D9: Hot Neutron Four-Circle Diffractometer

This is a guide to a standard single-crystal diffraction experiment on D9. It has been compiled as much as written, the principal sources being the D9 web page www.ill.fr/YellowBook/D9/EDIT94/D9.html and the manual in the big blue folder in the instrument area (although written in 1982, much it is still valid). Consequently, much of this document (particularly the computer program descriptions) has been plagiarised from the work of J.R. Allibon, M.S. Lehmann, G.J. McIntyre, A. Filhol, C. Wilkinson, M.-T. Fernandez-Diaz and others.

The computer programs described have evolved over the years under different operating systems, and consequently have many authors. To the best of my information, they can be attributed as follows:

`mad`	A. Barthelemy, A. Filhol, P.G. Rice, J.R. Allibon and C. Turfat in discussion with M.S. Lehmann, G.J. McIntyre, S.A. Mason, C. Zeyen and M. Thomas (as **`lsd`**); **`mad`** currently maintained for D9 by J.R. Allibon
`rafd9`	A. Filhol and M. Thomas, implemented for a PDP 11 by A. Barthelemy (as **`rafin`**); **`rafd9`** currently maintained by G.J. McIntyre
`index`	M.S. Lehmann and J.M. Savariault
`hklgen`	A. Filhol, S.A. Mason, A. Barthelemy and J.R. Allibon
`datap`	P. Coppens *et al.*, maintained by G.J. McIntyre
`racer`	C. Wilkinson and G.J. McIntyre
`fly`	C. Wilkinson and J.R. Allibon
`lamp`	D. Richard, M. Ferrand and G.J. Kearley

Things to do: write section on changing the wavelength plus several associated procedures (when we have a hot source again).

1 Characteristics of D9

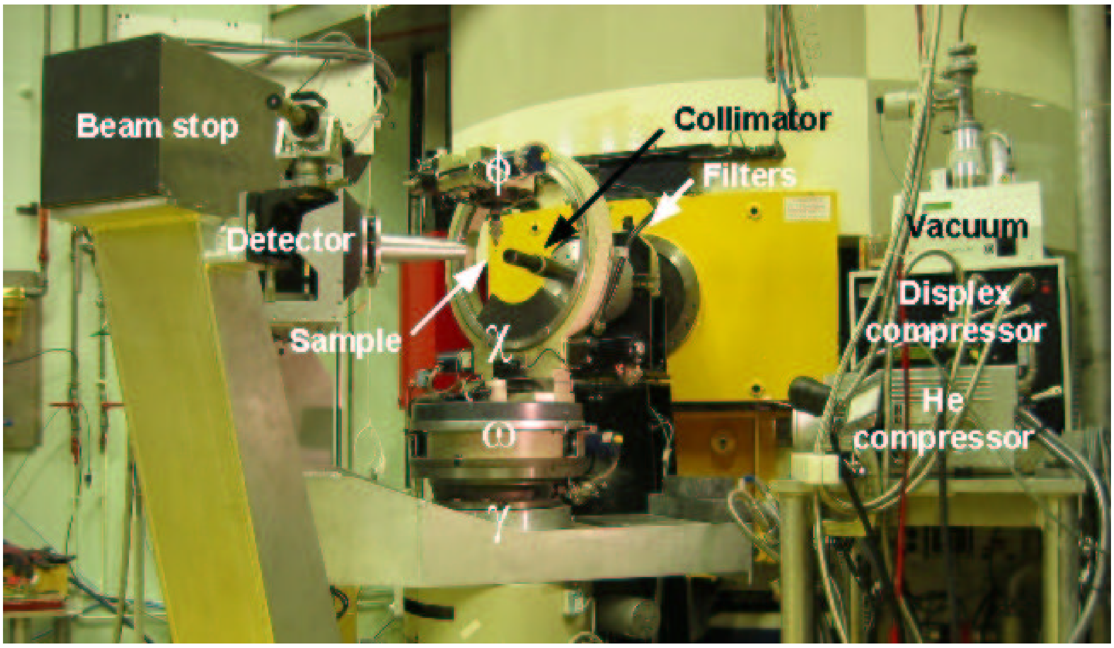
The diffractometer D9 is used for precise and accurate measurements of Bragg intensities up to very high momentum transfer. The resolution allows routine recording of extended data sets for the detailed study of atomic disorder and atomic thermal motions.

The wavelength of the neutrons is among the shortest available at any reactor in the world. The instrument is placed on a Tanzboden floor allowing a continuous choice of wavelengths in the range 0.35 - 0.85 Å.

In its standard ‘four-circle’ geometry the sample holder is an Eulerian cradle with offset χ-circle. A recent development also allows measurement in ‘normal beam’ (lifting-counter) geometry (*i.e.* measurement with cryomagnet, orange cryostat, *etc.*).

D9 is equipped with a small 2-D area detector which is particularly useful for rapid alignment and characterisation of samples and for studies of satellites and twinning. It covers 8 × 8 ° in 32 × 32 pixels. Programs are available for intensity integration using this detector.

The photo below shows D9 with a standard sample holder. Principal elements are labelled.



Measurement of one Bragg reflection normally takes a few minutes, and recording of a full set of data a few days. If the crystal is studied as a function of temperature, pressure *etc.*, the total measurement time is about a week.

The size of the beam at its exit from a movable ‘nose’ close to the sample can be adjusted by a system of interchangeable polyethylene plugs with boron carbide plastic ends. There is also a movable nose on the detector with a system of interchangeable boron carbide rings on the entrance.

The D9 monochromator is Cu(2 2 0) in transmission and is surrounded by a lead protection (γ-radiation shield). The main polychromatic beam enters this shielding and strikes the monochromator. Non-diffracted neutrons are stopped by a combination of

LiF absorber and polyethylene, placed after the monochromator. The take-off position with respect to the monochromator gives the different wavelengths.

Some key specifications of D9 are given below.

Monochromator	Cu(2 2 0) in transmission geometry
Accessible wavelengths	0.3 – 0.9 Å ($\lambda/2$ filters available)
Four-circle angle ranges	$5 < 2\theta < 120$ °, $-34 < \omega < 48$ °, $80 < \chi < 200$ °, $-179 < \phi < 179$ °
Lifting counter angle range	$-12.5 < \nu < 25$ °
Detector resolution	32 × 32 pixels
Detector pixel size	2×2 mm^2 or 0.25×0.25 °^{2}
Detector area	64 mm^2
Hot source – monochromator distance	10 m (partly in vacuum)
Monochromator – sample distance	2.29 m
Sample – detector distance	49 cm
Visible size of hot source	15 cm vertical, 5 cm horizontal

Further technical information about the geometry of the instrument (in relation to the integration of reflections from the 2-D detector) can be found in the attached document “RAFIN, RAFNB, RAFDB, RAFD19”.

2 Planning and preparation

Before arriving at ILL, complete and return all the forms sent to you by ILL concerning safety regulations.

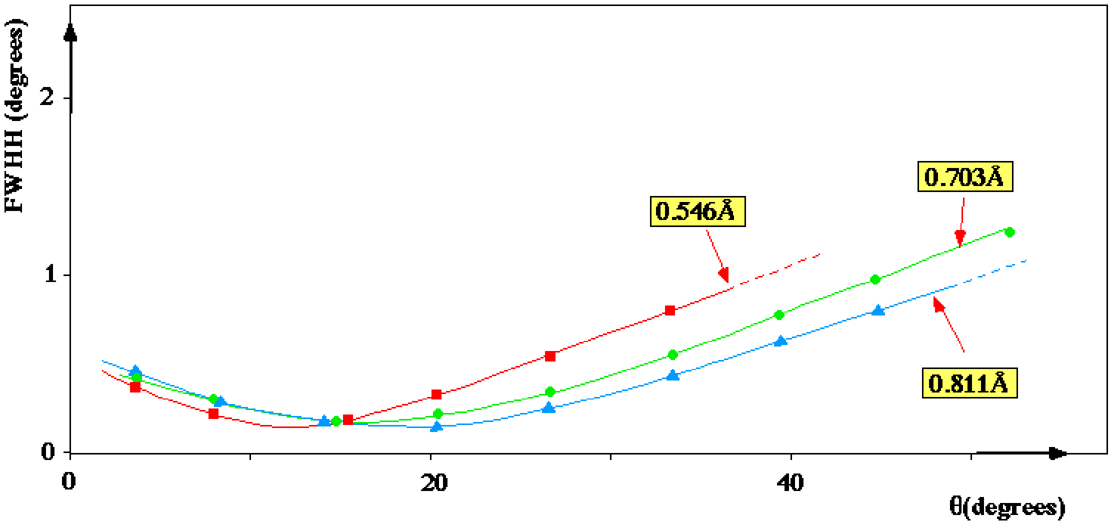
Make sure your local contact is up-to-date with your plans for instrument configuration, such as the sample environment and the wavelength you wish to use, so that as far as is possible, D9 can be prepared in advance.

2.1 Choosing the wavelength

Perhaps the single most important feature of D9 is its ability to use short-wavelength (high-energy, or 'hot') neutrons, and to vary the wavelength over a large range. Shorter wavelengths pack more *q*-space into an accessible angular range, allowing the measurement of very large numbers of reflections for detailed studies of atomic disorder and displacement parameters. Shorter wavelengths also minimise the neutron absorption problems associated with certain elements. Most D9 users have already thought about their desired wavelength in the context of these factors. There are, however, some further issues that also need to be considered when choosing a wavelength.

2.1.1 Resolution

There are certain disadvantages to conducting neutron diffraction measurements at short wavelengths. The intensity recorded by the detector is lower at shorter wavelengths, although the variation is not enormous (approximately 4 times the intensity at 0.8 Å compared to 0.5 Å). There is also a considerable sacrifice in resolution associated with short-wavelength measurements. The FWHM for a typical Bragg reflection as a function of θ is shown below for a number of different wavelengths (data from real crystals). Note that the resolution gets worse at shorter wavelengths as the monochromator take-off angle gets smaller. At 0.35 Å the resolution is relatively poor.



The choice of wavelength is a trade-off between resolution in reciprocal space (of individual reflections) and resolution in real space (accessible range of reciprocal space, especially important for displacement parameters). In order to make an informed choice of wavelength you therefore need to know the unit cell and Bravais lattice of your crystal.

2.1.2 *Scattering lengths and absorption*

Check the neutron scattering lengths and absorption cross-sections for the elemental isotopes contained in your crystal. Tables are available at www.ncnr.nist.gov/resources/n-lengths and in the International Tables for Crystallography. Isotopes with high coherent scattering lengths (*e.g.* ^{2}H) are desirable, contributing strongly to the reflection intensities. Those with high incoherent scattering lengths (*e.g.* ^{1}H) are undesirable, contributing strongly to the background. Those with high absorption cross-sections (*e.g.* ^{10}B) are particularly undesirable, reducing the intensities of all reflections (especially in large crystals) and affecting their relative intensities (especially with highly irregular crystals).

The most common isotopic substitution made as a result of the numbers in these tables is of deuterium (D or ^{2}H) for natural hydrogen (mostly ^{1}H), a substitution which can often be made relatively easily. For heavy metals, however, isotropic substitution is often too expensive and/or difficult and the better solution is to use shorter wavelength (‘hot’) neutrons to minimise the absorption problem. (Although the relationship between the absorption cross-section $\mu(\lambda)$ and wavelength λ is complex, it contains an approximately linear term whereby absorption decreases with wavelength.)

2.1.3 *Activation*

The extent to which the elements in your crystal become radioactive as a result of exposure to the neutron beam will affect your ability to take the crystal home immediately after the experiment. A table is available at www.ill.fr/YellowBook/D19/help/act_table.htm.

2.2 Choosing environmental apparatus

Be aware of limitations and safety implications of any environmental apparatus you wish to use. Make sure your local contact is up-to-date with your plans, so they can ensure the equipment will be available and any necessary preparations made. Be sure your crystal can cope with the intended conditions without melting, undergoing a destructive phase transition *etc*. For details of available environmental apparatus see www.ill.fr/YellowBook/D9/EDIT94/D9.html.

2.3 Choosing crystals

Bring several if at all possible. Even if you are positive that the crystal is perfect, there is always a possibility that it could get lost or damaged.

2.3.1 *Size*

Generally, crystals for D9 are in the range 5-25 mm^3. The ideal size is a balance among scattering power, absorption and extinction, however, in practice the crystal size used for an experiment depends largely on the results of crystal growth. Nonetheless, it can generally be said that ~ 1 mm^3 is the minimum size for which data collection can be carried out within a reasonable time, and ~ 50 mm^3 is the maximum due to absorption, extinction, saturation of the detector and the maximum size of the homogeneous beam (6 × 6 mm). Plans to use crystals outside this range should be discussed with your local contact before arrival.

2.3.2 Shape

There are a number of factors to consider when choosing crystals from a batch. If the compound is strongly absorbing, more isotropic (spherical) crystals will minimise the problem. In equatorial geometry, needle-shaped crystals mounted parallel to the pin will also minimise absorption. On the other hand, if the absorption is not a big issue, anisotropic crystals with well-defined faces are easier to index and orient in the beam. Well-defined faces are of course also a good sign of crystallinity, albeit occasionally misleading. A good way to judge a crystal is to put it on a single-crystal x-ray diffractometer; a single transmission Laue photograph also can provide a lot of information about crystal quality. Polarising optical microscopes can help judge the quality of transparent crystals.

2.4 Mounting crystals

The appropriate mounting method depends on the sample environment, particularly in the case of non-ambient temperatures. The pin (vanadium or aluminium) onto which the crystal will be directly fixed depends on the environmental apparatus that will be used. These pins are available from local contacts or instrument scientists as needed (being much more expensive than they appear, they tend to be used wastefully when left at the instrument in significant quantities). Consider the orientation of the crystal before gluing (see Section 3.3.2).

2.4.1 Choice of glue

Two principal types of glue are used for mounting crystals on pins for D9. Kwikfill, a two-part polymer glue, is the better choice at room and low temperature due to its flexibility, but cannot be used above ~200 ˚C. The ideal ratio of the green part in the tin to the yellow part in the tube is 40:1. For high-temperature experiments, Ceramabond, a single-compound ceramic glue, is normally used. Ceramabond is expensive and very air-sensitive, therefore you should seal the lid and wrap it with electrical tape immediately after use.

2.4.2 Mounting on the pin

Large crystals with well-defined faces can be easily glued in the desired orientation by hand, and left to dry sitting freely on the pin. For more difficult cases, there is a device on the D9 sample preparation table that allows the crystal to be held in place temporarily with plasticine or wax, and the pin lined up opposite on a goniometer head with the aid of a microscope. The pin then has glue applied to it and is translated into place on a thread, which holds it steady until the glue dries. If you have pre-mounted your crystal in a desired orientation on a goniometer head, that head can be placed in the apparatus in order to transfer the crystal directly to the D9 pin with a 180 ˚ rotation. If you pre-mount your crystal in this way, try to use an easily removable bond such as plasticine.

2.4.3 Sensitive crystals

Delicate crystals to which you do not wish to apply glue directly, or those which are sensitive to the solvents in the glue itself, can be wrapped in aluminium foil prior to gluing. Remember to measure the shape of the crystal first and to give the foil wrapping a shape that allows you to preserve its orientation.

Crystals which are sensitive to air or to vacuum, or which need to be kept moist, *etc.*, can be placed in quartz tubes. Quartz wool is used to keep such crystals oriented and in place away from the ends of the tube.

2.5 Documenting crystals

2.5.1 Crystallographic properties

Know as much about your crystals as possible before coming to ILL. You should definitely know the unit cell of the compound, from powder x-ray diffraction if nothing else. Ideally you should use single-crystal x-ray diffraction to assess the quality (to save time choosing a crystal) and index the faces (to save time aligning it). From the unit cell you should calculate a list of 2θ values of reflections for the wavelength you intend to use on D9, at ambient temperature and pressure (for orienting the crystal) as well as under the environmental conditions you intend to use. If you have a model for the structure you should calculate their neutron diffraction intensities in order to choose appropriate orienting reflections (remember that there can be enormous differences in relative intensities between neutron and x-ray diffraction). Whether or not you know enough about the structure to compile a complete list of expected reflection intensities, you should at least have an idea of the minimum number of reflections required for the problem you which to solve (minimum *d*-spacing and number of equivalents).

2.5.2 Crystalline faces

If your crystals have distinct faces, you should try to index these in terms of the unit cell directions, in order to facilitate the orientation of the crystals in the beam. Sometimes the identities of the faces are obvious (*e.g.* plate-like crystals where one axis of the unit cell is much longer than the others). If not, they can be determined by orienting the crystals on a lab single-crystal x-ray diffractometer, or by single-crystal x-ray photographic methods (Laue, Weissenberg or precession).

3 Setting up

3.1 Starting an experiment log file

One of the first things you should do is start a log of your experiment. A convenient way to do this is on the control computer, where you can leave the log file open on the desktop at all times and copy/paste important control parameters and output associated with the experiment, adding comments and notes about the crystal *etc*.. (See Section 0 if you are not sure how to do this on a Unix system.) Note that the control program **`mad`** creates daily log files in /users/d9/logfiles/mad*yymmdd*.log.

3.2 Setting the wavelength

Get your local contact to do (or arrange) this for you, if it wasn't possible before the experiment started (see Section 7.2). Don't forget to ensure that any necessary $\lambda/2$ filter is present for the wavelength being used.

3.3 Putting the crystal on the 4-circle

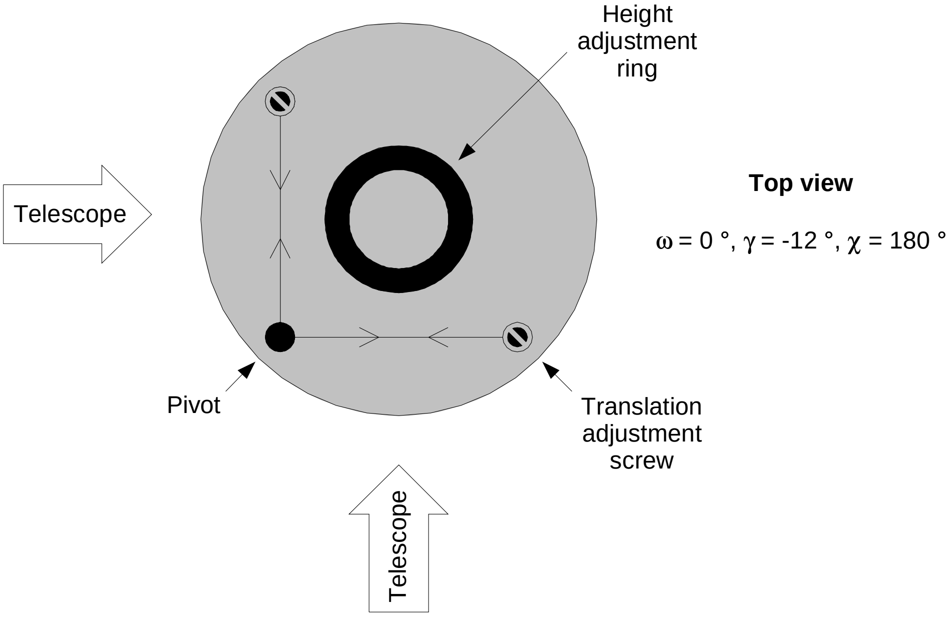
Attach the goniometer head with the crystal onto whatever environmental apparatus you are using. You generally want to start at $\gamma = 45$, $\omega = 0$, $\chi = 180$ and $\phi = 0$ ° [**`mad`**: `ppp 45 0 180 0`]. Note that γ is what would normally be labelled 2θ; the label γ is a reminder that, because the detector is 2-dimensional, γ and 2θ are only exactly the same at the centre of the detector. The label 2θ is reserved to denote twice the Bragg angle.

3.3.1 Centring in the beam with the Displex or furnace

Centre the crystal on the cross-hairs of the telescope: firstly with respect to the height, by turning the support ring; and secondly with respect to translation, by turning the two adjustment screws (get the local contact to identify these elements for you first, before risking misaligning other aspects of the instrument!) (see figure below). Only adjust the translation after moving ϕ such that the translation directions are perpendicular to the telescope [**`mad`**: `mph` *`angle`*]. The telescope is easily misaligned with respect to translation, therefore it is necessary to check the validity of the centring by moving ϕ back and forth by 180 ° and making sure that it is centred in both senses.

3.3.2 Orienting with respect to the beam

When orienting the crystal in the beam, bear in mind the conditions for reflection in four-circle geometry. A reflection of a particular index can be detected when the normal to the lattice planes (*i.e.* the normal to the crystal face of the same index) lies *parallel* to the detector plane (γ-ω plane, normally horizontal) and *perpendicular* to the beam (at $\gamma = \omega = 0$, *i.e.* perpendicular to the telescope at $\omega = 0$ and $\gamma = -12$ °). Thus, you need to know the indices of 2 crystal faces, in order to know immediately where to find all your desired orienting reflections (more or less easily depending on the faces and the symmetry). If you only know one or two faces, however, you can still optimise the search process because you can identify *planes* in which particular reflections should lie. The trick is to align the crystal so that a plane with one or more strong, generally low-order *i.e.* unambiguous [*e.g.* (1 0 0), (1 1 0)], reflections coincides with the equatorial (detector) plane at $\chi = 180$ °. This setting of χ puts the ϕ axis perpendicular to the equatorial plane. Hence, any reflections which are brought into diffracting position as ϕ is rotated will be scattered in the detector plane; you only have to know the 2θ values of the reflections expected, set $\gamma = 2\theta$ and $\omega = \theta$ and scan ϕ until they show up. The simplest case is when you know the index of one principal face [*e.g.* (1 0 0)] of an orthorhombic (or higher symmetry) crystal. By placing that face *parallel* to the detector plane at $\chi = 180$ °, principal reflections of the 2 other axes [(0 k 0) and (0 0 l)] will be found simply by scanning ϕ at appropriate 2θ settings of γ and ω. The (h 0 0) reflections themselves will be found at $\chi = 90$ °.



3.4 Opening / closing the beam shutter

Once the crystal is centred and you are ready to look for reflections, close the gate and lock it with the upper key, allowing you to remove the bottom key, which you place and turn in the shutter control box. Then press the black “open” button and hold it until the beam comes on (red light). To close the shutter, turn the key to the “close” position until the beam goes off (green light), allowing you to remove it from the control box and reverse the procedure to open the gate.

3.5 Setting up the computers

The instrument is controlled by a Silicon Graphics (SGI) computer with a Unix operating system. Most applications run interactively in X-windows but it is nonetheless useful to have a basic familiarity with Unix (getting around the directory structure, running applications and editing text files). Some of the most basic Unix commands are summarised in Section 0. Commonly used text editors are installed such as **`nedit`**, **`jot`** *etc*. Log on (if necessary) as `d9` and create your own working directory in `/users/d9`, *e.g.* `/users/d9/`*`yourname`*. You should run all the programs described below out of this directory because many of them depend on experiment-specific input files. You should copy into this directory some ‘template’ files to facilitate getting started with some other programs you will be using; these can be found in `/users/d9/ling/templates`.

You should also create a log file of your experiment, *e.g.* *`experiment`*`.log`. You should leave this log file open on the desktop at all times and copy/paste important control parameters and output associated with the experiment, adding comments and notes about the crystal *etc*.. Note that the control program **`mad`** also creates automatic log files in /users/d9/logfiles/madyymmdd.log.

A second SGI computer physically located at D9 is called `d9sgi.ill.fr`. When you log onto this computer as `d9` you will be directed to your own sub-area (which you will be invited to create the first time you log on). This computer should be used for CPU intensive data visualisation (particularly **`lamp`**), reduction of complete (*i.e.* big) data sets with **`racer`**, and activities not directly connected to the running of the instrument (particularly e-mailing and web-surfing).

The relationships among the various computer programs you will need to run on these computers (as described in this Section) are illustrated diagrammatically at the beginning of Section 8.

3.5.1 Instrument control

The principal instrument control program is **`mad`**, a command summary for which can be found in Section 8.5. It is used to move the diffractometer circles and collect data, both manually and automatically. It can be given instructions to move in terms of angles and in terms of a reciprocal unit cell, which is related to these angles by the UB matrix. Data from the 2-D detector can be recorded either as a function of time or of the number of counts recorded by the direct beam monitor. Commands can also be issued from **`mad`** to set scan parameters, sample temperature *etc*..

An important helper program associated with **`mad`** is **`parame`**, which should be run in a separate xterm to **`mad`**. It has an interactive menu and can be used to set all the parameters for scans, environment, UB matrix, diffractometer constants and zeros *etc*. Before starting, use **`parame`** to personalise general parameters for your experiment such as user name, compound, local contact and unit cell.

Note: the instrument can also be moved manually, however, you should not attempt to do this until your local contact or an instrument scientist/technician has shown you how, and made you aware you of the dangers. If one or more of the motors will not move under computer control, it may be because these motors are set in manual mode.

3.5.2 Experiment monitoring

There are a number of other X-applications that it helps to have running on the desktop. They will probably all be there to start with, but if not you can launch them yourself. They include: **`minspy`** (motor positions); **`dti liveplot`** (plot of temperature *versus* time); **`dti sh dyn`** (monitors temperature controller); **`msp`** (monitors data collection); **`viz`** (live plot of scan); **`x3d`** (live image of 2-D detector); **`xdet`** and **`ydet`** (live cross-sections of 2-D detector); and **`pcp`** (“stop” button for diffractometer motors). You should launch them from a Unix shell by typing their names followed by the Unix suffix “`&`” [*e.g.* **`viz`** `&`], which allows you to keep working in that shell.

3.6 Finding and scanning orientation reflections

Place an aperture on the nose of the beam guide just large enough that the whole crystal is in the beam at all angles, but not significantly larger (or the background will just increase). Ensure that any necessary $\lambda/2$ filter is present for the wavelength being used (see Section 7.2). Also, before starting, be aware that if the crystal is big and diffracts well, it may saturate or even damage the detector. If the detector records anything approaching 10000 counts/second, or it looks like that's about to happen, move the detector away from the reflection in question and/or close the shutter. If you have no choice but to use a crystal with exceedingly strong reflections, you can talk to your local contact about ways of attenuating the direct beam, but this is obviously not an ideal situation. Aside from the possibility of damaging the detector, its response begins to become non-linear above about 5000 counts/second, compromising data quality (although the integration program **`racer`** includes a deadtime correction if necessary).

Using your list of strong reflections at ambient temperature and pressure, and after giving **`mad`** (through **`parame`**) the unit cell and the wavelength being used, set $\gamma = 2\theta$ and $\omega = \theta$ [**`mad`**: `hkl1 h k l`] for the first reflection for which you wish to search. The exact search strategy depends on how much you know about the crystal (see Section 3.3.2), but it is usually based on running long ϕ scans at fixed χ values, starting at $\chi = 180$ ° [*e.g.* from $\phi = 0$ °, **`mad`**: `sph = 180 0.2 1000 0 0`]. (In practice, these ϕ-scans are usually carried out by moving the motor manually, however, you should not attempt this until you have been shown how by your local contact or an instrument scientist/technician.) When something shows up, you can stop the scan early with the "stop" button on the **`pcp`** tool. If nothing turns up, change χ in ~ 10 – 15 ° steps and try again, or ultimately try scanning slower.

Go back to the ϕ angle of the peak [**`mad`**: `pph angle`] and see where it is on the detector [*e.g.* **`mad`**: `acq 10000 1 0`]. Centre it on the 2-D detector by adjusting γ (horizontal) and χ (vertical) [**`mad`**: `mga angle, mch angle`] within the 8 × 8 ° range of the detector. Do some test scans [*e.g.* **`mad`**: `som = 3 0.1 10000 0 0`] to determine the appropriate scan type (see Section 4.1), width, step, and speed (remember that after the scan, the motor being scanned does not reset itself to the starting positions; you have to do it yourself). These scans will also give you a first sense of the crystal quality. Then scan them and keep the scans [*e.g.* **`mad`**: `som = 3 0.1 10000 0 1`], making a note of the number assigned to each scan.

3.7 Indexing reflections

If you find some reflections serendipitously, you will need to determine their indices in order to use them in the UB matrix determination. The program **`index`** can be helpful if the indices are not obvious. It requires the unit cell of your crystal (which you should know) and the wavelength, and tries to identify pairs of reflections based on their 2θ values and the angles between them. Modify the parameters in the input file `index.dat` appropriately as described in Section 8.3.

3.8 Determining the UB matrix

Once you have scanned a minimum number of unique reflections required to orient the cell (*e.g.* 2 for orthorhombic) for which you think you can assign indices, you can find the UB matrix. Note that this works best when the reflections are well separated; by at least 45 ° in ϕ and with at least one at $\chi > 135$ °).

In a separate X-terminal, in your working directory, copy and rename the file *`exp.`*`usr`. The last numbers in this file are the run numbers. Change them to the run numbers for your scanned reflections. Each row contains an inclusive sequence of runs, *e.g.* to use reflections 012345 and 012347-012349 enter the lines:

```
012345
012347 012349
```

Be careful not to leave a space before the first number in a row. The line above the run numbers tells the program where to find the data, which is grouped by instrument and reactor cycle, *e.g.* `D9 014` is D9 data from year 2001 cycle 4. Change these numbers to appropriate values.

Run the program **`racer`** with *`experiment`*`(.usr)` as the input file (**`racer`** can normally be used as a black box at this stage; however, if you have difficult peak shapes, you may need to consult the attached documentation). The output is *`exp.`*`raf`, which contains the integrated reflections indexed according to an old UB matrix, *i.e.* incorrectly in the context of the new experiment. Edit that file to give the reflections what you believe to be the correct indices. (The program **`racer`** is described in an attached document.)

Edit the file `rafd9.dat` to give it the correct title, unit cell, wavelength and `.raf` filename. Run the program **`rafd9`** choosing the default input (`rafd9.dat`) and file output, then look through the output file `rafd9.lpt` to see whether it worked or not. (A description of the file `rafd9.dat` can be found in Section 8.f. The geometrical considerations behind **`rafin`**, from which **`rafd9`** is descended, are described in an attached document). If it seems to have worked, put the new UB matrix (contained in the output file `ubfrom.raf`) into **`mad`** [**`mad`**: `getub`] and try to find some *different* reflections using this UB matrix [*e.g.* **`mad`**: `hkl0` *`h k l`*; `acq 10000 1 0`]. If it doesn't work, try different indices and/or reflections.

3.9 Refining the UB matrix and unit cell

Once you have this rough orienting UB matrix, you can use it to collect automatically a set of reflections with which to refine further the UB matrix. Create a text file *`exp.`*`hkl` containing a list of about 25 strong unique reflections, at moderate to high 2θ ($\theta > 20$ ° if possible) and as varied as possible in h, k and l (and hence in χ and ϕ). Set the scan parameters for automatic data collection to the appropriate values with **`parame`**, then collect the reflections [**`mad`**: `mes` *`exp.`*`hkl`].

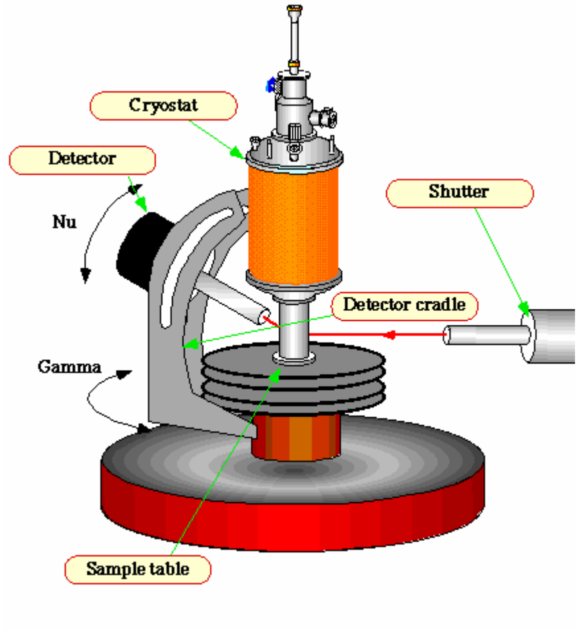
Edit *`exp.`*`usr` to give it the scan numbers for these reflections, re-run **`racer`**, then edit *`exp.`*`raf` so that it only contains these reflections. Re-run **`rafd9`**, after editing `rafd9.dat` to switch on the refinement flags (change the `0`'s to `1`'s) for the cell and for the vertical offset (see the file description at the end of this document). Ideally you want a "mean abs(DEV.)" < ~ 0.02 ° in the output file `rafd9.lpt`. Put the new UB matrix and cell into `mad` [`mad`: `getub`]. (Note: if you still have problems getting a decent refinement of the UB matrix, and you are sure the problem is not with your crystal, it may be necessary to refine and re-set the ω- and χ-zeros with the help of your local contact; see Section 7.3.)

3.10 Installing environmental apparatus

Before obscuring the crystal with environmental apparatus, ensure you know its orientation. Make a sketch of the crystal at $\chi = 180$ °, $\phi = \omega = \gamma = 0$ °. If possible, define the crystal shape completely in terms of faces and distances from the centre, in order that an absorption correction can be carried out posthumously in the event that the crystal is destroyed or lost (due to *e.g.* glue failure, sublimation, destructive phase transition) in the final stages of data collection. The `mad` command `inv` (see Section 8.5) is useful when indexing faces. Get your local contact to install (or arrange installation of) the environmental apparatus. As you change the environment (heat, cool *etc*.), it is a good idea to do repeated scans of a strong reflection so you can see reflections move and know if and when something dramatic happens [*e.g.* `mad`: `mes monitor.hkl`, where `monitor.hkl` contains the same reflection many times over].

3.11 Normal-beam geometry

D9 can now also operate in ‘normal-beam geometry’ using the lifting counter, which has a range $-12.5 < \nu < 25$ ° (although in practice this is limited by the diameter of the sample-environment. This is not a sufficiently standard procedure on D9 to warrant an instruction manual; if you will be working in normal-beam mode, you will have special help from the instrument scientists and technician. The geometry is represented below with an orange cryostat as the sample-environment.



4 Collecting data

If you have installed environmental apparatus, remember to re-check the crystal quality and re-refine the UB matrix under the data-collection conditions before starting. Move the nose of the beam guide as close as possible to the crystal. Check that you are filtering out any λ/2 neutrons effectively by looking for (*h*/2 *k*/2 *l*/2) where (*h k l*) is a strong reflection [*e.g.* **`mad`**: `hkl1 h/2 k/2 l/2`, `acq 100000 1 0`]. Check that the machine software limits [**`parame`** or **`mad`**: `par but`] lie within the hardware limits, and that the motors can move to these limits without straining any cables or hitting anything.

You should make an empirical check on the impact of absorption and extinction by measuring integrated intensities for both a strong and a weaker high-χ (~ 90 °) reflection in various orientations while turning around the scattering vector [**`mad`**: `ren0 h k l psimin psimax psistep`]. You should also check for multiple scattering by rotating around the scattering vector [**`mad`**: `psi0 h k l psimin psimax psistep`]. Note that before performing either of these scans, you should check that the motors will be able to reach the required angles [**`mad`**: `cal6 h k l psimin psimax psistep`].

4.1 Setting scan parameters

Before you begin data collection it is very important to spend some time choosing appropriate scan parameters. Mistakes made here can compromise your data and cannot be corrected retrospectively! These parameters are most easily set in the `scan` menu of **`parame`**.

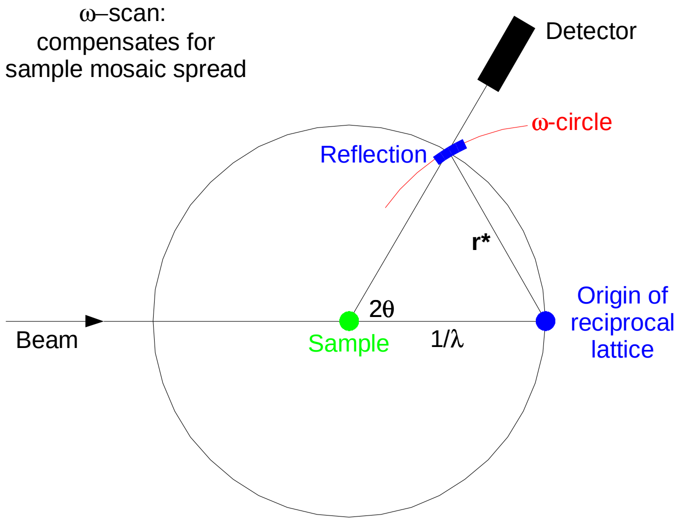
Note that in difficult cases it may be necessary to choose different parameters for different parts of reciprocal space, particularly different 2θ annuli. This can be done by including `par` commands in the `experiment.xbu` command file - see Section 4.3.2 below.

4.1.1 SCA parameters

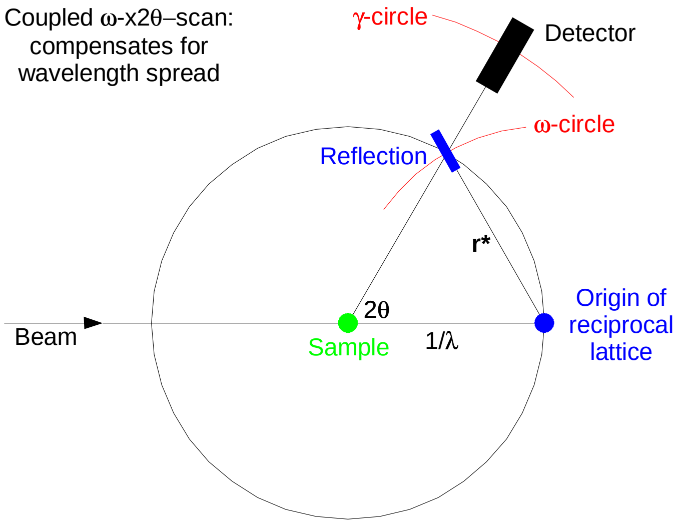
Scans are usually chosen to have 30 points as a compromise between resolution and the time spent stepping the motor positions.

A number of different scan modes are offered in the `scan` menu of **`parame`**. For normal data collection the choice is between ω-scans (option `2`) and coupled ω-*x*2θ scans (option `5`). Both types of scan have advantages and disadvantages and the best choice depends on the quality of the crystal, the wavelength and the 2θ angle. For this reason the coupling constant *x* can be set in **`parame`** as a function of 2θ, to allow for more ω-like scans at lower angles. Ideally, the type of scan is irrelevant because a reflection will only appear at the exact point where the crystal is in diffracting position for it; obviously, however, this is never the case.

In ω-scans (see figure below), the detector remains stationary at the ideal diffracting angle $\gamma = 2\theta$ while the ω-circle is used to move the peak through the diffracting position. This compensates for the mosaic spread of the sample crystal, which has the effect of mis-orienting the reciprocal lattice around its origin, turning the reciprocal lattice points into discs perpendicular to the origin of the reciprocal lattice. Since these discs lie on the same arc as is scanned by ω, ω-scans with the detector held steady will collect the maximum intensity at the middle of the detector (assuming this time that the Ewald sphere is a shell, *i.e.* the mosaic spread of the monochromator is less important that that of the sample crystal).



In coupled ω-*x*2θ scans (see figure below), the detector on the γ-circle (2θ-circle) moves simultaneously with ω. A coupling factor of *x* = 2 keeps the detector on a straight line from the sample to the reciprocal lattice point of the reflection, *i.e.* along the reciprocal lattice vector **r***. This compensates for the mosaic spread of the monochromator which leads to a small spread of incident neutron wavelengths, and hence different magnitudes of **r***.



4.1.2 XTH parameters

Mosaic spread of the sample has a greater effect at low angles, and the mosaic spread of the monochromator has a greater effect at high angles. It is therefore usual to compromise between these two types of scan by using ω-*x*2θ scans with the coupling factor (*x*) curve in `parame` set such that *x* increases with 2θ, giving the scans more ω-character at low angles. The greater the mosaic spread of the sample crystal, the more this curve will lean towards low *x*. The curve is determined empirically by doing test scans of strong reflections along different directions in reciprocal space at a range of 2θ values.

4.1.3 SPS parameters

The number of counts in each reflection is usually defined relative to the monitor rather than time, as they will need to be normalised to the monitor in any case. The optional monitor timeout is a safeguard against certain scenarios in which the system gets ‘stuck’ on one reflection.

4.1.4 MRE parameters

It is also necessary to set the resolution curve (scan widths in different 2θ ranges). These parameters are also determined empirically by doing as many test scans as you feel you have time for. Obviously these should include weak reflections (for determining minimum counting times) and strong reflections along different directions in reciprocal space at a range of 2θ values (for determining the resolution curve). As was seen in the figure in Section 2.1.1, the resolution curve is not a monotonic function of 2θ but has a minimum around $\theta \sim \theta_M$. The actual widths chosen for the scans depend on the forms of the peaks and must be judged on a crystal-by-crystal basis, but typically will be about four times the full-width at half-maximum of the strongest/broadest peaks in order to ensure a good background for integration of the weaker reflections.

4.2 Strategizing optimal use of beam time

4.2.1 Standard reflections

It is standard practice to collect 1-4 ‘standard’ reflections every 50-100 reflections, in order to monitor any possible deterioration in crystal orientation, crystal quality, detector stability of any other aspect of the experiment. The collection of standards is set up in **`parame`**. It is best to choose strong reflections in order that they can be collected more quickly than general reflections. The frequency, number and choice of standard reflections depend on your expectations for the crystal in the given environment. If the crystal is known to be resilient, 1 or 2 standards every 100 reflections is sufficient, mostly as a safeguard against the crystal moving in the beam. On the other hand, if there is a strong possibility of deterioration over the course of the experiment, closer monitoring is warranted.

4.2.2 Equivalent reflections

Collection of symmetry-equivalent reflections provides a check on data quality and reliability but takes time. For irregular and absorbing crystals, it provides an essential check on the significance and validity of any absorption correction applied. It is often best to divide the data collection into stages such that a unique set of reflections is collected before moving on to equivalents, as a contingency against unexpected loss of data-collection time. Note that the geometry of D9 is more restricted than that of a conventional 4-circle x-ray diffractometer, making it impossible to collect Friedel equivalents (-*h* -*k* -*l*) for many reflections (*h k l*), therefore a hemisphere is usually the largest sector that can be usefully collected. Furthermore, when asked to collect a reflection outside the allowed angle range, **`mad`** defaults to its Friedel equivalent, therefore, including Friedel pairs in your `exp.hkl` file will probably result in duplicated reflections.

4.2.3 Annuli

As a further contingency against as unexpected loss of beam time, it is good practice to split the sectors for data collection up into 2 or 3 annuli or shells with respect to 2θ. This can also help optimise the collection time if a large 2θ range is required, by allowing for longer counting times at higher angles.

4.2.4 Number of reflections vs. time per reflection

The number of reflections to be collected *versus* the collection time for each reflection depends on the purpose of the experiment (*e.g.* large numbers of high-angle reflections for accurate anisotropic displacement parameters, or good statistics on a relatively small number of weak reflections for modulated or magnetic structures). Bear in mind that a good ratio of observed data to refined parameters is about 10:1, and the minimum for a reliable refinement is about 5:1. When estimating what can be achieved in the time available, remember the standard reflections and motor driving time. A reasonable minimum time per reflection for a strongly diffracting crystal is around one minute.

4.3 Programming a run

4.3.1 Generating reflection files

Lists of reflections for automatic data collection can be generated with the program **`hklgen`**. It can be run interactively or through the input file `hklgen.dat`, which is described in Section 8.d. Using either input method, for each of the annuli/segments you wish to collect, supply **`hklgen`** with: θ angular limits; order of priority for data collection in terms of *h*, *k* and *l* (shortest to longest real space axis minimises motor driving time); limits in *h*, *k* and *l*; and systematic extinction conditions (only through the input file; see description at end of document). Run **`hklgen`** to get the file `normal.hkl`. You may need to do this a few times to get a version of `normal.hkl` containing an appropriate number of reflections. When you have a version you like, change the filename to something like `experiment_1.hkl`.

Note that **`hklgen`** does not do modulated structures; therefore for complicated cases you should try to prepare these files in advance rather than waste neutrons while figuring it out. However, even in these cases it is still useful to start from an **`hklgen`**-generated reflection list, because **`hklgen`** takes into account the machine software limits. Remember that if your reflection file contains a Friedel equivalent which is unreachable, **`mad`** may default to collect the same reflection twice.

4.3.2 Running the experiment

Data collection can be further automated by use of a macro file, which can contain any lines executable in **`mad`**, including scan parameters as well as reflection files. Create a text file `exp.xbu`, and list the commands in order on separate lines, *e.g.*

```
mes experiment_strong_reflections_low_angle.hkl
mes experiment_strong_reflections_high_angle.hkl
```

```
par sps 5000
mes experiment_weak_reflections_low_angle.hkl
mes experiment_weak_reflections_high_angle.hkl
mes experiment_weak_reflections_low_angle.hkl
mes experiment_weak_reflections_high_angle.hkl
```

Note the repetition of the final steps in order to utilise any remaining beamtime, in the event that the run finishes earlier than anticipated (*i.e.* don't waste neutrons!). You can also change environment variables such as temperature in `.xbu` files, however, remember that this usually affects the unit cell size, requiring the UB matrix to be re-refined (something that is not automated at the moment).

Check *all* the parameters in **`parame`** before starting, print them out (there is a "`LIST`" option on the top menu level of **`parame`**) and save them. Then execute the macro in **`mad`** [**`mad`**: `xbu` *`exp`*`.xbu`], making a note of the first reflection number. Monitor the progress closely to start with in order to confirm that your estimates of time per reflection (and hence for the whole programmed run) were approximately correct.

Note that reflections are saved as they are collected, therefore you can stop the procedure at any point with the red "stop" button on the **`pcp`** tool and only lose the reflection being collected. You can then 'restart' at that reflection [**`mad`**: `rst`], or start the `.xbu` file again (remembering to edit the *`experiment_n`*`.hkl` and/or *`exp`*`.xbu` files before you restart in order to avoid repeating everything from the start.)

4.4 Checking progress remotely

As well as physically visiting the instrument regularly during data collection (which is always necessary), you can monitor the experiment remotely through one of the SGI computers scattered around ILL. Log on to one of these computers as `d9`, then `telnet d9.ill.fr`. The easiest way to monitor the data collection from here is through status programs such as **`msp`**, **`dti sh dyn`** *etc*.

The best way to monitor the quality of the data you are collecting is to refine it periodically during the course of the experiment. The local CCSL and popular Shelx and Xtal packages are installed on `d9sgi` and all the SGI computers in the user computer room on the first floor of ILL4. At the very least you should integrate and pre-process the data from time to time to determine that its internal quality (R_{merge}, R_{int}, R_{stat} *etc.*) is sufficient for your purposes.

When the experiment is finished, print out the parameters from **`parame`** again and save them. Make sure you have documented the crystal shape and orientation before you dismount it.

5 Treating data

5.1 Visualising data

Programs exist for examining the contents of data files (**`datfil`**), quickly recovering scan profiles that appeared in the **`viz`** window (**`splot`**), and fitting curves to scans(**`diffit`**). These are noted in Section 0. They are all interactive and quite simple to use, and are worth getting to know if your experiment will require detailed examination of lots of individual peaks (*e.g.* weak or poorly formed reflections).

In difficult experiments, 3-D visualisation of individual reflections can help you decide which classes of data are worth collecting before beginning a long automated run. It is also a useful as a check on the efficacy of settings used in data reduction, *i.e.* visually comparing the observation of very weak peaks to their integration by **`racer`**. There are two alternative means of visualising reflections collected as 3-D data sets on D9.

The interactive program **`lamp`** has an on-line manual. You should run **`lamp`** on `d9sgi.ill.fr` or another SGI computer at ILL. Above the horizontal line, click on `data`, select `D9` from the instrument list, select the present cycle (unless you are viewing old data), enter the run number of the peak you which to visualise and click on `read` to load it into one of 20 workspaces (defaults to `W1`). You can then plot the contents of a workspace with the tools below the horizontal line. Selecting the ☐ `Surface` option gives a 3-D plot instead of individual slices. Playing around with the ☐ `Range` options is the easiest way to judge the regularity of peak shapes, and decide whether weak peaks are really observed or not.

The program **`fly`** is a simpler but faster means of visualising 3-D detector data in a frame-by-frame manner. It has the particularly useful capability of showing the integration volume used by **`racer`** superimposed onto the observed intensity. Detailed instructions are given in Section 8.9.

5.2 Reducing data

Integration of the 3-D data collected on D9 is carried out with **`racer`**. This is a powerful program, the effective use of which requires reading the attached documentation. It takes the file *`exp`*`.usr` as input and gives a number of output files, including *`exp`*`.col`, which contains the integrated reflection intensities. The reduction of complete data sets should be carried out on `d9sgi.ill.fr`.

5.3 Absorption correction

Absorption and/or extinction corrections are normally carried out with the program **`datap`**, which should be run on `d9sgi.ill.fr` rather than `d9.ill.fr`. The program uses a Gaussian grid system. In addition to the mean path lengths of the reflections $\overline{T}$ weighted by absorption, **`datap`** calculates the vectors needed for anisotropic extinction corrections and TDS calculations. The example input file `datap.dat` is described in detail at the end of this document, as well as the output file *`exp`*`.cor` (which is not as self-explanatory as most of the other output files). Instructions for **`datap`** are given in Section 8.7.

5.4 Data conversion

Conversion of data from “college 5” format (output by **`racer`** and **`datap`**) into input files for CCSL, Shelx *etc.* is carried out with the interactive program **`reform`**, which should be run on `d9sgi.ill.fr` rather than `d9.ill.fr`.

6 After the experiment

6.1 Getting data out of ILL

Because D9 raw data are 3-D intensity arrays, one reflection takes about 250kb of memory. It is therefore generally not usually practical to take raw data sets, consisting of thousands of reflections, home with you. Integrating the data reduces each reflection to one line of text in a `.col` file, making it much more portable. The easiest way to take these `.col` files away is to **`ftp`** them directly to a computer at your home institution. If you cannot do this, you can **`ftp`** them onto a PC or Mac in the user computer room on the first floor of ILL4, from where the data can be saved onto floppy or zip disks, burned onto a CD-R, ftp-ed to your home institution, e-mailed to yourself *etc*.. Users' directories are kept on `d9.ill.fr` and `d9sgi.ill.fr`, and ILL data is thoroughly backed up, therefore your local contact can always forward you the reduced data later if there is a problem. For further information about data storage at ILL see www.ill.fr/Computing/newuser.html.

6.2 Follow-up

After your experiment is over you should fill out a 'User Satisfaction Form', and after you have analysed your data you should fill out an Experimental Report. Both these forms can be obtained from and submitted through the ILL Visitors Club website at http://vitraill.ill.fr, to which you should have been given a password when your experiment was approved.

7 Additional information for local contacts

7.1 Checking the detector

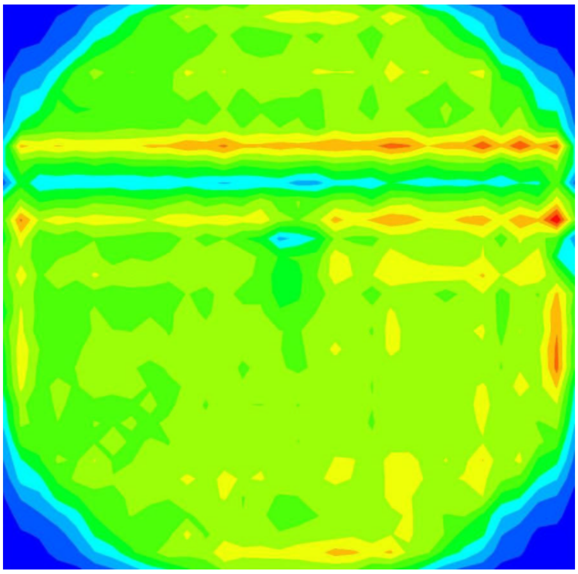
There is a Perspex sheet mounted on a goniometer head in the top draw of the D9 filing cabinet. Using the standard mount on the 4-circle, place this in the beam with a 10 mm collimator. Drive the motors to $\gamma = 90$, $\omega = 45$, $\chi = 180$, $\phi = 50$ °[**`mad`**: `ppp 90 45 180 50`] (the diffuse scattering being most evenly distributed perpendicular to the direct beam). Open the beam and record the diffuse scattering on the 2-D detector for about half an hour [**`mad`**: `acq 1800000 1 1`]. The homogeneity of the detector can also be checked by scanning a strong and sharp reflection [*e.g.* the (4 0 0) of the Ge standard] in χ (vertical) and γ (horizontal).

Ideally the recorded intensity distribution will be completely smooth except in the corners, where the circular collimator trims the square detector. Sometimes, however, inhomogeneities will be found which could affect data quality. The detector consists of 32 × 32 parallel horizontal and vertical wires; therefore, the most common problem is that of an individual wire recording too high or low a current. This must be corrected by adjusting the resistance of that wire.

The process of adjusting the resistances of the wires is incredibly tedious. While there should, in principle, be an optimal algorithm, in practice the system is only logical up to a point. The current detector is quite old (its replacement is currently under construction) and these illogicalities may well be due to shorting among the wires somewhere in the internal electronics. That said, some useful tips can be offered to minimise the pain of the experience.

- The resistances are adjusted *via* 16 small inset screws on each of four electronics cards, found on top of the detector. The four cards, from left to right, correspond to the following detector wires: upper vertical (X_0); lower vertical (X_1); left horizontal (Y_0); and right horizontal (Y_1).
- The adjustment screws are labelled 0-15 on each card, whereas they are labelled `1-32` on the live displays on the computer (**`xdet`** and **`ydet`**). Hence screw 0 on the leftmost card corresponds to vertical wire `1` on **`xdet`**, and screw 15 on the rightmost card corresponds to horizontal wire `32` on **`ydet`**.
- Adjust only one screw at a time and then collect some data to determine the effect. Keep a record of all adjustments so they can be reversed! The sensitivity of the screws is such that noticeable effects are achieved by turning in multiples of 180 °. Most of the time, a clockwise turn decreases the recorded intensity of a wire and an anticlockwise turn increases it. Mysteriously, however, this rule does not always seem to apply…
- The wires are interrelated in two ways. Firstly, adjusting the resistance of one wire will have a noticeable inverse effect on the resistance of its neighbours. Secondly, the total current through each card is constant, so if you try to get one group of resistances smooth by always adjusting them upwards, you will never be able to get the rest of the wires in the card on scale without messing up the initial group. A holistic approach is required.

- If one wire records drastically lower intensity than the others (with its neighbours consequently recording drastically higher intensities), as in the example below, DO NOT ATTEMPT TO ADJUST THE RESISTANCES. In this case there is a more serious problem causing the wire not to register at all, *e.g.* a faulty pre-amplifier or cable connection, requiring the attention of the ILL Detector Group (see attached list of phone numbers). Trying to adjust the resistances will be futile and only create a mess when the real problem is rectified. One thing you can try is cutting the power to the detector by disconnecting the high-tension cables (do not try this without either an instrument scientist or someone from the detector group!). If you try this then you will also need to read the following paragraph.



If the detector has been switched off, its communication with the cards in the electronics cabinet gets disrupted and the signal will not go through to the computer. You can usually reset the communication by pressing the button labelled “RAZ” on the back of the detector (a broom handle is kept in the electronics cabinet for this purpose). If this doesn’t work, you will need to follow the following. In the left hand side of the cabinet, the upper row of vertical cards contains 3 identical ones in positions 02, 03 and 04. Switch them off by pressing “abort”. Then “reset” the card in position 08. Then “reset” those in positions 02, 03 and 04 again. Finally, press the “RAZ” button on the back of the detector.

7.2 Setting and refining the wavelength

Describe the process of setting the wavelength.

The monochromator settings for the most commonly used wavelengths are:

λ	SA	ω_m
0.843	171.24	56.30
0.746	175.28	54.06
0.695	178.22	52.85
0.548	184.86	49.50
0.416	191.00	46.42

The tilt angles are a_1 = 49.25 and a_2 = 89.35.

Filters are necessary to remove higher-order beam contamination at longer wavelengths. They are placed on a wheel near the beam exit, downstream from the monitor. The motor module driving this wheel can be found in the right-hand side of the electronics cabinet. Drive it until the label associated with desired filter is in the vertical position, signifying that the filter is at the bottom. The filters can also be placed in position in **`mad`** (**`mad`**: `filter` *`filter`*). The available filters are:

Filter	**`mad`**: `filter` *`filter`*	Encoder reading	Effective
Er 0.25 mm	`er_0.25`		0.85/2
Er 0.62 mm	`er_0.62`		0.85/2
Ir 0.25 mm	`ir_0.25`		0.70/2
Hf 0.75 mm	`hf_0.75`		0.55/2
In 0.50 mm	`in_0.5`		0.50/2
Ta 0.25 mm	`ta_0.25`		???
B_4C	`b4c`		all
Li	`attenuat`		attenuator
-	`none`		-

Note that use of the Li attenuator precludes use of a filter; at longer wavelengths, if an attenuator is required then something else (normally Perspex) must be placed in the beam before the filter unit, and preferably with B_4C shielding blocking a direct view.

Describe the process of checking and setting the monochromator angles SA, ω_0, A1 *and* A2 *using the results of running* `opti84.com` *etc. after changing wavelength.*

Once the monochromator has been moved, the wavelength must be calibrated. This is normally done with a Ge standard crystal, which is kept mounted on a goniometer head in the D9 filing cabinet. Placed on the standard four-circle mount, the UB matrix for the Ge crystal on this goniometer head should be the UB matrix written on the container. Using **`parame`**, enter this UB matrix, the Ge cell constant (5.6575 Å) and the nominal wavelength, and check that all is well [*e.g.* **`mad`**: `hkl0 4 0 0`; `acq 10000 1 0`]. If the reflection is not there, check that ϕ is zeroed correctly. If it is there, check that the scan parameters in **`parame`** are sensible, find a `.hkl` file for Ge and collect reflections against which to refine the UB matrix and wavelength [*e.g.* **`mad`**: `mes align_ge84.hkl`]. Run these reflections through **`racer`** and **`rafd9`** as described in Section 3.9, however, in this case you need to keep the cell constants fixed and refine λ and the UB matrix, as well as the sample-detector distance if there are a sufficient number of reflections with centres significantly offset from the detector centre. If you have problems getting a decent result with these refined variables, it may be necessary to refine and re-set the ω- and χ-zeros, as described in the following Section. When the refinement is completed, the refined UB matrix and wavelength are entered through the command **`mad`**: `getub`.

7.3 Refining and setting diffractometer zeroes

At the beginning of each cycle, and at other times if necessary, the zeroes of the motors on D9 can be refined with **`rafd9`** against data from the Ge standard crystal. A full refinement in **`rafd9`** requires setting the refinement flags *`K`* to `1` (refine) in the input file `rafd9.dat` (see Section 8.6) for all parameters except the unit cell of Ge.

Obviously, a good refinement of all these parameters first requires the collection of a good set of Ge reflections, integrated carefully with **`racer`** (*i.e.* not just using it as a 'black box'). You then have to take the results of the refinement in `rafd9.lpt` and apply the necessary corrections to the zeroes with **`parame`**. The trick is that while **`rafd9`** gives the absolute value of the refined wavelength, it gives the *corrections that must be added to the zeroes* (rather than their absolute values). The relevant values in **`parame`** are in the column headed "small offsets" of the "machine parameters" menu.

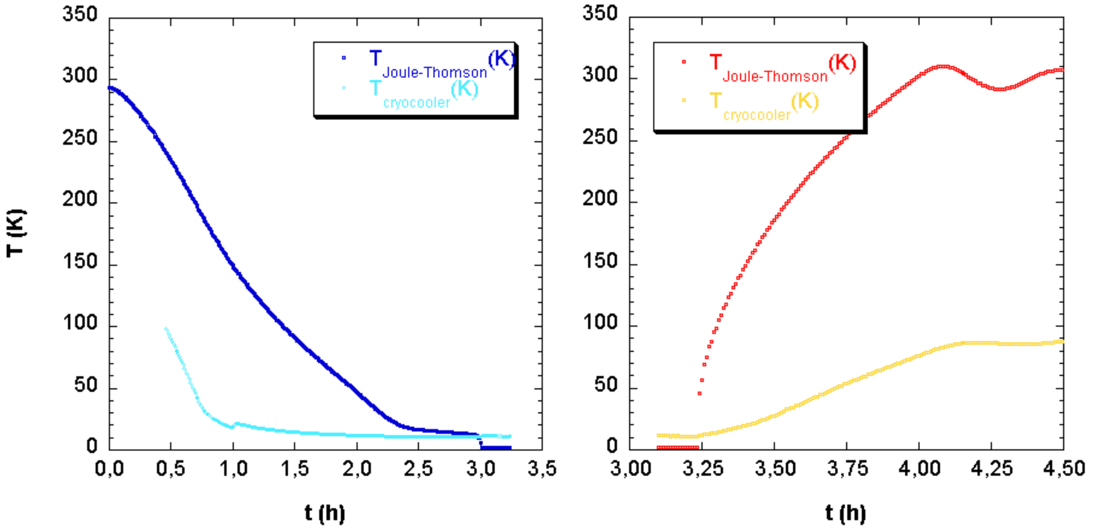
Refine (`rafd9.dat`)	Result (`rafd9.lpt`)	Action (**`parame`**)
x_zero	x_0 (mm), $\equiv \Delta\gamma_0$ (°)	$\gamma_0(\text{new}) = \gamma_0(\text{old}) + \Delta\gamma_0$
z_zero	z_0 (mm)	-
ds_zero	ds_0	-
ome_zero	$\Delta\omega_0$ (°)	$\omega_0(\text{new}) = \omega_0(\text{old}) + \Delta\omega_0$
chi_zero	$\Delta\chi_0$ (°)	$\chi_0(\text{new}) = \chi_0(\text{old}) + \Delta\chi_0$

Note that ϕ_0 is arbitrary.

You should record any changes to the zeroes in the Alignment Logbook in the instrument area.

7.4 Displex operating instructions

The standard low-temperature environment for D9 is a 2-stage Displex with an additional 'Joule-Thompson' stage. This modification allows temperatures of 2 K to be achieved without the use of an orange cryostat, *i.e.* without having to deal with liquid helium. The trade-off is slightly slower cooling times, especially when approaching ~ 15 K where the J-T stage kicks in (see plots of cooling and warming rates below).



In addition, the J-T stage requires a constant flow of helium gas through a liquid nitrogen trap, *even at temperatures above 15 K*, in order to keep it clean. Before starting, ensure that there is liquid nitrogen in the trap and that helium is flowing with a pressure of ~ 12 bar at the bottle. At room temperature, this is sufficient pressure that helium will flow into the Displex and out through the one-way pressure escape valve. The flow rate is very slow at moderate temperatures, however, at very low temperatures (especially 2 K) the flow rate and hence helium consumption is very high. If you intend to stay below 5 K for a considerable length of time, you must ensure you have a full helium bottle before beginning. An apparatus for changing helium bottles without interrupting flow will be installed soon. A full flask of liquid nitrogen is sufficient for about 5 days; refill it using the large bottle in the instrument area when necessary.

When your crystal is adequately aligned and its shape documented, install the inner heat shields and outer vacuum-tight sphere. Make sure the line to the rotary pump is closed (valve on back) and the line to the Displex is open (“vide isolé” on Displex) then switch on the pump (the button is on the front). The arrows on the front of the pump allow you so flip through a lot of menus, only two of which are normally of interest: 340 (vacuum) and 309 (speed). When you switch off this pump, make sure the speed is down below ~ 100 Hz (this takes a few minutes from full speed = 1500 Hz) before allowing air into it, otherwise the blades could be damaged. At low temperatures, you can close the Displex (“vide isolé” on Displex) and turn off the pump.

When you have a decent vacuum (<~ 10^{-3} bar), you can switch on the Displex (switch on the back) and begin cooling.

While cooling, turn the helium pressure up to ~ 20 bar in order to maintain a good flow though the Displex and out the pressure valve. Below 4.2 K, however, the vapour pressure of helium drops below 1 bar therefore it will not escape through the valve. In order to maintain a flow below this point, you need the assistance of the compressor (in front of the pump) which is switched on simply by plugging it in. Note that there is no reason not to switch the compressor on at higher temperatures, *i.e.* you can switch everything on including the compressor and go for lunch while it cools. Once you have reached your desired temperature, the helium flow rate can be reduced again to ~ 12 bar.

Note: do not at any stage open the helium line to air without first warming the nitrogen trap to room temperature (with the heat gun found in the left-hand cupboard if necessary), or the trap could become contaminated.

The controller for the Displex should be set up for you in advance and hence be completely transparent (just use `dti set temp` *`temp`* or **`mad`**: `pte` *`temp`*). If this is not the case, and you are familiar with the ILL temperature controllers, you can set it up yourself using the following parameters (`AUTOIDEN` does not yet work). Remember to check in the `MUX` menu that `external sensor` is set for sensors `A`, `B`, `C` and `D`. Note that the “sample” thermometer is not actually at the sample but rather at the tip of the cold finger, *i.e.* outside the J-T part, therefore it never gets colder than ~ 10 K.

Regulation	`A PT0`	`block 1500, unit # 150`
	`B C0`	
Sample	`C PT0`	`block 1501, unit # 150`
	`D C0`	
Set point limits	`0-300 K`	
Heating power limits	`0-5 W`	
PID settings	`regulation: J_Thompson`	

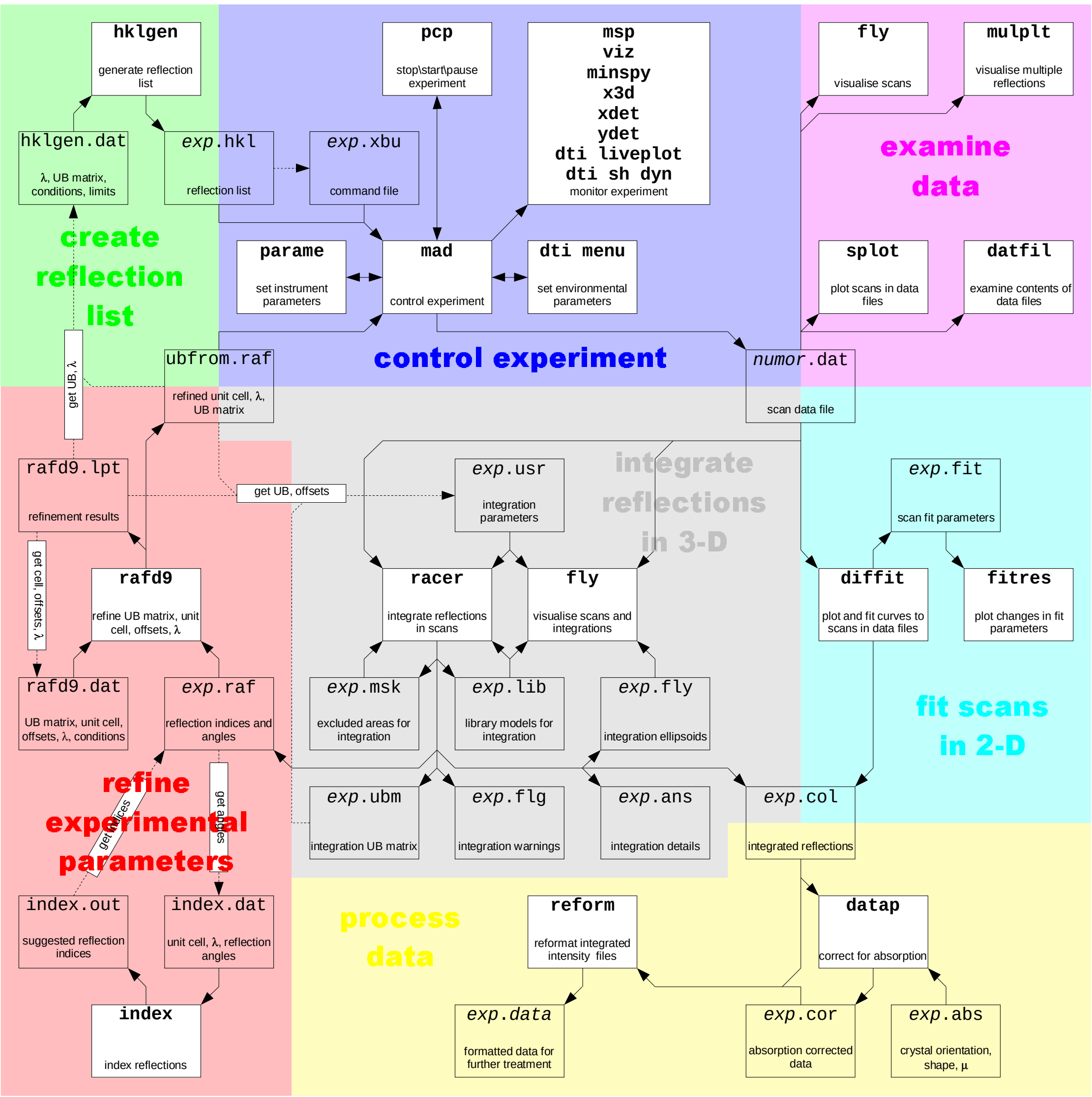
7.5 Furnace operating instructions

The standard high-temperature environment for D9 is a closed-shell furnace, which allows measurements up to 1250 K. The furnace will be installed for you by an instrument scientist or technician. As for the Displex discussed above, the controller should be set up for you in advance so that you can just use `dti set temp` *`temp`* or **`mad`**: `pte` *`temp`*. If this is not the case, and you are familiar with the ILL temperature controllers, you can set it up yourself using the following parameters. Again, remember to check in the `MUX` menu that `external sensor` is set for sensors `A`, `B`, `C` and `D`.

Regulation	`A PTDIN0`	`block 3, unit # -1`
	`B TYPEK`	
Sample	`C PTDIN0`	`block 4, unit # -1`
	`D TYPEK`	
Set point limits	`0-1250 K`	
Heating power limits	`0-25 %`	
PID settings	`regulation: four1375`	

8 Computer programs and commands

The flowchart below illustrates the relationships among the principal programs used in a D9 experiment (bold type in filled boxes) and their input and output files (plain type in open boxes). Dashed arrows indicate cases where parameters need to be cut-and-pasted from the output file of one program to the input file of another.



A graphical (tc/tkl) interface **d9tools** is currently being written with the aim of simplifying the use of these programs, for new users in particular.

8.1 Unix

Here are some useful Unix commands for those not familiar with this operating system. Note firstly that Unix is case-sensitive; this is a common cause of problems for first-time users. Note also that filenames in Unix cannot contain spaces (use "`_`" instead), commas *etc*.. On the other hand, periods are OK because suffixes such as `.txt`, `.dat`, `.dir` *etc*. (important in DOS) have no special significance to Unix; they just help you organise your files, so you can make them as long as you want or omit them altogether. Wildcards can be used to refer to one or multiple files in a directory, *e.g.* `crystal*` or `*.hkl` or even just `*`.

Note that on the SGI computers at ILL, an X-windows environment is used on top of Unix, Unix commands being entered through these windows. Some programs (particularly **`mad`** and **`parame`**) need to be run in xterms rather than regular Unix shells. To get a Unix shell, go to desktop menu in the toolbox and choose unix shell. You can launch an xterm (which many programs prefer or require) from the Unix shell with the command:

`xterm &`	xterm

The symbol `&` means "detach", putting the process into the background and allowing you to continue working in the original Unix shell or xterm. If you forget to detach a process and need to continue working in the original terminal, type ctrl-`z` in the original terminal to suspend the process and then `bg` to place it in the background. If this fails, or for some other reason a process gets stuck, you can use ctrl-`c` which is a 'harder' suspend. However, you should then track down the suspended process and kill it definitively by typing `ps`, finding its ID number in the list, and typing `kill` *`process_ID`*.

Unix has built-in help, provided you know the name of the necessary command:

`command`	(no arguments) see the format for using *`command`*
`man` *`command`*	see the help page for *`command`*
`help`	D9 help pages

Files are organised in a directory tree. To find out where you are in this structure:

`pwd`	print working directory (`/users/d9` after login)

To see what's in the present working directory:

`ls`	list names
`ls -l`	list details

You should make yourself a subdirectory in `/users/d9`:

`mkdir` *`your_dir`*	make directory

To change directory use:

`cd` *`directory`*	*e.g.* from `/users/d9`, `cd` *`your_dir`*

Some example directories and some with special names:

`/users/d9` or `~`	home directory when you login as `d9`
`/`	root directory
`/users/data/`	short-term data storage directory

`.`	present directory
`..`	parent of present directory

To copy, move, rename and delete files:

`cp original copy`	copy files for templates *e.g.* from `your_dir`, `cp ../ling/templates/* .`; save output from a program, *e.g.* `cp rafd9.lpt rafd9_save.lpt`
`mv oldname newname`	move (rename) a file; by including paths with the names you can also move files with this command
`rm file`	remove (delete) a file
`cat file1 file2 > newfile`	combine files

To look at and manipulate the contents of files:

`cat file` `more file`	list contents of `file` ("return" to advance a line, "space" to advance a page, "`q`" to quit; for more type "`h`" for help)
`less file`	as for `more` but starting at the end of `file`
`xpsview file`	view the postscript file `file`
`vi file` `emacs file`	edit `file` using the applications **`vi`** or **`emacs`** (these editors have a steep learning curve; however, they work without X-windows *i.e.* in a simple **`telnet`** session)
`nedit file` `jot file`	edit `file` using the X-windows text editing applications **`nedit`** or **`jot`** (much more intuitive for new users)

To print text or postscript files on the D9 printer:

`lpr file`	print `file` (text or postscript) on the D9 printer
`lpr -p file`	print a text file `file` with title, date, page # *etc.*

To work or transfer files through the network:

`telnet computer`	login remotely to `computer` (`logout` to exit)
`ftp computer`	login remotely for the purpose of file transfer

Within **`ftp`** you can navigate `computer` with `ls`, `cd` *etc.* as well as:

`get file`	copy `file` from `computer` to `d9`
`put file`	copy `file` from `d9` to `computer`
`mget files*`	multiple copy `files*` from `computer` to `d9`
`mput files*`	multiple copy `files*` from `d9` to `computer`
`bye`	quit **`ftp`**

Note that all programs requiring data files as input will by default look in `/users/data`, which contains data from the current cycle. If you need to access old data then you will need to create or place a file in your directory called `.storage` (which you can get from /users/ling/d9/templates), then change the `DI1` environment variable:

`setenv DI1 /usr/illdata/`*`YYC`*`/d9`	use archived data from cycle *`C`* of year *`YY`*

(Remember to change it back afterwards to `/users/data`.)

`xcalc`	calculator

8.2 Programs not requiring unique input files

Some useful interactive programs are available that do not require special input files.

`parame`	instrument and scan parameters for **`mad`** (note: should be run in an xterm rather than a normal Unix shell)
`dti menu`	interactive display and modification of control parameters for the environmental apparatus
`reform`	transform data from “College 5” format (output by **`racer`** and **`datap`**) to a variety of formats including input files for CCSL and Shelx.

A number of programs are available for examining data (when looking at old data remember to `setenv DI1 /usr/illdata/YYC/d9`).

`datfil`	interactive program for examining the contents of data files
`splot`	simple plotting program for examining scans (note: should be run in an xterm rather than a normal Unix shell)
`diffit`	interactive program for examining and fitting curves to scans – used for following phase transitions in conjunction with **`fitres`**
`fitres`	interactive program for following variations in the fit parameters of a series of scans output from **`diffit`**

There are also a number of non-interactive applications that can be launched from the Unix shell. Many of these are automatically launched along with **`mad`**. They should generally be launched by typing their names followed by the Unix suffix `&`, which allows work to continue in the shell from which the application is launched.

`pcp k`	kill **`mad`**, in the event that it is running remotely on another terminal
`dti set temp` *`temp`*	set Displex/furnace temperature to *`temp`*
`dti sh dyn`	monitors the temperature controller
`dti liveplot`	plot of temperature *vs.* time
`msp`	monitors data collection
`minspy`	monitors motor positions
`viz`	live plot of current scan
`x3d`	live image of 2-D detector
`xdet`, **`ydet`**	live cross-sections of 2-D detector
`pcp`	“stop” button for all motors

8.3 Program **`index`**

The program **`index`** attempts to determine the indices of a set of reflections based on the angles among them. You need only know the unit cell of your crystal and the wavelength at which the data were collected. When finding an initial UB matrix for your crystal, **`index`** can be used to correct the identity of the integrated orienting reflections in the **`racer`** output file *`exp`*`.dat` (since **`racer`** will have identified them using the UB matrix for the previous crystal).

The structure of the **`index`** input file `index.dat` is as follows.

`title`	title line
`a b c alp bet gam wavelength min_ang`	unit cell, wavelength, minimum angle between unique reflections
`gamma omega chi phi delta`	γ, ω, χ and φ and δ-γ of reflection 1
`gamma omega chi phi delta`	γ, ω, χ and φ and δ-γ of reflection 2
…	…
`gamma omega chi phi delta`	γ, ω, χ and φ and δ-γ of reflection *n*
`0`	terminating line

8.4 Program **`hklgen`**

The program **`hklgen`** is used to generate a list of reflections for data collection, taking into account the limits on the angles of the 4-circle.

The structure of the **`hklgen`** input file `hklgen.dat` is as follows.

`title`	title line
`wavelength e1 e2 e3 e4 e5 e6 e7 e8 e9`	wavelength and extinction conditions
`u11 u12 u13` *`u21 u22 u23`* *`u31 u32 u33`*	UB matrix
`th_min th_max ch_min ch_max ph_min ph_max om_min om_max`	angular limits
`h_order k_order l_order`	order of variation in the reflection list
`h_min h_max k_min k_max l_min l_max`	(h k l) limits
`out1 out2 out3 out4`	output control parameters

The meaning of the extinction conditions *`e1-9`* is as follows.

`e1` (h k l)	`0`	no conditions
	`1`	$h + k + l = 2n$
	`2`	h, k, l all even or all odd
	`3`	$-h + k + l = 3n$
	`4`	$h = k + l = 3n$
	`5`	$h + k = 2n$
	`6`	$k + l = 2n$
	`7`	$h + l = 2n$
	`8`	$h + k + l = 6n$
	`9`	h, k, l all even
	`10`	h, k, l all odd
	`11`	if $h - k = 3n$, then $l = 6n$
`e2` (h k 0)	`0`	no conditions
	`1`	$h = 2n$
	`2`	$k = 2n$
	`3`	$h + k = 2n$
	`4`	$h + k = 4n$
`e3` (0 k l)	`0`	no conditions
	`1`	$k = 2n$
	`2`	$k + l = 2n$
	`3`	$k + l = 3n$
	`4`	$k + l = 4n$
	`5`	$l = 2n$
`e4` (h 0 l)	`0`	no conditions
	`1`	$l = 2n$
	`2`	$h = 2n$
	`3`	$l + h = 2n$
	`4`	$l + h = 4n$

e5	$(h\ h\ l)$	0	no conditions
		1	$l = 2n$
		2	$h = 2n$
		3	$2h + l = 4n$
e6	$(h\ {-h}\ l)$	0	no conditions
		1	$l = 2n$
e7	$(0\ k\ 0)$	0	no conditions
		1	$k = 2n$
		2	$k = 4n$
e8	$(h\ 0\ 0)$	0	no conditions
		1	$h = 2n$
		2	$h = 4n$
e9	$(0\ 0\ l)$	0	no conditions
		1	$l = 2n$
		2	$l = 3n$
		3	$l = 4n$
		4	$l = 6n$

8.5 Program **`mad`**

The following is a summary of commands in the principal control program **`mad`**. Note that **`mad`** should be run in an xterm rather than a normal Unix shell. Parameters are entered in free format. The symbol "`=`" means "keep the present value" of a parameter and if a line ends prematurely it means "keep the present value" of any following parameters.

`cal0` *`h k l psi`*	calculate all the motor positions for the reflection (*`h k l`*) from the UB matrix and wavelength at ψ = *`psi`* (defaults to `0`)
`cal1` *`h k l`*	calculate 2θ and *d*-spacing (hence the ω and γ motor positions) for a reflection (*`h k l`*) from the lattice and wavelength
`cal5` *`exp.`*`hkl` *`psi`*	calculate all the motor positions for the reflections in the file *`exp.`*`hkl` from the UB matrix and wavelength at ψ = *`psi`* (defaults to `0`)
`cal6` *`h k l psi_min psi_max psi_step`*	calculate all the motor positions for a ψ rotation around the reflection (*`h k l`*) from the UB matrix and wavelength, ψ being stepped from *`psi_min`* to *`psi_max`* in steps of *`psi_step`*
`hkl0` *`h k l psi`*	calculate all the motor positions for the reflection (*`h k l`*) from the UB matrix and wavelength at ψ = *`psi`* (defaults to `0`) and drive there if possible
`hkl1` *`h k l`*	calculate 2θ for a reflections from the lattice and wavelength and drive the ω and γ motors there if possible
`inv` *`gamma omega chi phi`*	calculate (*h k l*) for these motor positions from the UB matrix and wavelength
`rga`	read present position of γ motor
`rom`	read present position of ω motor
`rch`	read present position of χ motor
`rph`	read present position of ϕ motor
`ppp`	read present position of γ, ω, χ and ϕ motors
`pga` *`gamma`*	move γ motor to position *`gamma`*
`pom` *`omega`*	move ω motor to position *`omega`*
`pch` *`chi`*	move χ motor to position *`chi`*
`pph` *`phi`*	move ϕ motor to position *`phi`*
`ppp` *`gamma omega chi phi`*	move γ, ω, χ and ϕ motors to positions *`gamma`*, *`omega`*, *`chi`* and *`phi`* respectively
`mga` *`angle`*	move γ motor by *`angle`*
`mom` *`angle`*	move ω motor by *`angle`*
`mch` *`angle`*	move χ motor by *`angle`*
`mph` *`angle`*	move ϕ motor by *`angle`*

`acq time code keep repeat`	collect data at the present position for `time` counts (if `code = 0`) or ms (if `code = 1`), and write it to memory if `keep = 1` (defaults to `0`), `repeat` times (defaults to `1`)

`sga angle1 angle2 step time code keep`
`som angle1 angle2 step time code keep`
`sch angle1 angle2 step time code keep`
`sph angle1 angle2 step time code keep`
	if `angle1` < `angle2`, scan γ, ω, χ or ϕ motors from `angle1` to `angle2`, in `step` ° steps with `time` counts (if `code = 0`) or ms (if `code = 1`), and write it to disk if `keep = 1` (defaults to `0`);
	if `angle1` > `angle2`, scan γ, ω, χ or ϕ motors respectively through `angle2` °, centred at `angle1`, in `step` ° steps with `time` counts (if `code = 0`) or ms (if `code = 1`), and write it to disk if `keep = 1` (defaults to `0`);
	either scan type can be forced by giving `start` or `from` as the first parameter

`qsc n h_min h_max k_min k_max l_min l_max`	perform a Q-scan of `n` points from (`h_min k_min l_min`) to (`h_max k_max l_max`) s
`mes h k l psi`	scan the reflection (`h k l`) according to the parameters set for automatic scans at ψ = `psi` (defaults to `0`)
`mes exp.hkl`	scan the reflections in the file `exp.hkl` according to the parameters set for automatic scans
`ren0 h k l psi_min psi_max psi_step`	perform a complete scan (as for `mes`) at values of ψ separated by `psi_step` between `psi_min` and `psi_max` around the reflection (`h k l`)
`psi0 h k l psi_min psi_max psi_step`	perform an acquisition (as for `acq`) at values of ψ separated by `psi_step` between `psi_min` and `psi_max` around the reflection (`h k l`)
`xbu exp.xbu repeat`	execute the **mad** commands in the buffer file `exp.xbu` `repeat` times (defaults to `1`)
`par xxx`	give the present value(s) of the parameter `xxx`
`par xxx value1 2 …n`	change the first `n` values associated with the parameter `xxx` to `value1, 2, …n`
`wait time code`	pause for `time` seconds (if `unit` is undefined) or minutes (if `code = minutes`) or hours (if `code = hours`)
`pte temp`	set Displex/furnace temperature to `temp`
`rst`	restart **mad** after stopping it with the **pcp** tool.

All the parameters directly modifiable in **`mad`** (**`mad`**: `par` *`xxx value1 2 …n`*) can also be changed through **`parame`**, which provides an interactive format for viewing and editing. It is still sometimes necessary, however, to enter parameters directly through **`mad`**, particularly as commands within `.xbu` buffer files. When only changing a few parameters in a long list, you can use "`=`" to skip a parameter and then stop at the last one you wish to change. A complete list of these parameters can be obtained by flipping through **`parame`** in "examine" mode. Some parameters you may want to change in this way are listed below.

`par sca` *`points type`*

scans have *`points`* points per peak, scan geometry *`type`* (`1` = γ, `2` = ω, `3` = χ, `4` = ϕ, `5` = ω-γ)

`par sps` *`time code max`*

scan for *`time`* monitor counts (if *`code`* = `0`) or ms (if *`code`* = `1`), up to a maximum of *`max`* (`0` = no timeout)

`par mre` *`the1 the2 the3 the4 the5 wid1 wid2 wid3 wid4 wid5`*

scan widths follow a resolution curve whereby reflections for which $\theta <$ *`the1`* have width *`wid1`* *etc*.

`par mbg` *`points step time`*

measure *`points`* additional background points either side of the peak in *`step`* ° steps with *`time`* counts or ms (already defined by `sps`)

`par xth` *`the1 the2 the3 the4 the5 fac1 fac2 fac3 fac4 fac5`*

coupling factor for γ-ω scans whereby *`fac1`* is used for $\theta <$ *`the1`* *etc*.

8.6 Program **`rafd9`**

The program **`rafd9`** is used to refine values used by **`mad`** when controlling the instrument to collect data. Most importantly, it is used to refine the wavelength and machine zeroes against data from a standard crystal, and to refine the unit cell and UB matrix of your crystal.

The structure of the **`rafd9`** input file `rafd9.dat` is as follows.

`title`	title line
`iout isav ioto amin andi ncyc icvt epsi`	**`rafd9`** program parameters
`x z ds`	x, z (co-ordinates of the detector centre), ds (sample-to-detector distance)
`K wavelength`	wavelength
`K x_zero K z_zero K ds_zero K ome_zero K chi_zero`	zero shifts of x, z, ds, ω and χ
`K a K b K c K alp K bet K gam`	unit cell constants
`exp.raf`	input reflection file
`-1`	terminating line

A code *`K`* precedes each refinable parameter. *`K`* = `0` means do not refine, *`K`* = `1` means refine. For the unit cell parameters in line 6, *`K`* = `2` means refine with constraints, *e.g.*

`1` *`a`* `2` *`b`* `2` *`c`* `0 90 0 90 0 90`	cubic cell constraints
`1` *`a`* `2` *`b`* `2` *`c`* `1` *`alp`* `2` *`bet`* `2` *`gam`*	rhombohedral cell constraints

The **`rafd9`** program parameters in line 2 have the following meanings.

`iout`	`0`/`1`/`2` for short/medium/full output
`isav`	`0`/`1`/`2` for save nothing/UB + wavelength/UB + wavelength + unit cell
`ioto`	`0`/`1`/`2` for orienting reflections optimised/1st pair/1st and 2nd reflections
`amin`	minimum angle between orienting reflections
`andi`	maximum discrepancy between observed and calculated angles
`ncyc`	maximum number of cycles
`icvt`	`0`/`1` for no/yes convergence tests
`epsi`	magnitude of the finite increase

8.7 Program **`datap`**

The program **`datap`** is used to apply absorption corrections to data. It also calculates the mean path lengths and direction cosines describing the incident and diffracted beams required for extinction corrections. You must know the size, shape and orientation of your crystal to use **`datap`** (although there are a number of different ways in which you can define these).

The structure of the **`datap`** input file *`exp`*`.abs` is as follows.

`title`	title line
`a b c alp bet gam wavelength`	unit cell and wavelength (real or reciprocal, angles or cosines; values < 1 are assumed to be cosines)
`h k l theta omega chi phi` *`h k l theta omega chi phi`* *`h k l theta omega chi phi`*	indices and angles (note *`theta`* (= γ/2) not *`gamma`*) of the reference reflections
`ICN1 ICN2 ICN3 ICN4 ICN5 ICN6`	control parameters (described below)
`h k l i` … *`h k l i`*	*`ICN6`* reflections for angle check (can be in hexagonal coordinates)
`start finish th_min th_max`	first and last reflection to be treated, minimum and maximum θ considered
`Agrid Bgrid Cgrid planes`	(only if *`ICN1`* > 0 or *`ICN2`* > 0); number of points in the Gaussian grid along a, b and c (must all be even and have a product < 4097; generally use 15μT, μ = absorption coefficient and T = thickness in that direction), and the number of bounding planes
`h k l d` … *`h k l d`*	(*`ICN1`* = `1`); indices and distance from origin (cm) of *`planes`* bounding planes
`omega chi phi d` … *`omega chi phi d`*	(*`ICN1`* = `2`); angles bringing plane into diffracting position and distance from origin (cm) of *`planes`* bounding planes
`a b c d` … *`a b c d`*	(*`ICN1`* = `3`); coefficients of the equations *`a`*x + *`b`*y + *`c`*z = *`d`* and distance from origin (cm) of *`planes`* bounding planes
`mu scale`	(*`ICN1`* = `3`); coefficients of the equations absorption coefficient μ and scale factor to be applied to F^2 (normally `1`)

The **`datap`** control parameters in line 4 have the following meanings.

`ICN1`	`0`: no absorption correction `1` (preferred): absorption correction using crystal bounding planes described by *h k l* and distance *d* in cm

	`2`: absorption correction using crystal bounding planes described by the angles γ, ω, χ and ϕ bringing them into diffracting position and distance *d* in cm `3`: absorption correction using crystal bounding planes described by the equation of planes in the Seattle system $ax + by + cz = d$ with distance *d* in cm
`ICN2`	`0`: no extinction correction `1`: extinction correction using quantities as for LINUS and TDS calculations `-1`: extinction correction using vector components but no TBAR
`ICN3`	`0`: normal `1`: calculate distances between pairs of vertices; vertex cards follow bounding plane cards
`ICN4`	`0`: normal; four-circle with observed setting angle given in data file with reflections `1`: normal beam geometry (experimental) `-1`: no angles given, therefore four-circle bisecting geometry assumed
`ICN5`	`0`: normal; output corrected data on a file `1`: no output of reflection data
`ICN6`	*n* setting angle checks (must be followed by *n* cards with *h k l*); program will give the angles assuming bisecting geometry

The **`datap`** output file *`exp.cor`* (I6,F9.5,F8.4,F9.5,6F7.4,‘ABTDS’) copies the input reflection information and adds a line containing the following formatted information.

I6	reflection number
F9.5	absorption correction factor
F8.4	mean path length $\overline{T}$
F9.5	TDS correction α (none = -99.0)
3F7.4	XN(I), I = 1,3: components (Å) of the unit vector normal to the plane containing the incident and diffracted beams, parallel to the real cell axes *a*, *b*, *c*
3F7.4	XNSO(I), I = 1,3: components (Å) of the unit vector normal to the incident beam in the plane of the incident and diffracted beams

8.8 Program **`racer`**[*]

The program **`racer`** is used to integrate raw data (3-D intensity maps) from the 2-D detector of D9 working in ‘single reflection’ mode, *i.e.* during measurement the centre of the detector is driven through the centres of individual Bragg reflections whose positions are calculated from the experimental UB matrix. It outputs several files including `experiment.col` (a list of reflection indices, intensities and angles for use by a structure refinement package). The method employed is based on that described by Wilkinson, Khamis, Stansfield & McIntyre (*J. Appl. Cryst.* **21**, 471-478) and uses *a priori* information obtained from strong reflections to improve the precision of integration of weak reflections. When looking at old data remember to `setenv DI1 /usr/illdata/YYC/d9`.

8.8.1 Input files

`experiment.usr`	Integration parameters.
`experiment.msk`	Excluded regions (can be empty or missing initially).
`experiment.lib`	Integration library for weak reflections (can be empty or missing initially).

The structure of the **`racer`** input file `experiment.usr` is as follows.

`cont monitor deadtime`	normalisation parameters
`actvol volfac sigbr fill difflim`	
	integration parameters
`iadlib irefine iprint icard`	range of 3-D data to use
`imin imax jmin jmax kmin kmax`	
	range of 3-D data to use
`drad horiz vert omega`	sample-to-detector distance, offsets
`U11 U12 U13`	UB matrix (optional)
`U21 U22 U23`	
`U31 U32 U33`	
`twinU11 twinU12 twinU13`	twin UB matrix (optional)
`twinU21 twinU22 twinU23`	
`twinU31 twinU32 twinU33`	
`D9 YYC`	instrument, year and cycle number
`180000`	reflection for integration
`180005 180015`	range of reflections for integration

The parameters have the following specific meanings. Recommended starting values are given in parentheses where necessary.

`cont (0.2)`	Fractional peak height of the contour modelled for strong reflections.
`monitor (10000)`	Nominal monitor count to which the intensities of all reflections are subsequently scaled. Intensities are also scaled to a ‘standard’ step width of 0.05 °.
`deadtime (0.0)`	Deadtime in seconds. The count on a particular pixel is corrected using the expression $Counts_{corr}$ = $Counts_{obs}$/(1 – `deadtime` × A) where A is the larger

---

[*] Taken almost verbatim from “A Guide to RACER” by C. Wilkinson and G.J. McIntyre.

	of the total counts on the two wires that intersect at that pixel.
`actvol (300)`	The ‘actual’ volume which has the same ellipsoidal shape as the modelling contour and will be used for the integration of weak reflections. This depends on the size and quality of the crystal. (Aim at an enclosed intensity of about 90 % of the total peak intensity; see the output file `experiment.ans`.)
`volfac (4.0)`	Factor by which `actvol` is multiplied to define the limit of integration for strong reflections. Background is measured in a shell of equal volume lying outside `volfac` ×`actvol`.
`sigbr (1.0)`	Signal to background ratio defining a ‘strong’ reflection, determined from a crude B/P/B analysis of the ‘omega’ scan given in `experiment.ans`.
`fill (0.8)`	Fractional filling factor of the ellipsoidal modelling shape by pixels above `cont` in a ‘strong’ reflection. Failure to reach a level of about 0.8 indicates a mis-shapen reflection, which is not then used as a library model and is subsequently treated as ‘weak’. Poor mosaic spreads, however, may mean that this factor has to be reduced below 0.8.
`difflim (0.2)`	A flag is raised if the (3-D) difference in degrees between the observed centre of gravity of a strong reflection and its centre of gravity calculated from the UB matrix and offsets is more than `difflim`.
`iadlib (1)`	On (=1) / Off (=O) switch for building a reflection library processing with an existing library. Weak reflections are fully processed only when `iadlib` = `0`.
`irefine (100)`	The number of strong reflections encountered in processing for the refinement of the experimental UB matrix and offsets. A calculation is done every `irefine` strong reflections. A negative number here results in UB matrix refinement only after the last reflection in the processing list.
`iprint (3)`	Print switch. If `iprint` = `0` minimum information is given in the `experiment.ans` file. If `iprint` = `1` the counts in each element of integration are written in map form in `experiment.ans` (not recommended except for single reflection diagnostics; the same plots are available on screen with the ‘eggs’ option of **`fly`**). If `iprint` = `3` a profile of peak-minus-background points inside the integration ellipsoid (*), and inside the rectangular box enclosing the background ellipsoid(.) is printed in `experiment.ans`.
`icard (0)`	Format option for `experiment.col`. If `icard` =

	`0` the ‘A’ card format is used with integer *hkl*, if `icard` = `1` the ‘R’ card format is used with real *hkl* for non-commensurate problems.
`imin imax jmin jmax kmin kmax` (`5 28 5 28 1` *n*)	(`N` is the number of frames in each scan) The range of vertical, horizontal and frame elements you wish to consider for processing. Looking toward the specimen from behind the detector, the point (`1 1 1`) is at the bottom left-hand corner of the first frame.
`drad horiz vert omega` (`488.0 0.0 0.0 0.0`)	The specimen to detector distance `drad` and the horizontal and vertical offsets of the detector in mm. These are obtained either from `ubfrom.raf` output by **`rafd9`** or from `experiment.ubm` output by this program. The ω offset is given in degrees (no need to change the order or signs of the values).
`U11-33`	UB matrix used for the processing. This is optional. If no matrix is, given, the measurement UB matrix is used. A process wavelength different from that used at measurement time may also be given at the end of the options line, after `icard`.
`twinU11-33`	UB matrix for a second crystallite. The UB matrix must also have been given above. No treatment for twinning will be made if `twinUB` is omitted.
`D9 YYC`	Instrument and the cycle (*e.g.* `891` is cycle number 1 in 1989). When looking at old data remember to `setenv DI1 /usr/illdata/YYC/d9`.
`180000`	A single number refers indicates a reflection to be processing. Two numbers on the same line indicate a range (inclusive) to be processed. These may be combined in any sequence. Note that they are quite sensitive to formatting; *do not leave spaces before numbers and do not add comments to these lines.*

8.8.2 Output files

`experiment.ans`	Details of the integration of each reflection.
`experiment.col`	The number of each scan, *hkl*, the integrated intensity and its standard deviation, $\gamma/2$, ω, χ, ϕ, the measurement temperature and ‘`A`’ or ‘`R`’ in column 80. The ‘A’ format is (16,3l4,2F10.2,4F8.2,F9.2, ‘A’) and the ‘R’ format is (16,3F6.2,18,14,4F8.2,F6.1,5X, ‘R’) Each time the program is run a line is added to the most recent `experiment.col` for each reflection processed. If no file is present at the start, one is created.
`experiment.raf`	Data in appropriate format for input to **`rafd9`**. A new line is added to the file for each strong reflection processed. A file is created if none is present.

`experiment.lib`	Details of library models used in the integration of weak reflections. The file is read at the beginning of **`racer`** and updated (in order of γ) for each suitable strong reflection. Re-processing a particular reflection causes the previous entry for that reflection to be overwritten.
`experiment.ubm`	Contains refined UB matrices and detector offsets for every `irefine` strong reflection processed. If it is not possible to perform a refinement (*e.g.* all the reflections are coplanar) the program waits until it has a non-singular normal-equations matrix and refinement may therefore be over more than `irefine` reflections.
`experiment.flg`	Advises of unusual features during processing.
`experiment.fly`	Integration ellipsoids for input into **`fly`**.
`experiment.msk`	A file created by the program if one does not already exist in the area from which you are working. It contains a 32×32 array of elements of `1` and/or `0`. Its function is to mask out (`0`) elements on the detector which you wish to exclude from your calculations. If it is created by the program it contains only `1`s. If it is there already, and contains any `0`s which mask out elements used by the program for the integration of a reflection, a flag is raised in `experiment.flg`.

8.8.3 Suggested strategies

The program will run and give reflection intensities for any sensible values of the input parameters, but to obtain the best results for weak reflections it is necessary to tune the variables in the `experiment.usr` file, to refine a good UB matrix and to have a full library.

Starting

Take a previous `experiment.usr` file and change the instrument, cycle and reflection ranges appropriate to process the first fifty (say) reflections. Adopt the `drad` and offset values used during the experimental measurement and use the experimental UB matrix (default). Set `iadlib = 1` and `iprint = 3`.

Tuning

Run the program and look at `experiment.ans`. If necessary, change the `sigbr` parameter in the `experiment.usr` file and re-run the program until you have a suitable number of 'strong' reflections. Check that these have backgrounds similar to the 'approximate background' given in the `experiment.ans` file. Another indicator is the position of the 'background' points (.) on the line printer graphs. If they lie consistently below the 'zero' level on both sides of the peak then the background is high and is probably being taken too near the peak centre. Check that the `actvol` ellipsoid contains about 90 % of the total peak intensity. Otherwise vary `volfac` and `actvol` until this is the case.

You will probably find the ‘eggs’ option of the **`fly`** program which displays peak and background integration regions superposed on the numerical data useful at this stage.

Library and UB

Set the `iadlib = 1`, `irefine` to a number which you estimate will give you a few UB matrix refinements when processing all your reflections (or a negative number to use the whole range) and include the full list of reflections you wish to process. Run the program and check `experiment.ans` to see that you have tuned the parameters sensibly. You should have now created a library over the whole reflection range and obtained better UB matrices and offsets.

Processing

Incorporate into `experiment.usr` either the offsets and UB matrices from `experiment.ubm` (produced by the program), or run `experiment.raf` with **`rafd9`** to obtain these *via* `ubfrom.raf` for a particular reflection range. Set `iadlib = 0` to freeze the library file and `sigbr` and `fill` to lower values (`0.3`, say) to allow as many as possible of the reflections to be treated as ‘strong’. Rename or delete `experiment.col` so a new one can be created. Change the reflection range as appropriate and process the data.

8.8.4 Tips for difficult cases

Wide crystallite spread

In this case it may not be valid to assume that reflections are ellipsoidal. The program can be fooled into performing a one-dimensional integration by having an empty library and setting `sigbr` artificially high, so that all reflections are treated as weak, but with no library models available. The ‘approximate’ intensity in `experiment.ans`, obtained by projecting data on to the frame axis and subtracting an iterated projected background, is then written to `experiment.col`. Only strong reflections will be well integrated by this method.

No strong reflections

A special version of **`racer`** (**`werner`**) is available to construct a library for this type of data. It superposes reflections until a sufficient statistic has been acquired to pass the (preferably relaxed!) good shape criterion for ellipsoids. Data can then be processed as normal with **`racer`**. Ask your local contact for more details.

Other reflections in the measurement box

In the spirit of data collection with a conventional single detector **`racer`** assumes that the user only intends to observe one reflection in each scan. For large unit cells, or extremely broad crystal mosaic, there may be other Bragg reflections in the measurement box. The program does mask off those parts of the integration volume which may contain counts due to neighbouring reflections. The indices of the four shortest reciprocal lattice vectors should be given in `experiment.usr`; **`racer`** then ignores the volume beyond the plane that is the perpendicular bisector of each vector when its tail is placed on the calculated centre of the scanned reflection. For a conventional primitive unit cell the shortest vectors would be (1 0 0), (0 1 0), (0 0 1) and $(1\ \bar{1}\ 0)$ (for a hexagonal cell). For centred cells longer vectors may be given, while for incommensurate cells factional indices may be needed. `imin`, `imax`, `jmin`, `jmax`, `kmin`, `kmax` and `experiment.msk` may also be used to limit the data treatment volume, but normally just to avoid non-uniform regions of the detector.

Uneven background (often from specimen environment)

If the background does not conform to a Poissonian statistical distribution, usually due to high points from powder lines *etc.*, then an alternative background subtraction is made. In this method the background is assumed to be different in each pixel, but invariant from frame to frame. It is determined over the detector area outside the ellipsoidal peak integration volume, an average being taken over pixels down the frame axis. It is thus better determined statistically on the edges of a frame than in the middle. The pixel by pixel background is subtracted from each peak element. A warning is given in `experiment.ans` and a flag is raised.

Twinned or multiple crystals

Extra peaks due to twinning, a second crystal or even a different unit cell can be masked. If orientation matrices for two crystals are included in `experiment.usr`, the first is used to define the centres of reflections in the usual way, the second is used to determine whether part of the integration volume should be excluded, in a similar manner to the masking of neighbouring reflections of the first crystal. At present only one twin is allowed, only the closest reflection of the second crystal is considered, and no systematic absences or lattice centrings are considered.

The integrated reflections are classed among three types: type 1 are those of the first crystal that are unaffected by the second crystal; type 2 are those where there is partial overlap of reflections of both crystals; and type 3 are those where there is exact superposition of reflections from the two crystals. The division between type 1 and type 2 is when the distance between the reflections of the two crystals is less than six lengths of a voxel (voxel = vertical pixel × horizontal pixel × scan step). The reflections are written to two files, `experiment.col` for the untwinned reflections (type 1) and `experiment.twn` for twinned reflections (type 2 and type 3).

The reflection coordinates intended for nudging of the matrix by **`racer`** or for unit-cell refinement by **`rafd9`** are handled slightly differently. Both type 1 and type 3 reflections are accepted and put into `experiment.raf`. Similarly, both type 1 and type 3 reflections are included in `experiment.lib`. Weak reflections of type 1 and type 3 are integrated using the minimum $\sigma(I)/I$ algorithm, but for weak reflections of type 2, for which the shape of the overlapped reflections cannot be estimated by the shapes of nearby strong reflections (even if the overlap is the same), the relative intensities of the contributions may be different, so the simple analysis of the ω projection is used.

For twins there usually exists a simple matrix transformation between the individual orientation matrices, but to allow for the possibility of two unrelated crystals both matrices are entered, rather than one matrix and the transformation.

There follows an example of the input parameter files, and selected output files, for ABP. This was a monoclinic structure which, it appeared, formed twins related by a 180 ° rotation around the *a* axis, *i.e.* $a^*_1 = a^*_2$, $b^*_1 = -b^*_2$, c^*_1 and c^*_2 18 ° apart.

For the first domain:

`dom1.usr`

`dom1.col`
`dom1.twn`
`dom1.ubm`
`dom1.raf`

For the second domain:

`dom2.usr`
`dom2.col`
`dom2.twn`
`dom2.ubm`
`dom2.raf`

Some care, and trials with **`mulplt`** and **`rafd9`** with a small subset of reflections, may be needed to get the initial orientation matrices of the two crystals, but once these are reasonable a possible data-treatment and analysis strategy is as follows.

1. Run through **`racer`** with the first orientation matrix placed first in *`experiment`*`.usr` to give `dom1.col` (untwinned reflections for crystal 1) and `dom1.twn` (twinned reflections).

2. Run through **`racer`** with the second orientation matrix placed first in *`experiment`*`.usr` to give `dom2.col` (untwinned reflections for crystal 2), and `dom2.twn` (twinned reflections). The filers `dom1.twn` and `dom2.twn` should be identical, except that the indices will correspond to the appropriate orientation matrix.

3. Repeat steps 1 and 2 with the new matrices from `dom1.ubm` and `dom2.ubm`, or better, from `dom1.raf` and `dom2.raf` run through **`rafd9`** to ensure that the same offsets are used for both crystals, until there are no significant improvements in the matrices.

4. To check that the masking is working, look in detail at a scan that clearly shows peaks for both crystals, and compare the **`racer`** profiles of runs for the two crystals with a run in which no masking is made.

5. The final result will be three reflection files:

`dom1.col`	reflections labelled $h_1k_1l_1$ with contributions $p_1(h_1k_1l_1)$;
`dom2.col`	reflections labelled $h_2k_2l_2$ with contributions $p_2(h_2k_2l_2)$;
`dom1.twn` (= `dom2.twn`)	reflections labelled $h_1k_1l_1$ but with contributions $p_1(h_1k_1l_1) + p_2(h_2k_2l_2)$;

where p_1 and p_2 are population scale factors for the two crystals, $h_1k_1l_1$, and $h_2k_2l_2$ are indices for crystals 1 and 2 respectively. The files `dom1.col` and `dom2.col` can be used together in most structure refinement programs, if included with independent scale factors, or if they are scaled and merged beforehand. The file `dom1.twn` can be used if the program allows for twinning. It would probably require a little recoding to allow `dom1.col`, `dom2.col` and `dom1.twn` to be included together.

8.9 Program **`fly`**

The program **`fly`** can be used as a stand-alone way of looking at 2-D detector data frame-by-frame, or in conjunction with a user (*`experiment`*`.usr`) and a library (*`experiment`*`.lib`) file from **`racer`** to superimpose the integration volumes used. When looking at old data remember to `setenv DI1 /usr/illdata/`*`YYC`*`/d9`.

In the stand-alone mode, simply type **`fly`** *`run_number`*. It will default to a 2-D colour-coded intensity map of the frame with the highest intensity, with the colours scaled for the whole scan. You can than use the following interactive commands.

`n`	next frame
`p`	previous frame
`#`	frame number
`num` *`12345`*	run number
`prx`	x-profile
`pry`	y-profile
`2d`	2-D plot
`m`	colour-coded intensity map
`num`	number map
`f`	scale the colour-coding for the present frame
`s`	scale the colour-coding for the whole scan
`max` *`max`*	scale the colour-coding to a maximum intensity *`max`*
`min` *`min`*	scale the colour-coding to a minimum intensity *`min`*
`e`	‘eggs’ – superimpose the integration volumes

To use the ‘eggs’ option (`e`), you must start the program with **`fly`** *`experiment`* in a directory containing *`experiment`*`.usr`, *`experiment`*`.lib` and *`experiment`*`.fly` (*i.e.* in which **`racer`** has been run). The ‘eggs’ option will then display number maps with superimposed colour representations of the integration volume. Pixels highlighted in red are contained within the *`actvol`* core of the peak, which is used for the integration of ‘weak’ peaks. Pixels highlighted in green are also used in the integration of strong reflections. Pixels highlighted in violet are contained within the integrated background volume annulus.

9 Support

There is an ILL phone book in a yellow folder in the instrument area, and on the ILL web site. Some phone numbers particularly relevant to a D9 experiment are listed below.

D9		71 29
Physicist	Maria-Teresa Fernandez-Diaz	76 06
	(H)	0-04 76 46 04 24
	Chris Ling	73 88
	(H)	0-04 76 43 86 02
Technician	John Archer	74 22
Secretary	Marie-Rose Guillermet	72 26
Reactor control room		11
Radioprotection (SPR)		31 or 75 59
General safety	Henry Schweitzer	70 05
Guardians	*Site entrance*	15
Medical service		33
Emergency	*CENG fire brigade*	18
Ambulance		0-15
Hall d'essai	Pierre Thomas	73 71
Detector	Giuliana Manzin	73 66
	Bruno Guérard	72 77
`mad`, VME/PCI	John Allibon	74 26
Instrument electronics	Jacques Munnier	71 31
Network, Ethernet	Armand Guellec	70 21
Unix workstation	*Helpdesk*	70 13
	Odile Tillier	73 89

Note that the Instrument Control Service (SCI) is currently conducting a trial of an Out-of-Hours Support (OHS) service. Further technical information about the instrument control systems and the status of the OHS trial can be found on the ILL intranet at http://dpt.ill.fr/SCI/sci_web3/welcome.html.

Interventions should be recorded in the appropriate log book in the instrument area: Electronic Intervention Log; Displex Calibration Log; or Alignment Log. There is also a User Log in which you should record anything unusual observed in the course of your experiment.