

# Chapter 7

## XDFOUR – A General 2-D And 3-D Fourier Synthesis Program

### 7.1 Overview

The aspherical atom model used in multipole refinement gives structure factor phases closer to the true phases for non-centrosymmetric crystals than does the spherical or independent atom model (SPH). This permits mapping of the density by Fourier synthesis in various ways. The *experimental deformation map* is obtained using the calculated multipole phases with the observed structure factors  $F_o$ :

$$\delta\rho^{\text{exp}}(\mathbf{r}) = \frac{1}{V} \sum_{\mathbf{h}} \left[ |F_o(\mathbf{h})| e^{i\phi_{mul}} - |F_{sph}(\mathbf{h})| e^{i\phi_{sph}} \right] e^{-2\pi i \mathbf{h} \cdot \mathbf{r}}$$

$F_{sph}(\mathbf{h})$  is computed with atomic positions and thermal parameters obtained from the multipole refinement. The *dynamic model map* is obtained from the calculated multipole structure factors, i.e. the Fourier coefficients are the difference of two values of  $F_c$ :

$$\delta\rho^{\text{dyn}}(\mathbf{r}) = \frac{1}{V} \sum_{\mathbf{h}} \left[ |F_{mul}(\mathbf{h})| e^{i\phi_{mul}} - |F_{sph}(\mathbf{h})| e^{i\phi_{sph}} \right] e^{-2\pi i \mathbf{h} \cdot \mathbf{r}}$$

(temperature factors are included in  $F_{mul}$  and  $F_{sph}$ ). This density distribution is free of experimental noise. The use of multipole phases makes the maps slightly model-dependent; to check that all significant density features of the experimental data are included in the model we compute the *residual map*:

$$\delta\rho^{\text{res}}(\mathbf{r}) = \frac{1}{V} \sum_{\mathbf{h}} \left[ |F_o(\mathbf{h})| - |F_{mul}(\mathbf{h})| \right] e^{i\phi_{mul}} e^{-2\pi i \mathbf{h} \cdot \mathbf{r}}$$

For good data this should be a flat, featureless map.

Crystallographic Fourier synthesis programs (except the FFT type) compute the density distribution as

$$\rho(\mathbf{r}) = \frac{1}{V} \sum_{\mathbf{h}} [A(\mathbf{h}) \cos 2\pi \mathbf{h} \cdot \mathbf{r} + B(\mathbf{h}) \sin 2\pi \mathbf{h} \cdot \mathbf{r}]$$

where  $A(\mathbf{h}) + iB(\mathbf{h}) = F(\mathbf{h})$ . For the *total density map* the Fourier coefficients are

$$A(\mathbf{h}) = |F_o(\mathbf{h})| \cos \phi_{mul}$$

$$B(\mathbf{h}) = |F_o(\mathbf{h})| \sin \phi_{mul}$$

For the *experimental deformation map*

$$A(\mathbf{h}) = |F_o(\mathbf{h})| \cos \phi_{mul} - |F_{sph}(\mathbf{h})| \cos \phi_{sph}$$

$$B(\mathbf{h}) = |F_o(\mathbf{h})| \sin \phi_{mul} - |F_{sph}(\mathbf{h})| \sin \phi_{sph}$$

For the *dynamic model map*

$$A(\mathbf{h}) = |F_{mul}(\mathbf{h})| \cos \phi_{mul} - |F_{sph}(\mathbf{h})| \cos \phi_{sph}$$

$$B(\mathbf{h}) = |F_{mul}(\mathbf{h})| \sin \phi_{mul} - |F_{sph}(\mathbf{h})| \sin \phi_{sph}$$

For the *residual map*

$$A(\mathbf{h}) = \left[ |F_o(\mathbf{h})| - |F_{mul}(\mathbf{h})| \right] \cos \phi_{mul}$$

$$B(\mathbf{h}) = \left[ |F_o(\mathbf{h})| - |F_{mul}(\mathbf{h})| \right] \sin \phi_{mul}$$

XDFOUR computes a 2-D or 3-D Fourier summation on a grid oriented either with respect to a general (non-rational) plane, without interpolation errors, or with axes parallel to the crystallographic axes. The value of  $F(000)$  is taken from `xd.fou`, and is included in the Fourier summations, ensuring the electron density is on a correct scale.

## 7.2 Files used and created by XDFOUR

Input: `xd.mas`, `xd.res`, `xd.fou`  
Output: `xd_fou.grd`, `xd_fou.out`

## 7.3 Input instructions for XDFOUR

### 7.3.1 SELECT

**SELECT** (\*)**fobs** (\*)**fmod1** (\*)**fmod2** (\*)**print** (\*)**snlmin** *snlmin* (\*)**snlmax** *snlmax*

**fobs**, **fmod1**, **fmod2**

The coefficients for the summation are defined here. The reflection file contains the observed structure factors  $F_o$ , and two sets of calculated structure factors.  $F_c$  may be computed by the least squares program according to various alternative density models, *e.g.* independent atom, multipole, static, anharmonic, and any two of these may be selected for output of the corresponding  $F_c$  (see **FOUR** instruction for XDLSM.) The starred options in the **SELECT** line signify the type of coefficient to be used. If two are starred then the coefficients are formed from the difference of the corresponding  $F_c$  sets. If only one is starred, it forms the coefficients. In the example below, the **SELECT** line specifies a residual map with coefficients  $F_o - F_{multipole}$ . In this case the  $F_c$  set labelled **fmod1** has been defined in the least squares program as  $F_{multipole}$ .

```
SELECT *fobs *fmod1 fmod2
```

Note that  $F_c$  must be calculated without anomalous dispersion (so that the scattering factors are real) as signified by zero as the third parameter after **fmod1** in the XDLSM section. Fourier maps are computed without dispersion, hence it is removed from  $F_o$  also.

As a further example, one would obtain a dynamic model map by including

```
FOUR fmod1 4 2 0 0 fmod2 -1 2 0 0
```

in the XDLSM section, and

```
SELECT fobs *fmod1 *fmod2
```

in the XDFOUR section. The Fourier coefficients are then formed as the difference between the dispersion-removed multipole  $F_c$  and the dispersion-removed independent atom  $F_c$ . Note that in this example no extinction correction is applied. This is signified by the final zero in the **fmod1** and **fmod2** options of XDLSM.

## print

The results listing file will include the grid of density values if this option is starred. A grid file (xd\_fou.grd) suitable for input to the graphical programs is always written.

**snlmin** *snlmin* **snlmax** *snlmax*

These options define the  $\sin(\theta)/\lambda$  range for which Fourier coefficients are included in the calculation. The default values are *snlmin* 0.0, *snlmax* 2.0. Note that the specified ranges are *only* applied if the corresponding items are starred - otherwise the default ranges are used.

## 7.3.2 APPLY

**APPLY** *symm* *is trans tx ty tz (atoms ... | all)*

This command is used to apply a symmetry operation to the pseudoatoms in the asymmetric unit. This is only used to include the additional atoms in the gridfile for plotting purposes. The symmetry operations are referenced according to the sequence in which they are listed at the start of the program output. For example,

```
APPLY SYMM 3 TRANSLATIONS -2 -1 1 O(1) N(1) C(1) H(1) H(2) H(3)
```

applies the third symmetry operation, with the lattice translations shown, to the six atoms whose labels are given. If a symmetry operation is to be applied to *all* atoms in the asymmetric unit, then the keyword "all" may replace the pseudoatom labels. More than one line beginning with **APPLY** may be present.

## 7.3.3 GRID

**GRID** (\*)**3-points** (\*)**perp** (\*)**cryst**

The **GRID** line should specify one of the following options: **3-points**, **perp**, or **cryst** preceded by an asterisk, *e.g.*

```
GRID *3-POINTS PERP CRYST
```

This means that the option **3-points** is selected.

### 3-points and perp

When either of the options **3-points** or **perp** are chosen, at least 3 points must be specified. There are 3 types of formats.

1. **ATOM** *label atom* (**symm** *symop*) (**trans** *tx ty tz*) (\*)**mark**  
*atom* must be identical to an atom label given in the parameter file.
2. **ATOM** *no no* (**symm** *symop*) (**trans** *tx ty tz*) (\*)**mark**  
*no* is the sequence number of an atom in the parameter file.
3. **XYZ** (*label label*) *x y z* (**symm** *symop*) (**trans** *tx ty tz*) (\*)**mark**  
*label* is a label which may be marked on the plot. The fractional coordinates *x y z* are free format real numbers.

Common for the three formats are the options:

- **symm** *symop* where *symop* is the sequence number of an operation in the list of space group operations which appears at the start of the program output.
- **trans** *tx ty tz* indicating three lattice translations (positive or negative integers).
- **(\*)mark** If flagged with a star, the position is marked on the plot.

The first 3 points are used to define a right-handed orthonormal coordinate system in the following way. The origin of this coordinate system is the centroid of the three points in the list. For the option **3-points**, the points define the *xy* plane; the *x*-axis is parallel to the vector from point 1 to point 2 and the third point is in the half-plane *y*>0. For the option **perpendicular**, the *xy* plane is perpendicular to the vector from point 1 to point 2; the projection of the third point onto the *xy* plane defines the direction of the *x*-axis, *x*>0.

### cryst

In this case the grid has oblique axes parallel to the crystallographic axes *a*, *b* and *c*. Of the commands described below, only **LIMITs**, **TRAN**slate and **PERM**ute should be used in this case.

#### 7.3.4 TRAN

Having defined a first orthonormal coordinate system, we may define the final grid-coordinate system by any of the following operations on it, in any order, and as many of them as you like. The operations are:

- Translation of the coordinate system origin, command **TRAN**
- Rotation around axes through the origin, command **ROTA**
- Permutation of the axes, command **PERM**

As soon as a line is read, the operation is performed on the coordinate system, and the next operation acts on this new coordinate system with respect to its axes. The command for translation is

**TRAN** *tx ty tz*

The interpretation of *tx ty tz* depends on the option chosen on the **GRID**-line :

- **\*3-points**: *tx ty tz* are in Ångströms
- **\*perp**: *tx ty* are in Ångströms, *tz* in fractions of the vector from point 1 to point 2.
- **\*cryst**: *tx ty tz* are in fractional coordinates

#### 7.3.5 ROTATE

**ROTA eulerian** *alpha beta gamma*

**ROTA** *axis angle*

$\alpha$ ,  $\beta$ ,  $\gamma$  are the Eulerian rotation angles: first a rotation of  $\alpha$  degrees about the *z*-axis, then  $\beta$  degrees about the new *y*-axis, and finally  $\gamma$  degrees about the new *z*-axis.

*axis*: *x*, *y* or *z*

*angle* equals the angle (degrees) which the coordinate system is rotated about the coordinate axis given by *axis*. An example: the result of these four instructions is to leave the coordinate system unchanged.

```
ROTATE EULERIAN 45 -54.5 90
ROTATE Z -90
ROTA Y 54.5
ROTATE Z -45.
```

### 7.3.6 PERM

**PERM** *new-x new-y new-z*

*new-x, new-y, new-z*: some permutation of x, y or z. An example: How to generate a left-handed system by turning z into -z? Here is one way of doing it:

```
ROTation Y 90
PERMute Z Y X
```

### 7.3.7 LIMITS

This command defines the limits of summation. A 3-dimensional grid is obtained when all three axes have more than one grid point.

**LIMI** (*keyword value*) ...

Where *keyword* is any of the following (default values in brackets):

```
xmin [0.] xmax [1.] nx [41]
ymin [0.] ymax [1.] ny [41]
zmin [0.] zmax [0.975] nz [40]
```

*?min* and *?max* are the limits of the box dimensions along the respective coordinate axis. *n?* is the number of grid points in this direction. It is stressed that the limit information is only applied to the final grid-coordinate system. Example: a 2-D grid is defined. The sense of the *y*-axis is inverted since *ymin > ymax*.

```
LIMITS XMIN -2. XMAX 2. NX 41
LIMITS YMIN 2. YMAX -1. NY 31
LIMITS ZMIN 0. ZMAX 0. NZ 1
```

Default values assumed by the program divide the cell up into 40 sections, each having  $41 \times 41$  points. There are no arbitrary limits to the grid size which may be calculated - the only limits are those imposed by the system memory. However please note that the graphical programs may not be able to display extremely large grid files.

