

# Refinement of flipping ratios using FULLPROF

Juan Rodríguez-Carvajal<sup>1</sup> & Carlos Frontera<sup>2</sup>

Laboratoire Léon Brillouin (CEA-CNRS), CEA/Saclay, 91191 Gif sur Yvette Cedex, France  
Instituto de Ciencia de Materiales de Barcelona (ICMAB-CSIC), Bellaterra, Spain.

Work performed at the *Service de Physique Statistique, Magnétisme et Supraconductivité*, MDN, CEA/Grenoble.

## Purpose

Here we give the essential recipes for working with FULLPROF in treating flipping ratios. Using incoming polarised neutrons into a sample, that is in a vertical magnetic field, one measures the ratio between the diffracted intensity of a Bragg reflection for spin-up neutrons ( $I^+$ ) to the same measured intensity when the incoming neutrons are spin-down ( $I^-$ ), the quantity  $R=I^+/I^-$  is called flipping ratio. No polarisation analysis is performed. The first Blume-Maleyev equation, simplified with the suppression of the chiral term, is the basis for the analysis. A relatively detailed description of the treatment of magnetisation density using polarised neutrons is given in the article “Analysis of charge and spin densities” by P. Coppens and P.J. Becker in the International Tables for Crystallography, Volume C, pp 627-645. Here we shall give only the necessary expressions to be known for preparing the input control file that we shall refer hereafter as the “PCR file”. This part of FULLPROF is under test and it is highly probable that changes, in future releases, will take place. The most probable modifications may concern the normalisation factors in order to provide refined parameters that could be compared directly to those obtained with other programs. Moreover the inclusion of the Schwinger effect, nuclear polarisation and a tensorial approach for the local susceptibility (for highly anisotropic materials) are planned.

## The expression used to calculate the flipping ratio

We shall call  $I_h$  the intensity (scattering cross-section) of the diffracted beam and, if the incident polarisation is noted as  $\mathbf{P}_i$ , the equation is:

Cross-section (Equation 1):

$$I_h = N_h N_h^* + \mathbf{M}_{\perp h} \mathbf{M}_{\perp h}^* + \left( N_h \mathbf{M}_{\perp h}^* + N_h^* \mathbf{M}_{\perp h} \right) \mathbf{P}_i \quad (1)$$

Where  $N_h$  is the nuclear structure factor of reflection  $\mathbf{h}$  and  $\mathbf{M}_{\perp h}$  is the magnetic interaction vector: perpendicular component of the magnetic structure factor to the scattering vector ( $\mathbf{Q}=2\pi \mathbf{h}$ ).

The particular geometry used in flipping ratio measurements allows using a scalar form for the vectorial quantities in the above expression. The polarisation direction is the same as the applied magnetic field, along the vertical  $\mathbf{z}$ -axis, and the magnetic structure factor  $\mathbf{M}_h$  is also along the vertical  $\mathbf{z}$ -axis ( $\mathbf{M}_h=M_h\mathbf{z}$ ), so, if we call  $\alpha$  the angle between the vertical axis and the scattering vector  $\mathbf{h}$ , we have:

$$\mathbf{M}_{\perp h} = \mathbf{e} \times \mathbf{M}_h \times \mathbf{e} = \mathbf{M}_h - \mathbf{e} (\mathbf{e} \cdot \mathbf{M}_h) = M_h \mathbf{z} - e M_h \cos \alpha$$

$$\mathbf{e} = \mathbf{h}/h = \mathbf{Q}/Q = (e_x, e_y, e_z) = (e_x, e_y, \cos \alpha) \quad \mathbf{P}_i = p(0, 0, 1); \quad p = \frac{n^+ - n^-}{n^+ + n^-} \quad (2)$$

$$\mathbf{M}_{\perp h} = M(\mathbf{h})(\mathbf{z} - \mathbf{e} \cos \alpha), \text{ so that}$$

$$\mathbf{M}_{\perp\mathbf{h}}\mathbf{M}_{\perp\mathbf{h}}^* = M(\mathbf{h})(\mathbf{z} - \mathbf{e} \cos \alpha) M^*(\mathbf{h})(\mathbf{z} - \mathbf{e} \cos \alpha) = M(\mathbf{h})M^*(\mathbf{h}) \sin^2 \alpha$$

So, dropping the scattering vector index, the incident polarisation dependent intensity is given by:

$$\begin{aligned} I^+ &= NN^* + MM^* \sin^2 \alpha + (NM^* + N^*M) p \sin^2 \alpha \\ I^- &= NN^* + MM^* \sin^2 \alpha - (NM^* + N^*M) p \sin^2 \alpha \end{aligned} \quad (3)$$

The quantity  $p$  may be lower than 1 due to the fact that the incident beam may be not fully polarised. In the expressions (2) the quantities  $n^+$  and  $n^-$  represent the number of neutrons with spin up and down, respectively, in the incident beam. Let us supposed that the incident beam is formed by spin-up neutrons, a flipper transforms this incoming beam into a spin-down neutron beam before the scattering process. We have to take into account the possibility of an inefficient flipper, calling  $e$  the fraction of neutrons that are effectively flipped the above expressions have to be modified as:

$$\begin{aligned} I^+ &= NN^* + MM^* \sin^2 \alpha + (NM^* + N^*M) p \sin^2 \alpha \\ I^- &= NN^* + MM^* \sin^2 \alpha - (NM^* + N^*M) p(2e-1) \sin^2 \alpha \end{aligned} \quad (4)$$

The flipping ratio is given by:

$$R = \frac{I^+}{I^-} = \frac{NN^* + MM^* q^2 + (NM^* + N^*M) p^+ q^2}{NN^* + MM^* q^2 - (NM^* + N^*M) p^- q^2} \quad (5)$$

where we have written  $q^2 = \sin^2 \alpha$  and have called  $p^+$  and  $p^-$  the effective polarisation for spin-up and spin-down of the incident beam. In the Appendix I we give a more detailed expression taking into account the extinction correction.

### What FULLPROF can do?

The program FULLPROF can be used to refine, by least squares, a list of flipping ratio measurements. The minimised function is:

$$\chi^2 = \sum_i \frac{\{R_i(obs) - R_i(calc, \boldsymbol{\beta})\}^2}{\sigma_i^2(R(obs))} \quad (6)$$

where the vector  $\boldsymbol{\beta}$  represents the list of free parameters. FULLPROF calculates the nuclear structure factor  $N$  from the structural model provided by the user and it is corrected for extinction. The magnetic structure factor is calculated in a simplified manner because it is supposed that only the component along the  $\mathbf{z}$  axis is contributing, however complex forms of magnetic form factors are available. The simplified expression of the magnetic structure factor (without explicitly taking into account the symmetry) is given by:

$$M(\mathbf{h}) = p_B \sum_{j=1}^n O_j T_j f_j(\mathbf{h}) \exp\{2\pi i \mathbf{h} \cdot \mathbf{r}_j\} \quad (7)$$

The sum is extended to all atoms with unpaired electrons in the unit cell, the factor  $p_B = \mu_B / 2 = 0.2695$  allows to convert Bohr magnetons,  $\mu_B$ , to scattering length amplitudes (in  $10^{-12}$  cm). The form factor  $f_j(\mathbf{h})$ , Fourier transform of the unpaired magnetisation density, may depend on the scattering vector  $\mathbf{h}$  (and not only on its amplitude  $|\mathbf{h}| = h$ ). The form factor contains implicitly the effective number of unpaired electrons (magnetic moment) in  $\mu_B$ . The coefficients  $O_j$  and  $T_j$  are, respectively, the occupation factor and the Debye-Waller factor.

It is supposed that the crystallographic structure and the extinction parameters have been refined at the same temperature as that of the flipping ratio experiment using a large number of reflections. The structural and extinction parameters should be fixed in the flipping ratio refinement. So the specific content of the  $\beta$  vector depends on the particular model selected for refinement. The different models existing in FULLPROF correspond to different ways of calculating  $f_j(\mathbf{h})$ . The particular kind of form factor is accessed by giving to the atom/ion a particular reserved name that we will made explicit at the description of each model. The program is also able to calculate the flipping ratios without experimental data. We shall see that in the forthcoming paragraphs.

## Spherical refinements

This is the simplest case, and the first to be tried, for modelling spin densities. It is supposed that the magnetisation density is dominated by the radial wave function of the unpaired electrons. If we consider that the wave function is not very much changed with respect to that of the free atom (ion) we can use tabulated values and refine only a small number of coefficients per magnetic atom. The magnetic form factor depends only of the magnitude of the scattering vector. If we use the common variable  $s=\sin\theta/\lambda$ , then the expression of the form factor used in FULLPROF to be inserted in the expression (7) is the following:

$$f(|\mathbf{h}|/2) = f(s) = \sum_{l=0,2,4,6} W_l \langle j_l(s) \rangle \quad (8)$$

Where the functions  $\langle j_l(s) \rangle = \int_0^\infty U^2(r) j_l(4\pi sr) 4\pi r^2 dr$  are tabulated in form of approximations (see

International Tables for Crystallography, Volume C, sect. 4.4.5) inside FULLPROF. The refinable parameters are the coefficients  $W_l$ . The unit of these coefficients is Bohr magnetons and  $W_0$  gives the value of the magnetic moment.

The expressions of the approximations for calculating  $\langle j_l(s) \rangle$  are the following:

$$\begin{aligned} \langle j_l(s) \rangle &= s^2 \left( A_l \exp\{-a_l s^2\} + B_l \exp\{-b_l s^2\} + C_l \exp\{-c_l s^2\} + D_l \right) \quad \text{for } l = 2, 4, 6 \\ \langle j_0(s) \rangle &= A_0 \exp\{-a_0 s^2\} + B_0 \exp\{-b_0 s^2\} + C_0 \exp\{-c_0 s^2\} + D_0 \end{aligned} \quad (9)$$

The coefficients  $A_l$ ,  $B_l$ ,  $C_l$ , and  $D_l$ , can be found in the above reference or in the output file of FULLPROF. An atom/ion is treated using this kind of approach if the name for its form factor is set to **MPOL** within the PCR file. Another way of doing spherical refinements is to consider a multipolar refinement in which we set to zero a part of the coefficients (see next paragraph).

## Multipolar refinements

For this kind of refinement a local Cartesian frame is required for defining the spherical harmonics. By default, FULLPROF defines (when using the flipping ratio option) a Cartesian frame attached to the unit cell with the same origin with the  $\mathbf{x}$ -axis aligned with the  $\mathbf{a}$ -axis,  $\mathbf{y}$  is in the plane  $(\mathbf{a}, \mathbf{b})$  and  $\mathbf{z}$  is defined as  $\mathbf{z} = \mathbf{x} \times \mathbf{y}$ . The local Cartesian frame, attached to each atom, to which the vectors  $\mathbf{r}$ ,  $\mathbf{r}_0$ , and spherical angles of the forthcoming formula are referred, can be selected by the user by a simple rotation of the above frame around a provided crystallographic direction. We shall see this in the section dedicated to the description of the input file.

## Density approach

In cases of departure of the magnetisation density from a spherical distribution more sophisticated approximations to the magnetisation density have to be used. If we consider the magnetisation density around one atom sited in the origin, we can write it as a series of real spherical harmonics:

$$m(\mathbf{r}) = \sum_{l=0}^4 R_l^{dens}(r) \sum_{m,p} \beta_l^{m,p} y_l^{m,p}(\mathbf{r}_0) \quad (10)$$

Where the unit  $\mathbf{r}_0$  vector along  $\mathbf{r}$  is parameterised in terms of the local spherical angles  $(\theta, \varphi)$ . The real spherical harmonics are defined as in M.Kara and K. Kurki-Suonio, Acta Cryst. **A37**, 201 (1981):

$$y_l^{m,p}(\theta, \varphi) = \frac{1}{\sqrt{2(1+\delta_{m0})}p} \{(-1)^m Y_l^{+m}(\theta, \varphi) + p Y_l^{-m}(\theta, \varphi)\}$$

with  $0 \leq m \leq l$  and  $p=\pm 1$ . The Fourier transform of expression (10) is the form factor that writes:

$$f(\mathbf{h}) = \sum_{l=0}^4 i^l \langle j_l(2\pi h) \rangle \sum_{m,p} \beta_l^{m,p} y_l^{m,p}(\mathbf{h}_0) \quad (11)$$

where,  $\langle j_l(s) \rangle$  is the Fourier-Bessel transform of the radial density:

$$\langle j_l(s) \rangle = \int_0^\infty R_l^{dens}(r) j_l(sr) r^2 dr \quad (12)$$

and  $j_l$  are the spherical Bessel functions. The coefficients  $\beta_l^{m,p}$  are adjustable occupancies of the different spherical harmonics. The radial density function can be taken to be Slater-type functions:

$$R_l^{dens}(r) = \text{Slater}(n_l, \zeta_l; r) = \frac{\zeta_l^{n_l+3}}{(n_l+2)!} r^{n_l} \exp\{-\zeta_l r\} \quad (13)$$

The volume factors  $\zeta_l$  can also be adjusted. An atom/ion is treated using this kind of approach if the name for its form factor is set to **MULT** within the PCR file.

Within FULLPROF the expression used for calculating the form factor is similar to (11) except that two  $l=0$  terms are allowed and the real coefficients are noted  $A$  instead of  $\beta$ . With the new notation the magnetisation density is written as:

$$m(\mathbf{r}) = A_0^{0+} \text{Slater}(n_{s1}, \zeta_{s1}; r) y_0^{0+}(\mathbf{r}_0) + B_0^{0+} \text{Slater}(n_{s2}, \zeta_{s2}; r) y_0^{0+}(\mathbf{r}_0) + \sum_{l=1}^4 \text{Slater}(n_l, \zeta_l; r) \sum_{m=0, \dots, l; p=\pm} A_l^{m,p} y_l^{m,p}(\mathbf{r}_0) \quad (13')$$

If one wants to start using a spherical approach it is necessary to consider only the first (and/or the second) term  $l=0$  in the expression (11), or in the transformed of the expression (13') so that the form factor takes the form  $f(\mathbf{h}) = \langle j_0(2\pi h)_A \rangle A_0^{0+} y_0^{0+}(\mathbf{h}_0) + \langle j_0(2\pi h)_B \rangle B_0^{0+} y_0^{0+}(\mathbf{h}_0)$ .

### Orbital approach

In this approach the contribution to the magnetisation density coming from each ion can be described in terms of an orbital centred in that ion. This orbital is taken to have a well defined orbital moment (except in the case of a hybrid s-p orbital). Thus, in general:

$$m(\mathbf{r}) = |\phi_l(\mathbf{r})|^2 \quad (14)$$

where the orbital  $\phi_l$  can be chosen to be a  $p$ ,  $d$  or  $f$  orbital expressed in terms of real spherical harmonics:

$$\phi_l(\mathbf{r}) = R_l(r) \sum_{m,p} A_l^{m,p} y_l^{m,p}(\mathbf{r}_0) \quad (15)$$

In general, the coefficients  $A_l^{m,p}$  are complex numbers ( $A_l^{m,p} = |A_l^{m,p}| \exp\{i\phi_l^{m,p}\}$ ), an overall phase is meaningless so we fix  $\phi_l^{l+}$  arbitrarily to zero. Each type of orbital leads to a magnetisation density that can be expressed in a way similar to equation (10):

$$m(\mathbf{r}) = |\phi_l(\mathbf{r})|^2 = \left| R_l(r) \sum_{m,p} A_l^{m,p} y_l^{m,p}(\mathbf{r}_0) \right|^2 = R_l^2(r) \sum_{l'=0,2,\dots,2l} \beta_{l'}^{m,p} y_{l'}^{m,p}(\mathbf{r}_0) \quad (16)$$

The magnetic form factor of each magnetic ion can now be written as:

$$f(\mathbf{h}) = \sum_{l'=0,2,\dots,2l} i^{l'} \langle j_{l'}(2\pi h) \rangle \sum_{m,p} \beta_{l'}^{m,p} y_{l'}^{m,p}(\mathbf{h}_0) \quad (17)$$

where, in this case,  $\langle j_{l'}(s) \rangle$  is related to  $R_l(r)$  by

$$\langle j_{l'}(s) \rangle = \int_0^\infty r^2 R_l^2(r) j_{l'}(sr) dr \quad (18)$$

The radial part is handled within FULLPROF in three different ways.

The first approach consists of taking the  $\langle j_{l'}(s) \rangle$  just, as in the case of simple spherical refinements, using the tabulated approximations, see equations (9).

The second approach is to approximate  $R_l(r)$  to a hydrogen-like radial function with principal quantum number  $n$  ( $n \geq l+1$ ), with  $l$  the same quantum number as the angular part. The form of such a function can be found in text books:

$$rR_{nl}(r) = \left\{ \frac{\xi}{2n} (n-l-1)!(n+l)! \right\}^{1/2} \exp\left\{ -\frac{\xi r}{2} \right\} (\xi r)^{l+1} \times \sum_{\nu=0}^{n-l-1} \frac{(-\xi r)^\nu}{\nu!(\nu+2l+1)!(n-l-1-\nu)!} \quad (19)$$

where  $\xi = \frac{2Z_{eff}}{na_0}$ , with  $a_0=0.52918 \text{ \AA}$ . In this approach  $n$  can be chosen by the user (with the obvious restriction that  $n \geq l+1$ ) and  $Z_{eff}$  can be adjusted.

The third approach is to approximate  $R_l(r)$  to a Slater-type radial function, written in this case as:

$$R(r) = \text{Slater}(n, \eta; r) = \frac{(2\eta)^{2n+3}}{(2n+2)!} r^n \exp\{-\eta r\} \quad (19')$$

In the case of orbitals of type  $p$ ,  $d$  or  $f$ , the atom/ion using this kind of approach should have the name for its form factor set as **DIPO**, **QUAD** or **HXAP**, respectively, within the PCR file.

Another case that can also be treated is the hybrid  $s$ - $p$  orbital. For this, it is assumed that the expression of the orbital has the form:

$$\phi_{s-p}(\mathbf{r}) = R_p(r) \sum_{m,p} A_1^{m,p} y_1^{m,p}(\mathbf{r}_0) + R_s(r) A_0^{0+} y_0^{0+}(\mathbf{r}_0) \quad (20)$$

In this case the expression for the magnetisation density is similar to that of equation (16) but with  $l'=0, 1, 2$ . If this type of orbital is chosen,  $\langle j_{l'}(s) \rangle$  can be calculated only by assuming that  $R_p$  and  $R_s$  are hydrogen-like functions with the same principal quantum number  $n$ . An atom/ion is treated using this kind of approach if the name for its form factor is set to **DISP** within the PCR file.

The selection between the three approaches for treating the radial part is selected in the PCR file by the value of the `Radial_Model` variable (see below).

In the Appendix II we give the development of the magnetisation density as a function of the orbital coefficients.

### How to calculate flipping ratios in FULLPROF

The format of the input control file for flipping ratio calculations is exactly the same as that for single crystal calculations except that the option **JBT=10** is compulsory. To get the complete explanation of all the items appearing in the PCR file the reader is invited to consult the manual of the program. FULLPROF knows that flipping ratio calculations have to be performed only by reading the type of crystallographic data (see below) or by explicit indication in the PCR (for simulations). Let us start with the simulation of flipping ratios when we do not have experimental data. For doing such a simulation one needs only the PCR file.

#### *Simulations*

For performing a flipping ratio simulation the user should put **JOB=3** and **CRY=4**. The program generates reflections according to the provided  $2\theta_{\min}$ ,  $2\theta_{\max}$  angles. The wavelength and other characteristics of the simulation are provided at end of the PCR file. The simplest way to understand how to perform a simulation is by looking at a simple example.

In the example given below in Panel 1, the used wavelength is 0.711 Å, the variable `itypd=2` means that the program will calculate flipping ratios from the model given in the same file. `Polarp` and `Polarm` are the polarisation degree for neutrons "up" and "down" respectively, `UB_mat=1` means that the program will read the orientation matrix just below the current line. The same type of file serves also for calculating structure factors, instead of flipping ratios, for a single crystal, so if `itypd=0, 1` the program will calculate structure factors and the output `*.int` file will contain either the square of structure factors (`itypd=0`) or simply the module of structure factors (`itypd=1`).

#### *Description of atomic parameters for different models of the magnetisation density*

The option **JBT=10** is compulsory for calculating flipping ratios, at this point the atoms are described as for the usual crystallographic jobs. An atom is considered to have unpaired electrons if its scattering factor label has a particular selected name and its kind of atom is of type `n_t=4` or `n_t=5`. These last values instruct the program that the form factor for the atom is of special type.

The first line of an atom having a special form factor contains the following items:

Label	ScattLab	Mag	Vek	x	y	z	Biso	Occ	n_t	Spc
-------	----------	-----	-----	---	---	---	------	-----	-----	-----

After this line another line with the codes of refinements must be given. For flipping ratio refinements this line of codes must be a list of zeros.

Where `Label` must start with the chemical symbol of the atom followed by the valence state. The program extracts from this label the scattering length for calculating the structure factor. The valence is needed for some of the special form factors available.

The item `ScattLab` is used for select different modes of unpaired electron density, we shall discuss this in more detail below.

### Panel 1: Important items for simulations of flipping ratios

The header of the PCR file adapted for single crystal simulations looks like:

```
COMM Simulation of Flipping ratios
! Files => DAT-file: flipr-sim, PCR-file: flipr-sim
!Job Npr Nph Nba Nex Nsc Nor Dum Iwg Ilo Ias Res Ste Nre Cry Uni Cor Opt Aut
   3  0  1  0  0  0  0  0  0  0  0  0  0  0  4  0  0  0  0
!
!Ipr Ppl Ioc Mat Pcr Ls1 Ls2 Ls3 NLI Prf Ins Rpa Sym Hkl Fou Sho Ana
   0  0  1  0  1  0  0  0  0  1  0  0  0  0  4  0  0
!
!NCY Eps R_at R_an R_pr R_gl Thmin Step Thmax PSD Sent0
   1  0.01 1.00 1.00 1.00 1.00 10.0000 0.1000 89.9000 0.000 0.000
!
```

Where we have emphasised in bold the needed parameters for performing a simulation. Putting **Fou=4** (in the case of flipping ratios) instruct the program to calculate a density map, according to the provided model, that can be visualised using GFOURIER.

This is followed by the description of the crystallographic phase under study. This part of the PCR file is similar to that for powder diffraction, except that more sophisticated descriptions of the form factors are allowed. We shall see that in the next section.

The tail of the PCR file adapted for single crystal simulations. We have emphasised in bold the important part for performing simulations.

```
! Scale Factors
! Sc1 Sc2 Sc3 Sc4 Sc5 Sc6
   1.000 0.000 0.000 0.000 0.000 0.000
   0.00 0.00 0.00 0.00 0.00 0.00
! Extinction Parameters
! Ext1 Ext2 Ext3 Ext4 Ext5 Ext6 Ext7 Ext-Model
   42.0 42.0 74.0 0.000 0.000 0.000 0.000 4
   0.00 0.00 0.00 0.00 0.00 0.00 0.00
! a b c alpha beta gamma #Cell Info
   6.197000 6.197000 7.013000 90.000000 90.000000 120.000000
   0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
! x-Lambda/2 + Not yet used parameters
   0.03000 0.00000 0.00000 0.00000 0.00000
   0.00 0.00 0.00 0.00 0.00
! Parameters for Single X-tal calculations
! Lambda itypd ipow Polarp Polarm UB_mat
   0.7110 2 0 0.89000 0.89000 1
! UB_matrix
   -0.017950 0.000610 0.188620
   -0.042370 -0.173200 -0.003470
   0.127620 -0.031460 0.012250
```

The items Mag and Vek are not used in the case of flipping ratio calculations. These are integer variables; both must be fixed to zero. The items x, y, z, Biso and Occ have the same meaning as in normal refinements (fractional coordinates, isotropic temperature factor and occupation factor).

The item Spc is not used for flipping ratio calculations.

The admissible (integer) values for  $n\_t$  in the case of flipping ratio calculations are the following:

- $n\_t = 0$       Isotropic atom without unpaired electrons. It contributes only to the nuclear structure factor.
- $n\_t = 2$       Anisotropic atom without unpaired electrons. It contributes only to the nuclear structure factor.
- $n\_t = 4$       Atom with unpaired electrons, for which a special form factor is given. It contributes to both nuclear and magnetic structure factor. The atom has only an isotropic thermal parameter.
- $n\_t = 5$       Atom with unpaired electrons, for which a special form factor is given. It contributes to both nuclear and magnetic structure factor. This case allows the use of anisotropic temperature factors in calculating flipping ratios or structure factors of complex object described by different kinds of form factors. Two additional lines (one with  $\beta_{11}, \beta_{22}, \beta_{33}, \beta_{12}, \beta_{13}, \beta_{23}$  and the other with the corresponding codes) are read for an atom after the reading of form factor coefficients (see below).

Examples of the two first lines characterising an atom of the list are given in Panel 2.

For flipping ratio calculations the most important part is the model for the form factor to be used. This is indicated to the program by the special names for the variable `ScattLab` (that is indicated in the PCR file by the name `Type`). Let us describe in detail the different cases available in `FULLPROF`.

**Panel 2: Examples of the first lines describing atoms.**

*Example 1:*

!Label	ScattLab	Mag	Vek	x	y	z	Biso	Occ	$n\_t$	Spc
N1	N	0	0	0.69232	0.12932	0.73347	0.32982	1.00000	0	0
				0.00	0.00	0.00	0.00	0.00		
...										
C1	MULT	0	0	0.69232	0.12932	0.73347	0.32982	1.00000	4	0
				0.00	0.00	0.00	0.00	0.00		

The atom N1 contributes only to nuclear structure factors. The atom C1 is described by a multipole expansion (the coefficients are read after these two lines, these are described below).

*Example 2:*

An example of complete description for an anisotropic atom contributing only to the structure factor is as follows:

Sulf	S	0	0	0.87879	0.00620	0.74224	0.00000	1.00000	2	0
				0.00	0.00	0.00	0.00	0.00		
		0.01387	0.02232	0.00073	0.00064	0.00081	0.00039	<-Betas		
		0.00	0.00	0.00	0.00	0.00	0.00			

1: `ScattLab = MPOL`



This is the simplest case corresponding to spherical atoms in which the form factor is calculated as linear combinations of the form given by equation (8). An example of atom for which the MPOL option is used is given in Panel 3.

**Panel 3:** In the example below an ion of uranium,  $U^{3+}$ , is treated for flipping ratio calculations with a form factor of the form:  $f(s) = W_0 \langle j_0(s) \rangle + W_2 \langle j_2(s) \rangle$ . The internal tables (see equations 9) are used and the refined parameters  $W_0$  and  $W_2$  are the two first coefficients of the list. A maximum of four coefficients can be refined.

U3	<b>MPOL</b>	0	0	0.00000	0.00000	0.00000	0.09825	0.04167	<b>4</b>	0
				0.00	0.00	0.00	0.00	0.00		
	<b>0.03596</b>	<b>0.06116</b>		0.00000	0.00000	0.00000	0.00000	0.00000	<-FormFactor	
	<b>11.00</b>	<b>21.00</b>		0.00	0.00	0.00	0.00	0.00		
	0.00000	0.00000		0.00000	0.00000	0.00000	0.00000	0.00000		
	0.00	0.00		0.00	0.00	0.00	0.00	0.00		

In the Panel 3 it is readily visible that the atom is isotropic from the point of view of the thermal Debye-Waller factor. In this case we have  $B_{iso} \approx 0.098 \text{ \AA}^2$ . The occupation factor is given by the quotient between the multiplicity of the site and the general multiplicity of the space group. The user can use a multiplicative factor to get more simple numbers.

It is important to realise that the coefficients  $W_l$  can be related to the magnetic moment of the atom and to the orbital contribution to it. Comparing the expression given in Panel 3 with the most common notation in the dipolar approximation:

$$f(s) = \mu_T (\langle j_0(s) \rangle + C_2 \langle j_2(s) \rangle)$$

It is easy to calculate the orbital contribution  $\mu_L = \mu_T C_2 = W_2$  and the spin contribution ( $\mu_S = \mu_T - \mu_L$ )  $\mu_S = W_0 - W_2$  that are readily obtained from the  $W_l$  refined values.

2: ScattLab = **MULT**

This is the most general case corresponding to a multipolar expansion of the magnetisation density around an atom. The form factor is calculated as linear combinations of radial functions and real spherical harmonics of the form given by equation (11). An example of atom for which the MULT option is used is given in Panel 4.

Notice that the coefficients correspond to the expression (10'), where we have written in the PCR file the notation  $A_l^{m,p}$  as  $A(lmp)$  and  $\zeta_l$  as  $Zed(l)$ . The presence of two spherical ( $s_1$  and  $s_2$ ) terms is needed for simulate hybrid  $s$ - $p$  orbitals using the density approach.

It is shown that after reading all the 32 possible free parameters characterising the multipolar expansion in , that there is a line for selecting the exponent of the Slater functions used to calculate the radial parts.

After reading the exponent of the Slater functions there is another line containing the following four items:

Rot\_angle, Rot\_Direction\_x, Rot\_Direction\_y, Rot\_Direction\_z

Where is the rotation angle in degrees to be applied to the standard Cartesian frame of FULLPROF, see the introduction to multipolar refinements, around the axis **n** with components **n**=(Rot\_Direction\_x, Rot\_Direction\_y, Rot\_Direction\_z). Internally the vector **n** is normalized to be a unitary vector.

**Panel 4:** In the example below a carbon atom C1 is treated for flipping ratio calculations with a multipolar form factor. In this particular case only an effective spherical coefficient is refined. Notice that n<sub>t</sub>=5, so that the ADP's are read after all parameters characterising the multipolar form factor.

```

C1  MULT      0  0      0.80030  0.03509  0.68730  0.93435  1.00000      5  0
      0.00      0.00      0.00      0.00
!   A(00+)  Zed(s1)  B(00+)  Zed(s2)  A(11-)  A(10+)  A(11+)
0.01403  6.50743  0.00000  0.00000  0.00000  0.00000  0.00000 <-Multipole
11.00      0.00      0.00      0.00      0.00      0.00      0.00
!   Zed(1)  A(22-)  A(21-)  A(20+)  A(21+)  A(22+)  Zed(2)
0.00000  0.00000  0.00000  0.00000  0.00000  0.00000  0.00000
0.00      0.00      0.00      0.00      0.00      0.00      0.00
!   A(33-)  A(32-)  A(31-)  A(30+)  A(31+)  A(32+)  A(33+)
0.00000  0.00000  0.00000  0.00000  0.00000  0.00000  0.00000
0.00      0.00      0.00      0.00      0.00      0.00      0.00
!   Zed(3)  A(44-)  A(43-)  A(42-)  A(41-)  A(40+)  A(41+)
0.00000  0.00000  0.00000  0.00000  0.00000  0.00000  0.00000
0.00      0.00      0.00      0.00      0.00      0.00      0.00
!   A(42+)  A(43+)  A(44+)  Zed(4)
0.00000  0.00000  0.00000  0.00000
0.00      0.00      0.00      0.00
! Exponents of Slater Functions (ns1,ns2,n1,n2,n3,n4):
2  0  0  0  0  0
! Angle and axis of rotation matrix to local orthogonal frame
0.0      0.0000  0.0000  1.0000
! Anisotropic displacement parameters (bet11, bet22, bet33, bet12, bet13, bet23)
0.01007  0.00423  0.00510  -0.00020  -0.00017  0.00181 <-Betas
0.00      0.00      0.00      0.00      0.00      0.00

```

3: ScattLab = **DIPO**, **QUAD**, **HXAP**, **DISP**

These are the cases corresponding to the orbital approach for  $l=1$  ( $p$ ),  $l=2$  ( $d$ ) and  $l=3$  ( $f$ ) respectively. The expression used for the orbitals within FULLPROF is slightly different of expression (15). We use an occupation factor  $O_l$  making the complex  $A_l^{m,p}$  coefficients not all freely refinable.

$$\phi_l(\mathbf{r}) = O_l R_l(r) \sum_{m,p} |A_l^{m,p}| \exp\{i\phi_l^{m,p}\} y_l^{m,p}(\mathbf{r}_0)$$

where we must have

$$\left( \sum_{m,p} A_l^{m,p} y_l^{m,p}(\mathbf{r}_0) \right) \left( \sum_{m,p} A_l^{m,p} y_l^{m,p}(\mathbf{r}_0) \right)^* = \sum_{m,p} |A_l^{m,p}|^2 = 1$$

An examples of this kind of approach is written in the Panel 5, for the cases  $p$  and  $d$ . The general input for the orbital approach can be summarised as follows.

After the first two lines describing the position, thermal and occupation factors of the atom we have a set of pairs of lines (the second line correspond always to the refinement codes) with seven parameters per line ordered in the sequence:

$$O_l, A_l^{l+}, A_l^{l-}, \phi_l^{l-}, A_l^{(l-1)+}, \phi_l^{(l-1)+}, A_l^{(l-1)-}, \phi_l^{(l-1)-}, \dots, A_l^{0+}, \phi_l^{0+}, Z_{\text{eff}}(\text{or } \eta)$$

so that there are a maximum of  $4l+2$  refinable parameters. For the case of a p-orbital it is needed (even if not used) two set of pairs of lines because the minimum number read of form-factor parameters when  $n_t=4, 5$  is 14. In the case of a hybrid sp-orbital the list of parameters is that shown for the C1 atom in the Panel 5.

After the lines containing the refinable coefficients, another line containing the items Exponent (value of  $n$  in expressions (19) or (19')) and Radial\_Model is read. When the value of Radial\_Model is  $0 \leq \text{Radial\_Model} < l+1$ , the first approach (tabulated coefficients for calculating  $\langle j_i \rangle$ ) for the radial function treatment is selected by the program. When  $\text{Radial\_Model} > l+1$  the second approach is selected, so a hydrogen-like radial function in which  $n = \text{Radial\_Model}$  is used. When  $\text{Radial\_Model} < 0$  the third approach (Slater-type function as in expression (19')) is used.

**Panel 5:** In the example below a vanadium ion  $V^{3+}$  is treated for flipping ratio calculations with a form factor corresponding to a d-orbital, an oxygen atom treated by using a p-orbital and a carbon atom using a hybrid sp-orbital. Notice the presence of a pair of lines (with zeros) that are needed for the p-orbital and for the hybrid-sp orbital for compatibility with other uses of FULLPROF when  $n_t=4, 5$ . The values of the parameters are completely artificial and have no meaning; they are given only for illustration purposes.

```
V3  QUAD      0  0      0.50000  0.00000  0.00000  0.38910  0.50000  4  0
      0.00      0.00      0.00      0.00
!      Occ  A(22+)  A(22-)  Phi(22-)  A(21+)  Phi(21+)  A(21-)
      0.01959  0.17991  0.10000  0.00000  -0.00636  0.00000  0.00557 <- d-orbital
      11.00      0.00      0.00      0.00      21.00      0.00      0.00
! Phi(21-)  A(20+)  Phi(20+)  Zeff
      0.00000  0.10000  0.00000  5.17957
      0.00      0.00      0.00      0.00
! Exponent(n) & Radial Model :
      3  0
! Angle and axis of rotation matrix to local orthogonal frame
      45.0000  0.0000  0.0000  1.0000
O1  DIPO      0  0      0.94539  0.35284  0.68806  0.83457  1.00000  4  0
      0.00      0.00      0.00      0.00
!      Occ  A(11+)  A(11-)  Phi(11-)  A(10+)  Phi(10+)  Zeff
      0.03130  0.06429  0.42415  146.77235  0.31569  -8.48253  3.99334 <- p-orbital
      141.00  211.00      0.00      221.00      0.00      0.00      0.00
      0.00000  0.00000  0.00000  0.00000  0.00000  0.00000  0.00000
      0.00      0.00      0.00      0.00      0.00      0.00      0.00
! Exponent(n) & Radial Model :
      2  -1
! Angle and axis of rotation matrix to local orthogonal frame
      24.3291  0.0000  0.0514  0.0249
C1  DISP      0  0      0.34539  0.15284  0.28806  0.73488  1.00000  4  0
      0.00      0.00      0.00      0.00
!      Occ  A(11+)  A(11-)  Phi(11-)  A(10+)  Phi(10+)  A(00+)
      0.02140  0.01429  0.22318  49.43221  0.31569  -1.28277  0.19131 <- sp-orbital
      141.00      0.00      0.00      0.00      0.00      0.00      0.00
! Phi(00+)  Zeff(p)  Zeff(s)
      0.00000  3.34000  3.10000  0.00000  0.00000  0.00000  0.00000
      0.00      0.00      0.00      0.00      0.00      0.00      0.00
! Exponents(np, ns) & Radial Model :
      3  2  -1
! Angle and axis of rotation matrix to local orthogonal frame
      0.0000  0.0000  0.0000  1.0000
```

The description of the atom finishes with a line containing the transformation to the local Cartesian frame as in the case of multipolar refinement based in the density approach. Of course, if  $n_t=5$  a

further line (and its corresponding line of refinement codes set to zeros) containing the anisotropic displacement (temperature) parameters is needed.

### **The experimental flipping ratios file (codfil.int)**

The extension of this file should be \*.int as for a single crystal integrated intensity file. The reading of this file is activated by putting the value IRF=4 for the corresponding phase. The content of this file is described below line by line.

#### **1: Title**

#### **2: Format of the input data, using Fortran syntax, without quotes**

e.g.: (3i4,2f10.6,i4,5f8.5)

Notice always, at least, five real numbers ("5f") are read even if they are of no use, so the format should finish with 5f8.5 or 5f10.4 instructions. In some cases it could be six numbers, so the instruction must be of the form 6f8.5 or whatever adequate for six real numbers (see below).

#### **3: Lambda, itypdata, ipow, polarp, polarm, readub, ext\_calc, read\_strf**

The line number 3 contains the above items that are described below.

Lambda : wavelength in Angstroms (real number)  
itypdata : should equal to 2 for flipping ratio data (integer number)  
ipow : should equal to 0 for flipping ratio data (integer number)  
polarp : polarisation degree of up neutrons (max=1.0, real number)  
polarm : polarisation degree of down neutrons (max=1.0, real number)  
readub : If=1 read the UB-matrix in the next line  
ext\_calc : Extinction correction of Becker-Coppens type

ext\_calc = 0, The extinction correction coefficients c1, c2, c3 are read with each reflection  
c1=Tbar\*1000\* $\lambda^3/(V^2 \sin 2\theta)$   
c2=A( $\theta$ ) Becker-Coppens coefficients  
c3=B( $\theta$ )  
ext\_calc = 1 The extinction correction coefficients c2, c3 are calculated for Gaussian  
Becker-Coppens corrections  
ext\_calc = 2 The extinction correction coefficients c2, c3 are calculated for Lorentzian  
Becker-Coppens corrections

If ext\_calc = -1, -2 the coefficient c1 is also calculated from Tbar (calculated) for each reflection (not implemented yet, a constant value of Tbar is taken at present). In this case the UB matrix should be given as if readub=1.

The effective use of extinction correction depends ultimately of the value of iextinc(iph) in the PCR file. Only if iextinc(iph)=2, 3 the coefficients c1, c2, c3 are used in the correction of the structure factors. This part of the extinction correction is still under testing. Phenomenological extinction correction models 1 (isotropic Shelx-like) and 2 (anisotropic Shelx-like) are more stable for refinement at present.

read\_strf : If different from zero the real and imaginary part of the nuclear structure factors are read in the flipping ratios file.

If readub=1 or ext\_calc < 0 then the following line is read:

4: ub11, ub12, ub13, ub21, ub22, ub23, ub31, ub32, ub33

Example of the header of a \*.int file:

```
Single crystal data of My_Crystal: Flipping ratios at 220K
(3i4,2f10.6,i4,4f8.5)
0.711  2    0  0.9998 0.9997 1 0
0.0051992 -0.1439303 -0.0781113 -0.0020301 -0.1044526 0.1072490 -0.1893921 -0.0027108 -0.0034442
```

The rest of lines are the indices of reflections, the flipping ratio values with the corresponding standard deviations, a code number, and additional items depending of the value of the variable readub defined above. The format of these items corresponds to that given by the user in line 2.

If readub =0 the items are:

If read\_strf =0

h k l flipr sigma code q2 c1 c2 c3

If read\_strf ≠ 0

h k l flipr sigma code q2 Nreal Nimag c1 c2 c3

If readub =1 the items are

If read\_strf =0

h k l flipr sigma code c1 c2 c3

If read\_strf ≠ 0

h k l flipr sigma code Nreal Nimag c1 c2 c3

q2 is the value of  $\sin^2 \alpha$ , where  $\alpha$  is the angle between the scattering vector and the applied field defining the polarisation direction (vertical axis). If the orientation matrix is given q2 must not be provided, it is calculated directly from the indices of the reflections and the UB matrix. The coefficients c1, c2, c3 are used only if the Becker-Coppens extinction correction is to be applied.

## Projects

These notes may serve as the manual for FullProf for flipping ratios refinements. This is a part of the program that is under intensive testing, so things can change in the near future.

## Appendix I: Expression of the calculated flipping ratio taking into account the extinction

In this appendix we provide the explicit expression of the flipping ratio when using the extinction correction. We shall decompose the expressions (4) and (5) in order to make explicit the application of the extinction correction within FULLPROF.

Let us call the real and imaginary parts of the nuclear structure factor as  $A_N$  and  $B_N$ ; for the magnetic structure factor we write  $A_M$  and  $B_M$ . Let us call the extinction correction factors  $y_p$ ,  $y_m$  and  $y_{pm}$  for the scattering processes  $\uparrow\uparrow$ ,  $\downarrow\downarrow$  and  $\uparrow\downarrow$ , the expression of  $I^+$  and  $I^-$  when we apply the extinction correction are:

$$\begin{aligned} p_p^+ &= \frac{1}{2} \left( (1+p^+)y_p + (1-p^+)y_m \right); & p_m^+ &= \frac{1}{2} \left( (1+p^+)y_p - (1-p^+)y_m \right) \\ p_p^- &= \frac{1}{2} \left( (1+p^-)y_p + (1-p^-)y_m \right); & p_m^- &= \frac{1}{2} \left( (1+p^-)y_p - (1-p^-)y_m \right) \\ I^+ &= (NN^* + MM^* \sin^2 \alpha) p_p^+ + 2(A_N A_M + B_N B_M) p_m^+ \sin^2 \alpha + MM^* y_{pm} \sin^2 \alpha \cos^2 \alpha \\ I^- &= (NN^* + MM^* \sin^2 \alpha) p_p^- + 2(A_N A_M + B_N B_M) p_m^- \sin^2 \alpha + MM^* y_{pm} \sin^2 \alpha \cos^2 \alpha \end{aligned}$$

The values of  $p^+$  and  $p^-$  have to be provided by the user. Normally they are close to 1.  
(to be completed and detailed writing the different extinction corrections available)

## Appendix II: Multipolar development of the orbital approach

In this appendix we provide the explicit expressions of the general development for the magnetic density when using the orbital approach for the different cases.

### A. $p$ Orbital (**DIP**O)

A general  $p$  orbital is expressed as:

$$\phi_p(\mathbf{r}) = R_p(r) \left[ a y_1^{1+}(\mathbf{r}_0) + b e^{iph_1} y_1^{1-}(\mathbf{r}_0) + c e^{iph_2} y_1^{0+}(\mathbf{r}_0) \right] \quad (\text{A-1})$$

so that the spin density is given by

$$|\phi_p(\mathbf{r})|^2 = R_p^2(r) \left[ \sum_{m,p} Dpl2(p,m) y_2^{m,p}(\mathbf{r}_0) + Dpl0(0) y_0^{0+}(\mathbf{r}_0) \right] \quad (\text{A-2})$$

The following Fortran code summarises the calculations of the coefficients  $Dpl2$  and  $Dpl0$  as a function of the parameters  $a$ ,  $b$ ,  $c$ ,  $ph_1$  and  $ph_2$  of expression A-1.

```
real, PARAMETER :: &
    a01= 0.2185097040, a02= -0.2185097040, a03= -0.1261566430, &
    a04= 0.2523132860, a05= 0.2820947770

Dpl2( 2)= a01 *a*a +a02 *b*b
Dpl2( 1)= a01 *2.0*a*c*cos(ph2)
Dpl2( 0)= a03 *a*a +a03 *b*b +a04 *c*c
Dpl2(-1)= a01 *2.0*b*c*cos(ph1-ph2)
Dpl2(-2)= a01 *2.0*a*b*cos(ph1)
Dpl0( 0)= a05 *(a*a + b*b + c*c)
```

### B. Hybrid $s$ - $p$ Orbital (**DISP**)

A general  $s$ - $p$  orbital is expressed as:

$$\phi_{sp}(\mathbf{r}) = R_p(r) \left[ a y_1^{1+}(\mathbf{r}_0) + b e^{iph_1} y_1^{1-}(\mathbf{r}_0) + c e^{iph_2} y_1^{0+}(\mathbf{r}_0) \right] + R_s(r) d e^{iph_3} y_0^{0+}(\mathbf{r}_0) \quad (\text{B-1})$$

so that the spin density is given by

$$\begin{aligned} |\phi_{sp}(\mathbf{r})|^2 = & R_p^2(r) \left[ \sum_{m,p} Dpl2(p,m) y_2^{m,p}(\mathbf{r}_0) + Dpl0(0) y_0^{0+}(\mathbf{r}_0) \right] + \\ & + R_s(r) R_p(r) \left[ \sum_{m,p} Dspl1(p,m) y_1^{m,p}(\mathbf{r}_0) \right] + R_s^2(r) Dsl0(0) y_0^{0+}(\mathbf{r}_0) \end{aligned} \quad (\text{B-2})$$

The following Fortran code summarises the calculations of the coefficients Dspl1, Dpl2, Dpl0 and Dsl0 as a function of the parameters  $a, b, c, d, ph_1, ph_2$  and  $ph_3$  of expression B-1.

```
real, PARAMETER ::
  a01= 0.2185097040, a02= -0.2185097040, a03= -0.1261566430, &
  a04= 0.2523132860, a05= 0.2820947770

  Dpl2( 2)= a01 *a*a +a02 *b*b
  Dpl2( 1)= a01 *2.0*a*c*cos(ph2)
  Dpl2( 0)= a03 (a*a + b*b) +a04 *c*c
  Dpl2(-1)= a01 *2.0*b*c*cos(ph1-ph2)
  Dpl2(-2)= a01 *2.0*a*b*cos(ph1)
  Dpl0( 0)= a05 * (a*a + b*b + c*c)
  Dspl1( 1)= a05 *2.0*a*d*cos(ph3)
  Dspl1( 0)= a05 *2.0*c*d*cos(ph2-ph3)
  Dspl1(-1)= a05 *2.0*b*d*cos(ph1-ph3)
  Dsl0( 0)= a05 *d*d
```

### C. $d$ Orbital (QUAD)

A general  $d$  orbital is expressed as:

$$\phi_d(\mathbf{r}) = R_d(r) \left[ a y_2^{2+}(\mathbf{r}_0) + b e^{iph_1} y_2^{2-}(\mathbf{r}_0) + c e^{iph_2} y_2^{1+}(\mathbf{r}_0) + d e^{iph_3} y_2^{1-}(\mathbf{r}_0) + e e^{iph_4} y_2^{0+}(\mathbf{r}_0) \right] \quad (\text{C-1})$$

so that the spin density is given by

$$|\phi_d(\mathbf{r})|^2 = R_d^2(r) \left[ \sum_{m,p} Ddl4(p,m) y_4^{m,p}(\mathbf{r}_0) + \sum_{m,p} Ddl2(p,m) y_2^{m,p}(\mathbf{r}_0) + Ddl0(0) y_0^{0+}(\mathbf{r}_0) \right] \quad (\text{C-2})$$

The following Fortran code summarises the calculations of the coefficients Ddl4, Ddl2, and Ddl0 as a function of the parameters  $a, b, c, d, e, ph_1, ph_2, ph_3$  and  $ph_4$  of expression C-1.

```
real, PARAMETER ::
  a01= 0.2384136320, a02= -0.2384136320, a03= 0.1685838850, &
  a04= -0.1685838850, a05= 0.1802237630, a06= -0.1802237630, &
  a07= 0.1560783540, a08= -0.0637187213, a09= 0.2207281290, &
  a10= 0.0402992591, a11= -0.1611970370, a12= 0.2417955550, &
  a13= 0.0637187213, a14= -0.1560783540, a15= 0.0901118815, &
  a16= 0.2820947770

  Ddl4( 4)= a01 *a*a +a02 *b*b
  Ddl4( 3)= a03 *2.0*a*c*cos(ph2) +a04 *2.0*b*d*cos(ph1-ph3)
  Ddl4( 2)= a05 *c*c +a06 *d*d +a07 *2.0*a*e*cos(ph4)
  Ddl4( 1)= a08 *2.0*a*c*cos(ph2) +a08 *2.0*b*d*cos(ph1-ph3) +a09 *2.0*c*e*cos(ph2-ph4)
  Ddl4( 0)= a10 *a*a +a10 *b*b +a11 *c*c +a11 *d*d +a12 *e*e
  Ddl4(-1)= a13 *2.0*a*d*cos(ph3) +a08 *2.0*b*c*cos(ph1-ph2) +a09 *2.0*d*e*cos(ph3-ph4)
```

```

Ddl4(-2)= a07 *2.0*b*e*cos(ph1-ph4) +a05 *2.0*c*d*cos(ph2-ph3)
Ddl4(-3)= a03 *2.0*a*d*cos(ph3) +a03 *2.0*b*c*cos(ph1-ph2)
Ddl4(-4)= a01 *2.0*a*b*cos(ph1)
Ddl2( 2)= a07 *c*c +a14 *d*d +a06 *2.0*a*e*cos(ph4)
Ddl2( 1)= a07 *2.0*a*c*cos(ph2) +a07 *2.0*b*d*cos(ph1-ph3) +a15 *2.0*c*e*cos(ph2-ph4)
Ddl2( 0)= a06 *a*a +a06 *b*b +a15 *c*c +a15 *d*d +a05 *e*e
Ddl2(-1)= a14 *2.0*a*d*cos(ph3) +a07 *2.0*b*c*cos(ph1-ph2) +a15 *2.0*d*e*cos(ph3-ph4)
Ddl2(-2)= a06 *2.0*b*e*cos(ph1-ph4) +a07 *2.0*c*d*cos(ph2-ph3)
Ddl0( 0)= a16 *( a*a + b*b + c*c + d*d + e*e )

```

## D. $f$ Orbital (HXAP)

A general  $f$  orbital is expressed as:

$$\phi_f(\mathbf{r}) = R_f(r) \left[ \begin{aligned} &a y_3^{3+}(\mathbf{r}_0) + b e^{iph_1} y_3^{3-}(\mathbf{r}_0) + c e^{iph_2} y_3^{2+}(\mathbf{r}_0) + d e^{iph_3} y_3^{2-}(\mathbf{r}_0) + e e^{iph_4} y_3^{1+}(\mathbf{r}_0) + \\ &f e^{iph_5} y_3^{1-}(\mathbf{r}_0) + g e^{iph_6} y_3^{0+}(\mathbf{r}_0) \end{aligned} \right] \quad (\text{D-1})$$

so that the spin density is given by

$$|\phi_f(\mathbf{r})|^2 = R_f^2(r) \left[ \sum_{m,p} Dfl6(p,m) y_6^{m,p}(\mathbf{r}_0) + \sum_{m,p} Dfl4(p,m) y_4^{m,p}(\mathbf{r}_0) + \sum_{m,p} Dfl2(p,m) y_2^{m,p}(\mathbf{r}_0) + Dfl0(0) y_0^{0+}(\mathbf{r}_0) \right] \quad (\text{D-2})$$

The following Fortran code summarises the calculations of the coefficients Ddl4, Ddl2, and Ddl0 as a function of the parameters  $a, b, c, d, e, ph_1, ph_2, ph_3$  and  $ph_4$  of expression C-1.

```

real, PARAMETER ::
a01= 0.2548005880, a02= -0.2548005880, a03= 0.1801712210, &
a04= -0.1801712210, a05= 0.1881827120, a06= -0.1881827120, &
a07= 0.1214714200, a08= -0.1214714200, a09= 0.1086473390, &
a10= 0.1629710050, a11= -0.1629710050, a12= 0.1717865170, &
a13= -0.1717865170, a14= -0.0443550907, a15= 0.1774203480, &
a16= 0.0221775454, a17= -0.0858932585, a18= 0.2217754420, &
a19= -0.0118543962, a20= 0.0711263791, a21= -0.1778159290, &
a22= 0.2370879200, a23= -0.0221775454, a24= 0.0858932585, &
a25= 0.0443550907, a26= 0.1517177520, a27= -0.1517177520, &
a28= -0.1175200640, a29= 0.1175200640, a30= -0.2035507110, &
a31= 0.0678502396, a32= -0.0678502396, a33= 0.1146878450, &
a34= -0.1146878450, a35= 0.1332552280, a36= -0.0444184095, &
a37= -0.0993225873, a38= 0.1025799290, a39= 0.0993225798, &
a40= 0.0769349411, a41= -0.1795148700, a42= 0.0256449804, &
a43= 0.1538698820, a44= -0.1025799290, a45= -0.1332552280, &
a46= 0.1456731260, a47= -0.1456731260, a48= -0.0940315947, &
a49= -0.1880631890, a50= 0.1486770060, a51= 0.1151647120, &
a52= 0.0594708063, a53= -0.2102610470, a54= 0.1261566130, &
a55= 0.1682088380, a56= -0.1486770060, a57= -0.1151647120, &
a58= 0.0940315947, a59= 0.2820948060

Dfl6( 6)= a01 *a*a +a02 *b*b
Dfl6( 5)= a03 *2.0*a*c*cos(ph2) +a04 *2.0*b*d*cos(ph1-ph3)
Dfl6( 4)= a05 *c*c +a06 *d*d +a07 *2.0*a*e*cos(ph4) +a08 *2.0*b*f*cos(ph1-ph5)
Dfl6( 3)= a09 *2.0*a*g*cos(ph6) +a10 *2.0*c*e*cos(ph2-ph4) +a11 *2.0*d*f*cos(ph3-ph5)
Dfl6( 2)= a12 *e*e +a13 *f*f +a14 *2.0*a*e*cos(ph4) +a14 *2.0*b*f*cos(ph1-ph5) + &
a15 *2.0*c*g*cos(ph2-ph6)
Dfl6( 1)= a16 *2.0*a*c*cos(ph2) +a16 *2.0*b*d*cos(ph1-ph3) +a17 *2.0*c*e*cos(ph2-ph4) +&
a17 *2.0*d*f*cos(ph3-ph5) +a18 *2.0*e*g*cos(ph4-ph6)
Dfl6( 0)= a19 *a*a +a19 *b*b +a20 *c*c +a20 *d*d +a21 *e*e +a21 *f*f +a22 *g*g
Dfl6(-1)= a23 *2.0*a*d*cos(ph3) +a16 *2.0*b*c*cos(ph1-ph2) +a24 *2.0*c*f*cos(ph2-ph5) +&
a17 *2.0*d*e*cos(ph3-ph4) +a18 *2.0*f*g*cos(ph5-ph6)
Dfl6(-2)= a25 *2.0*a*f*cos(ph5) +a14 *2.0*b*e*cos(ph1-ph4) +a15 *2.0*d*g*cos(ph3-ph6) +&
a12 *2.0*e*f*cos(ph4-ph5)
Dfl6(-3)= a09 *2.0*b*g*cos(ph1-ph6) +a10 *2.0*c*f*cos(ph2-ph5) +a10 *2.0*d*e*cos(ph3-ph4)
Dfl6(-4)= a07 *2.0*a*f*cos(ph5) +a07 *2.0*b*e*cos(ph1-ph4) +a05 *2.0*c*d*cos(ph2-ph3)
Dfl6(-5)= a03 *2.0*a*d*cos(ph3) +a03 *2.0*b*c*cos(ph1-ph2)

```



```

Df16(-6)= a01 *2.0*a*b*cos(ph1)
Df14( 4)= a26 *c*c +a27 *d*d +a28 *2.0*a*e*cos(ph4) +a29 *2.0*b*f*cos(ph1-ph5)
Df14( 3)= a30 *2.0*a*g*cos(ph6) +a31 *2.0*c*e*cos(ph2-ph4) +a32 *2.0*d*f*cos(ph3-ph5)
Df14( 2)= a33 *e*e +a34 *f*f +a35 *2.0*a*e*cos(ph4) +a35 *2.0*b*f*cos(ph1-ph5) + &
a36 *2.0*c*g*cos(ph2-ph6)
Df14( 1)= a37 *2.0*a*c*cos(ph2) +a37 *2.0*b*d*cos(ph1-ph3) +a38 *2.0*c*e*cos(ph2-ph4) +&
a38 *2.0*d*f*cos(ph3-ph5) +a39 *2.0*e*g*cos(ph4-ph6)
Df14( 0)= a40 *a*a +a40 *b*b +a41 *c*c +a41 *d*d +a42 *e*e +a42 *f*f +a43 *g*g
Df14(-1)= a39 *2.0*a*d*cos(ph3) +a37 *2.0*b*c*cos(ph1-ph2) +a44 *2.0*c*f*cos(ph2-ph5) +&
a38 *2.0*d*e*cos(ph3-ph4) +a39 *2.0*f*g*cos(ph5-ph6)
Df14(-2)= a45 *2.0*a*f*cos(ph5) +a35 *2.0*b*e*cos(ph1-ph4) +a36 *2.0*d*g*cos(ph3-ph6) +&
a33 *2.0*e*f*cos(ph4-ph5)
Df14(-3)= a30 *2.0*b*g*cos(ph1-ph6)+a31*2.0*c*f*cos(ph2-ph5)+ a31 *2.0*d*e*cos(ph3-ph4)
Df14(-4)= a28 *2.0*a*f*cos(ph5) +a28 *2.0*b*e*cos(ph1-ph4) +a26 *2.0*c*d*cos(ph2-ph3)
Df12( 2)= a46 *e*e +a47 *f*f +a48 *2.0*a*e*cos(ph4) +a48 *2.0*b*f*cos(ph1-ph5) + &
a49 *2.0*c*g*cos(ph2-ph6)
Df12( 1)= a50 *2.0*a*c*cos(ph2) +a50 *2.0*b*d*cos(ph1-ph3) +a51 *2.0*c*e*cos(ph2-ph4) +&
a51 *2.0*d*f*cos(ph3-ph5) +a52 *2.0*e*g*cos(ph4-ph6)
Df12( 0)= a53 *a*a +a53 *b*b +a54 *e*e +a54 *f*f +a55 *g*g
Df12(-1)= a56 *2.0*a*d*cos(ph3) +a50 *2.0*b*c*cos(ph1-ph2) +a57 *2.0*c*f*cos(ph2-ph5) +&
a51 *2.0*d*e*cos(ph3-ph4) +a52 *2.0*f*g*cos(ph5-ph6)
Df12(-2)= a58 *2.0*a*f*cos(ph5) +a48 *2.0*b*e*cos(ph1-ph4) +a49 *2.0*d*g*cos(ph3-ph6) +&
a46 *2.0*e*f*cos(ph4-ph5)
Df10( 0)= a59 *( a*a + b*b + c*c + d*d + e*e + f*f + g*g )

```