Supplementary information for “Angle calculations for an area detector on a 2-axis arm: application to powder diffraction”

1 Experimental details

The powder sample comprised around 2 mg of γ-Fe$_2$O$_3$ nanoparticles that were supported by an Ir thin film on a Si substrate. The incident beam (tuned to 30 eV below the Fe K absorption edge) struck the sample in grazing incidence, and the scattered X-rays exited the sample chamber via a rectangular (Kapton) window. Because the dimensions of the window limited the (measurable) scattered beam directions, the area detector was moved to $(\nu,\delta) = (24°,20°)$, where (roughly) maximum fractions of two strong diffraction cones [those due to the (220) and (311) reflections of the cubic γ-Fe$_2$O$_3$ phase] were captured. The distance between the sample and detector was $D =$ 232.4 mm. With the detector at $(\nu,\delta) = (0,0)$, approximately half of this distance was helium and the remainder was air (post chamber).

The detector was a MAR165 charge coupled device (CCD). The pixel size was calibrated at 77.3 μm × 77.3 μm. Although the MAR165 has a circular active area, it outputs a square, $N \times N$ (pixels) data matrix (in the high resolution mode, $N = 2048$). The matrix elements in the corner regions, that lie outside the active area, each contain (by definition) zero counts. Only matrix elements within the detector's active area are plotted in Fig. 2.

2 Angle calculations and 2-D to 1-D pattern conversion in MATLAB

The position vectors corresponding to all CCD pixels–and (corner region) ‘pseudo pixels’–may be calculated in the form of an $N \times N \times 3$ array, `R`, by evaluating equation (4) as follows:

```
>> for index = 1:3
R(:,:,index) = ones(N,1)*((([1:N] + 0.5 - jbc)*xpix)*Xd(index)) +...
            ((([1:N]' + 0.5 - ibc)*ypix)*Yd(index))*ones(1,N) -...
            ones(N,N)*(D*Zd(index));
end
```

where `Xd`, `Yd` and `Zd` are the vectors $\hat{\mathbf{x}}'_d$, $\hat{\mathbf{y}}'_d$ and $\hat{\mathbf{z}}'_d$ defined by equations (2) and (3). The magnitudes of the position vectors may then be calculated in the form of an $N \times N$ matrix:

```
>> Rmag = sqrt(R(:,:,1).^2 + R(:,:,2).^2 + R(:,:,3).^2);
```

An $N \times N \times 2$ array, `K`, containing the 1st and 2nd components of the outgoing scattering vector directions, defined in equation (5), may then be computed by:

```
>> for index = 1:2; K(:,:,index) = R(:,:,index)./Rmag; end
```

from which matrices of 2θ and ψ values (in degrees) corresponding to the CCD pixels (and pseudo pixels) can be calculated by equating the 1st and 2nd vector components corresponding to equations (1) and (5):

```
>> TTH = acos(K(:,:,2)) * (180/pi);
>> PSI = acos(K(:,:,1)./sin(TTH)) * (180/pi);
```

At this point, a plot of the transformed 2-D diffraction pattern [as in Fig. 2(b)] can be made using the `surf` function, after reading in the MAR165 CCD data file using the `importdata` function, and ‘flipping’ this matrix vertically in order to be make it consistent with the chosen detector coordinate system of Fig. 1 (in which y_d points vertically *upwards* for $\delta = 0$) together with equation (4), where the row index i increases from bottom to top:

```
>> COUNTS = importdata('datafile.mccd');
>> COUNTS = flipud(COUNTS);
>> surf(TTH,PSI,COUNTS); view(2); shading interp;
```

The 2θ and ψ values in the matrices `TTH` and `PSI` are ordered according to detector pixel position, just as the count values are ordered in the 2-D diffraction data matrix, `COUNTS`. To allow a straightforward extraction of a 1-D (integrated) pattern, the `COUNTS` matrix should be converted to a matrix of count values ordered according to $\{2\theta,\psi\}$, *e.g.* with rows corresponding to constant ψ and columns corresponding to constant 2θ. This may be achieved by a 2-D interpolation (of the set of 'points' with values COUNTS located at the positions in the $\{2\theta,\psi\}$ plane given by TTH and PSI) onto a uniform rectangular grid of 2θ and ψ values, the corners of which are defined by the maximum and minimum values in the matrices TTH and PSI. The grid may be defined by the *vectors* TTH_ord and PSI_ord calculated as follows:

```
>> max_tth = max(TTH(:)); min_tth = min(TTH(:));
>> max_psi = max(PSI(:)); min_psi = min(PSI(:));
>> step_tth = (max_tth - min_tth)/(N - 1);
>> step_psi = (max_psi - min_psi)/(N - 1);
>> TTH_ord = min_tth:step_tth:max_tth;
>> PSI_ord = [min_psi:step_psi:max_psi]';
```

The reordered diffraction data matrix can then be (2-dimensionally) interpolated using the griddata function:

```
>> COUNTS_ord = griddata(TTH,PSI,COUNTS,TTH_ord,PSI_ord);
```

In general, the rectangular grid contains points that lie outside the region of the $\{2\theta,\psi\}$ plane spanned by the matrices TTH and PSI, the latter region being generally non-rectangular, as indicated by Fig. 2(b). The attempted interpolation (via the griddata command) onto such points produces NaN (not a number) entries in the reordered matrix, COUNTS_ord. To set such NaN values to zero, so that a 1-D diffraction patterns can be extracted from COUNTS_ord by summing count values of the same 2θ (same column), the following command may be issued:

```
>> COUNTS_ord(isnan(COUNTS_ord)) = 0;
```

Incidentally, the following surf plot of the reordered matrix will be (as expected) identical to the (above) surf plot of the matrices TTH, PSI and COUNTS, but for the former plot's rectangular shape (with the zeros at its edge regions that lie outside the $\{2\theta,\psi\}$ region spanned by the detector):

```
>> surf(TTH_ord,PSI_ord,COUNTS_ord); view(2); shading interp;
```

Finally, the reduction of any selected region of the reordered matrix to a 1-D (integrated) pattern may be carried out as follows:

```
>> c1 = 1 + round((tth1 - min_tth)/step_tth);
>> c2 = 1 + round((tth2 - max_tth)/step_tth);
>> r1 = 1 + round((psi1 - min_tth)/step_psi);
>> r2 = 1 + round((psi2 - max_tth)/step_psi);
>> INTEG_COUNTS = sum(COUNTS_ord(r1:r2,c1:c2),1);
>> TTH_VALUES = TTH_ord(c1:c2);
```

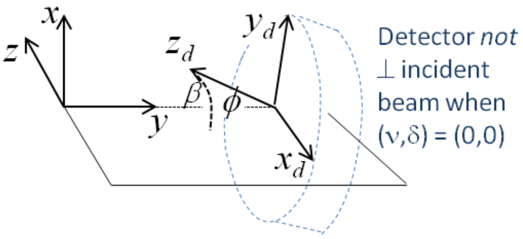
where the maximum and minimum 2θ and ψ values that define the selected region are given by the values of the variables tth1, tth2, psi1 and psi2, and correspond to column and row indices in COUNTS_ord denoted by c1, c2, r1 and r2. The highlighted (by the dashed-line rectangle) region of Fig. 2(b) corresponds to (tth1,tth2,psi1,psi2) = (30°,44.5°,40°,50°).

3 Detector non-orthogonality

An anonymous referee is gratefully acknowledged for raising the question as to how the calculation procedure would be modified to account for detector non-orthogonality (with respect to the incident beam axis). In Hammersley *et al.*'s paper (see reference in main article) the detector non-orthogonality is parameterized by two angles, ϕ and β, which are referred to in the FIT2D package (as well as in other packages, such as GSAS-II) as the *angle of detector tilt in plane* and the *rotation angle of tilting plane*, respectively. The former is the angle between the normal to the detector face and the incident beam axis [when, for the present purposes, $(\nu,\delta) = (0,0)$], whereas the latter angle (β) can be understood as a rotation about that axis. That is, starting from the ideally orthogonal detector in Fig. 1(a), the detector is rotated anticlockwise about its y_d axis (which is parallel to the x axis of the laboratory frame) by the angle ϕ, then subsequently rotated clockwise about the y (laboratory frame) axis by the angle β. The detector is then rotated about its normal in a clockwise manner until the x_d and y_d axes lie in horizontal and vertical planes, respectively. The rotations ϕ followed by β modify the direction of the detector normal (expressed in the laboratory coordinate system) from the unit vector $\hat{\mathbf{z}}_d$ in equation (2) to the following (unit) vector:

$$\hat{\mathbf{z}}_d = \begin{pmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix} = \begin{pmatrix} \sin\phi\sin\beta \\ -\cos\phi \\ -\sin\phi\cos\beta \end{pmatrix}$$

The following schematic illustrates the final (non-orthogonal) detector orientation, where the size of the angle ϕ is exaggerated for clarity (typical values of ϕ may vary up to only several decimals of a degree):



Another way to produce the same (general) non-orthogonal orientation as that shown in the above schematic is (starting from the ideally orthogonal detector) to first rotate clockwise about x_d (anticlockwise about z) by some angle, let us call it α_δ, and then anticlockwise about an axis parallel to x that passes through the origin of the detector coordinate system, by an angle, let us call it α_ν. The resulting direction of the detector normal is then be expressed as:

$$\hat{\mathbf{z}}_d = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_\nu & -\sin\alpha_\nu \\ 0 & \sin\alpha_\nu & \cos\alpha_\nu \end{pmatrix} \begin{pmatrix} \cos\alpha_\delta & -\sin\alpha_\delta & 0 \\ \sin\alpha_\delta & \cos\alpha_\delta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix} = \begin{pmatrix} \sin\alpha_\delta \\ -\cos\alpha_\nu\cos\alpha_\delta \\ \sin\alpha_\nu\sin\alpha_\delta \end{pmatrix}$$

By equating similar components of the two different parameterizations [*i.e.* the (ϕ,β) one and the (α_ν,α_δ) one] for the detector normal direction, the angles corresponding to the latter parameterization may be determined given those from the former. This would be the case of a detector whose non-orthogonality were initially calibrated using FIT2D (or another package adopting a similar parameterization, such as GSAS-II), using the 2-D diffraction pattern acquired from some calibration sample (Si, LaB$_6$, ...) with the detector at (ν,δ) = (0,0). Subsequently, once the angles α_ν and α_δ were determined, the $\hat{\mathbf{x}}_d$ and $\hat{\mathbf{y}}_d$ vectors [modified from the ideal case in equation (2)] are determined as:

$$\hat{\mathbf{x}}_d = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_\nu & -\sin\alpha_\nu \\ 0 & \sin\alpha_\nu & \cos\alpha_\nu \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ \sin\alpha_\nu \\ -\cos\alpha_\nu \end{pmatrix}$$

and

$$\hat{\mathbf{y}}_d = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_\nu & -\sin\alpha_\nu \\ 0 & \sin\alpha_\nu & \cos\alpha_\nu \end{pmatrix} \begin{pmatrix} \cos\alpha_\delta & -\sin\alpha_\delta & 0 \\ \sin\alpha_\delta & \cos\alpha_\delta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} \cos\alpha_\delta \\ \cos\alpha_\nu\sin\alpha_\delta \\ \sin\alpha_\nu\sin\alpha_\delta \end{pmatrix}$$

The calculation procedure presented in the main article would thus be modified to account for any known (previously calibrated) detector non-orthogonality by applying the transform in equation (3) to each of the above modified vectors, $\hat{\mathbf{x}}_d$, $\hat{\mathbf{y}}_d$ and $\hat{\mathbf{z}}_d$, rather than to the versions of these vectors given in equation (2).

A different approach would be to treat α_ν and α_δ as unknowns, together with the beam center coordinates i_{bc} and j_{bc} in equation (4), and use a modified calculation procedure to test detector non-orthogonality at some alignment of (ν,δ) $\neq$ (0,0). The approach would be as follows. A calibration powder sample would be measured with the detector displaced in ν and δ. An attempt to fit the locus of points corresponding to a given diffraction ring (of known 2θ) in the 2-D pattern would then be made using equations (1) and (3)–(5), along with the above modified forms of the vectors, $\hat{\mathbf{x}}_d$, $\hat{\mathbf{y}}_d$ and $\hat{\mathbf{z}}_d$. That is, for a series of ψ values (but a constant 2θ) similar components of the vectors in equations (1) and (5) would be equated to each other in order to determine (this time) the components of the position vectors in equation (5) (rather than the angles 2θ and ψ). The values of the 'fitting parameters' (α_ν,α_δ,i_{bc},j_{bc}) would be adjusted until the resultant (series of) values of the indices i and j, from equation (4), matched the locus of points comprising the diffraction ring.

Such a method could find an application, *e.g.*, in single-crystal studies using an area detector on a pseudo six-circle diffractometer. Namely, the powder (calibration) sample would be mounted before or after the single-crystal study, to check the detector non-orthogonality at ν and δ values relevant to that study. The same calibrant would then be measured with the detector at (ν,δ) = (0,0), and the detector non-orthogonality would this time be determined using existing software (*e.g.* FIT2D). A comparison of the angle parameters (that parameterize the non-orthogonality) determined by each method would constitute a check of the robustness of the detector mount, *i.e.* a check of the possibility that small tilts were introduced upon moving the detector in ν and δ. [Of course, it should be confirmed beforehand that the two methods agree–*i.e.* give consistent $\hat{\mathbf{z}}_d$ directions– when applied to the same calibration pattern acquired at (ν,δ) = (0,0) or at small values of these angles such that the incident axis still intercepted close to the center of the detector face.]