Win-IntegrStp V3.53

Users manual

RJ ANGEL

rossangelsoftware@gmail.com

http://www.rossangel.com

INTRODUCTION

WinIntegrStp is a Windows program that allows single-crystal X-ray intensity data collected in step-scan mode with a point detector to be examined and integrated by full-profile fitting. It runs in both manual and automatic modes.

Version 3.3 of WinIntegrStp was first made available in April 2002, and updated September 2002. It was a "beta-test" version. It ran on Windows-NT platforms and some Win-98 machines. Due to some compiler problems it did not run on other Windows systems. Feedback from users resulted in version 3.4 which has been demonstrated to run correctly on Windows NT, ME, XP, 98 and 95. Version 3.4 was released in April 2003. New file formats and utilities were added to version 3.4 in summer 2003.

Version 3.5 incorporates some bug fixes, a variety of increased reporting to the log file, and uses v6 of the GINO library, instead of the version 5 used by previous compilations of WinIntergStp. It was developed over the period from Spring 2004 to Fall 2005. Version 3.5 was released at a beta-test version to some users in December 2005 and, after testing, was released for general use in January 2006. Subsequent minor versions incorporate minor bug fixes – see Section 1 for details of the latest changes.

The WinIntegrStp program is distributed on a non-commercial basis and the author would appreciate its use being acknowledged in any publications that make use of it by reference to:

> Angel RJ (2003) Automated profile analysis for single-crystal diffraction data. *Journal of Applied Crystallography*, 36:295-300.

If you would like to receive program updates (including bug fixes), please register with me as a user by e-mail (rossangelsoftware@gmail.com). If you discover apparent bugs in the program, please send me the input files, the output file and a full description of the problem by e-mail. Other suggestions for improvements and modifications will also be considered. Further information will be posted on the web site http://www.rossangel.com.

The manual is divided into three major sections. The background to the program is described in "Methods". New users are encouraged to follow the instructions in the "Program" section to install the program and to follow through the worked example before attempting to integrate their own data. The Appendices contain further details of the operation of the program and the format of the *instrument parameter files*.

ACKNOWLEDGEMENTS

The code has been developed from an original VAX-VMS program written by Larry Finger. Many users have provided significant feedback and undertaken testing of the program, especially David Allan, Tiziana Boffa-Ballaran, Fernando Camara, Diego Gatta, Mario Koch, Nancy Ross, Mark Welch, and Jing Zhao. Thankyou!

DISCLAIMER

While I try to ensure that the WinIntegrStp software is free of bugs and errors, people use it at their own risk. I cannot accept any responsibility whatsoever for either incorrect results or for any physical, mental or other damage arising from use of this software, nor from errors in this documentation.

1. VERSION NOTES

The current version of WinIntegrstp is 3.53. It includes developments and changes made in the period of October 2011 through to March 2012. The significant changes are listed here, along with indications as to where to find further details in this manual:

- The alpha1 and alpha2 components of the calculated peak profile are plotted.
- More information added to the log file.
- The log file is opened automatically.

Some changes were made to the code to correct very small inconsistencies in the peak positions and unit cell parameters determined on the same data by the SINGLE program and WinIntegrstp. The current versions of WinIntegrstp, SINGLE and the zref utility all now produce results that are the same within the rounding precision of the output data format (e.g. 0.0001 in unit-cell volumes):

- Circle parities are no longer applied to the data when it is read in, but are inly used in the calculation of the orientation matrix. This is now consistent with SINGLE.
- Changed the card that describes the goniometer size in the exp file. It was called RCRS, but is renamed GONCON and the order of the paramateres has been changed to be consistent with the usage in Single *difprof.dat* file. See the Appendix for details of GONCON.
- Corrected the peak position calculation with monochromator data.
- Changed the output format of the angles in *int* files from 4F8.3 to 2F8.4,2F8.3 to accommodate the higher precision from omega and 2theta circles with motor steps that are not exact multiples of 0.001deg.

2. METHODS

Introduction

Profile fitting methods are widespread in the analysis of powder diffraction patterns, but are rarely used in the extraction of intensities from single-crystal datasets collected by step-scan methods. Instead, most step-scanned datasets are integrated by "counting" methods in which a certain percentage of both ends of each scan is pre-defined as background, and the counts at the steps between these limits are assigned to the integrated intensity of the peak from which the background is subsequently subtracted. Some integration programs implement a dynamic setting of the background limits, often following the algorithm proposed by Lehmann and Larsen (1974). This algorithm has the advantage over fixed background methods that it improves the $I/\sigma(I)$ ratio by only integrating the peak area. The dynamic setting of background limits also allows for the peaks being slightly offset from the middle of the scans.

However, while these various integration methods for step scans provide good estimates of integrated intensities for peaks with good signal-to-noise ratios, they have several drawbacks. They provide no information about peak shape with which to test whether the maximum in a step scan actually arises from the crystal diffraction, or whether it is an artifact, perhaps arising from diffraction from an environmental cell. Neither do "counting" methods provide a mechanism for the recovery of weak reflections from datasets. As a result, experience has shown that in order to obtain high-quality refinements from crystals held at high-pressures in diamond-anvil cells, every reflection profile in a dataset must be examined graphically by the experimentalist – a long, and tedious process.

By contrast, peak-profiling methods in which the step-scan data for each reflection is fitted with an appropriate profile function provide the possibility of addressing these issues (Pavese and Artioli 1996). And the tests for peak rejection can be quantified, based on the refined peak shape parameters, thereby allowing the entire integration process to be automated. The WinIntegrStp program is a utility to perform full peak-profiling of step-scan data with the methods of Pavese and Artioli (1996). In particular WinIntegrStp provides:

- full interactive control of profile refinement
- full set of tests of peak shape and position
- full interactive control of peak rejection tests
- various methods of recovery of weak peaks from noisy data.

For a full historical review of peak-profiling methods as applied to single-crystal diffraction data, the reader is encouraged to read Pavese and Artioli (1996) and the references therein.

Profile Function

In WinIntegrStp each step scan is fitted by the method of weighted least-squares to a pair of pseudo-Voigt functions to describe the α_1 and α_2 contributions to the profile, plus a constant background (e.g. Pavese and Artioli 1996):

$$I(\omega) = \mathrm{PV}_1(\omega) + \mathrm{PV}_2(\omega) + constant \qquad (1)$$

Each single pseudo-Voigt is written as:

$$PV(\omega) = \frac{I\eta\Gamma}{2\pi(\omega-\omega_0)^2+\Gamma^2} + \frac{2I(1-\eta)\sqrt{\ln 2}}{\Gamma\sqrt{\pi}} \exp\left[-\left(\frac{2(\omega-\omega_0)\sqrt{\ln 2}}{\Gamma}\right)^2\right] \qquad (2)$$

The refineable parameters for a single pseudo-Voigt are:

- position ω_0
- total integrated intensity I
- full width of the peak at half-maximum Γ
- mixing parameter η which is zero for a pure Gaussian peak and 1 for a pure Lorentzian.

The numerical constants π and $\sqrt{\ln 2}$ in (2) serve to normalise the profile so that the parameter I is the true integrated intensity.

When both α_1 and α_2 wavelengths contribute peaks to the profile (as in Equation (1)) it is reasonable to assume that both peaks have the same η and Γ parameters. The position of the α_2 peak is calculated from that of the α_1 peak plus an offset due to the known α_1/α_2 dispersion (see below). In WinIntegrStp the relative integrated intensity of the α_2 peak is expressed in terms of a refineable parameter $I_{ratio} = I(\alpha_2)/I(\alpha_1)$. The peak function defined by Equations (1) and (2) then provides a reasonable representation of the peak shape over a wide range of Bragg angles. This formulation therefore results in six refineable parameters for each reflection scan:

- background term
- full width of the both peaks at half-maximum, Γ
- integrated intensity of the α_1 peak
- position ω_0 of the α_1 peak
- mixing parameter η of both peaks
- intensity of the α_2 peak as a fraction of the α_1 peak, $I_{ratio} = I(\alpha_2)/I(\alpha_1)$

The offset of the position of the α_2 peak from the α_1 peak depends on the geometry of the incident beam optics on the diffractometer, as follows:

- For an unmonochromated source, and also for one with a monochromator whose diffracting plane is 90° from that of the diffractometer, the offset of the α_2 peak is $\arcsin(\lambda_2 \sin\theta_1/\lambda_1) - \theta_1$. The frequently used approximation $\Delta\lambda/\lambda \tan\theta_1$ for this offset is in error by ~10^{-4} degrees at $2\theta = 40°$ and by ~10^{-3} degrees for $2\theta > 80°$.
- For a monochromator in parallel geometry there is an additional offset of the α_2 peak due to the α_1-α_2 dispersion by the monochromator crystal. This additional offset is added to the sample dispersion when the detector lies on one side of the direct beam, and subtracted when the detector lies on the other side. WinIntegrStp handles these contributions to the α_1-α_2 dispersion through the parameters provided on the wavel and mono cards in the *instrument parameter file*:
  - The parameter nwave on the wavel card specifies whether or not there the α_2 component is present, and the value(s) of the wavelength(s).
  - The mono card specifies the monochromator geometry, including the 2θ(mono), the dihedral angle. For parallel geometry monochromators it also specifies the sense of the monochromator dispersion with respect to the dispersion from the sample crystal.

Default values for the peak parameters are provided to the fitting routine in WinIntegrStp by the PKPAR card in the *instrument parameter file,* but can be changed interactively by the user. The

user also retains complete control over which parameters are refined and which are left fixed at default values.

Constraining Parameters

In order to obtain the best estimates of intensities from the integration process, it is useful to constrain some of the other six parameters in the least-squares refinement of each scan. This is particularly important for weak reflections, for which parameter correlations can lead in a full refinement to meaningless parameter values such as negative intensities. In WinIntegrStp the user has complete control over which parameters are refined and which are fixed to preset values. In this section we discuss which parameters to fix, and how to obtain the preset values. The underlying principle is that the shapes of the profiles of all of the reflections from one crystal on one diffractometer can be described by either constant values for the parameters, or values that vary smoothly and systematically across the dataset.

The values of the peak-shape parameter η and the intensity ratio I_{ratio} are usually the same for all crystals on a given diffractometer. Indeed, variation of these values is a sensitive indicator of a change in the diffractometer alignment. The parameters cannot be reliably determined from low-angle reflections because the overlap of the α_1 and α_2 components leads to strong correlations in the least-squares refinement of the profile function. The parameters η and I_{ratio} are therefore best determined for a sample crystal by fitting a few slow scans of strong, high-angle peaks from a good-quality crystal in which the doublet is well resolved. In the subsequent integration of datasets the values of η and I_{ratio} are kept fixed at the predetermined values (see also Pavese and Artioli 1996). It is recommended that a high-quality standard crystal is used to check these parameters at regular intervals, and that the correct values are loaded into the *instrument parameter file* on the PKPAR (for η) and WAVEL (for I_{ratio}) lines. The user can, however, change these values manually by entering them in the dialogue box under **Parameters|Pkvals**.

Pavese and Artioli (1996) noted that the refined intensity parameter is strongly correlated with the background parameter, so they sought for ways in which to provide more reliable integrated intensities through fixing or constraining the background parameter. Because the level of the radiation background away from the positions of diffraction often varies smoothly with the setting angles of the diffractometer, Pavese and Artioli (1996) used a polynomial function of the diffractometer setting angles to define the background value for each reflection. WinIntegrStp provides several alternative methods (controlled by the **Parameters|Background** dialogue box) for fixing the background:

- None.
- A spline function of 2θ (but not other angles). This is obtained by taking the counts at the end of each reflection scan, binning them into intervals of 1^o in 2θ, and then fitting a spline function to the minimum values in each data bin.
- An exponential function $back = a.\exp(-b.2\vartheta) + c$ in which *a, b,* and *c* are parameters that can be obtained from the counts at the end of each scan.
- Fixing the background of each scan to the average of the counts of a selected number of steps at both ends of each individual scan. If the difference in the total counts from the two sides of the scan exceeds three mutual *esd*'s (based upon counting statistics) only the side with the lower value is used to set the background value, and the "U" flag is set for the reflection. No R_{wp} test is performed on reflections with uneven backgrounds.

Experience so far has shown that the two global functions of 2θ often result in slight overestimates of the background level, and thus underestimates of intensities. This is probably a result of the background levels varying with the orientation of the crystal on the diffractometer, and not just as a function of 2θ (Pavese and Artioli 1996). For datasets such as those from a diamond-anvil cell that have a strong local variation in background levels we find that the "fixed background method" provides the most reliable estimates of intensities.

The peak widths of reflections from a single crystal usually remain constant, or vary slightly with 2θ. In WinIntegrStp the peak width variation is expressed as a function $\Gamma = a + b\tan\vartheta$ where the parameters *a* and *b* are preset in the *instrument parameter file* on the pkpar line. The user can change these values manually by entering them in the dialogue box under **Parameters|Pkvals**. The values for a dataset can also be obtained by fitting the values of the stronger reflections, as illustrated in the worked example in this manual; **Utilities|Preprocess** must be run to fit the strong reflections before **Utilities|Calc FWHM** is available. Strong variations of peak widths, especially with orientation, are indicative of a problem with the sample crystal, often the presence of pseudo-merohedral twinning.

The UB matrix can be reliably determined from the setting angles of the strong reflections, as described in detail in a following section. Although the positions (i.e. ω values of the peak maxima) should be described by the UB matrix, tests have shown that the calculated position of a peak is often displaced from the observed position by a few hundredths of a degree. These errors can be attributed to setting-angle errors and other experimental uncertainties. The errors are often smaller if the effects of crystal-offsets are included in the calculation of reflection positions, but the offsets are still sufficient to bias the integrated intensity values of strong reflections. Therefore, it is not recommended to use the position calculated from the UB matrix in the first attempt at fitting a reflection, even in automatic integration mode.

Peak Discrimination/Tests

As noted above, the refinement of a peak profile function provides the opportunity to automatically test for abnormal peak shapes and to reject the corresponding intensity data. The tests are controlled by the user through the **Fitting Controls and Results** dialogue box. The tests fall into several groups, as indicated below, and the test sequence is terminated after the first failure, which is indicated by the program through the setting of an error flag (given in **bold** below, and also listed in Appendix 2). The listed sequence of tests appears to be the one which is the most reliable in terms of both rejecting aberrant profiles and not rejecting correct profiles.

The first group of tests is applied to all refinements:

- Least-squares errors, including non-convergence in the refinement (**C**), invalid data (**D**) and various math errors (**E**).
- $R_{wp} = \sum w_i \left(y_{i,obs} - y_{i,calc}\right)^2 / \sum w_i y_{i,obs}^2$ is tested against the limit set by the user (**F**). Tests indicate that R_{wp} is a more sensitive test of peak shape than alternative measures of fit. The test is not applied if the fixed-background option has been selected and the reflection has uneven backgrounds.

The remainder of the tests are only applied if the user has requested them on the **Fitting Controls and Results** dialogue box:

- Background level against the values set by the user (**B**).
- Intensity against the minimum value (often 0) set by the user (**I**).

If the refined intensity is less than 6 esd's then the next two tests are not applied, even if requested by the user, because they are not reliable for low intensities:

- If η is refined and a test has been requested by the user, it's value is tested against the limits set by the user (**S**).
- If tests for either I_{ratio} or Γ have been requested by the user, a second refinement is performed in which the Intensity and I_{ratio} are refined. If the α_1-α_2 doublet is well-resolved, then Γ and the peak position are also refined simultaneously. If the value of Γ or I_{ratio} from this re-refinement fall outside of the user-set limits an error flag (**W** or **R** respectively) is set.

The last test, if requested, is applied to all reflections:

- If ω_0 (the peak position) has been refined and a test requested by the user it's value is tested against the limits set by the user (**P**). This test is applied to the original refinement if the refined intensity is less than 6 esd's, or to the re-refinement if $I>6\sigma(I)$.

UB refinement

The refinement of the UB matrix is based upon the angles and indices of the strong reflections selected and fitted by the **Utilities|Preprocess** option. The 2θ, χ and ϕ angles of each reflection are copied from the input data file and are the setting angles at which the data-collection scan in ω was performed. The ω angle assigned to each reflection is the position of the α_1 component of the diffraction peak, as obtained from the least-squares fitting of the diffraction profile. All unit-cell parameters determined by this fitting utility therefore correspond to the α_1 peak positions and the α_1 wavelength as set in the *instrument parameter file*. The unit-cell parameters so obtained should therefore be independent of the range of 2θ values of the reflections used in the refinement (Angel et al. 2000). The UB matrix is calculated for the setting angles converted to the parities and axial system of Busing and Levy (1967).

WinIntegrstp provides several options for determining the UB matrix and the unit-cell parameters:

- Refinement of unconstrained unit-cell parameters by the method of least squares.
- Refinement of symmetry-constrained unit-cell parameters by the method of vector-least-squares (Ralph and Finger 1982).
- Refinement of the crystal offsets and the unconstrained unit-cell parameters by the iterative method of Dera and Katrusiak (1999).
- Refinement of the omega-zero position and the unconstrained unit-cell parameters by an iterative method.

The iterative methods to refine crystal offsets or omega-zero proceed by iterating the following operations until convergence:

0) the UB matrix is refined in the normal way to the reflection data;
1) the differences between the observed ω_{obs} values and those predicted by the UB matrix and the indices are attributed to the crystal offsets or the omega-zero (as selected);
2) the crystal offsets or omega-zero are calculated from these differences;
3) the contribution of the crystal offsets or omega-zero, ω_{corr} to the ω of each reflection is calculated;
4) a new UB matrix is obtained by refinement to the reflection data but with the ω values replaced by ω-ω_{corr}. This new UB is used in 1) and the iteration 1) through 4) is repeated until convergence.

Tests have shown that the simultaneous refinement of omega-zero and crystal-offsets does not normally converge. Unless omega-zero is significantly different from zero, then its refinement alone with the UB matrix does not converge. WinIntegrStp does not at this time allow refinement of crystal offsets with a symmetry-constrained refinement.

Reflection recovery

Conventional integration methods involving counting provide no mechanism by which weak reflections can be recovered from the dataset, but by fitting profiles and constraining the profile parameters a greater proportion of weak intensities are successfully integrated on a first pass. Nonetheless for weak reflections, or reflections in areas of high and structured background, even the simultaneous refinement of peak position and intensity alone is unstable and often results in negative integrated intensities even though a weak maximum is present in the scan range.

Therefore there is an option within WinIntegrStp to constrain the peak position for reflection profiles that fail the profile tests on the first attempt. The procedure is invoked by selecting both the **R**efine and the **D**efault boxes for peak position on the **Fitting controls** dialogue box. The first attempt to fit a reflection profile proceeds as described above, followed by tests of the parameter values. If the profile fails a test, the vector $\mathbf{h}_\phi$ = **UB.h** is calculated and the value of ω consistent with the values of the components of $\mathbf{h}_\phi$ and the setting angles 2θ, χ and ϕ for the reflection is found. If crystal offsets or an ω circle zero were determined along with the refinement of the **UB** matrix, the effect of these is included in the calculation. The peak profile function is then refined again with the ω position of the α_1 peak fixed at this calculated value. After this refinement, the same series of tests are applied to the refined parameters.

Data Corrections and Output

In WinIntegrStp the profile parameters are fitted to the raw peak profile as read from the input scan data file without any prior scaling, except to add 1 to each step count to ensure no zero count entries are present (Blessing 1987). This is intended to preserve the counting statistics, as weights are assigned to each data point in a scan as the inverse of the counts. The profile displayed in the graphics window of WinIntegrStp is also the raw data.

The intensity and its *esd* provided in the output file are derived directly from the integrated intensity parameter of the α_1 peak as obtained by the least-squares fit. They are both scaled by such factors as scan time, attenuator etc as indicated in the raw data file. A Lorentz-polarisation factor is applied for an ideally imperfect crystal (Azaroff 1955), including the effect of the monochromator if one is indicated by a MONO card. No corrections are made for decay of intensities during the experiment. In principle, the intensity values in the step scans from a crystal held at high pressure should be corrected for absorption by the diamond-anvil cell prior to fitting as the absorption will vary as the cell is rotated during the scan. In practice such corrections are small and it has not been found necessary to implement them in WinIntegrStp.

Conventions

Internally the WinIntegrStp program uses the axial conventions and circle parities defined by Busing and Levy (1967):

When all circles are at their zero positions:

- the 2θ arm lies in the position of the undiffracted direct beam (2θ = 0)
- the plane of the χ circle is perpendicular to the direct beam (ω = 0)
- the ϕ-axis is perpendicular to the diffraction plane (χ = 0)
- the choice of ϕ = 0 is arbitrary.

These conventions also define the "normal-beam equatorial geometry" of Arndt and Willis (1966) subsequently generalised by Dera and Katrusiak (1998). In these zero positions the Cartesian basis of the "ϕ-axis" coordinate system (Busing and Levy 1967) has its axes defined as follows:

- the origin is at the centre of the diffractometer
- the positive *y*-axis extends from the crystal towards the detector (i.e. along the undiffracted direct beam)
- the positive *z*-axis is parallel to the ϕ axis, perpendicular to the diffraction plane, and away from the ϕ-axis carrier
- the positive *x*-axis makes a right-handed set, and corresponds to an imaginary diffraction vector at 2θ = 0.

The sense of positive rotations of the four diffractometer circles under the Busing and Levy (1967) convention are left-handed for all axes except for the χ-axis. To be explicit, when viewed *from the +z direction* (looking down on the diffractometer from above), positive movement of the 2θ, ω and ϕ axes away from their zero positions is clockwise. When viewed *from the +y direction* (looking towards the crystal from the detector arm) positive movement of the χ-axis is anti-clockwise. These senses of rotations are hereinafter defined as having *positive parities*. Circles on diffractometers that rotate in the opposite sense will be said to possess *negative parities*.

Eulerian setting angles in the input data file are converted to bisecting geometry. Kappa geometry angles in the CAD4 data files are converted to Eulerian geometry through the following equations taken from Schagen et al. (1988):

$$2\theta_E = 2\theta_K \qquad \chi_E = 2\arcsin(\sin\alpha \sin(\kappa/2))$$

$$\omega_E = \omega_K - \theta_K + \delta \qquad \phi_E = \phi_K + \delta$$

where $\delta = sign(\sin(\kappa/2))\arccos(\cos(\kappa/2)/\cos(\chi_E/2))$, ω_K is absolute and ω_E is bisecting.

The setting angles of the reflections are otherwise not modified, except for calculations involving the orientation matrix. For these calculations, the angles are modified internally by the Busing-Levy parities given on the PARITY card in the *instrument parameter file*. These parities therefore only affect the calculation of the orientation matrix, the crystal offsets, and the calculation of setting angles from the orientation matrix.

3. THE PROGRAM

Installation

Download the winintegrstp.zip file. It contains the executable, some *dll* files, *gino.con*, some test datasets, some *parameter files*, this manual as a *pdf* file and a *pdf* reprint of the paper by Angel (2003).

Unzip all of the *dll* files, *gino.con*, and the *pdf* file containing the manual into one folder. It is recommended that this folder is not used for data files. If you are upgrading from a version 3.4 or earlier of WinIntegrStp, you should use a new folder for v3.51 of the program. Earlier versions will not work with the new *dll* files, and v3.51 will not work with the old *dll* files.

Create a shortcut to executable by right-clicking on the file, dragging to the location for the shortcut (e.g. the desktop) and then selecting "create shortcut". The shortcut will be created with the WinIntegrStp icon.

Unzip the remainder of the files into a working directory. These include example datasets:
- R130901.dca: a ruby dataset collected with the Xcalibur diffractometer at Virginia Tech Crystallography Laboratory (VTX).
- R130901.rfl: the same ruby dataset in the WinIntegrStp/Single "standard" format.

and *instrument parameter files*:
- Xcalibur.par: Xcalibur diffractometer (Oxford Diffraction) with perpendicular monochromator, of the original KUMA-design.
- Xcalibur2.par: Xcalibur-2 diffractometer (Oxford Diffraction) with parallel monochromator (appropriate for Enhance optics).
- Xscans.par: P4 diffractometer with parallel monochromator (Siemens). If you are collecting data on a P4 whose monochromator has been removed, remove the mono card from this file.
- CAD4.par: CAD4 diffractometer with perpendicular monochromator (Enraf-Nonius).
- Huber.par: Diffractometer without monochromator.

Test the installation by working through the example described in detail below.

Input data files

WinIntegrStp reads the following step-scan data file formats produced by various single-crystal diffractometers:

Manufacturer	Machine/Software	File format
Oxford Diffraction	Xcalibur/Crysalis	Ascii (*.dca)
Enraf Nonius	CAD4	Ascii (*.dat)
Siemens/Bruker	Xscans	Ascii (*.p4t)
-	Old version Single	Ascii (*.dat)
Philips	Febo	Ascii (*.feb)
Stoe	Stadi	Ascii (*.d00)
WinInt standard format	Single-03,Single-04	Ascii (*.rfl)

Note: The program has not been fully tested for the Stoe Stadi datafiles, especially with respect to scaling the intensities for step time, attenuators etc, and should not be used to integrate Stoe diffractometer data at this time.

Other file formats could be incorporated into WinIntegrstp. Contact the author. When the reflection data file is opened the data in it is read, converted to Busing-Levy conventions using the information in the experiment file, and is written out to a "standard" format scratch file.

To save the input datafile in the standard data file format select **Files|Write Std datafile**.

Standard data file format

The standard format data file is an Ascii file. All lines of the file begin with an identifying code letter in the first column. All lines are written in *fixed format.* An example is provided in the distribution package. The file is divided into two parts, the *header* information, and the *step scan data*.

The *header* block must occur at the top of the file, and only at the top of the file. The information provided in the *header* is not used in the calculations by the WinIntegrStp program. The recognised types of line (as indicated by the first character) in the *header* are:

H can be used for any comment (H=header).
T title – WinIntegrStp displays this on the GUI.
U three lines for the UB matrix, in Busing-Levy conventions.
A the value for the attenuator factor (> 1). Not used in WinIntegrStp.

The *header* block must be terminated by the word **DATA** on a single line by itself.

Each reflection present as step scan data is represented by a single *reflection header line*, labelled **R**, followed by the intensity counts at each scan step, arranged 10 steps per line with each line labelled with an **S**.

The *reflection header line* is fixed format with variables and format as follows:

```
ISEQ, IH, ANG_PSI, ANGL, NSTEP, STEP_SIZE, SCALE, NSTD, SCANCODE
FORMAT('R',I6,3I4,5F10.4,I4,2F10.5,I4,A1)
```

The variables are:

ISEQ	Sequence number
IH	Three values h,k,l
ANG_PSI	Psi-rotation angle (often left as zero, as explicit position is given by ANGL)
ANGL	Four Eulerian angles $2\theta,\omega,\chi,\phi$, ω relative to bisecting
NSTEP	Number of steps in the scan
STEP_SIZE	Step size of ω in degrees
SCALE	Scale factor to multiply step counts to put them on a relative scale: see below
NSTD	0 if a data collection reflection, = standard number if a standard
SCANCODE	Type of reflection scan: see below.

The scancodes associated with each reflection indicate how the reflection was collected during a "constant precision mode" data collection. They do not affect the processing of the data except for "W" reflections which are ignored if the "skip weak reflection" option is selected on the GUI.

W	Weak reflection not rescanned after initial scan
M	Reflection rescanned at minimum speed
R	Reflection rescanned at intermediate speed
S	Strong reflection not rescanned
F	Strong reflection collected with the attenuator set.
I	Standard reflection

If this information is not present, or the data were not collected in constant precision mode, then the scancode 'N' can be used for data collection scans, and 'I' for standard reflections.

The dataset may contain reflections collected with different scan times, scan speeds, or with the attenuator set. These parameters are subsumed into the value of 'scale'. For true step scans, the value of scale should be set to:

$$scale = \frac{Atn\_factor \times step\_size}{time\_per\_step}$$

For continuous scans with intensity "dumped" every step, the expression becomes:

$$scale = Atn\_factor \times step\_size. \times speed$$

where *Atn_factor* = 1 when the attenuator is not set, and is >1 when the attenuator is set.

Note that WinIntegrStp works by fitting and displaying the counts as listed in the scan records and then multiplies the integrated intensity of each scan by the *scale* so as to place all output F^2 values on the same relative scale.

The *reflection header line* must be followed by int((nstep+9)/10) lines of *step scan data*. Each line of *step scan data* must start with an 'S'. All except the last line must contain 10 values of consecutive step counts in I8 format after the initial 'S'. The last line may be incomplete if nstep is not an exact multiple of 10 steps.

Instrument parameter files

WinIntegrStp requires information about the geometry of the diffractometer on which the step-scan data was collected, and a large number of default parameters for the peak-fitting process. The use of these parameters is described in detail in the "worked example" and "methods" sections in this manual. These parameters are stored in an *instrument parameter file*, with a default extension *par*.

The *instrument parameter file* is an Ascii text file that can be edited by any text editor, such as Notepad. The general *format* of the *instrument parameter file* is that the first six characters of each line are read as a label. The label defines the content of the rest of the line. If the first six characters of a line are blank, then the remainder of the line is ignored; blank labels can therefore be used to space out the information or to add comments (see the example files). All

numeric fields in labelled lines should be separated by commas. The information is read by Fortran read statements, so floating-point values should include a decimal point, and integer values must not contain a decimal point.

Details of the format of the *instrument parameter file* are given in the Appendix.

Output data files

WinIntegrstp produces an output file containing integrated intensity data in one of three different Ascii text formats that can be selected by the user with the **Parameters|Output format** menu. The program defaults to the format used the previous time the program was run, otherwise the *Rfine int* format.

- **RFINE "int" format**, suitable for reading into programs such as ABSORB (available from http://www.rossangel.com). The file contains one line per reflection with: hkl, setting angles, F^2, esd(F^2), reflection number, flags, and continuation counter set to zero. The flags are explained in Appendix 2.

- **RFINE extended format.** The first line is the same as the RFINE "int" format. The subsequent lines display all of the refined profile parameters and esd's, and various indicators of fit. Also readable by ABSORB.

  For both Rfine format files, the setting angles are converted from the angles in the input data file to Eulerian geometry (2θ, ω, χ, ϕ), with ω defined as the deviation from bisecting position. The value of ω is the value for the position of the α_1 peak maximum as obtained from the least squares fit of the profile.

  The sequence number for reflections is derived from the sequence number of the reflection in the input step-scan data file. This number is multiplied by 10 and then, if the reflection is a standard, the number of the standard is added. Thus a "normal" reflection with sequence number 201 in the original file will have a sequence number 2010. If the 203rd reflection was also the second standard, its sequence number would be 2032.

- **SHELX HKLF4** format, with hkl, F^2, esd(F^2), a group number set to 1, and the direction cosines of the incident and diffracted beams *with respect to the crystal axes, not Busing-Levy. Important note*: these direction cosines can only be calculated if the UB matrix has been calculated in WinIntegrStp, or read in with the input datafile. If the UB matrix is not available, then the program does not write the direction cosines. Note also that the SHELX file format does not distinguish between data collection reflections and standard reflections.

Notes for all of these data file formats: Intensities are scaled for instrument parameters such as scan time, attenuator etc as indicated in the raw data file. A Lorentz-polarisation factor is applied for an ideally imperfect crystal (Azaroff 1955), including the effect of the monochromator if one is indicated by a MONO card. No corrections are made for decay of intensities during the experiment. The *esd* of the intensity is derived directly from uncertainty in the intensity parameter refined in the least-squares fitting of the profile.

Log file

A log file is created automatically when you select an *input datafile.* Information about the *instrument parameter file*, and various intermediate results from the utilities within WinIntegrStp are written to this file. It can be read with any text editor.

Experiment file

The utility **Files|Create Exp File** will create an *experiment file* that includes the current UB matrix, unit-cell parameters, wavelengths, and diffractometer parameters that are needed for input to the Absorb program. If data have been integrated, a summary of the integration is also provided. The format of the *experiment file* is the same as the *instrument parameter file*. Further details are provided in the manual for the ABSORB program.

Cif

The utility **Files|Create Cif** will create a fragment of a *crystallographic information file* with a filename based on that of the *input datafile* with the string "*_winint.cif*" appended. If a UB matrix and unit-cell parameters have been refined the *cif* will contain that information in the relevant _cell and _diffrn_orient sections. Otherwise only information on the data integration method is provided.

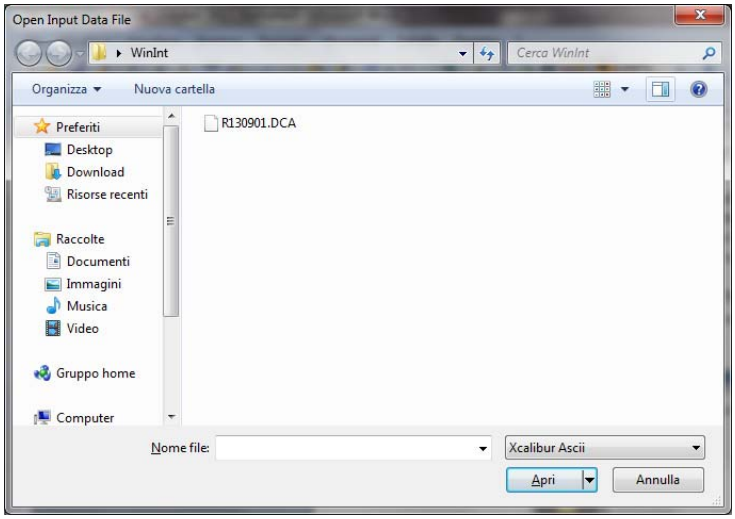
Export file

The **Export** button on the main window will write the current reflection scan into an Ascii data file suitable for importing into plotting and analysis programs. The file has some header information and three columns of data with one step-scan point per line. The columns correspond to the ω position, the observed intensity and the calculated intensity at the ω position.

4. USING WinIntegrStp – A WORKED EXAMPLE

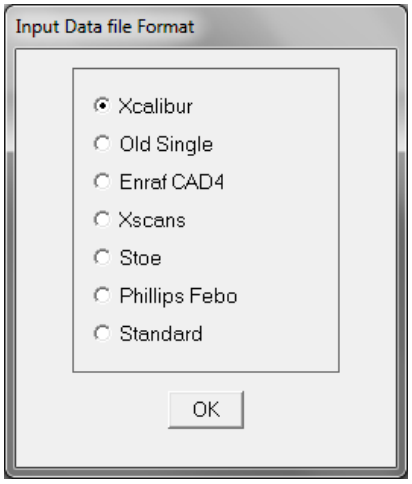
In this worked example, an intensity dataset collected on the Xcalibur instrument in the Virginia Tech Crystallography Lab is used to demonstrate how to examine a dataset and to prepare and execute an automatic integration. This exercise is long-winded and probably unnecessary for this high-quality dataset, but the example serves to illustrate the steps by which a poorer-quality dataset might be integrated.

Start the program by double-clicking on the WinIntegrStp shortcut. The program should start and display a file browser window, overlain on the WinIntegrStp main window, which looks something like this:



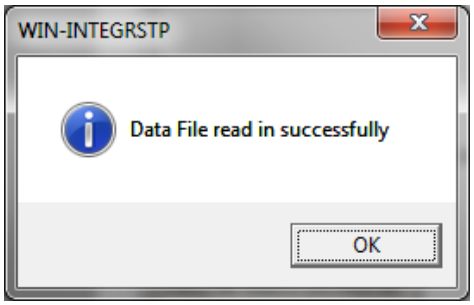
Use the file browser to navigate to the location of the test datasets and select the file *R130901.DCA.* Note that the "File Type" pull-down list controls which types of file are displayed by the browser *not* the actual file format which is specified in the next dialogue window. Here you need to set the *type* to *Xcalibur Ascii* for this example so as to display files with the *dca* extension. If you use a non-standard file extension for your datafiles, select *All formats* from this list to display all files. Select the file and click on the the Open button. The file format dialogue box now appears.

The program interprets the *dca* extension as being a file produced by the Xcalibur diffractometer. But a different file format can be chosen with this dialogue. *Important:* it is this dialog that tells the program the file format, *not* the list in the browser shown above.



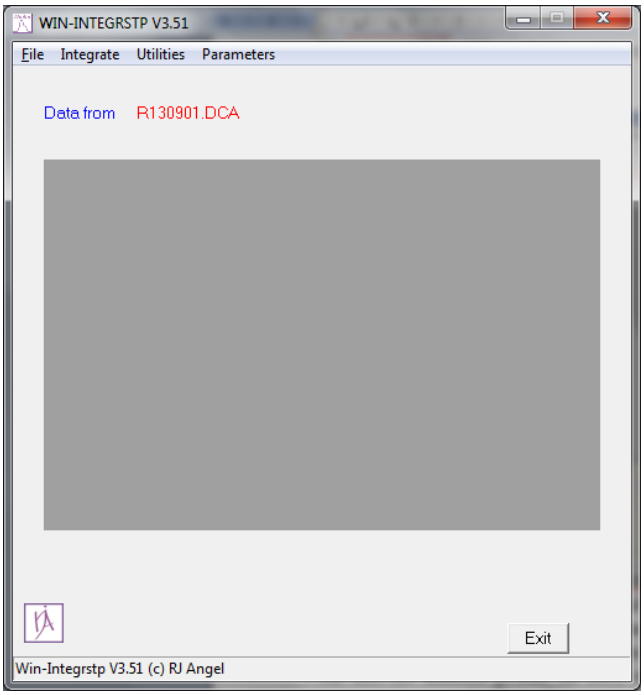
Click the **OK** button. Another file browser appears to allow you to select an *instrument parameter file*. Select the *Xcalibur.par* file provided in the distribution.

After clicking on the **open** button, the data file will be read in. You may see a "please wait" box during this process, followed by:



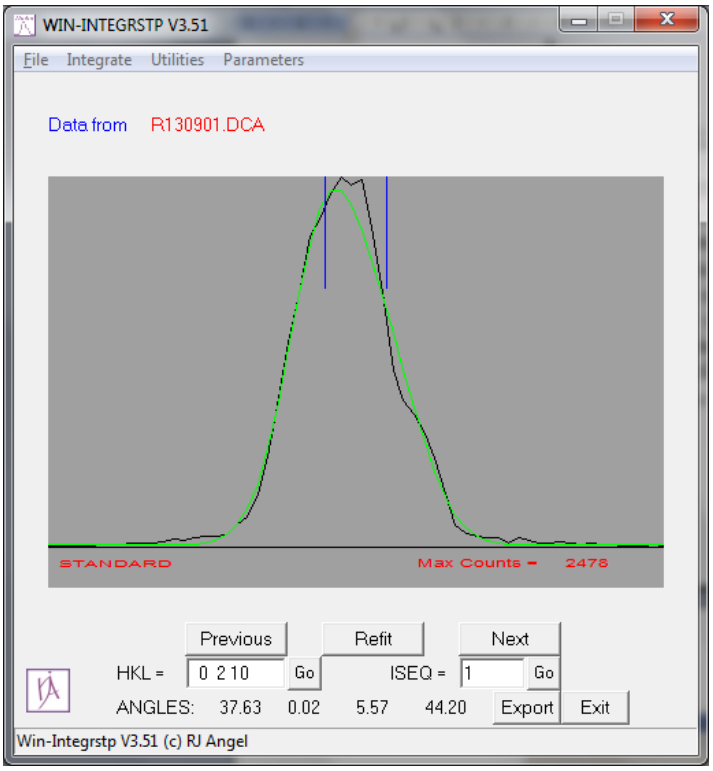
If you see an error box, then there is a problem with the format of your datafile. In some cases you will get a report that some data cannot be read in. This is often due to the step counts for strong reflections overflowing the format of the data file. These reflections are not stored or used by WinIntegrStp. Or maybe you selected the wrong file format?

When you click *OK* on the box shown above, you should now see the WinIntegrStp master window:



Before proceeding further select **Files|Open Log file** and create a new log file to contain extra output from the program, such as information on the input dataset and the results of the integration and the least-squares refinement of cell parameters.

The first step before integrating the data is to examine it. Select the **Integrate** entry on the menu bar and the **manual profile fit** entry on the pull-down menu. The manual-fitting part of the program is started. This allows you to examine the data and to experiment with profile parameters and test values. While fitting the data, there are two windows visible. The first is theWinIntegrStp master window that displays the scan (black line) and the fitted peak profile (green line) and the calculated positions of the α_1 and α_2 peaks:



The master window also provides controls to move through the dataset:

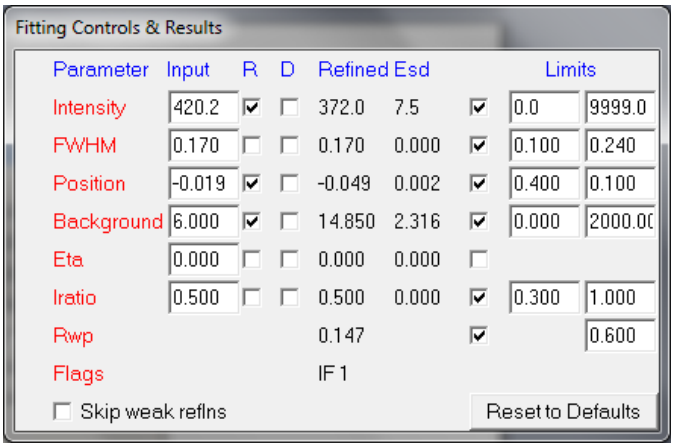
Previous moves back one reflection and fits it.
Refit allows you to refit the current reflection after changing the fit parameters.
Next moves forward to the next reflection, and fits it.
Export allows you to export the displayed reflection scan as an Ascii data file.
Exit exits from the manual fitting mode.

You can also move to a specific reflection by entering either the *hkl* or the sequence number, and clicking on the adjacent **Go** button or hitting a carriage return. Try it!

The angles displayed are the ones recorded in the input datafile as being the centre of the scan. They have the parities of the diffractometer. For data from kappa-geometry goniometers, the displayed angles have been converted from kappa to Eulerian angles.

If you go past the end of the dataset the program will exit from manual fitting mode. If this happens, just start it again by selecting **Integrate|Manual profile fit** from the menu bar.

In addition to the master window, a second window is displayed:



This window allows you to control the peak profile refinement by:

- setting parameter values (Input column)
- selecting which values should be refined (R column)
- selecting which parameters should be set to default values (D column)
- selecting which parameters should be tested against limits (Limits and adjacent tick boxes).

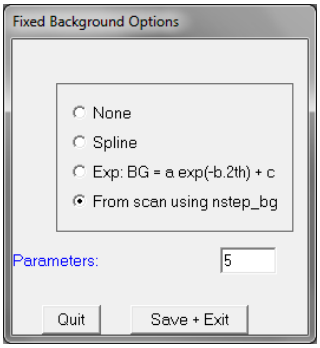
For a complete description of the program actions as controlled by this box, please read the section on Methods earlier in this manual, and section 2 of the Appendix.

In addition, the values of the refined parameters and their esd's are displayed, along with the R_{wp} of the fit and the flags indicating the results of the fits. Reset some of the "R" flags and explore the results. After changing the parameters you must select **Refit** on the master window to see the results. You should discover that some combinations of parameters cannot be refined simultaneously in a stable manner – for example, the intensity ratio of the α_2 component to the α_1 (I_{ratio}) and the Gaussian/Lorentzian mixing parameter (η). Therefore, in order to obtain the most reliable estimates of the integrated intensities, especially from weak reflections, it is desirable to determine the values of some of these parameters and to fix them during the final integration.

The intensity ratio of the α_1:α_2 doublet and the mixing parameter η can usually only be reliably determined by fitting well-resolved diffraction doublets that have been collected with long count times per step. These values are then stored in the *instrument parameter file*. Therefore reset the parameter values to the defaults supplied in the *instrument parameter file* by selecting the **Reset to Defaults** button on the Fitting Controls and Results window. This resets all of the parameters and flags back to the values in the *instrument parameter file.*

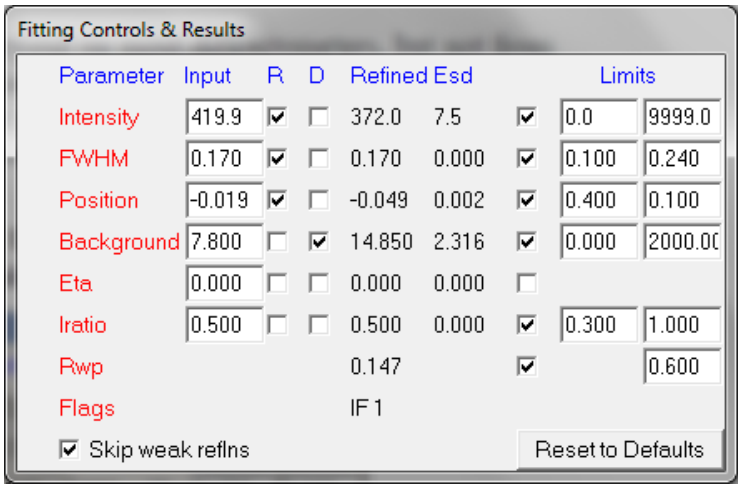
The peak backgrounds will vary from reflection to reflection. In some cases the variation with setting angle can be parameterised and used to fix the background for each reflection (Pavese and Artioli 1996). However, an even simpler method is available provided the peaks are always centred in the scan, and do not extend to the edges of the scan region. Then we can set the background to be fixed at the values at the end of each scan. To do this, **Exit** from the manual peak fitting. Select **Parameters** from the menu bar, and **Backgrounds** from the pull-down menu. The background dialogue box appears.

The default background option is set by the BACKGR card in the *experiment file.* If the last option **From scan** is not already selected, select it and enter **5** in the **Parameters** box, as shown. This means that the counts in the first and last 5 steps of each scan will be averaged and used to set the default background value for each scan. If the backgrounds are significantly different on the two sides, then only the lower one is used and an "uneven background" flag (U) is set. Select **Save + Exit**.



The peak widths will vary from crystal to crystal on most diffractometers, but not from reflection to reflection from one crystal, except for a possible variation with 2θ, unless twinning or anisotropic peak broadening are present. Therefore, the peak widths can be determined from the stronger reflections and then used as fixed values in the final integration.

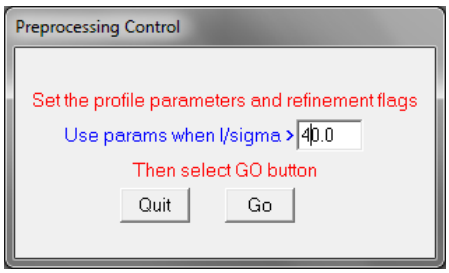
To obtain the peak widths, we first select **Utilities|Preprocess**. This brings up two windows. First ensure that the **Fitting controls** box looks like this:



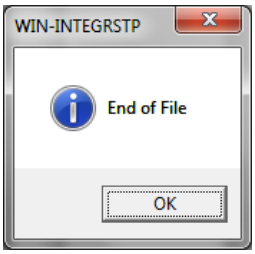
The numbers in some Input fields and the columns "Refined" and "esd" may be different. But ensure that the boxes in columns "R" and "D" are correct and that the "Limits" are the same as in the example. These settings mean that the intensity, peak width and position will be refined for each reflection, and the background will be fixed to that calculated from the end steps of each scan.

When the values and selections are correct, go to the **Preprocessing Control** window and set the **I/sigma** value to 40.0.

This means that only the parameters from fitting the strongest reflections with I/σ(I) > 40 in the dataset will be used in subsequent calculations of the peak widths and positions. Select the **Go** button and watch the program fit the peaks! The process can be interrupted and restarted with the **Stop** and **Start** buttons on the master window.

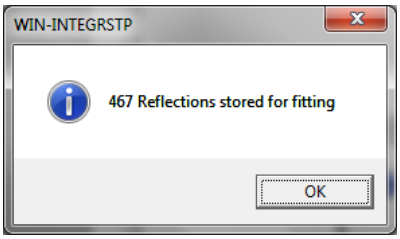


At the end, after 928 reflections, this little window appears: Select **OK**.



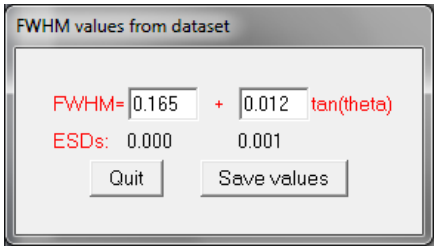
The last scan in the dataset disappears from the graphics window and this information appears:

When you select **OK** the parameters from the strong reflections are now stored.

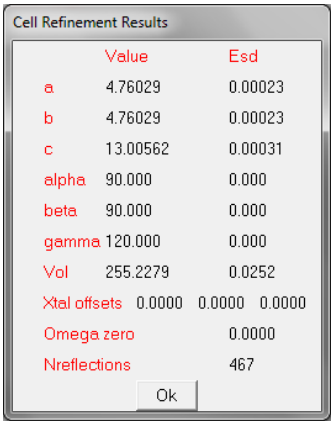


To calculate the average widths select **Utilities|Calc FWHM** and the results of fitting the FWHM of the strong reflections will be displayed:

Select **Save values** to store these values. You can check that the values have been correctly stored by selecting **Parameters|Pkvals** from the menu bar.

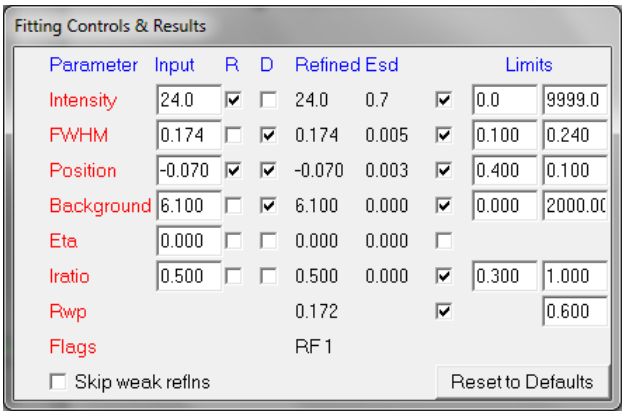


The positions of the strong reflections are also stored, and can be used to refine the unit-cell parameters. Try the various options available under **Utilities|Calc UB**. These include a symmetry-constrained least-squares procedure that, for this dataset, should provide the results shown here when the symmetry is constrained to be hexagonal.



An alternative procedure is to run the **Preprocess** utility again, with the FWHM set to the default values you have already determined in order to obtain more reliable peak positions, from which a UB matrix can be refined.

Now we have a model for the background, for the FWHM and for the positions of the reflections (a UB matrix) we are now ready to set up the automated integration procedure. First we shall set the controls and default values and check them by the manual fitting option that we used earlier. Select **Integrate|Manual Profile Fit** and set the **Fitting Controls** as shown here.



Note that:

- we are only refining two parameters, Intensity and position, for each reflection. This makes the integration procedure much more robust and reliable.
- the FWHM (peak width) is set to the **D**efault value (determined from the strong reflections) and is not refined.
- the position will be refined. If the integration fails on the first pass, the peak will be reintegrated with the position constrained to that calculated from the UB matrix.
- the background value is set to the **D**efault value (determined from the ends of each scan) and is not refined.
- The η and I_{ratio} parameters are fixed to pre-determined values from the *instrument parameter file.*

Use the buttons on the master window to look through some of the reflections. The idea is that you are seeing exactly what the automatic mode will do. Most reflections will have a flags displayed such as **SF1** or **NF1**; the **F1** means that the reflection was successfully fit (**F**) on the first (**1**) attempt (the **S** and **N** come from the diffractometer output). A full explanation of these codes is provided in Appendix 2.

Go to the reflection with sequence number 203 (*hkl* = -3 -4 1). It is clearly the wrong shape, with the α_2 peak more intense than the α_1 peak (presumably as the result of a double-diffraction event). See that the reflection flag is **MRR2**. This means that the reflection was a remeasured scan at minimum speed (**M**), but was rejected (first **R**) because the test of the intensity ratio (second **R**) of the doublet showed it was anomalous on the second (**2**) attempt. The second fit was attempted because the reflection was also rejected on the first pass and the **D** flag for position is set. The latter means that all reflections rejected on the first pass through fitting are refitted with constrained position in an attempt to recover the weak reflections. Reflections such as this one that are rejected both times will appear in the output file, but flagged as rejected. This example also illustrates the fact that the program will perform some tests even when the relevant peak parameter is not refined so as to provide a mechanism for rejection of aberrant scans such as this one. If you don't want this kind of peak shape test to be employed, untick the box next to the limits display for the line **Iratio**. Select **Refit** on the main window and see that this bad reflection is now passed by the program.

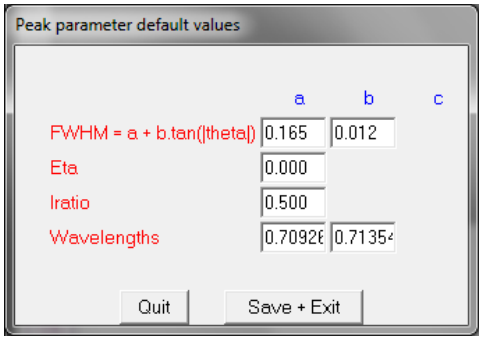
Reflection number 225 suffers from the same problem, and the offset of the observed and calculated profiles clearly illustrates that the peak position is not refined on the second integration pass.

The process of recovery is illustrated by reflection number 575. The flags are **MF2** which mean it was another reflection rescanned at minimum speed on the diffractometer, and was successfully fitted on the second pass with the position constrained to that calculated from the UB matrix. If you untick the **D** box for **Position** and then **Refit** the reflection you can see that the refinement of position and intensity on the first pass is not stable and gives a negative intensity (and wrong position).

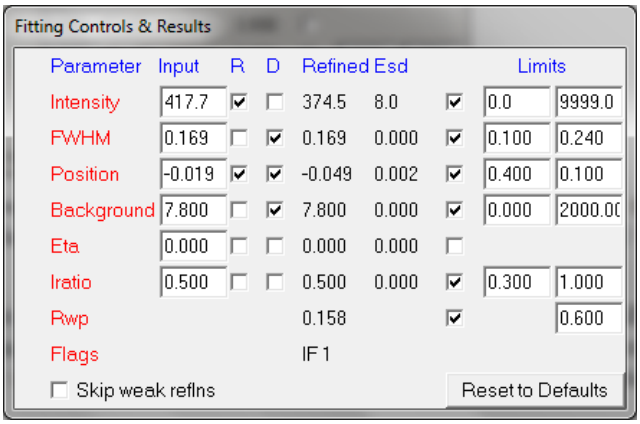
When you are ready to perform the automated integration, **Go** back to the first reflection, and then **Exit** from the manual program.

Select **File|Open output file** from the menu bar, and type in a new name (e.g. **example.int**) and **open** the file. If you use an existing file, you will be asked whether you want the new data to be appended to the end of the file, or whether you want the previous data to be over-written.

Now select **Integrate|Auto Profile Fit**. The Peak parameter dialogue box appears:



If the parameters are correct as shown, select **Save and Exit**, which will lead to the display of the **Fitting Controls and Results** dialogue box. The controls should look just as before:



Note that with the **Skip weak reflns** option selected, those weak reflections that the diffractometer did not collect on a second pass (in constant precision mode) will be ignored.

If you forgot to go back to the first reflection, type in the number "1" into the *ISEQ* window and click on the smaller **Go** button alongside of it. You can also choose to start the auto integration at any reflection by number or hkl by using these two navigation tools.

Once you are looking at reflection 1, start the automated integration with the **Start** button on the master window, watch the integration performed until the "End of File" message is displayed. Select **OK**.

Select **Files|Create Exp File** to create an *experiment file*. This includes details of the integration and some controls necessary for using the ABSORB program**.** Both this file and the *integrated data file* that you have created can be examined with a text editor. In the *experiment file* you will see that 13 reflections are listed as being "refitted", which means they were fitted on a second pass after failing a test on the first integration pass. Of these 13 reflections, 5 were recovered (including #575) and 8 were rejected (including #203 and #225) on the second pass.

Also in the *experiment file* you will see that the circle parities and blaxes cards indicate that the angles in the *integrated data file* have been set to Busing-Levy parities, and that the UB matrix in the *experiment file* has been defined with the Busing-Levy axial system.

Select **Files|Create cif.** This will create a *cif* with the name *R130901_winint.cif*. This contains the results of the *last* unit-cell refinement that you performed (whereas the log file will contain the results of all of the refinements).

APPENDICES

1. The Instrument Parameter File

The *instrument parameter file* is a text file that can be edited by any text editor, such as Notepad. The general *format* of the *instrument parameter file* is that the first six characters of each line are read as a label. The label defines the content of the rest of the line. If the first six characters of a line are blank, then the remainder of the line is ignored; blank labels can therefore be used to space out the information or to add comments (see the example files). The lines can appear in any order within the file.

The information is read by Fortran read statements but without format statements – so called "free format". Numerical values should be separated by commas or spaces.

There are two basic types of information that are provided in the *instrument parameter file*. First, there is basic information about the diffractometer, and these entries should not need to be changed once they are set up for your diffractometer. Second, there are default values that are used in fitting the peak profiles in WinIntegrStp and for testing the results. You may choose to change these values for certain crystals, or groups of crystals, although all of the values are settable via the user interface in WinIntegrStp.

The recognised labels and information are as follows, complete with examples:

WAVEL 2,0.709316,0.713606,0.50
Format: A6,I5,3F10.0

Variables:		
	nwave	number of wavelengths, 1 or 2
	wave(1)	wavelength in Angstrom of α_1 component
	wave(2)	wavelength in Angstrom of α_2 component (if nwave=2)
	wratio	Intensity ratio of α_2/α_1 (if nwave=2)

ATTEN 8.0
Format: A6,F8.0

Variable: atn_factor attenuation factor of detector attenuator (if present)

MONO 1,12.2,90.0,0
Format: A6,I5,2F10.0,I5

Variable:		
	lmono	=1 if monochromator present, 0 if not
	tth_mono	2θ value of the monochromator
	Dihed	dihedral angle, monochromator to crystal
	iparallel	for dihed =0, iparallel = +1 means dispersion of monochromator adds to the dispersion from the crystal at +2θ, iparallel = -1 means dispersion of monochromator adds to the dispersion from the crystal at -2θ. *Set as 0 if dihed≠0*

PARITY 1,1,-1,1
Format: A6,4I5

Parities for the four diffractometer circles in the order 2θ, ω, χ, ϕ as defined by Busing and Levy (1967). This example has the χ axis rotating in the opposite direction to that defined in by Busing and Levy (1967).

BLAXES -2,1,3
Format: A6,3I5

Defines the sense of the axes of the orthogonal coordinate system used by the diffractometer control software in calculating the UB matrix, relative to the positive axial system defined by Busing and Levy (1967). The three digits refer to the *x,y,z* axes of the diffractometer coordinate system. In the example:

- the –2 in the first position indicates that *+x(diffractometer) = -y(Busing-Levy)*
- the 1 in the second position indicates that *+y(diffractometer) = +x(Busing-Levy)*
- the 3 in the third position indicates that *+z(diffractometer) = +z(Busing-Levy)*

While WinIntegrStp works only in the Busing-Levy system, the information on this card is used to convert any UB matrix written in the datafile header to the Busing-Levy coordinate system.

GONCON 200.,200.
Format:2F10.0

Variables:	U0	Diffractometer size: Source to crystal distance in millimetres.
	V0	Crystal to detector distance in millimetres

These variables are only required if you need to refine the crystal offsets during least-squares refinement of the UB matrix to the peak positions. If no values given, 200mm are used as default values.

IMPORTANT: In previous versions this information was given by the RCRS card, with the order of the parameters reversed!

PKPAR 0.17,0.00,0.000
Format:3F10.0

Variables:	*a,b*	Default peak width parameters, $\Gamma = a + b\tan\theta$
	η	Default Lorentzian/Gaussian profile mixing parameter.

BACKGR 3,5
Format: depends on first item which is *iback*

If *iback* = 0 no fixed background is set. This is the same as omitting the BACKGR card. No other parameters.
If *iback* = 1 the background will be estimated from a spline function in 2θ. No other parameters.

If *iback* = 2 the format is I5, 3F10.0, with the parameters *a,b,c* for the expression $back = a.\exp(-b.2\vartheta) + c$.

If *iback* = 3 the format is 2I5 with the second variable defining the number of steps at the both ends of each scan used for setting the background.

RFLAGS 1,0,1,1,0,0
Format:6I5

Variables: Flags to indicate whether the peak parameters should be refined (1) or not (0). Given in the order: background, Γ, Intensity, position, η, I_{ratio}.

TFLAGS 1,1,1,1,0,1,1
Format:6I5

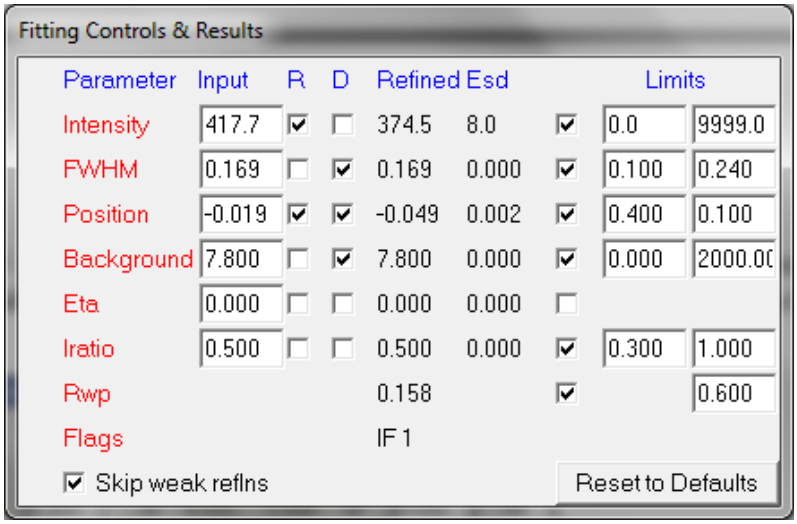
Variables: Flags to indicate whether the values of refined parameters should be tested against limits. The details of the tests are given in section 2 of this Appendix and in the section of the manual concerning peak discrimination and tests. The first six flags apply to the variables in the order background, Γ, Intensity, position, η, I_{ratio}. The seventh flag is the test on R_{wp}.

FITLIM 0.0,2000.,0.10,0.24,0.0,9999.,0.4,0.1,0.,1.,0.3,1.0,0.6
Format:13F10.0

Variables: Default test limits for the peak parameters given in min,max pairs in the order background, Γ, Intensity, position, η, I_{ratio}. The last value is the test limit for R_{wp}.

---

2. Fitting Controls and Results box



This dialogue box provides the controls for both the fitting of the profile function to each individual reflection scan by the method of least-squares, and the tests that are applied to the refined parameters. Note that not all of the controls that appear the same really work in the same way!

Input. This column displays the starting values of the parameters in the refinement of the current reflection. The starting values used for the *initial fit* of the reflection are set as follows:

- The starting value for the Intensity is estimated from the data scan itself.
- If the corresponding default flags are ticked the starting values for FWHM, η and I_{ratio} are calculated from the stored default values.
- The starting value for position is estimated from the scan, unless the position is not refined, in which case it is calculated from the UB matrix (if available).
- The starting value for the background is estimated from the end points of scan unless the default is selected, in which case the background is calculated according to the selected "fixed background" option (in **Parameters|Backgrounds**).

If the parameter is to be refined, then the exact input value is not critical, so long as the program can refine the peak profile function from the starting value. In manual fit mode the user can enter values into the **Input** fields and these will be used as starting values when the **Refit** button is pressed.

Refined. This column lists the refined parameter values. If input values or the controls for refinement or defaults are changed, the refined values are only updated after the **Refit** button is selected in the master window.

Esd. This column lists the estimated standard deviations of the refined parameter values, as estimated from the diagonal elements of the variance-covariance matrix obtained in the least-squares process. If χ^2_w of the fit is greater than unity, the elements of the variance-covariance matrix are multiplied by χ^2_w. If input values or the controls for refinement or defaults are changed, the displayed esd values are only updated after the **Refit** button is selected in the master window.

R and D columns. The basic principle is that if the R column box is ticked for a variable, it is refined. When the R column is not ticked, then the fixed value of the parameter depends on the state of the D column. The settings of the R and D columns interact differently for different variables. Default values for the R flags are taken from the RFLAGS line of the *instrument parameter file*.

- Intensity: only refined or not. No default value. If not refined it is fixed at the display value.
- FWHM, I_{ratio}, η: If R is ticked, the parameter is refined, otherwise it is fixed to the value displayed.
- Background. If D is ticked, then the initial value is calculated by the method selected in the **Parameters|Backgrounds** menu, otherwise it is set to the average of the first and last steps of the scan. If R is ticked, the parameter is refined, otherwise it is fixed to the value displayed.
- Position; the action depends on the state of both the R and the D boxes:

	R box ✓	R box ☐
D box ✓	Refine position, but if refinement fails tests, re-refine with position calc from UB	Fix position to that calculated from UB
D box ☐	Refine position	Fix to position given in Input value

Limits. These consist of a tick box and a pair of editable displays for each parameter, and a single display for R_{wp}.

If the tick box is selected for a given variable, that variable will be tested according to the protocol described below. Default settings are given by the TFLAGS line of the *instrument parameter file*.

If testing of a parameter is selected then it is tested against the limits set in the two edit boxes. The default values are given by the FITLIM line of the *instrument parameter file*.

The sequence in which the tests are applied is described in the section on Peak Discrimination earlier in this manual.

Variable	**Test**	**Conditions**	**Method**
Intensity	I > lower limit	R,T	Orig. refinement
FWHM	LL>Γ>UL	T, I>6 σ(I)	Re-refinement
Position	See note	R,T	Orig. refinement if I<3 σ(I), otherwise re-refinement.
Background	LL>*back*>UL	R,T	Orig. refinement
η	LL>η>UL	R,T, I>3 σ(I)	Orig. refinement
$\boldsymbol{I_{ratio}}$	LL>I_{ratio}>UL	T , I>3 σ(I)	Re-refinement

Notes: The limits on ω are expressed as the deviation from the centre of the scan. Since the refined ω position is that of the α_1 component, this usually lies at slightly lower ω than the centre of the scan, and the limits should be slightly asymmetric to reflect this. The program swaps the limits if the scan has been measured at negative 2θ values so as to preserve the functionality of the test.

The user should use the manual mode to determine the optimal limits for these tests prior to running an automatic integration.

In the table LL means lower limit, UL means upper limit. R means refine flag set, T means test flag set.

Flags. The letters and numbers on this line indicate both the way in which the reflection was scanned and the results of fitting and tests of the fit parameters, as follows:

First letter: The type of scan. See the description of the standard reflection file format.
Second letter: 'F' for fitted, 'R' for rejected.
If the second letter is 'R' the third letter indicates the reason for rejection:

- E Error in the least-squares procedure
- C No convergence in the least-squares procedure
- D Data error (e.g. not enough data points)
- B Background exceeds set limits
- I Intensity refined to less tan zero
- PRefined position is outside set limits
- W Refined width is outside set limits

- R Intensity ratio is outside of set limits
- F R_{wp} exceeds set limit
- Sη parameter is outside of set limits

The digit indicates on which pass through the fitting procedure the previous two flags were set, either 1 (first attempt) or 2 (second, constrained attempt).

The appearance of a "U" indicates that the background at the two ends of the scan were significantly different, i.e. uneven.

These same codes appear in the RFINE format *output data file*.

3. ZREF utility

This is an additional utility for those with diffractometer systems that can do, or can be tricked into doing, 8-position centering of reflections using the Hamilton method. The input data file must consist of sets of eight consecutive ω scans of a reflection in the eight equivalent positions on the diffractometer, in a specified order:

1	2θ	ω	χ	ϕ
2	-2θ	ω	χ	ϕ
3	-2θ	-ω	χ+180	ϕ
4	2θ	-ω	χ+180	ϕ
5	2θ	-ω	180-χ	ϕ+180
6	-2θ	-ω	180-χ	ϕ+180
7	-2θ	ω	-χ	ϕ+180
8	2θ	ω	-χ	ϕ+180

The first step is to use **Preprocess** to fit the reflections individually. Then the **Zref** utility will calculate the true angles and crystal offsets from each set of eight equivalent reflections, and put the true angles back into the peak list for use with the **Calc UB** utility. Full output from ZREF appears in the log file.

At VTX this facility is used and the delta-d/d plot (below) to help align the Huber Eulerian-cradle diffractometer and determine the optimal peak-profile parameters. After mechanical alignment of the diffractometer, a zref is performed on the diffractometer with the Single software. This software also stores the profiles from the final ω scan of each individual reflection. This **Zref** utility in WinIntegrStp is then used to determine the effect on the refined unit-cell parameters of using slightly different peak profile parameters **Iratio** and η. The set of parameters that gives the smallest esd in the refined unit-cell volume and the smallest spread in delta-d/d values are then chosen for use for measurements.

4. delta-d/d plot

This utility will plot the values of $\frac{d_{obs} - d_{calc}}{d_{obs}}$ against their 2θ values for all of the reflections in the current reflection list generated either from the **Preprocess** utility or the **Zref** utility. The value of d_{obs} for each reflection is calculated from the stored 2θ value, and d_{calc} is calculated

from its Miller indices and the current UB matrix. If there is no dependence of the fitted peak position with 2θ then the plot will have no significant slope with 2θ.

REFERENCES

Angel RJ (2003) Automated profile analysis for single-crystal diffraction data. Journal of Applied Crystallography 36:295-300.

Angel RJ, Downs RT, Finger LW (2000) High-pressure, high-temperature diffractometry. In: R T Downs (Ed) High-pressure and high-temperature crystal chemistry, MSA. pp 559-596.

Arndt UW, Willis BTM (1966) Single crystal diffractometry. Cambridge University Press

Azaroff LV (1955) Polarization correction for crystal-monochromatized X-radiation. Acta Crystallographica 8:701-704.

Blessing RH (1987) Data reduction and error analysis for accurate single crystal diffraction intensities. Crystallography Reviews 1:3-58.

Busing W, Levy H (1967) Angle calculations for 3- and 4- circle X-ray and neutron diffractometers. Journal of Applied Crystallography 22:457-464.

Dera P, Katrusiak A (1998) Towards general diffractometry. I. Normal-beam equatorial geometry. Acta Crystallographica A 54:563-660.

Dera P, Katrusiak A (1999) Diffractometric crystal centering. Journal of Applied Crystallography 32:510-515.

Lehmann MS, Larsen FK (1974) A method for location of the peaks in step-scan measured Bragg reflexions. Acta Crystallographica A30:580-584.

Pavese A, Artioli G (1996) Profile-fitting treatment of single-crystal diffraction data. Acta Crystallographica A52:890-897.

Ralph R, Finger LW (1982) A computer program for refinement of crystal orientation matrix and lattice constants from diffractometer data with lattice symmetry constraints. Journal of Applied Crystallography 15:537-539.

Schagen JD, Straver L, van Meurs F, Williams G (1988) CAD4 operators guide. Delft Instruments., Delft