FAULTS manual

Montse Casas Cabanas (CIC Energigune), Jokin Rikarte Ormazabal
(CIC Energigune), Marine Reynaud (CIC Energigune)
and Juan Rodriguez Carvajal (ILL)

December 2015

Contents

Chapter 1

The FAULTS program

FAULTS is a Fortran90 program to refine X-ray Powder Diffraction (XRD) and Neutron Powder Diffraction (NPD) patterns of crystal systems with any type of coherent planar defect. This program is based on the DIFFaX program [1, 2] and on the library CrysFML (Crystallographic Fortran Modules Library) [3, 4].

The DIFFaX program (Diffracted Intensities From Faulted Xtals) calculates diffraction intensities from defective layered crystals. This tool has been widely used to interpret the diffraction data of one-dimensionally disordered systems and it is based on an algorithm that exploits the recursive nature of the patterns found in randomized stacking sequences to compute the average interference wavefunction scattered from each layer type occurring in a faulted crystal (mathematical details can be found in references [5, 6]). However, approximate or merely qualitative results sometimes are not sufficient for a thorough microstructural characterization, a computerized comparison of the DIFFaX calculated intensities with experimental data has been developed. The resulting code is the FAULTS

program [7, 8]. Note that FAULTS, as DIFFaX, can also simulate 2D diffraction patterns and diffuse streaks in reciprocal space for single crystals.

The program is distributed in the hope that it will be useful, but without any warranty of being free of internal errors. The authors acknowledge all suggestions and notification of possible bugs found in the program.

1.1 Download and installation

FAULTS can be obtained either as part of the FullProf Suite [9, 10, 11] at www.ill.eu/sites/fullprof/, for Windows, Linux or MacOS, or as a separate program for Windows by downloading the compressed file FAULTS.zip at www.cicenergigune.com/en/areas-investigacion/power-storage-batteries-and-supercaps/lineas-investigacion/li-ion/faults/.

In case the user chooses to install FAULTS with the FullProf Suite, the program FAULTS can be launched directly from the FullProf Suite Toolbar or by opening a terminal and invoking the program as: Faults myinputfile.flts. If FAULTS is downloaded as a separate Windows program, it should be launched from a command-line window (cmd.exe), after having copied the standard scattering factor data file *data.sfc* into the working folder.

1.2 Program specifications

Conventionally, crystals are thought in terms of unit cells, atoms in the asymmetric unit and a determined symmetry space group. Nevertheless,

to use FAULTS, as in the case of DIFFaX [12], one needs to think of crystals in terms of atomic sheets, or layers, which can be interconnected via stacking operations occurring with a certain probability. By means of this description, planar defects can be described as different layer types and/or transition vectors. Frequently, the plane of layers will not coincide conveniently with any of the unit cell faces of the parent crystal. Then, a transformation of atom coordinates to a new cell system is required.

The structural information as well as the refinement details are read by FAULTS from a free format input data file, with *.flts* extension. Each value used to describe the structure is associated to a refinement code that allows the possibillity of restrictions. The user must be aware of the way he(she) can control the refinement procedure: the number of parameters to be refined, fixing parameters, making constraints, etc. The experimental XRD or NPD patterns can be read from many different formats and background subtraction can be achieved by linear interpolation or polynomially after applying the scale factor. Impurities or other phases are treated as background as well, and their pattern information must be given by the user. Another major feature of FAULTS is the implementation of a more adequate isotropic size broadening treatment which takes into account the Gaussian and Lorentzian contributions to the FWHM in addition to the consideration of a finite number of layers per crystallite already present in DIFFaX [1, 2, 12]. The refinement is carried out using a Levenberg Marquardt fit. The quality of the agreement between observed and calculated profiles is given by the R-Factor and Chi^2 agreement factors that are calculated at the end of each refinement cycle and are defined as follows:

Profile factor:

$$Rp = 100 \frac{\sum_{i=1,n} \mid y_i - y_{ic} \mid}{\sum_{i=1,n} y_i}$$

where y_i is the profile intensity and y_{ic} is the number of calculated counts at the ith step.

Reduced Chi square:

$$Chi^2 = \left[\frac{R_{wp}}{R_{exp}} \right]^2$$

being $R_{\omega p}$ and R_{exp} the Weighted Profile Factor and the Expected Weighted Profile Factor respectively, which are defined as:

$$R_{wp} = 100 \left[\frac{\sum_{i=1,n} w_i \mid y_i - y_{ic} \mid^2}{\sum_{i=1,n} w_i y_i^2} \right]^{1/2}$$

and

$$R_{exp} = 100 \left[\frac{n-p}{\sum_i w_i y_i^2} \right]^{1/2}$$

where n-p is the number of degrees of freedom and $w_i = 1/\sigma_i^2$, with σ_i^2 referring to the variance of the "obserbation" y_i.

1.3 Input files

1.3.1 The input control file: CFILE.flts

The input control file is a free format file that contains all the structural data and the type of calculation to be done. This file must be written by the user.

The FAULTS input file is similar to the one used in DIFFaX [12], although there are some differences that should be taken into account to avoid errors. The DIFFaX2FAULTS convertor, which permits to convert DIFFaX input files into FAULTS ones, is available at www.cicenergigune.com/en/areas-investigacion/power-storage-batteries-and-supercaps/lineas-investigacion/li-ion/faults/.

Free format means that the input file is not case sensitive and that the different sections do not have to follow a concret sequence, however, a space is needed between each item and all the section headings and subsection keywords must be present. Each line must not exceed 132 characters, taking into account the blank spaces. When the program is run, mistakes will normally generate error messages. Empty lines as well as lines starting with the exclamation symbol (*!*) in the first column are considered as comments and are ignored by the program. Also, comments can be placed at the end of a line if they are in braces ({}). It is recommended not to use tabulator to introduce blank spaces when editing the file, as some editors do not consider it as blank spaces.

An example of input control file can be found in figures 1.1, 1.2 and 1.3. The different section headings that constitute the input control file appear in orange color, comments appear in italics (and of course after a *!* symbol), subsection keywords in red, parameters in blue and refinement codes in green. The latter are codewords that allow the control of the refined parameters. As in the FullProf program [10, 11], these are the numbers C_x that are entered for each refined parameter. A zero codeword means that the parameter is not being refined. For each refined parameter, the codeword is formed as:

$$C_x = sign(a)(10p + |a|)$$

where p specifies the ordinal number of the parameter x and a is the factor by which the computed shift (the parameter variation in each refinement cycle) will be multiplied before use.

Below, each section, which has to begin with the corresponding heading, is explained in more detail.

Title **section**

The document must begin with this section. TITLE heading must be followed in the next line by the title chosen by the user.

Instrumental and size broadening **section**

The first line must contain the heading INSTRUMENTAL AND SIZE BROADENING. The data that follow describe the type of radiation and the instrumental conditions as well as the parameters needed for size broadening, which are refinable parameters. To refine the parameters that constitute this section it is not necessary to recalculate the diffraction pattern in each cycle, so, for computational economy it is strongly recommended to refine them separately.

- **Radiation type:** The line must begin with the keyword *radiation*, followed by one of the different three options, which are X-RAY, NEUTRON and ELECTRON (the latter for the simulation of selected area diffraction patterns -SADP-).

```
TITLE
LiNiO2 ideal structure

Instrumental And Size Broadening
!Type of Radiation
 Radiation     X-Ray
!          lambda1   lambda2   ratio
 Wavelength    1.5406   0.0      0.0
!Instrumental aberrations     zero     sycos     sysin
 Aberrations              0.0000   0.0000   0.0000
                          0.00     0.00     0.00
!Instr. broadening      u         v         w         x         Dg        Dl
 Pseudo-Voigt          0.11     -0.0036   0.009     0.7000    479.26    459.87   Trim
                       0.00      0.00     0.00      0.00      0.00      0.00

 Structural
SPGR   R -3 m
 !              a         b          c       alpha   beta  gamma
 Avercell  2.8659    2.8659    14.253     90    90    120.0
 !          a          b          c          gamma
 Cell     2.8659     2.8659     14.253     120.0
         11.00      11.00      1.00       0.00
! FullProf Studio commands
FST_CMD  SEQ  9 1 2 3 1 2 3 1 2 3
FST_CMD  CONN Ni O 0.0 2.3
FST_CMD  box  -2 2 -2 2 -0.01 1.01
FST_CMD  stack_vect  0 0 0.5
!Laue symmetry
 Symm  unknown
!Number of layer types
 Nlayers  3
!Layer width
 Lwidth  infinite

 Layer 1
!Layer symmetry
 LSYM   none
!Atom name   number  x       y       z       Biso    Occ
 Atom Ni3+      1     0.000   0.000   0.000   0.8     1.0
                      0.00    0.00    0.00    1.00    0.00
!Atom name   number  x       y       z       Biso    Occ
 Atom O2-       2     1/3     2/3     -0.075  1.0     1.0
                      1.00    0.00    0.00    0.00    0.00
!Atom name   number  x       y       z       Biso    Occ
 Atom O2-       3     2/3     1/3     0.075   1.0     1.0
                      0.00    0.00    0.00    0.00    0.00
```


Figure 1.1: **Example of input control file, corresponding to the ideal structure of $LiNiO_2$.**

```
!Atom name   number  x       y       z        Biso    Occ
Atom Li1+     4      2/3     1/3     -0.167   0.8     1.0
                     0.00    0.00    1.00     1.00    0.00

Layer 2 = 1

Layer 3 = 1

Stacking
!Stacking type
 Recursive
!Number of layers
 infinite

Transitions
!Layer  1 to layer  1
LT  0.00000000  0.000000  0.000000  1/3
   0.000000   0.000000  0.000000  0.000000
FW     0.00     0.00     0.00     0.00     0.00     0.00
      0.00     0.00     0.00     0.00     0.00     0.00
!Layer  1 to layer  2
LT  1.00  2/3      1/3      1/3
   21.000000  0.000000  0.000000  0.000000
FW     0.00     0.00     0.00     0.00     0.00     0.00
      0.00     0.00     0.00     0.00     0.00     0.00
!Layer  1 to layer  3
LT  0.000000  0.000000  0.000000  0.000000
   0.000000  0.000000  0.000000  0.000000
FW     0.00     0.00     0.00     0.00     0.00     0.00
      0.00     0.00     0.00     0.00     0.00     0.00
!Layer  2 to layer  1
LT  0.000000  0.000000  0.000000  0.000000
   0.000000  0.000000  0.000000  0.000000
FW     0.00     0.00     0.00     0.00     0.00     0.00
      0.00     0.00     0.00     0.00     0.00     0.00
!Layer  2 to layer  2
LT  0.00000  0.000000  0.000000  0.0000
   0.000000  0.000000  0.000000  0.000000
FW     0.00     0.00     0.00     0.00     0.00     0.00
      0.00     0.00     0.00     0.00     0.00     0.00
!Layer  2 to layer  3
LT  1.00  2/3      1/3      1/3
  31.000000  0.000000  0.000000  0.000000
FW     0.00     0.00     0.00     0.00     0.00     0.00
      0.00     0.00     0.00     0.00     0.00     0.00
!Layer  3 to layer  1
```


Figure 1.2: **Cont. Example of input control file, corresponding to the ideal structure of $LiNiO_2$.**

```
LT  1.00  2/3      1/3     1/3
   41.000000  0.000000  0.000000  0.000000
FW     0.00     0.00     0.00     0.00     0.00     0.00
       0.00     0.00     0.00     0.00     0.00     0.00
!Layer  3 to layer  2
LT  0.000000  0.000000  0.000000  0.000000
   0.000000  0.000000  0.000000  0.000000
FW     0.00     0.00     0.00     0.00     0.00     0.00
       0.00     0.00     0.00     0.00     0.00     0.00
!Layer  3 to layer  3
LT  0.00000  0.000000  0.000000  0.0000
   0.000000  0.000000  0.000000  0.000000
FW     0.00     0.00     0.00     0.00     0.00     0.00
       0.00     0.00     0.00     0.00     0.00     0.00

Calculation
 Lma
 Corrmax     30
 Maxfun     2400
 Tol      0.100000E-04
 Nprint     0

Experimental
!Filename                        Scale factor    code
 FILE  LiNiO2_ ideal.dat          1.0            0.00
 Excluded_Regions  0
 FFORMAT  free
!Linear interpolation
 Bgrinter    sim.bgr
!Polynomial  Number of coefficients
 Bgrcheb    2
!Polynomial coefficients
   1.00000    0.20000
   0.0        0.0
!number of pattern backgrounds
 Bgrnum  1
!Pattern file        Filename     Scale factor    code
Bgrpatt         phase2.sub      0.026000      1.00   phase2.hkl
```


Figure 1.3: **Cont. Example of input control file, corresponding to the ideal structure of $LiNiO_2$.**

- **Wavelength:** In addition to λ_1, FAULTS reads also the value of λ_2 and the intensity ratio I_2/I_1 (although in the case of monochromatic radiation it is only necessary to introduce λ_1). λ unit is Å and this line must begin with the keyword *wavelength*.

- **Aberrations:** Keyword *aberrations* followed by values for the instrumental aberrations (zero, sycos and sysin) must be given. As they are refinable, refinement codes must be entered in the next line. The parameters sycos and sysin relate to systematic 2Theta shifts having a $cos\theta$ and $sin\theta$ dependency, respectively, that originate from different physical and/or geometrical problems depending on the diffraction geometry (see FullProf manual for more details [11]).

- **Profile parameters:** Instrumental and size broadening are treated differently than in DIFFaX. In both cases, the calculated pattern is convoluted with a profile function (indicated by the keyword *Gaussian*, *Lorentzian* or *Pseudo-Voigt*) that takes into account both instrumental and size effects. The FWHM of the Gaussian (H_G) and Lorentzian (H_L) components of the peak profile have an angular dependence given by:

$$H_G^2 = Utan^2\theta + Vtan\theta + W + \frac{4ln2\lambda^2}{\pi D_G^2 cos^2\theta}(\frac{180}{\pi})^2$$

$$H_L = Xtan\theta + \frac{2\lambda}{\pi D_L cos\theta}(\frac{180}{\pi})$$

U, V, W and X, which must be introduced by the user, are refinable parameters that constitute the instrumental resolution function (IRF). It is always advisable to know their values a priori and fix them during the refinement (although U and X can be refined to account

for strains). They must be followed by D_G and D_L, which are the parameters that are used to model the isotropic size broadening due to a finite size of the crystallites (≤ 1 μm). They are also refinable. If the sample is not affected by size broadening, D_G and D_L can take any value ≥ 5000 Å.

- **TRIM keyword:** As in DIFFaX [12], the optional keyword *trim*, which has to be put in the same line as the profile parameters, tells the program to ignore intensity information close to the origin when simulating the instrumental broadening for a powder diffraction pattern. The peak at the origin is usually many orders of magnitude more intense than any other peak of the diffraction pattern, and the background that is generated by the tail of this peak, when broadened, can easily swamp peaks close to the origin. When *trim* is specified, the unbroadened pattern that is written to file retains the data at the origin, whereas the broadened data that is written to file suppresses the peak at the origin. It is generally desirable to specify *trim* whenever the powder pattern angular range includes the origin. *trim* has no effect if the powder pattern angular range requested by the user does not include the origin.

***Structural* section**

This section must begin with a first line containing the heading STRUCTURAL.

- **Cell parameters:** The keyword which this line has to begin with is *cell*. The value of **c**, perpendicular to the a-b plane (and therefore parallel to the stacking direction), is needed to provide a reference scale along the stacking direction with which to define the

z-component of the stacking vectors. **c** does not have to correspond to any special periodic dimension along the stacking direction for any of the layers. However all the layers must share the same dimensions, **a**, **b** and γ. These are all refinable parameters and the units are Å for **a, b** and **c**, and degrees for γ.

- **Laue symmetry:** In this case the keyword is *symm*. There are 12 options : **-1, 2/M(1) , 2/M(2), MMM, -3, -3M, 4/M, 4/MMM, 6/M, 6/MMM, AXIAL** and **UNKOWN**. Axial constraints the program to integrate only along $00l$ (useful in turbostratic structures) and if UNKNOWN is specified the program will establish the symmetry by randomly sampling the reciprocal space. In this case, the user can specify a percentage of tolerance on intensity deviations for this search. Otherwise the program tests the user option and, if it is not consistent with the cell parameters, it is automatically changed.

- **Number of layer types:** The keyword is *nlayers*. It is the number of different layer types needed to describe the structure. If two layers are structurally identical but their stacking vectors are different, they are considered as two different layer types.

- **Layer caracteristic width:** Unlike in DIFFaX, this line is compulsory, and it must begin with the keyword *lwidth*. Then the layer characteristic width is especified, in Å. There are two possible formats for this line: width along a and width along b, both refinable values, or *infinite*. For example,

```
!layer width
Lwidth 120.00 300.00
       0.0     0.0
```

or

```
!layer width
Lwidth infinite
```

- **Optional generation of the Bragg positions of the average cell:** In case the cell parameters defined for the FAULTS refinement are different from the original cell (the cell from which the disordered model comes from; for instance, in the case of a monoclinic or triclinic average cell), the user can ask the program to generate the Bragg positions corresponding to the average cell in the output file CFILEn.prf (see section 1.4.3). To do so, the user has two options:
(1) either he(she) indicates manually the space group (after the keyword *SPGR*) and the cell parameters a, b, c, α, β, γ (after the keyword *Avercell*) of the average cell by adding the following optional lines in the *Structural* section:

```
!Average cell parameters (for Bragg positions in .prf file)
Avercell 3.12 4.15 4.60 90.00 118.79 90.0
!Average space group (for Bragg positions in .prf file)
SPGR C 2/m
```

(2) or, by means of the keywords *CalcAverCell* and *SPGR*, he(she) asks the program to calculate the average cell from a sequence of layers that he(she) specifies with the following optional lines in the *Structural* section:

```
!Number of stacking vectors needed to calculate the average cell
CalcAvercell 2
!Specific sequence to calculate the average cell
 1 2
!Average space group (for Bragg positions in .prf file)
SPGR C 2/m
```

Note that the average cell parameters and space group are only used for the Bragg positions calculation for the .prf file and are not taken into account for the pattern calculation during the refinement or simulation.

- **Optional commands for generation of a .CIF file and visualization of the structure with FullProf Studio and VESTA:** To simplify the construction of the structural section and to avoid errors, it is recommended to regularly draw the structure with the help of a visualization program, such as FullProf Studio [13, 14] or VESTA [15, 16]. For this, the program describes the stacking of layers in a big artificial box containing all the atoms of the layers that are asked to be painted by the user. To do so, the user should ask the program to generate additional the output files *.fst*, *.cif* and *.vesta* (see section 1.4.4) suitable for FullProf Studio and VESTA, respectively , by entering an optional line beginning with the keyword *FST_CMD*, followed by the keyword *SEQ*, the number of layers and the sequence of the layers to be drawn; for example:

```
!FullProf Studio commands
FST_CMD SEQ 5    1 1 2 3 1
```

Then the user can add as many optional lines as needed, each of them beginning with the keyword *FST_CMD*, followed by the required FullProf Studio commands (see the FullProf Studio manual for more details [14]). For example, to control respectively the drawing of Ni-O bonds (of length between 0.0 and 2.3 Å) polyhedra of the example input file, the limits of the box and the position of the first layer in the artificial box, the user may add the following lines:

```
FST_CMD CONN   Ni O   0.0 2.3
FST_CMD BOX   -2.0 2.0   -2.0 2.0   -0.01 1.01
FST_CMD stack_vect   0 0 0.5
```

***Layers* section**

- **Layer number:** The layers have to be described with the header *LAYER* followed by the ordinal corresponding number. If two layers are structurally identical it will be enough to write *LAYER* $j = i$ where i and j are the ordinal numbers that correspond to each layer (a blank space is necessary between the number of layer and the = symbol). Then, for layer j the rest of the section can be avoided.

- **Layer symmetry:** The keyword *lsym* must be followed by one of the two options: **NONE** or **CENTROSYMMETRIC**. If the latter is specified, only the asymmetric half of the atom coordinates need to be entered, and the program will create another identical atom in (-x,-y,-z) per each atom described in the layer. Take care to halve the occupancy in case of the atoms located at any inversion center.

- **Atomic data:** For each atom the keyword *atom* must be written, followed by a specific 4 characters name. The first two characters correspond to the atomic symbol and the last two represent the valence.

If the symbol has only one character, contrary to DIFFAX, there must not be a space between the symbol and the charge (Ex. F1-, Ni2+). Some atoms do not require any valence (ex. C). Below is the list of the names accepted by FAULTS (this list is found in the standard scattering factor data file *data.sfc*):

H ; H. ; D ; H1- ; He ; Li ; Li1+ ; Be ; Be2+ ; B ; C ; C. ; N ; O ; O1- ; O2- ; F ; F1- ; Ne ; Na ; Na1+ ; Mg ; Mg2+ ; Al ; Al3+ ; Si ; Si. ; Si4+ ; P ; S ; Cl ; Cl1- ; Ar ; K ; K1+ ; Ca ; Ca2+ ; Sc ; Sc3+ ; Ti ; Ti2+ ; Ti3+ ; Ti4+ ; V ; V2+ ; V3+ ; V5+ ; Cr ; Cr2+ ; Cr3+ ; Mn ; Mn2+ ; Mn3+ ; Mn4+ ; Fe ; Fe2+ ; Fe3+ ; Co ; Co2+ ; Co3+ ; Ni ; Ni2+ ; Ni3+ ; Cu ; Cu1+ ; Cu2+ ; Zn ; Zn2+ ; Ga ; Ga3+ ; Ge ; Ge4+ ; As ; Se ; Br ; Br1- ; Kr ; Rb ; Rb1+ ; Sr ; Sr2+ ; Y ; Y3+ ; Zr ; Zr4+ ; Nb ; Nb3+ ; Nb5+ ; Mo ; Mo3+ ; Mo5+ ; Mo6+ ; Tc ; Ru ; Ru3+ ; Ru4+ ; Rh ; Rh3+ ; Rh4+ ; Pd ; Pd2+ ; Pd4+ ; Ag ; Ag1+ ; Ag2+ ; Cd ; Cd2+ ; In ; In3+ ; Sn ; ; Sn2+ ; Sn4+ ; Sb ; Sb2+ ; Sb5+ ; Te ; I ; I1- ; Xe ; Cs ; Cs1+ ; Ba ; Ba2+ ; La ; La3+ ; Ce ; Ce3+ ; Ce4+ ; Pr ; Pr3+ ; Pr4+ ; Nd ; Nd3+ ; Pm ; Pm3+ ; Sm ; Sm3+ ; Eu ; Eu2+ ; Eu3+ ; Gd ; Gd3+ ; Tb ; Tb3+ ; Dy ; Dy3+ ; Ho ; Ho3+ ; Er ; Er3+ ; Tm ; Tm3+ ; Yb ; Yb2+ ; Yb3+ ; Lu ; Lu3+ ; Hf ; Hf4+ ; Ta ; Ta5+ ; W ; W6+ ; Re ; Os ; Os4+ ; Ir ; Ir3+ ; Ir4+ ; Pt ; Pt2+ ; Pt4+ ; Au ; Au1+ ; Au3+ ; Hg ; Hg1+ ; Hg2+ ; Tl ; Tl1+ ; Tl3+ ; Pb ; Pb2+ ; Pb4+ ; Bi ; Bi3+ ; Bi5+ ; Po ; At ; Rn ; Fr ; Ra ; Ra2+ ; Ac ; Ac3+ ; Th ; Th4+ ; Pa ; U ; U3+ ; U4+ ; U6+ ; Np ; Np3+ ; Np4+ ; Np6+ ; Pu ; Pu3+ ; Pu4+ ; Pu6+ ; Am ; Cm

After the name, the atom will be identified by an ordinal number. Then the atomic coordinates *x, y, z* will be detailed as well as the atomic displacement factor and finally the occupation, which has to

take a value between 0 and 1. If the layer is centrosymmetric, the atom located in the centre of symmetry will have 1/2 for full occupancy. All the parameters are refinable. They can be written as real numbers or as fractions, which ensures maximum machine precision (except for the atomic displacements, which cannot be given as a fraction).

***Stacking* section**

The first line of this section must start with the heading STACKING.

- **Stacking type:** There are two possibilities: EXPLICIT and RECURSIVE. For an explicit sequence of layers, the diffracted intensities will be calculated for a unique layer sequence that will be specified in the next line. For a recursive sequence of layers, the diffracted intensities are to be calculated for a statistical ensemble of crystallites, each with a distinct stacking sequence, but weighted by the probability that such a sequence will occur.

- **Explicit sequence:** If the stacking type is explicit, there are three possibilities. One of them is SPECIFIC, in which the user has to introduce a stacking sequence, up to 5000 layers. The list can occupy more than one line, and between each layer type there must be a space. Each line must not exceed 132 characters, taking into account the blank spaces. For example,

  EXPLICIT
  SPECIFIC
  1 1 2 1 3 1 2

  corresponds to a crystallite that consists of 7 layers of three different types. However, care has to be taken not to write a forbidden

transition (with a zero stacking probability). If this happens, the program will stop and give an error message.

Another possibility is the RANDOM M option which tells the program to generate a sequence of M layers (with a maximum of 5000). The probability that a specific layer-to-layer transition will occur in the sequence will be weighted by the transition probabilities listed in the TRANSITIONS section, described next.

Finally, the program can also generate semi-random sequences of two different layer types, that is to say, the user can control certain parts of the sequence. To do so, the option SEMIRANDOM M must be used, where M is the number of layers (also with a maximum of 5000). Below this line, the user can specify parts of the sequence by the keyword SEQ followed by the position of the first layer in the sequence, the position of the final layer and the two types of layers that will be alternated. For example,

```
EXPLICIT
SEMIRANDOM 60
SEQ 11 20 1 3
```

will generate a random sequence of layers but from layer 11 to layer 20 the sequence will be 1 3 1 3 1 3 1 3 1 3. If the sequence is incompatible with the last layer of the sequence that is generated automatically (due to the stacking probabilities), in this case layer number 10, this layer would be eliminated and the fixed sequence would move a position, starting in layer 10. If that was still not compatible with the stacking

probabilities this process would be repeated until the program found a compatible sequence.

- **Recursive sequence:** In this case, the user can indicate the mean number of layers the crystallites contain. Any number larger than M $\geq$1022, will be treated as infinite. Alternatively, an infinite number can be specified by the keyword INFINITE. If the latter is chosen, the refinement code underneath must be eliminated. If the number of layers is lower than 1022, the program will apply a line broadening due to a finite size in the stacking direction. This value (which can be refined) will have no physical meaning, but it is the only way to introduce a certain anisotropy in size broadening. However, if the crystallites induce a purely isotropic size broadening (represented by D_G and D_L , that will be lower than 5000) this value should be INFINITE as size broadening has already been taken into account in the *Instrumental and size broadening* section.

***Transitions* section**

First line must contain the heading TRANSITIONS. This section contains as many subsections as different layers constitute the system; and each subsection will contain as many lines (without taking into account the refinement codes) as different layers constitute the system. Each subsection refers to the transitions of layer i to the rest of layers, including itself. Each line must contain, after the keyword LT: the stacking probability α_{ij}, that will be a value comprised between zero (forbidden transition) and 1 (unique possible transition) and the stacking vector R_x, R_y, R_z, relative to **a, b** and **c** respectively. A set of anisotropic Debye-Waller type factors (C_{11}, C_{22}, C_{33}, C_{12}, C_{23}, C_{31}) is specified after the stacking vector, in a separate line

and beginning with the keyword FW. These factors are equivalent to specifying an ellipsoidal error spread for the stacking vectors and are useful for modeling systems where there is some coherence between nearest neighbor layers, but no long-range coherence, as in the case of liquid crystals or pillared clays. If the stacking probability α_{ij} is zero, everything on the line after it is ignored and set to zero. All the parameters constituting this section are refinable parameters and can be written as real numbers or as fractions.

***Calculation* section**

First line must contain the heading CALCULATION. This section contains the type of calculation the program is asked to perform. There are two options: SIMULATION and LMA. The first one corresponds to the pattern calculation, and thus no parameter will be refined. The second option indicates that the refinement is to be done by means of the Levenberg Marquardt fit. Depending on the calculation type, the next lines must contain:

- **Simulation:** In order to simulate a powder diffraction pattern the user has to specify the keyword *Powder* followed by the values of $2\theta_{min}$, $2\theta_{min}$ and the step (in degrees). Optionally, a scale factor and an offset value can be specified on the same line to simulate a pattern with background and noise generated from a Poisson distribution. For example,

  CALCULATION
  SIMULATION { type of optimization }
  POWDER 5.0 90.0 0.02 ScaleF 1.00 Bckg_Level 150.00

  If a selected area diffraction pattern is to be simulated, *SADP* key-

word must be followed by the values of i_plane, l_upper, loglin and brightness parameters, also used in DIFFaX. The i_plane parameter refers to the plane in reciprocal space to view. Choosing *1* views the plane containing h0l spots; *2* the plane containing 0kl spots; *3* the plane containing hhl spots and *4* the plane containing h-hl spots. The l_upper parameter corresponds to the maximum value of l to go out to. The maximum values of h and k will be automatically defined by this l value, since the calculated diffraction pattern will be square. The intensity data can be plotted on a linear or logarithmic scale, by defining loglin to be *0* or *1* respectively. In linear mode, the value of brightness refers to an intensity scaling factor. A brightness factor of 1 means that the intensity range (excluding the zero order beam) will be compressed so as to fit within the range 0 to 32767 and a birghtness factor of 10 means that the intensity range will be fitted within the range 0 to 327670, but all values above 32767 will be saturated, and will be set equal to 32767. The file created is an unsigned 16 bit image of size 256 x 256 with a **.sadp* extension. To display the file a specific software like ImageJ can be used [17].

For example:

> CALCULATION
> SIMULATION { type of optimization }
> SADP 1 4.0000 1 50.0000 { i_plane, l_upper, loglin, brightness }

For the streak simulation the user should put the *STREAK* keyword followed by the next values: adapt_quad, h, k, l0, l1 and dl. Optionally, one can put *UNBROADEN* keyword on the next line for calculating unbroadened streak data. Scale factor is equal to one for the streak

calculations. Data is written in free format and can be visualized using WinPLOTR. For example:

```
CALCULATION
SIMULATION { type of optimization }
STREAK 1 1 0 -5 5 0.01 { adapt_quad, h, k, l0, l1, dl }
UNBROADEN
```

- **LMA:** The user has to specify the maximum correlation parameter (which defines the minimum correlation used to build the correlation matrix that will be given in the output file; keyword *Corrmax*), the maximun number of function evaluations (keyword *Maxfun*), the tolerance (i.e. convergence criterion; keyword *Tol*) and the print control parameter (keyword *Nprint*). Each of these parameters must be placed in a new line. For example,

```
CALCULATION
LMA
Corrmax 30
Maxfun  2400
Tol     0.1E-04
Nprint  0
```

If the print control parameter *Nprint* is set to 0, the values of the agreement factors at the end of each evaluation are displayed in the FAULTS window (see subsection 1.5) during the refinement process. If *Nprint* is negative, the FAULTS windows displays the previous and new values of the refinable parameters at each evaluation step, in addition to the agreement factors, at the end of each evaluation.

Finally, whatever the kind of calculation to be performed (SIMULATION or LMA), the user can add an optional line with the instruction *Replace_Files* at the end of the *Calculation* section to make the program replace existent output files by the ones generated at the end of a new run, instead of creating new files.

Experimental **section**

This section is only necessary in case a refinement is to be done. The section must begin with a first line containing the heading EXPERIMENTAL, followed by:

- **Filename:** In this line, which must begin with the keyword *file*, the user indicates the file name of the experimental intensity data file, which must not exceed 20 characters, extension included (**.dat* or other ; see section 1.3.2). After that, the scale factor and its refinement code must be given, in the same line.

- **Excluded regions:** Optionally the user can exclude some regions of the experimental data. To do so, *Excluded_Regions* must be written, followed by the number of excluded regions. In the next lines, 2θ min and 2θ max of each excluded region must be specified (in a separate line per excluded region). If these are not detailed, the program will use the whole 2θ range of the experimental data.

- **File format:** In this line the user writes a code to tell the program which is the format of the intensity data file, that depends on the instrument. The possibilities (which are detailed in the next subsection 1.3.2) are: D1B, D20, NLS, G41, D1A, D2B, D1AOLD, D1BOLD, DMC, SOCABIM, XYSIGMA, GSAS, PANALYTICAL, TIMEVARIABLE and FREE. The line must begin with the keyword *Fformat*.

- **Background file:** To determine the background, three different possibilities are available, and at least one of them must be present. First, the background can be modeled by linear interpolation. To do so, the user must write the keyword *Bgrinter* followed by the name of the file (extension included) that contains background data. This file should contain a list of pairs of values: scattering variable \ intensity (see subsection 1.3.3). Note that background points are not refinable parameters. For example,

  ```
  !Linear interpolation
  Bgrinter    Sim.Bgr
  ```

  If the background is to be treated as a polynomial, the keyword is *Bgrcheb* and the number of coefficients must be specified after it. In the following line the polynomial coefficients must be entered. These are refinable parameters, so the user must remember to write their refinement codes below them. For instance,

  ```
  !Polynomial Number of coefficients
  Bgrcheb     2
  !Polynomial coefficients
  1.00000     0.20000
  1.0         1.0
  ```

  Finally, a third background subsection is available in order to deal with possible secondary phases. In case the sample to be analysed has secondary phases, these will give extra peaks in the experimental pattern. In order to take them into account, FAULTS program can treat them as background. Thus, a file containing the diffraction pattern

in free format (see section 1.3.4) must be added for each secondary phase present in the experimental pattern. This information must be given as follows: first the number of pattern backgrounds is to be specified, preceded by the keyword *Bgrnum*; then the user must write in a separate line the keyword *Bgrpatt* followed by the filename, scale factor and its refinement code. Optionally, the name of an *.hkl file can be given after the scale's factor refinement code so that the Bragg reflections of each additional phase appear in the calculated *.prf file. A separate line will be writen for every given pattern. For example,

```
!Number of pattern backgrounds
Bgrnum 1
!Pattern file  Filename     Scale factor  Code  hkl file
Bgrpatt        phase2.sub   0.026000      1.00  phase2.hkl
```

1.3.2 Experimental pattern files

The experimental intensity data file can have different formats that depend on the instrument. The files FAULTS is able to read are the following:

- **D1B o D20 (Ins=3 in FullProf):** These are the data file from D1B and D20 diffractometers (ILL) and have a **.dat* extension.

  Line 1-3: comments
  Line 4: 2θ min, step (deg.) + other parameters
  Line 5: number of points in the data file (npts)
  Lines1 : npts pairs D Y1 D Y2 ... D Y10 (where D is the detector number and Y the measured intensity)

- **NLS (Ins=4 in FullProf):** Data file from N.L.S.(Brookhaven) synchrotron radiation with **.dat* extension.

Line 1:2θmin, step size, 2θ max (deg.)
Lines1: pairs of lines with 10 items like
Y1 Y2 ... Y10 (intensities)
S1 S2 ... S10 (sigmas)

- **G41 (Ins=5 in FullProf):** Data file from G41 multidetector neutron diffractometer at LLB with **.dat* extension.

  Line 1-3: comments
  Line4: npts
  Line 5:2θmin, step size, 2θ max (deg.)
  Lines1: Y (one column)

- **D1A o D2B (Ins=6 in FullProf):** "Multicounters diffractometers data with **.dat* extension.

  Line 1: Comments
  Line 2: step size (deg.)
  Line 3: 2θmin
  Line 4: Comments
  Lines1: D Y1 D Y2 ... D Y10

- **D1AOLD o D1BOLD (Ins=1 in FullProf):** These are the data file from D1B and D20 diffractometers (ILL) (old format).

  Line 1:2θmin, step size, 2θ max (deg.)
  Lines1: npts pairs D Y1 D Y2 ... D Y10

- **DMC (Ins=8 in FullProf):** These are files from DMC diffractometer from Paul Scherrer Institute with **.dat* extension.

  Line 1-3: comments
  Line 4: 2θmin, step size, 2θ max (deg.)
  Lines1: Lines1: pairs of lines with 10 items like
  Y1 Y2 ... Y10 (intensities)

S1 S2 ... S10 (sigmas)

- **SOCABIM (Ins=9 in FullProf):** Thee are XRD files with **.uxd* extension generated by Socabim software.

  Line 1-29: comments
  Lines1: Y1, Y2, ..., Yn.

- **XYSIGMA (Ins=10 in FullProf):** Files with **.dat* extension

  Line 1: keyword XYSIGMA
  Line 2-6: comments
  Lines1: X, Y, S

- **GSAS (Ins=12 in FullProf):** data file for the GSAS analysis data program.

  Line 1: text
  Line 2: npts
  Lines1: intensities (diverse formats)

- **TIMEVARIABLE (Ins=11 in FullProf** data from variable time data collection.

  Line 1: 2θmin, step size, 2θ max (deg.)
  Lines1: T1 I1 T2 Y2 ... T10 I10 (T = time)

- **FREE (Ins=0 in FullProf):** free format files generally with **.dat* extension. Up to 7 lines of comments are accepted, the first real numbers are $2\theta_{min}$, step size and $2\theta_{max}$ (deg.), and the *npts* following lines are the intensity values.

1.3.3 Background: CFILE.bgr

If the user chooses to treat the background as a linear interpolation between N given points, he/she should provide a file which contains the list of points that will be used to calculate the background at each value of the scattering variable. The user can create such a file from the experimental data using the program WinPLOTR [18, 19] of the FullProf Suite [9, 11]. The extension of this file is usually **.bgr*, but can be any other. In this file, comments be added if they are written after a *!* symbol. Example:

```
! background: positions intensity:
!_____________________________

    10.0 40.6745
    15.0 74.8976
    45.0 74.759
    ...
```

1.3.4 Secondary phases: CFILE.sub and CFILE.hkl

To take into account secondary phases, it is necessary to provide one or several files CFILE.sub in free format (see subsection 1.3.2) that contain the calculated profile of these phases. The extension of such files is usually **.sub*, although the program accepts any kind of extension. These files can for instance be generated by the program FullProf [9] at the end of a refinement or a simulation when the option Ipr=2 or Ipr=3 is activated (for more details about this option, please refer to the FullProf Manual [11]). Moreover, the user can optionally provide a file CFILE.hkl for each CFILE.sub file, which contains the list of the hkl reflections for each phase. The user can create such a file by activating the option hkl=5 when carrying out a refinement or a simulation with the program FullProf [9, 11]. Example:

>Phase no.: 1 CFILE 19 reflections, N&T: 0 0.00
! SPGR: R -3 m; CELL: 2.860 2.860 14.227 90.0 90.0 120.0

h	k	l	Mult	S.Factor	2T
0	0	3	2	2474.23438	18.7107
1	0	1	6	1799.86182	36.8334

...

1.4 Output files

Once the program has finished, FAULTS will create the following output files, depending on the users requirements. These files will be saved to the same directory as the input data files.

1.4.1 Calculated profile: CFILEn.dat

If a powder pattern simulation is to be done, the program writes the calculated profile in a **.dat* file that is written in free format (see subsection 1.3.2).

1.4.2 Calculated SADP: CFILEn.sadp

If a selected area diffraction pattern simulation is to be done, the program writes the calculated pattern in a **.sadp* file. It contains an unsigned 16-bit image of size 256x256. To display this file, a specific software such as ImageJ can be used [17].

1.4.3 Observed and calculated profile: CFILEn.prf

If the user is performing a refinement, the program creates a file with the experimental pattern, the calculated one with the best fit, their difference

plot and the Bragg reflections. This file is to be fed into a visualisation program, such as WinPLOTR [9, 18]. It is essential to plot frequently the observed and experimental patterns. The examination of the difference pattern is a quick and efficient method to detect blunders in the model or in the input file controlling the refinement process. It may also provide useful hints on the best sequence to refine the whole set of model parameters for each particular case.

1.4.4 Crystallographic structure files: CFILE.fst, CFILE_flts.cif and CFILE_flts.vesta

If the user has specified *FST_CMD* commands for the drawing of the structural model, the program creates 3 output files: a file **CFILE.fst** to be fed into the visualization program FullProf Studio [13, 14], a second file **CFILE_flts.vesta** to be opened with the visualization program VESTA [15, 16], and a short Crystallographic Information File **CFILE_flts.cif** which can be read with other visualization programs.

Details on the format of the file **CFILE.fst** can be found in the FullProf Studio manual [14].

It is highly recommended to regularly draw the structure during the construction of the structural model as well as during the refinement.

1.4.5 Progress report: CFILEn.out

When the user run a simulation or a refinement with FAULTS, an **.out* file is generated, which contains information about different steps of the calculation. If a refinement is done, the last part of the file contains information about the refinement process: the values the refinable parameters have taken during refinement, the values of the agreement factors for each evaluation and the best values of each parameter obtained during refinement.

1.4.6 Final report: CFILE_new.flts

When the refinement is finished a **.flts* file is generated. It contains the information of the refinement's final result. Its format is similar to the input file, with the same structure (section headings, keywords, parameters, refinement codes), but with the refined values for the refinable parameters. Thus, this file can be used as the input file in case the user wants to continue the refinement process. If the file is used as input file directly, at the end of the refinement it will be overwritten.

1.5 Running FAULTS

The program is invoked from a terminal (or a DOS window) and it will appear as in figure 1.4.

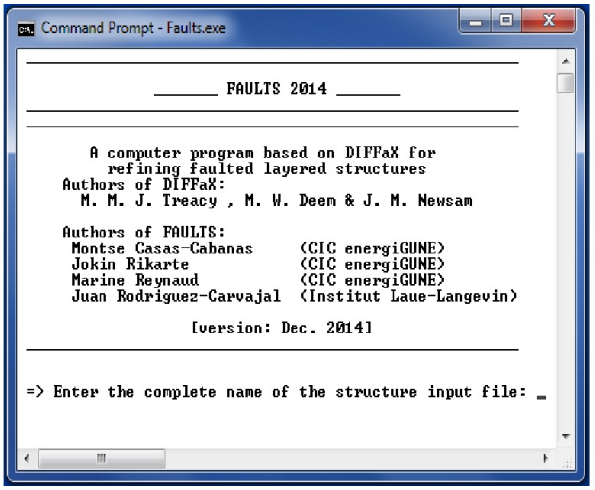

Figure 1.4: **FAULTS layout when invoked**

Once the user enters the name of file CFILE.flts, if a refinement is to

be done, the layout of the program will appear as in figure 1.5. During the refinement process, depending on the option chosen by the user (see the comments on keyword *Nprint* in the *Calculation* subsection page 20), the terminal will display the values of agreement factors at the end of each evaluation, and optionally the list of previous and new refinable parameters at each evaluation step.

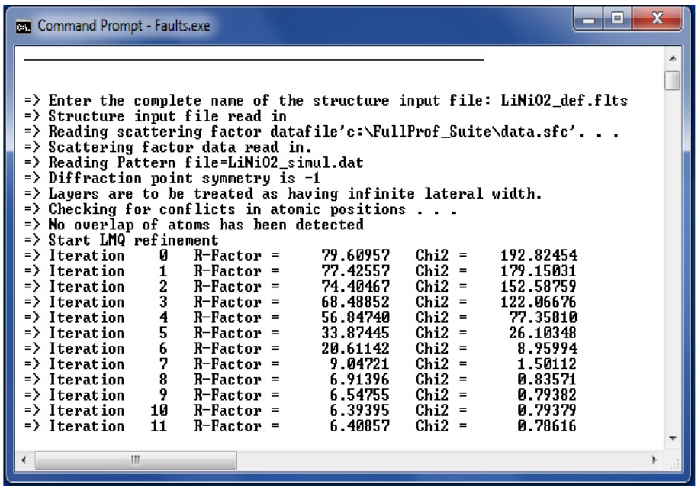

Figure 1.5: **FAULTS layout during refinement**

Once the refinement has finished, the list of the best values obtained for the refinable parameters is displayed (see figure 1.6). The user can then open the CFILEn.prf to see the goodness of the refinement.

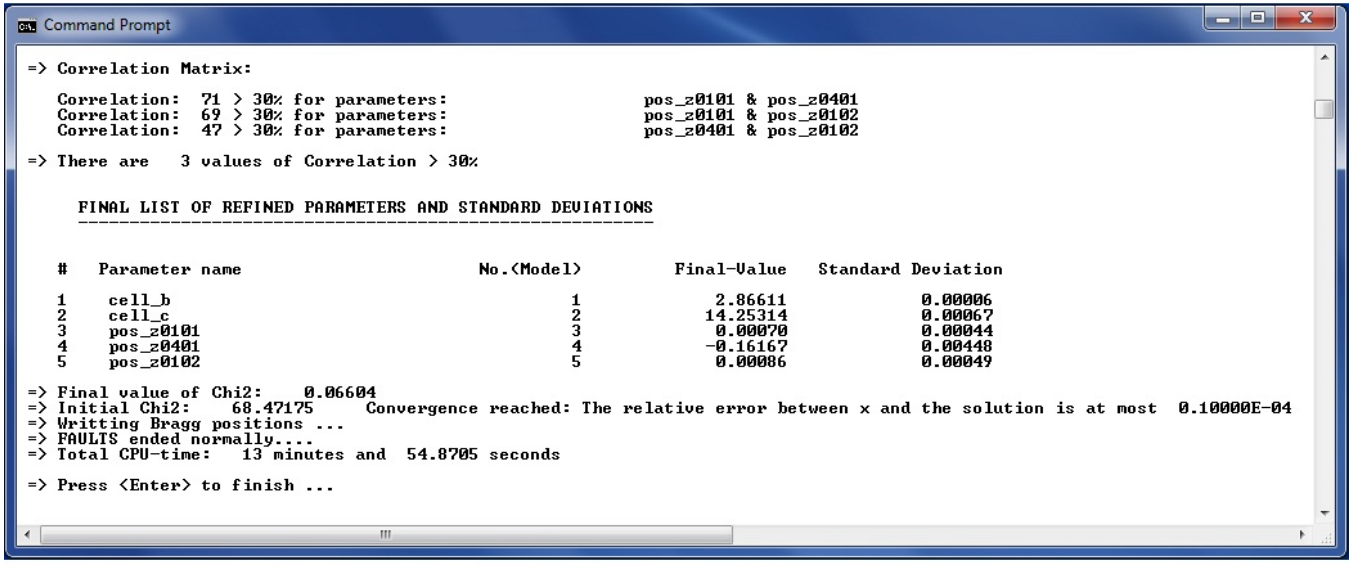

Figure 1.6: **FAULTS layout at the end of the refinement**

Chapter 2

Example

2.1 Stacking faults in $LiNiO_2$

Lithium nickel oxide has been intensively studied as a positive electrode material in Li-ion batteries [20]. As other lithium transition metal oxides, it crystallizes in an O3-type layered structure consisting in three NiO_2 slabs per unit cell, with an ABCABC oxygen stacking sequence, and lithium ions located in the octahedral sites of the interlayer spaces (see figure 2.1). In order to explain the significant broadening of the $(10l)$ and $(01l)$ diffraction lines, DIFFaX simulations have been used for this material with the hypothesis of the existence of O1 stacking faults in the structure at low lithium concentrations ($Li_\epsilon Ni_{1.02}O_2$, $\epsilon \leq 0.3$) [21]. These stacking faults represent a break in the normal stacking sequence of the structure, with a local ABAB oxygen stacking sequence.

The ideal structure can be described with three layers, each one containing a NiO_2 slab and a lithium interslab. The three layers are structurally identical, but shifted with respect to each other, resulting in transition vectors $\vec{t}_{12}=\vec{t}_{23}=\vec{t}_{31}=(2/3\ 1/3\ 1/3)$ and a stacking probability of 1 for each

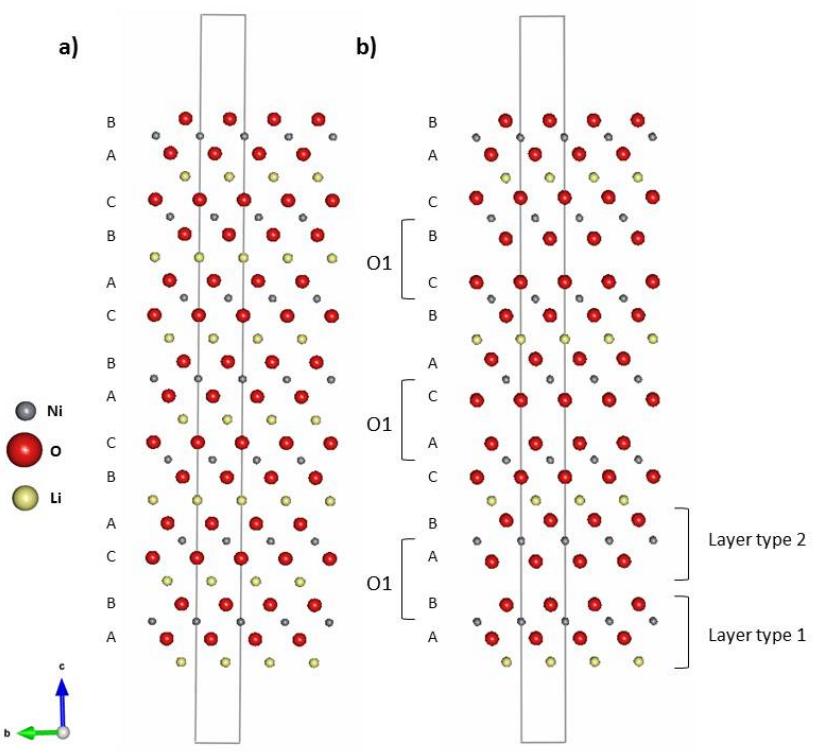

Figure 2.1: **a) Ideal structure of lithium nickel oxide ($LiNiO_2$) with O3 stacking, and b) defective structure with O1 stacking faults ($Li_cNi_{1.02}O_2$).**

transition. O1 defects require the definition of a new type of layer, structurally different, since these defective layers do not contain any lithium (see figure 2.2). Therefore, three more layers are defined in the faulted structure, structurally identical between each other but different from the previous ones, and new transitions are allowed in order to describe the defects (see figure 2.3).

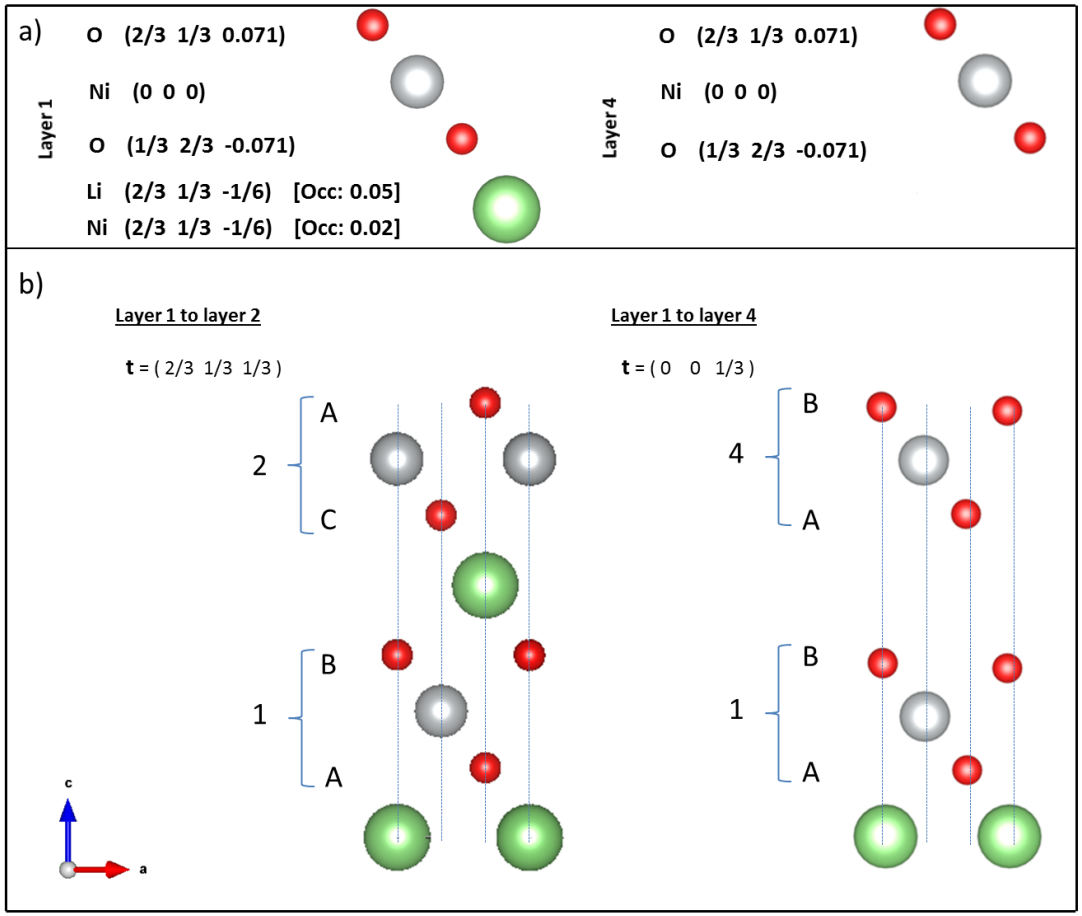

Figure 2.2: **a) Schematic representation and atomic coordinates of the layers required for describing stacking faults for $Li_{0.05}Ni_{1.02}O_2$ in the FAULTS program. b) Graphic representation of the different transition possibilities from layer 1 and transition vectors.**

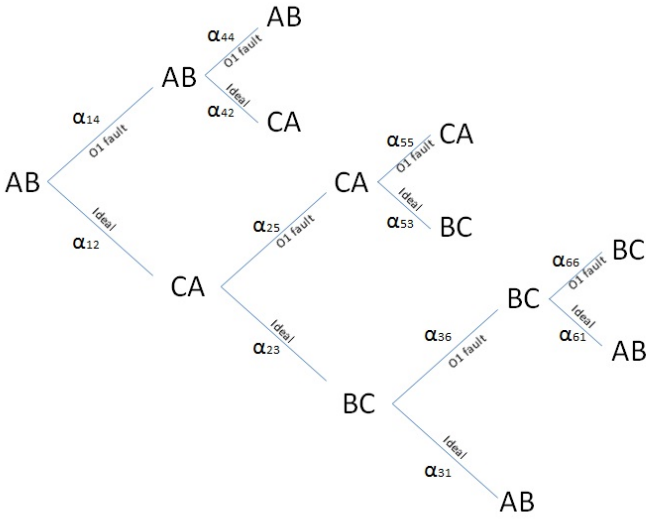

Figure 2.3: **Possible layer transitions for $Li_\epsilon Ni_{1.02}O_2$ containing O1-type stacking faults, where α_{ij} is the probability transition from layer *i* to layer *j*.**

2.2 Analysis of simulated data

By means of the stacking description described above, FAULTS has been used to simulate a diffraction pattern with the parameter's values described in table 2.1. The obtained simulated XRD pattern for $Li_\epsilon Ni_{1.02}O_2$ ($\epsilon \leq 0.3$) has then been used to analyse and test out the program. This are also the example files (LiNiO2_simul.flts and LiNiO2_refine.flts) included with the program. The refinement has been done by means of the Levenberg Marquardt minimization algorithm, restraining the program to a maximum of 2400 function evaluations and a criterion of convergence 0.1e-4. As an

Refined parameter	Simulation	Initial value	Final value(Std dev.)
a,b	2.81540	2.86540	2.81510(10)
c	13.363000	13.26300	13.3638(7)
Scale	1.00000	1.00000	1.010(3)
z_{O1},z_{O3}	0.07133	0.17133	0.0716(2)
z_{O2},z_{O4}	-0.07133	-0.17133	-0.0716(2)
α_{12},α_{23},α_{31}	0.8580	1.0000	0.8569(10)
α_{14},α_{25},α_{36}	0.1420	0.0000	0.1431(10)
α_{42},α_{53},α_{61}	0.8580	1.0000	0.842(6)
α_{44},α_{55},α_{66}	0.1420	0.0000	0.158(6)

Table 2.1: **Starting and final values of the parameters refined in the analysis of simulated data.**

example of the refinement process, the initial values of the refined parameters, which have been chosen far enough from the correct ones not to bias the result, and the final refined values are also shown in table 2.1. A visual comparison between the calculated and the simulated powder patterns and their difference is shown in figure 2.4. Starting Rp and Chi^2 values were 45.62% and 186.38 respectively and reached a final value of 4.86% and 1.03 respectively. The values of the refined parameters are very close to those used in the simulation and lead to a XRD pattern practically identical to the one obtained with the values of these parameters used in the simulation (see figure 2.4).

An example of the evolution of a and b cell parameters and of Chi^2 and Rp throughout a run of 12 iteration cycles is shown in 2.5.

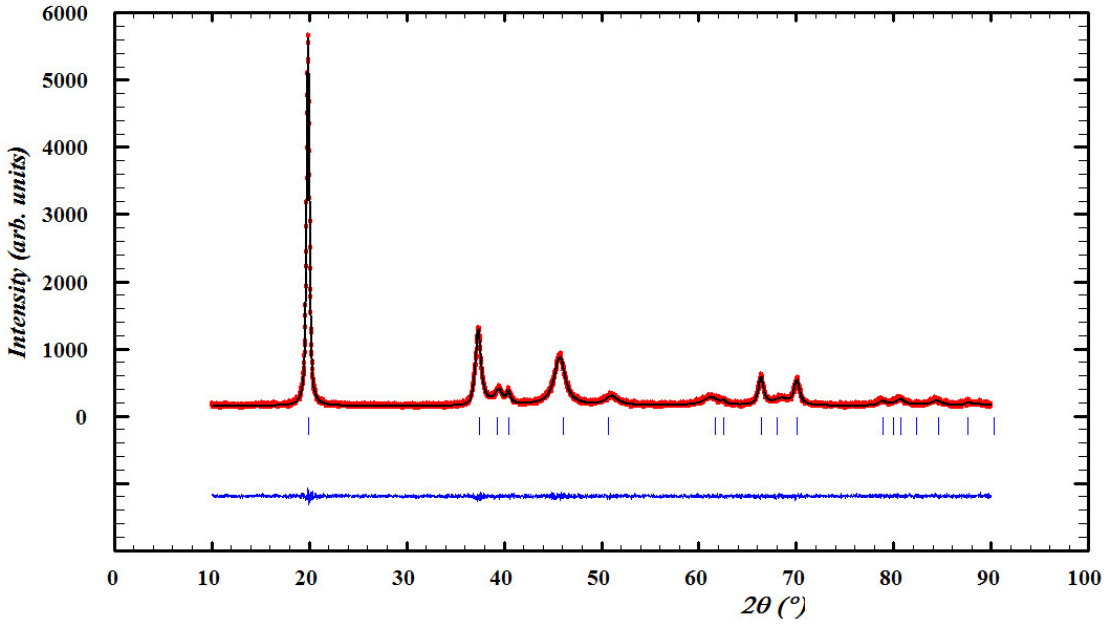

Figure 2.4: **Comparison of the X-ray diffraction patterns corresponding to the FAULTS analysis of the simulated data: simulated pattern (dotted curve) and calculated pattern using the FAULTS refinement (continuous curve). The diagram underneath shows the difference between them. The ticks show the position of the Bragg reflections of the R-3m average unit cell (the one used to describe the ideal O3 structure of $Li_{\epsilon}Ni_{1.02}O_2$).**

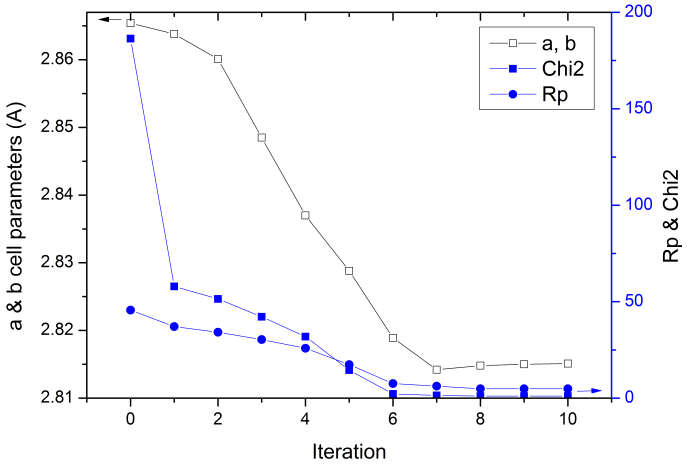

Figure 2.5: **Evolution of the functions Rp and Chi^2 and of the cell parameters a and b, versus the cycle number.**

Chapter 3

Frequent errors

This part of the manual is a tentative of compiling frequent mistakes made by new users of FAULTS. This list is far from being exhaustive. It is only intended to help beginners in starting working correctly with FAULTS and doesn't dispense the user of a carefull reading of the entire manual. The authors would be grateful if users could notify them about other possible sources of errors not reported below or about possible bugs in the program (faults*at*cicenergigune.com or fullprof*at*ill.fr).

First of all, the users should always read carefully the messages written by FAULTS in the command window and in the output file .out. Most of the errors can be identified from these messages.

3.1 Input control file .flts

- Tabs are not allowed in the input files of FAULTS. Only spaces can be used.

- All the non-optionnal sections and keywords have to be present in the .flts input file.
- Contrary to the .pcr input files used in FullProf, a missing refinement code will not considered as zero but will produce an error, so all the refinement codes have to be present in the .flts input file.
- Contrary to the .pcr input files used in FullProf, empty lines can be used in the .flts input file.
- The combination of the profile parameters U, V, W, X should not led to a profile function with negative values (no negative Gaussian and/or Lorentzian FWHM). The user can check this by calculating the Gaussian and and Lorentzian FWHM (H_G^2 and H_L) with the formula given page 10 using the program WinPLOTR [18, 19] (Menu *Calculations => I.R.F. (U,V,W,X,Y,Z)*) or any spreadsheet.

3.2 Input data files

- Make sure to comply with the description of each file type.
- No intensity can be zero o negative.
- Empty lines should be avoided.

3.3 Other remarks

- The way of working with FAULTS requires especial care of the user. Before starting to do refinements it is advisable to make simulations in order to start with an initial model that is not too far from the experimental diffraction pattern.

- If the structural model is complicated, the first calculations can take time, so the user should make sure that he/she lets the program some minutes before concluding that it is blocked. Keep in mind that the derivatives are calculated numerically by calling two times the total function per free parameter and this calculation may be expensive in CPU-time.

Bibliography

[1] Treacy M.M.J., Newsam J.M. and Deem M.W. A general recursion method for calculating diffracted intensities from crystals containing planar faults. *Proc. R. Soc. Lond. A*, 433:499-520, 1991.

[2] Treacy M.M.J., Newsam J.M. and Deem M.W. DIFFaX. Available at: http://www.public.asu.edu/~mtreacy/DIFFaX.html.

[3] Rodríguez-Carvajal J. and Gonzalez-Platas J., Crystallographic Fortran 90 modules library (CrysFML): a simple toolbox for crystallographic computing programs. *IUCr CompComm. Newsletter*, 1:50-58, 2003. Available at: http://www.iucr.org/iucr-top/comm/ccom/newsletters/.

[4] Rodríguez-Carvajal J. and Gonzalez-Platas J., CrysFML repository. Available at: http://forge.epn-campus.eu/projects/crysfml/repository/.

[5] Treacy M.M.J., Deem M.W., and Newsam J.M. DIFFaX Manual, 2005, Chap 8: How DIFFaX works (p48). Available at: http://www.public.asu.edu/~mtreacy/DIFFaX$_$manual.pdf.

[6] Hendricks S. and Teller E. X-ray interference in partially ordered layer lattices. *J. Chem. Phys.*, 10:147-167, 1942.

[7] Casas-Cabanas M., Rodríguez-Carvajal J., Palacín M.R., *Z. Kristallogr. Suppl.*, 23:243-248, 2006.

[8] Casas-Cabanas M., Rikarte Ormazabal J., Reynaud M., Rodríguez-Carvajal J., *in preparation.*

[9] Rodríguez Carvajal J., FullProf Suite. Available at: http://www.ill.eu/sites/fullprof.

[10] Rodríguez Carvajal J., Physica B.(1993), 192, Recent advances in magnetic structure determination by neutron powder diffraction. *Physica B.*, 192:55-69, 1993.

[11] Rodríguez Carvajal J., FullProf Manual. Available at http://www.ill.eu/sites/fullprof/php/tutorials.html.

[12] Treacy M.M.J., Deem M.W., and Newsam J.M. DIFFaX Manual, 2005. Available at: http://www.public.asu.edu/~mtreacy/DIFFaX$_$manual.pdf.

[13] Rodríguez Carvajal J. and Chapon L., FullProf Studio. Available at http://www.ill.eu/sites/fullprof/index.html.

[14] Rodríguez Carvajal J. and Chapon L., FullProf Studio Manual. Available at http://www.ill.eu/sites/fullprof/index.html.

[15] Momma K. and Izumi F. VESTA3 for three-dimensional vizualization of crystal, volumetric and morphology data. *Journal of Applied Crystallography*, 44:1272-1276, 2011.

[16] Momma K. and Izumi F., VESTA (Visualization for Electronic and STrcutural Analysis). Available at http://jp-minerals.org/vesta/en/.

[17] http://imagej.nih.gov/ij/index.html

[18] Roisnel T. and Rodríguez Carvajal J., WinPLOTR: A windows tool for powder diffraction pattern analysis. *Mater Sci Forum*, 378(3):128-123, 2000.

[19] Roisnel T. and Rodríguez Carvajal J., WinPLOTR: A graphic tool for powder diffraction. Available in the FullProf Suite or as an independant program at http://www.cdifx.univ-rennes1.fr/winplotr/.

[20] Whittingham M. S., Lithium Batteries and Cathode Materials. *Chemical Reviews*, 104:4271-4301, 2004.

[21] Croguennec L., Pouillerie C., Mansour A. N. and Delmas C., Structural characterisation of the highly deintercalated $Li_xNi_{1.02}O_2$ phases (with x$\leq$0.30). *Journal of Materials Chemistry*, 11:131-141, 2000.