *lapw2* program

lapw2.F In the main routine `lapw2.F` we check whether the running mode of the program (MODUS variable read from `case.in2`) is equal to 'QDMFT' (l.201). If this is the case, then `ifqdmft` variable is set to `.TRUE.` and the subroutine `readdata_qdmft` of module `qdmft` is called. This routine reads, for each **k**-point, the range of bands (`nb_bot, nb_top`) included to \mathcal{W} and the geometrical weights w_k (from `case.oubwin` written by *dmftproj*) as well as the corresponding density matrix N^k (from `case.qdmft` written by *pyDMFT*). Then $H_U - E_{DC}$ is also read from `case.qdmft`.

l2main.F In the subroutine `l2main.F`, which calculates the charge density within the MT spheres and kinetic energy, for each **k**-point, we calculate $S_{lmi}^{\nu,\alpha}(\mathbf{k}, \sigma)$ coefficients (for all radial functions i used) from the lowest valence band to `nb_top` to include all bands below and within \mathcal{W} (l. 706). Then, in the main cycle for calculation of the charge/kinetic energy densities (ll. 1010-1341), we restrict summation over bands to the range from the the lowest valence band to `nb_bot-1`, to obtain contributions $\rho_{\sigma\alpha}^{bw,lm}(r)$ and $\epsilon_{\sigma\alpha}^{bw,lm}(r)$ from the states below \mathcal{W} . In order to get contributions from the bands within \mathcal{W} we call (within the cycle): the `sum_bands_dmft` subroutine from `qdmft` (l.1226) to compute $\sum_{mm'} C_{lml_1m_1}^{l'm'} \sum_{\nu\nu'}^{\mathcal{W}} S_{lmi}^{\nu,\alpha}(\mathbf{k}, \sigma) (S_{l'm'j}(\mathbf{k}, \sigma))$ and store the result in different (ZSA, ZSB, ...) temporary arrays, and the `add_dmft_contr` subroutine (l.1294) to multiply those arrays by $N_{\nu\nu'}^k w_k$ (for charge density) and $N_{\nu\nu'}^k w_k \epsilon_{k\nu}$ (for the kinetic energy density) and to sum over ν, ν' . The rest of the cycle, which finish computing the sums in (3.17) and (3.18) (by multiplying the corresponding contributions by $x_{li}^{\alpha\sigma}(r) x_{lj}^{\alpha\sigma}(r)$ and summing them over different indices) does not need to be changed.

In addition, the kinetic energy is corrected by the subroutine `dmft_etot` (l. 1351) by removing the contribution from the bands within \mathcal{W} calculated in the "usual way" by the `read_vec`(eq. (3.12) and adding the correct kinetic energy contribution computed in accordance with (3.21)

fourir.F In the subroutine `fourir.F`, for each **k**-point, we make the program to read the coefficients $a_G^{k\nu}$ from the lowest valence band to `nb_top` (l.186), then we restrict summation (l.271) in the cycle over bands (ll. 281-310) to `nb_bot-1` to obtain the below-window contributions $\rho^{bw,I}(\mathbf{R})$ and $\epsilon^{bw,I}(\mathbf{R})$ in accordance with eqs. (3.9,3.11). Then the subroutine `add_dmft_intr` is called (l.315) to compute the corresponding contributions within \mathcal{W} in accordance with eqs. (3.19) and (3.20).

3.2.2 Changes in Wien2k shell scripts

x This script now recognizes the mode `-qdmft` for the *lapw2* program. QDMFT is then inserted in `case.in2` input file as MODUS parameter. In order to launch *lapw2* in this mode type `x lapw2 -qdmft ...`, where ... may be other options (`-so`, `-up`, etc.). In addition, the units 32 and 34 are assigned to input files `case.qdmft$updn` and `case.outbwin$updn`, respectively, which are read by *lapw2* launched in `-qdmft` mode.

run/runsp This scripts now also recognize the mode `-qdmft`. In this mode the usual sequence of Fortran program calls within the Wien2k self-consistent cycle is altered. Instead of the usual sequence (for `run`):

```
lapw0 // Generation of one-electron potential from input charge density
lapw1 // Contruction and diagonalization of the Hamiltonian
lapw2 // Calculation of the valence charge density and kinetic energy
lcore // Calculation of the core charge density
mixer // Mixing charge density and calculating the total energy
```

the new sequence is as follows:

```
lapw0 // Generation of one-electron potential from input charge density
lapw1 // Contruction and diagonalization of the Hamiltonian
lapw2 -almd // Calculation of coefficients for the Wannier construction
dmftproj // Construction Wannier orbitals
```

```

pytriqs case.py carries out the DMFT cycle
lapw2 -qdmft Calculation of LDA+DMFT charge density/total energy
lcore // Calculation of the core charge density
mixer // Mixing charge density and calculating the total energy

```

For `runsp` script the changes are similar, `lapw2` and `dmftproj` are called separately for *up/dn* spins with flags `-up/dn`. With `-so` `lapw2` and `dmftproj` will be called with `-so` flag.

run_para/runsp_para Those are new scripts for running the DMFT cycle (`pytriqs case.py`) in parallel, with the rest of programs running in the serial mode. Inside those scripts `pytriqs` is called using an appropriate launcher for the parallel MPI calculations. On the Hubbard cluster the corresponding command line for `pytriqs` is

```
mpirun -c $NSLOTS -x PYTHONPATH ./pytriqs $file.py --same_dir
```

where `$NSLOTS` is the number of processors used. This line should be adapted to a particular cluster.

3.3 Running self-consistent LDA+DMFT calculation with Wien2k

To launch an LDA+DMFT job one needs:

- to create usual Wien2k input files, for example, by using `init/initso` scripts
- to create `dmftproj.def` and `case.indmftpr` as described in the previous chapter
- to have the Python script `case.py` for running the DMFT cycle
- Then launch LDA+DMFT calculations by typing one of the following command lines:

```

run -qdmft // paramagnetic serial LDA+DMFT calculations
runsp -qdmft // spin-polarized serial LDA+DMFT calculations
runsp -qdmft -so // spin-polarized spin-orbit serial LDA+DMFT calculations
run_para -qdmft -np N // paramagnetic parallel LDA+DMFT calculations on N nodes
runsp_para -qdmft -so -np N // spin-polarized spin-orbit parallel LDA+DMFT calculations
on N nodes

```

Those command lines may be supplemented with any usual Wien2k options you need (`-in1line #`, `-i #` etc.).