

1 Introduction

The FitAllB program, an add-on to the FABLE suite of programs for analysis of 3DXRD data (<http://fable.wiki.sourceforge.net/>), is tailored to do centre-of-mass (COM) refinements of grain orientations, positions and strain tensors from far-field images of a polycrystalline material. The name FitAllB originates because the routine fits all B_i (1), the grain specific reciprocal space metric which contains information about the strain state of the grain.

The aim is to be able to handle several hundred illuminated grains and obtain the strain tensors to an accuracy of 10^{-4} . The strain tensors are output both in the Cartesian grain coordinate system relative to the grain orientation and in the sample system for overall comparisons, and if the components of the stiffness tensor \mathbf{C} are provided the stress tensors in the same two representations will also be output. FitAllB includes an error estimation routine to give standard deviations of all refined parameters. Finally the relative volumes of the grains are refined, so in principle a 3D orientation and stress/strain map of the polycrystal can be obtained using tessellation.

2 Mathematical equations

2.1 Basic equations

We aim to minimise the following χ^2 -function using MINUIT [James, 1972]:

$$FCN = \sum_{i,j(i)} \left(\Gamma_{ij}^{-1} \bar{G}_{ij} - \frac{\lambda}{2\pi} U_i B_i \bar{G}_{hkl,ij} \right)^T V_{ij}^{-1} \left(\Gamma_{ij}^{-1} \bar{G}_{ij} - \frac{\lambda}{2\pi} U_i B_i \bar{G}_{hkl,ij} \right) \quad (1)$$

i.e. the deviation from the ideal diffraction equation. The sum is taken over i grains with $j(i)$ reflections in the i 'th grain. $\bar{G}_{hkl,ij}$ is the assigned hkl of the reflection, B_i is the grain specific reciprocal space metric, which is connected to the strain tensor as described in Section 2.2, and U_i gives the orientation of the grain in the rotated system. $\Gamma_{ij} = \Phi_x \Phi_y \Omega(\omega_{ij})$ relates the rotated system to the laboratory system, V_{ij} is the covariance matrix of $\Gamma_{ij}^{-1} \bar{G}_{ij}$, and the unrotated scatter vector in reciprocal space, \bar{G}_{ij} , can be expressed in terms of direct space quantities:

$$\bar{G}_{ij} = \left[\frac{\bar{d}_{ij}}{|\bar{d}_{ij}|} - \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \right], \quad (2)$$

$$\bar{d}_{ij} = R \begin{pmatrix} 0 \\ (y_{det,ij} - y_{det,0})p_y \\ (z_{det,ij} - z_{det,0})p_z \end{pmatrix} + \begin{pmatrix} D \\ 0 \\ 0 \end{pmatrix} - \Gamma_{ij} \begin{pmatrix} x_{0,i} \\ y_{0,i} \\ z_{0,i} \end{pmatrix} \quad (3)$$

Refineable are 12 parameters per grain (3 positional ($x_{0,i}$ $y_{0,i}$ $z_{0,i}$), 3 rotational (U_i) and 6 strain (B_i)) and the 10 global parameters: tilt (R , 3 parameters), sample-to-detector distance (D), pixels sizes (p_y and p_z), beam centre on detector ($y_{det,0}$, $z_{det,0}$), and the tilt of the ω -axis around x and y (Φ_y and Φ_z).

The measured quantities are ω_{ij} , $y_{det,ij}$ and $z_{det,ij}$, thus 3 observables per reflection.

2.2 Strain tensor definition

The grain specific reciprocal space metric of (1) relating reciprocal space to the Cartesian grain system is defined from the reciprocal lattice constants as [Poulsen, 2004]:

$$B_i = \begin{pmatrix} a_i^* & b_i^* \cos(\gamma_i^*) & c_i^* \cos(\beta_i^*) \\ 0 & b_i^* \sin(\gamma_i^*) & -c_i^* \sin(\beta_i^*) \cos(\alpha_i) \\ 0 & 0 & c_i^* \sin(\beta_i^*) \sin(\alpha_i) \end{pmatrix} \quad (4)$$

and its direct space counterpart relating any direct space coordinate system to the Cartesian grain system is given as:

$$A_i = \begin{pmatrix} a_i & b_i \cos(\gamma_i) & c_i \cos(\beta_i) \\ 0 & b_i \sin(\gamma_i) & -c_i \sin(\beta_i) \cos(\alpha_i^*) \\ 0 & 0 & c_i \sin(\beta_i) \sin(\alpha_i^*) \end{pmatrix} \quad (5)$$

Seing that the elastic strain tensor gives the perturbation of the local lattice in real space we define it as:

$$\epsilon_i = \frac{1}{2} (A_i A_0^{-1} + (A_i A_0^{-1})^T) - I, \quad (6)$$

where A_0 is (5) for the undeformed lattice and I is the identity matrix. From this it is clear that there is a one-to-one correspondence between the strain tensor components and the elements of B_i if the undeformed lattice constants are known.

The strain tensor as defined in (6) is sometimes termed the linear Lagrangian strain tensor [Schlenker et al., 1978].

2.2.1 Rotating the strain tensor

The strain tensor as defined above relates to the Cartesian grain system which is rotated by U_i relative to the sample system. Below is sketched how the different coordinate systems are related:

$$\begin{array}{ccc}
\text{undeformed} & U_{i,0} & \text{undeformed} \\
\text{sample} & \longleftrightarrow & \text{Cartesian grain} \\
\text{coordinate system} & U_{i,0}^T & \text{coordinate system} \\
\\
& A_0^{-1} \downarrow \uparrow A_0 & \\
& \text{direct space reference} & \\
& \text{coordinate system} & (7) \\
\\
& A_i \downarrow \uparrow A_i^{-1} & \\
\\
\text{deformed} & U_i & \text{deformed} \\
\text{sample} & \longleftrightarrow & \text{Cartesian grain} \\
\text{coordinate system} & U_i^T & \text{coordinate system}
\end{array}$$

From this it can be derived that if ϵ_i is the strain in the Cartesian grain coordinate system describing the difference between the undeformed and deformed crystal lattices, then:

$$\epsilon_i^{sample} = U_i \epsilon_i U_i^T \quad (8)$$

is the strain tensor in the sample coordinate system. Here it has been assumed that $U_i = U_{i,0}$ since only U_i and not $U_{i,0}$ can be obtained from the diffraction experiment. In the case of elastic deformation this is a good approximation, but it does not hold in the plastic deformation regime where grain rotations are non-negligible.

2.3 Error estimation

The covariance matrix of $\Gamma_{ij}^{-1} \overline{G}_{ij}$, V_{ij} , is assumed to be diagonal (thus no covariance) and equivalent for all reflections corresponding to a specific grain. Assuming that the experimental error on ω_{ij} is significantly larger than the error on the detector coordinates $y_{det,ij}$ and $z_{det,ij}$ and remembering that:

$$\Gamma_{ij}^{-1} \approx \begin{pmatrix} \cos(\omega_{ij}) & \sin(\omega_{ij}) & 0 \\ -\sin(\omega_{ij}) & \cos(\omega_{ij}) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (9)$$

we must have $V_{ij}^{11} \approx V_{ij}^{22} > V_{ij}^{33}$. This relative weighting of the errors in x , y and z is taken into account by setting $V_{ij}^{11} = V_{ij}^{22} = 4 \cdot 10^{-8}$ and $V_{ij}^{33} = 10^{-8}$. If these error estimates are correct the contribution from grain i to (1) must be equal to the number of observation minus the number of refined parameters [Eadie et al., 1971],

thus in most cases $3 \cdot j(i) - 12$. In case the contribution from grain i to (1) differs from the predicted value, the estimated experimental errors are scaled to make the observed contribution equal to the prediction. The errors on the refined parameters are then calculated by the MINUIT Variable Metric Minimisation to be the change in each parameter causing the value of (1) to increase by 1.

3 Algorithm

The FitAllB program package contains 3 different script which basically run the same program in 3 different preset modes. The reason for this is that in order to obtain accurate grain resolved strain tensors it is of the utmost importance that the global parameters relating to the experimental setup are well calibrated. The calibration or refinement of global parameters must be performed using an undeformed polycrystalline sample. A good start guess on both the global parameters (ImageD11) and grain orientations and positions (GrainSpotter) is a prerequisite. The refinement is then carried out by alternately refining the global parameters and the grain parameters (positions and orientations) for a preset number of cycles to minimise correlations. When fitting the global parameters the tilt of the sample stage corresponding to a rotation around the beam (ImageD11: chi, FitAllB: wx, c.f. Table 4.1.3) is kept fixed as this correlates with the detector tilt component corresponding to a rotation around the same axis (ImageD11: tilt_x, FitAllB: tx). Furthermore the beam centre in the z -direction (ImageD11: z.center, FitAllB: cz) is kept fixed as this correlates with the grain positions along z . After every cycle the refined global parameters are output in an ImageD11 format detector.par file so that the information is readily available for running FitAllB.

fitallb.py is the main script for refining COM positions, orientation and strain tensors after having calibrated the global parameters. Only grain specific parameters are refined. If nearfield diffraction data are available these are initially used to fit the grain positions, which are then kept fixed in the subsequent farfield refinement of orientations and strains.

fitglobalgrain.py is used to refine the global parameters for each grain. The final values and estimated errors are then obtained as the average and spread over all grains.

fitglobal.py is used to refine the global parameters for all grains in one go. This procedure is more time consuming, but less affected by outlier grains.

3.1 FitAllB algorithm

3.1.1 Farfield detector only

Read files and set defaults

Outlier rejection

Refine COM positions, orientations and strain tensors for each grain (**grain**)

Outlier rejection

While more outliers are being rejected

 For all grains from which outliers have just been rejected

 Refine COM positions, orientations and strain tensors for each grain (**final**)

 Outlier rejection

3.1.2 Farfield and nearfield data

Read files and set defaults

Match spots on nearfield with grains indexed on farfield

Nearfield outlier rejection

Refine COM positions (and orientations) for each grain on nearfield(**rotpos**)

Nearfield outlier rejection

While more outliers are being rejected

 For all grains from which outliers have just been rejected

 Refine COM positions (and orientations) for each grain on nearfield (**rotpos**)

 Nearfield outlier rejection

Initialise farfield refinement

Farfield outlier rejection

Refine COM orientations and strain tensors for each grain on farfield (**grain**)

Farfield outlier rejection

While more outliers are being rejected

 For all grains from which outliers have just been rejected

 Refine orientations and strain tensors for each grain on farfield (**final**)

 Farfield outlier rejection

3.2 Algorithms for fitting global parameters

Read files and set defaults

Farfield outlier rejection

For user defined number of cycles ($\# \in \{0, \dots, \text{number_of_cycles}-1\}$)

Refine COM positions and orientations for each grain on farfield (**rotpos#**)

Farfield outlier rejection

While more outliers are being rejected

For all grains from which outliers have just been rejected

Refine COM positions and orientations for each grain on farfield (**rotpos#**)

Farfield outlier rejection

fitglobalgrain: Refine global parameters for each grain and average (**globals#**)

or

fitglobal: Refine global parameters for all grains simultaneously (**globals#**)

Farfield outlier rejection

4 Input

file	origin	contents
grainspotter.log	GrainSpotter	number of grains, for each grain: U , and for all assigned peaks: $h, k, l, 2\theta, \eta, \omega, \text{peak_id}$
peaks.tXXXX.ftt	peaksearch	filtered peaks, for each peak: $\text{peak_id}, y_{det}, z_{det}, \omega, \sigma_y, \sigma_\omega, I$
detector.par	ImageD11	experimental parameters: unstrained unit cell, wavelength, tilt, sample-to-detector distance etc.
crystalinfo.pdb <i>or</i> crystalinfo.cif	structure database	optional crystallographic information for volume/intensity based outlier rejection
fitallb.inp	in time: GUI	name can be anything, see examples below

4.1 Example of FitAllB input files

4.1.1 Farfield detector only

```

title 'IF steel, experimental data, undeformed layer 50'
##### Mandatory input files, farfield strain fitting
log_file S1_e0_050_grainspotter.0.35.log
flt_file S1_e0_050.flt
par_file frelon_fitted_july08.par
##### Additional file to be read for volume based outlier rejection
structure_file IF_steel.cif
##### Mandatory experimental parameters
dety_size 2048      # detector dimension in pixels
detz_size 2048      # detector dimension in pixels
w_step 0.5          # step size in omega
w_limit -22.5 22.5 67.5 112.5  # limits of omega intervals
crystal_system cubic
##### on/off possibilities for fitting
# Global parameters, do not fit
w 0                # Fit omega stage tilt parameter wy (1=yes/0=no)
tilt 0             # Fit detector tilt parameters tx, ty, tz (1=yes/0=no)
pixel 0            # Fit pixel size py and pz (1=yes/0=no)
center 0           # Fit beam centre on detector along y, cy (1=yes/0=no)
L 0                # Fit sample-to-detector distance (1=yes/0=no)
# Grain parameters, fit all
rod 1              # Fit orientations and thus Rodrigues vectors (1=yes/0=no)
xyz 1              # Fit cms positions (1=yes/0=no)
eps 1              # Fit strain tensors (1=yes/0=no)
##### Outlier rejection information
rej_ia 0.2         # Outlier limit in deg for gvector misorientations
rej_vol 5          # Outlier limit for volume/intensity based rejection
rej_resmedian 10   # Outlier limit for robust residual based rejection
rej_resmean 5      # Outlier rejection limit in terms of average peak contribution
overlap 1          # grain overlap tolerance
skip 31 45         # skip grains 31 and 45 in the GrainSpotter log_file
##### Tolerances
tol_grain 1e-3
##### Parameters for strain-to-stress conversion
c11 23.7e10
c12 14.1e10
c44 11.6e10
##### Parameters used for resuming refinement
#resume grain
#res_file S1_e0_050/S1_e0_050_grain.txt
#rej_file S1_e0_050/S1_e0_050_rej.txt

```

4.1.2 Farfield and nearfield data

```

title '10 grains of IF steel, 2 detectors'

```

```

##### Mandatory input files, farfield strain fitting
log_file if100_t50_10.log
flt_file if100_t50.flt
par_file if100.par
##### Additional files for nearfield position fitting
near_flt_file if100_near_t50.flt
near_par_file if100_near.par
##### Additional file to be read for volume based outlier rejection
structure_file IF_steel.cif
##### Mandatory experimental parameters
w_step 0.5          # step size in omega
w_limit -22.5 22.5 67.5 112.5 # limits of omega intervals
dety_size 2048      # farfield detector dimension in pixels
detz_size 2048      # farfield detector dimension in pixels
near_dety_size 1536 # nearfield detector dimension in pixels
near_detz_size 1024 # nearfield detector dimension in pixels
crystal_system cubic
##### Parameters for strain-to-stress conversion
c11 23.7e10
c12 14.1e10
c44 11.6e10
##### Outlier rejection information
# farfield
rej_ia 0.2          # Outlier limit in deg for gvector misorientations
rej_vol 5           # Outlier limit for volume/intensity based rejection
rej_resmedian 10    # Outlier limit for robust residual based rejection
rej_resmean 5       # Outlier rejection limit in terms of average peak contribution
min_refl 36         # Grains with fewer reflections than min_refl are skipped
# nearfield
# use same parameters as for farfield, thus no need to specify near_rej_ia, etc..
##### on/off possibilities for fitting
# Global parameters, do not fit
w 0                 # Fit omega stage tilt parameter wy (1=yes/0=no)
tilt 0              # Fit detector tilt parameters tx, ty, tz(1=yes/0=no)
pixel 0             # Fit pixel size py and pz (1=yes/0=no)
center 0            # Fit beam centre on detector along y, cy (1=yes/0=no)
L 0                 # Fit sample-to-detector distance (1=yes/0=no)
# Grain parameters, farfield
xyz 0               # Do not fit cms positions on farfield
rod 1               # Fit orientations and thus Rodrigues vectors on farfield
eps 1               # Fit strain tensors on farfield
# Grain parameters, nearfield
near_xyz 1          # Fit cms positions on nearfield
near_rod 0           # Do not fit orientations and thus Rodrigues vectors on nearfield
near_eps 0          # Do not fit strain tensors on nearfield
##### Tolerances
tol_xyz 1e-1
tol_grain 1e-3
tol_rotpos 1e-2

```


4.1.3 Global parameter fitting, farfield only

```

title 'fitgloball 5 grains of IF steel'
##### Mandatory input files
log_file if100_globals_wedge_5.log
flt_file if100_globals_wedge.flt
par_file if100_globals_wedge.par
##### Additional file to be read for volume based outlier rejection
structure_file IF_steel.cif
##### Mandatory experimental parameters
w_step 0.5
w_limit -22.5 22.5 67.5 112.5
detx_size 2048      # detector dimension in pixels
detz_size 2048      # detector dimension in pixels
crystal_system cubic
##### Number of refinement cycles
cycle 20
##### Outlier rejection information
min_refl 45
rej_ia 0.2
rej_resmedian 5
rej_resmean 10
rej_vol 5
##### Global parameters
w 1      # Fit omega stage tilt parameter wy
tilt 1    # Fit detector tilt parameters tx, ty, tz
pixel 0   # Fit pixel size px and py
center 1  # Fit beam centre on detector along y-axis, cy
L 1       # Fit sample-to-detector distance
##### Grain parameters
xyz 1     # Fit grain positions
rod 1     # Fit orientations and thus Rodrigues vectors
eps 0     # Fit strain tensors
##### Tolerances
tol_global 1e-1
tol_rotpos 1e-2

```

The name of the file can be anything and the format is simply ascii. The commands can be in any order. Some of them are mandatory, the rest will be given default values if nothing is specified. `#` is used for commenting.

Table 1: Input files and overall information about experimental setup and sample. **Mandatory** commands are given in boldface.

command	usage
title	Anything to specify the refinement NB! Remember to use quotation marks
log_file	Name of the log file from GrainSpotter to be read
flt_file	Name of the filtered peaks file from peaksearch to be read
par_file	Name of the detector.par file from ImageD11 to be read
near_flt_file	Name of the nearfield filtered peaks file
near_par_file	Name of the nearfield detector.par file
structure_file	(possible formats .cif or .pdb) Contains information about the crystal structure that is used for outlier rejection based on intensities (Section 6.2)
w_step	Step size in ω
w_limit	Limits of the ω -intervals collected (max and min for every interval must be given, order irrelevant). If the data are collected as only one interval w_limit can be omitted as the limits can be determined from the filtered peaks file. Used to reject peaks on the edges of the ω -intervals.
bg near_bg	Average background levels on far- and nearfield detectors. Used to reject overflown peaks, i.e. peaks with a maximum intensity above $2^{16} - \text{bg}$. Defaults: bg 1000, near_bg 100
dety_size detz_size near_dety_size near_detz_size	Farfield detector size along y and z in pixels default values are 2048 and 2048. Nearfield detector size along y and z in pixels default values are 1536 (y) and 1024 (z). Used to reject peaks close to the edges of the detector and to convert to the ImageD11 beam centre convention in case of a flipped detector, so it is important to check whether the defaults are correct.
crystal.system	Used for setting up stiffness tensor, allowed possibilities as of present: isotropic, cubic, hexagonal, orthorhombic, monoclinic, triclinic. NB! All crystal systems can be treated as triclinic. This information is also important for converting the orientation into the fundamental zone, therefore it is mandatory.
c11, c12, etc.	Stiffness constants. The program will stop if the supplied c's are not enough to fulfil the requirements of the crystal system.

Table 2: Fit options: Parameters and tolerances. NB! The global parameters also exist in a near_ version, for instance the tilt of the sample stage should be the same for both detectors, so fitting global parameters for 2 detectors using fitglobalgrain/fitglobal one should use w 1 and near_w 0.

command	usage
w	Fit omega stage tilt parameter wy (1=yes/0=no)
tilt	Fit detector tilt parameters tx, ty, tz (1=yes/0=no)
pixel	Fit pixel size py and pz (1=yes/0=no)
center	Fit beam centre on detector along y, cy (1=yes/0=no)
L	Fit sample-to-detector distance (1=yes/0=no)
xyz	Fit cms positions on farfield (1=yes/0=no)
rod	Fit orientations and thus Rodrigues vectors on farfield (1=yes/0=no)
eps	Fit strain tensors on farfield (1=yes/0=no)
near_xyz	Fit cms positions on nearfield (1=yes/0=no)
near_rod	Fit orientations and thus Rodrigues vectors on nearfield (1=yes/0=no)
near_eps	Fit strain tensors on nearfield (0=no only option)
tol_step	where step can be rotpos, grain or global. Fit tolerance used in the corresponding refinement step. Defaults are: tol_rotpos = tol_global = 1e-2 and tol_grain = 1e-3. Note that the program sets tol_final = tol_grain·0.1. The refinement is considered converged when the estimated distance to the minimum is less than 10^{-3} times the specified tolerance.
res_file	Read starting values of grain parameters from this .gff (grain file format). If used the peaks assigned to each grain in the grainspotter.log are replaced by matches made by fitallb based on the grain parameters in res_file.
cycle near_cycle	Number of cycles for fitglobalgrain/fitglobal, default=20

Table 3: Outlier rejection commands. NB! All `rej_` commands can be used in a `near_rej_` version to specify that the parameters used for the nearfield refinement should be different from the farfield case. The same goes for the `overlap` and `min_refl` commands, but not for `skip`.

command	usage
<code>rej_ia</code>	Outlier rejection limit in degrees for gvector misorientation, Section 6.1. All peaks with misorientations larger than this are rejected. Default: 0.2.
<code>rej_vol</code>	Outlier limit for volume/intensity based rejection, Section 6.2. NB! Requires a valid <code>structure_file</code> command. Default: 5.
<code>rej_resmedian</code>	Outlier limit for robust residual based rejection, Section 6.3.1. Default: 5.
<code>rej_resmean</code>	Outlier rejection limit in terms of average peak contributions, Section 6.3.2. Default: 10.
<code>rej_multi</code>	Remove peaks assigned to more than one grain (1=yes/0=no), Section 6.4. If yes (=default) the peak is removed from the grain where the fit is poorest.
<code>overlap</code>	Criteria for merging grains, Section 6.5. Grain with a larger fraction of overlapping grains than specified are merged. Default: 1., i.e. no merging.
<code>min_refl</code>	If the number of reflection assigned to a grain drops below the specified value the grain is skipped in subsequent refinement steps. Defaults: <code>min_refl</code> 9, <code>near_min_refl</code> 3.
<code>skip</code>	Used to rule out specific grains from the GrainSpotter log file that for some reason or another are known to be poor.

5 Output

All output files will be saved in `i` directory with the name of the used input file up to the first punctuation, so in the example using `fitallb.inp` as input file, the output directory will be named `fitallb`. This directory name will also be used as the stem of the generated output files which are given in Table 5.

The parameters `eps11`, `eps22`,... are the components of the strain tensor in the Cartesian grain coordinate system; `eps11_s`, `eps22_s`,... are the strain tensor components in the sample system; `sig11`, `sig22`,... and the the components of the stress tensor in the Cartesian grain coordinate system; and finally `sig11_s`, `sig22_s`,... are the strain tensor components in the sample system.

The MINUIT program puts certain limitations on how the function FCN to be minimised is implemented. To make sure that the function has the right specifications it is necessary to build it before every refinement step. The building is performed by the `build_fcn.py` script and the result is a file called `fcn.py` placed in the output directory (in our example `fitallb`) which contains the functions to be minimised in the correct syntax. `fcn.py` is read by the program during execution, but deleted upon normal termination of the fitting.

Besides generating the described output files (in directory `fitallb`), the program will give a lot of info in the dialog window from which it was called. To start out with the number of reflections rejected based on the different criteria are listed. If nearfield data are used in the refinement or grain parameters are read from a `.gff` the program will use the grain parameters and do a forward projection onto the detector to harvest diffraction spots rather than using the ones assigned in the `grainpotter.log`. It may happen that the number of assigned peak is higher than actually possible because very loose cuts are being used, but these falsely assigned peaks will eventually be removed as outliers. Later on follows list of all grain with the number of assigned reflection, the total contribution to (1) from each grain and the average contribution per per reflection (should be roughly 3 according to the arguments in Section 2.3). While the program is running a counter specifies which grain is being refined now to give an idea of the refinement progress.

Table 4: List of output files from FitAllB, FitGlobalGrain and FitGlobAll

file	contents
fitallb_log.log fitallb_near_log.log	Main output files for far- and nearfield: Global parameter values and standard deviations Residual values, refinement times and number of outliers for each refinement step
fitallb_rej.txt fitallb_near_rej.txt	List of rejected peaks and skipped grains for each refinement step The reason to reject the peak, i.e. intensity or residual, is specified
fitallb_cor.txt fitallb_near_cor.txt	Correlation matrix of the refined parameters for each grain New matrices appended after each grain and final step
fitallb_cov.txt fitallb_near_cov.txt	As fitallb_cor.txt, except this is the covariance matrices
fitallb_global.txt	correlation and covariance matrices of the refined global parameters are appended to this file after each cycle
fitallb_globals#_fab.par	ImageD11 type detector.par file with refined global parameters
fitallb: nearfield: fitallb_near_rotpos.gff farfield: fitallb_grain.gff fitallb_final.gff fitglobalgrain/fitglobal: fitallb_rotpos#.gff fitallb_globals#.gff	For each step in the algorithm the values of the parameters for each grain is listed in these (grain file format) files. Parameter order: grainno mean_IA grainvolume x y z rodx rody rodz U11 U12 U13 U21 U22 U23 U31 U32 U33 eps11 eps22 eps33 eps23 eps13 eps12 eps11_s eps22_s eps33_s eps23_s eps13_s eps12_s sig11 sig22 sig33 sig23 sig13 sig12 sig11_s sig22_s sig33_s sig23_s sig13_s sig12_s NB! If more than one final step is required the output file is overwritten in each step
fitallb_errors.txt	Estimated standard deviation of all the grain parameters given in the same order as listed above for the values The file is updated after each refinement step. Step specific output files are names fitallb_step_errors.txt

6 Outlier rejection

Several different outlier rejection schemes are implemented into FitAllB, and the philosophy behind each of these will be outlined below. It should be noted that grains with less than a user defined number of assigned peak (min_refl) will be skipped to avoid spending time on refining non-physical ghost grains.

Median Absolute Deviation

Median absolute deviation (MAD) is used a robust measure for outlier rejection within FitAllB. Given the data set X_1, X_2, \dots, X_n the median absolute deviation is defined as:

$$\text{MAD} = \text{median}_i(|X_i - \text{median}_j X_j|)$$

For normally distributed data more than 99% of the observations should fall between the $\text{median} \pm 5\text{MAD}$. In FitAllB the method is used iteratively:

While more outliers are present in the data set

Calculate median and MAD of data set

Reject data point differing by more than $\text{madlimit} * \text{MAD}$ from median

6.1 Based on internal angles

The internal angle (measured in degrees) is the angle between $\Gamma_{ij}^{-1} \bar{G}_{ij}$ and $\frac{\lambda}{2\pi} U_i B_i \bar{G}_{hkl,ij}$ of (1). All reflections with an internal angle greater than `rej_ia` will be rejected as outliers. The default value for undeformed materials is 0.2, this must be increased for deformed samples.

6.2 Based on intensities

This check is only carried out if a valid structure file is given in `fitallb.inp`. In this case the measure $\frac{I}{L|F^2|}$ is calculated for each peak. Here I is the intensity, L is the Lorentz factor and $|F^2|$ is the theoretical structure factor squared determined based on the structure file. This quantity is directly proportional to the grain volume and should thus be roughly the same for all peaks assigned to a given grain. Thus a MAD outlier rejection with a `madlimit` of `rej_vol` is performed on these volume measures within each grain. The default value of 5 was chosen based on the arguments in MAD Section.

6.3 Based on residuals

The residual is the refined quantity and should thus be as small as possible. Sometimes a single peak contributes abnormally much to the residual. This could either be because the peak belongs to a grain for which the parameters are still poor (in which case usually all peaks assigned to this grain will contribute rather much to the residual) or because the peak is assigned to a wrong grain. It is the peaks of this second kind that the residual based outlier rejection is tuned to spot and get rid of.

6.3.1 Based on residuals, robust

At any given time the contribution to the residual from each peak can be calculated. Within each grain the mean and median of these are determined and the peaks with the largest residual contributions are rejected until the mean is less than `rej_resmedian` times the median. The default value is 5, but it can also be lowered.

6.3.2 Based on average residuals

After each refinement step the peaks contributing more than `rej_resmean` (default=10) times the mean peak contribution to the residual within each grain are rejected. This specific outlier rejection scheme is, contrary to the others, not used until after the first refinement step as it is not very robust towards poor starting values of the grain parameters.

6.4 Peaks assigned to multiple grains

Occasionally a number of peaks will be assigned to more than one grain, especially if many grains are illuminated simultaneously, the sample is textured or very loose cuts have been applied in the indexing/forward projection. `FitAllB` identifies these peaks and looks at how they contribute to each of the assigned grains. If it is found that the peak gives rise to a higher residual in `grain1` than in `grain2`, and at the same time the volume calculated for the peak is further from the mean volume in `grain1` than in `grain2`, then the peak will be rejected from `grain1`.

6.5 Merging of grains

It sometimes happens that the `GrainSpotter` comes up with two grains that are said to be different but in fact have the same orientation and many of the same peaks assigned. `FitAllB` contains a routine for identifying and merging grains of this type. In `fitallb.inp` the `overlap` command tells how large a fraction of overlapping

peaks is allowed before the grains are considered to be identical. The default value is 1, i.e. do not merge anything. Grains where a larger fraction of the assigned peaks are identical are cornered out and the grain with the fewest assigned peaks is skipped. In case the two grains contain the same number of peaks, only the identical peaks are kept in the refinement.

7 Installation and execution

FitAllB can be downloaded from:

<https://fable.svn.sourceforge.net/svnroot/fable/FitAllB>

Go to the trunk and install the program by:

```
python setup.py install
```

After setting the relevant paths the program can then be called from the directory containing all input files in the following way:

```
fitallb.py -i fitallb.inp
fitglobalgrain.py -i fitglobalgrain5_5.inp
fitglobal.py -i fitglobal5_5.inp
```

For more information on the standard 3DXRD geometry the user is referred to the geometry document [Poulsen et al., 2009].

7.1 Provided test examples

All files required to perform the below describe test runs are found in trunk/test.

if10.inp: 10 grains of IF steel simulated using PolyXSim within a volume of $0.7 \times 0.7 \times 0.01$ mm with random orientations, grain sizes and Gaussian strains with a mean of 0 and a spread of 0.001. The refinement is an example of using both near- and farfield data, and the runtime is approx. 1 min per grain on a 2.2 GHzm 3.5 GB RAM PC.

if10.inp_res: Same data as for if10.inp, but only the farfield refinement is performed. The starting values of all grain parameters is read from the if10_near_rotpos.gff file generated in the if10 refinement to show how resuming a refinement can be done. Note that the grain positions refined on the nearfield detector are kept fixed in this farfield refinement. Runtime approx. 40 sec per grain.

fitglobalgrain5_5: 5 grains of undeformed IF steel simulated within a volume of $0.7 \times 0.7 \times 0.01$ mm with random orientations and grain sizes. The input file of global parameters, if100_globals_wedge.par, has a slight offset in all of these to demonstrate the powers of the fitglobalgrain.py. The true parameters used for the simulation are given in if100_globals_wedge_true.par for comparison. 10 refinement cycles performed. Runtime: min.

fitgloall5_5: Same input as for fitglobalgrain5_5, but intended for use with fitgloall.py to demonstrate the difference between the two. Runtime:

fitgloall5_res2: Resuming the fitgloall refinement of fitgloall5_5 using as input the final grain parameters, fitgloall5_5_rotpos4.gff, and globals, fitgloall5_5_globals4_fab.par, as input to do 2 more cycles.

7.2 Required packages

python 2.4 or later

numpy (<http://www.numpy.org/>)

pyminuit (<http://code.google.com/p/pyminuit/>)

From FABLE: ImageD11, xfab, polyxsim

7.3 Contact

Questions, comments and suggestions are most welcome!

Jette Oddershede

Risø DTU

April 2009

jette.oddershede@risoe.dtu.dk

References

- [Eadie et al., 1971] Eadie, W. T., Dryard, D., James, F. E., Ross, M., and Sadoulet, B. (1971). *Statistical Methods in Experimental Physics*. North-Holland Publishing Company.
- [James, 1972] James, F. (1972). <http://www.cern.ch/minuit>.
- [Poulsen, 2004] Poulsen, H. F. (2004). *Three-Dimensional X-ray Diffraction Microscopy*. Springer.
- [Poulsen et al., 2009] Poulsen, H. F., Schmidt, S., Wright, J., Sørensen, H. O., Oddershede, J., and Alpers, A. (2009). 3DXRD and TotalCryst Geometry http://fable.wiki.sourceforge.net/space/showimage/Geometry_version_1.0.7.pdf.
- [Schlenker et al., 1978] Schlenker, J. L., Gibbs, G. V., and Boisen Jr., M. B. (1978). *Acta Cryst.*, A34(1):52–54.