Using HKL3000R for Data Collection

Hardware Note

Our system has a Rigaku microfocus generator (MicroMax-007HF), great Osmic optics (Varimax HF) and, in addition to the traditional R-Axis IV++ image plate detector, a hybrid pixel array detector (Dectris Pilatus 200K) that shows very high detective quantum efficiency (DQE) and a very fast millisecond readout time (that allows the shutter to stay open during 'continuous' data collection). The Pilatus detector is small and this makes the Rigaku AFC-11 partial 4-axis goniostat (ω, φ, 2θ and partial χ) critical to mapping the desired slices of reciprocal space onto the detector surface.

Controlling the Machine

The control computer runs a Centos Linux operating system to control the machine, and the upstream source is a Red Hat Enterprise Linux operating system running HKL3000R and the X-ray Generator Control (XG Control).

HKL3000R effectively controls the goniostat and detector, indexes data, calculates strategy and processes the data. The XG Control program controls the generator including start/stop and power ramping.

A quick outline of workflow (using the Pilatus side)

1. Initiate communication with device using Connect button in Collect tab.
2. Goniostat in mount position (Collect>Align); mount crystal and center (turn on lights and check that the crystal/loop is centered in the cross-hairs during rotation).
3. Test shots at 0° and +89° for 10 seconds per 0.25° or 0.50° (increase time for R-Axis IV++).
4. If OK then index using Index tab, select *primitive triclinic* in Bravais Lattice.
5. Refine as Fit Basic and Fit All, then click Bravais Lattice to select likely correct lattice (symmetry).
6. Refine with Fit All and then Mosaicity.
7. In Strategy tab first select space group (if known), then launch the Rigaku Strategy (this takes into account the small detector surface of the Pilatus, the 4-axis goniostat and guarantees redundant and complete data collections avoiding crashes by disallowing certain positions using the built-in set limits), edit to taste, exit strategy and Load DC to put strategy in Collect tab.
8. Start data collection via Collect Tab.
9. Check integration parameters (spot size, profile fitting radius etc.), then start integrating data runs. Scale data to check for quality and potential point groups.

   Finalize scaling (Error Scale Factor to change χ^2), educated guess on space group.

Using the Graphical User Interface (HKL3000R)

The computer is always left running, usually logged in. Upon login (user name: rigaku; password: rigaku), the default Desktop has icons for the software on the left-hand side. There are two icons labeled "HKL3000R" and "HKL-3000R P300K Chi". You want to use the **HKL-3000R P300K Chi** to control the machine. The **HKL3000R** version of the program can be used to analyse data you have *already* collected and cannot control the machine.
Other icons of interest are **XG Control** for controlling the generator.

On top of the window is the icon bar, which can be used to open a Terminal (you can also right-click the mouse and select 'open terminal') if you want to use standard Unix commands (e.g. for looking in your home directory or managing data). That's the icon right next to the System menu. The web browser (Firefox) is the icon next to that. Since this is a data collection/processing machine, **do not install any additional software on this system without permission**.
The HKL-3000 combines all the functions of HKL-2000 and incorporates in addition a number of other crystallographic computer programs such as the CCP4 suite, the ARP/wARP suite, the SHELX-97 suite, the SOLVE/RESOLVE suite, Coot, PyMOL and Buccaneer.

Quit any existing instances of HKL3000R to avoid accidentally overwrite other people's data. Start the software by clicking on the **HKL-3000R P300K Chi** icon. Underneath the top menu bar is a series of tabs labeled Project, Collect, Data, Summary, Index, etc. Click on each tab to reach the function for each. If you are screening crystals you will just use **Project** and **Collect**. If you are collecting data or want to auto-index your data you will use more of the tabs further to the right. If you've used HKL2000 then the general layout is familiar. HKL2000 is a data processing program that uses the underlying programs Denzo and Scalepack, and HKL3000R is an extension of that program to control the machine as well as process the data.

Use the **Project** and **Crystal** framework to keep images for each protein under a distinct project name, with different crystals being a new crystal within a project. Avoid making names absurdly long. Click on the **Project Tab** to manage projects and samples. You can create new projects, save existing ones, load old ones. When HKL3000 starts it defaults to a generic project name - it doesn't save your old project info so you should remember to save it manually before quitting and load it after starting. It's possible to collect data and process it without changing the default project name. Just make sure you specify a separate and unique data directory name for your data and processing directories. I generally recommend against doing this if you're actually *collecting* data since it makes more sense to cluster related datasets into one project. Also, NEVER modify someone else's project in this way.

To reload an existing project, use the Load button within the Project tab and locate the relevant .sav file in the /data directory (scroll past the subdirectory names to find them).

An alternative to this interface is to use the **Project, Crystal and Base Directory** fields in the Collect tab. In all cases the data filenames are

```
/data/Project_Crystal/Project_Crystal_0000.img
```

where Project and Crystal are taken from the values that you've given. The "0000" gets automatically incremented, starting at 1, when you collect images. You can use the Data Directory field in the Collect tab to provide a different directory structure (e.g. grouped by name, lab etc).

The **Collect** is by far the most important tab. To start doing something productive you should mount your crystal and click on the **Collect Tab**. The buttons on the left are the important ones. You need to press the **Connect button** to get the program to talk to the machine - this establishes connections to the servers in the generator itself. It also initializes the detector and all axes on the goniostat when it does so - this takes a while since the goniostat is not all that fast. LOCK DOWN THE PHI (φ) AXIS BEFORE CLICKING ON CONNECT.

Below the Connect button the radio buttons select what type of experiment you are doing. **Crystal Check** loads default parameters for the sort of images we shoot to test crystals. **Data Collection** loads the sort of parameters you would use if you actually want to collect an entire dataset. Make sure you define a Project and Crystal either via the fields in this tab or via the Project tab. For Crystal Check you should set distance, number of frames, omega (ω) start, frame width (0.25° or 0.50°), exposure time (10 seconds is a good start) and click the toggle if you want to shoot orthogonal pairs of images separated by 90 degrees. Limitations of the goniostat mean that sometimes it is better to separate them by 80° or perhaps -90° since at 2θ=0° an ω=+90° is in the forbidden range but ω=89° or ω=80° is acceptable. Once you have changed the values the blue button at left marked **Collect Sets** will be enabled and you can collect the images. The new system is quite fast - moving the goniostat is one of the slower steps but typical exposure times are short (seconds) and readout is very fast (milliseconds).

With the old single axis goniostat on the RAXIS system, the rotation axis was called phi (φ). Note that with the multi-circle goniostat the scan axis is always omega (ω) rather than phi - this is typical of multi-axis of goniostats where ω is the more accurate axis used for scanning and the axes χ and φ are used for repositioning only.

For actual data collection use the **Data Collection** button, which has many of the same parameters as Crystal Check. The current frame is now displayed using the QT version of XDISP - where the contrast and image zoom are the only significant features. This has a different GUI behavior than HKL's X-windows based XDISP program and runs independently of it. With the smaller detector and a more vesatile goniostat it is likely that you'll be collecting data with multiple runs - in fact you'll probably get *better* data if you collect with two different φ/χ combinations rather than just one.

Additional features on this tab are: **"Status" button** at the lower left is the most useful. The Status button is sometimes a more reliable source of information.

Indexing and Generating a Strategy

In the **Data tab** HKL3000 will load all data runs associated with the project - both test images and data collection runs, and also for data runs that are about to be collected. The blue boxes mark the individual blocks of data. To remove irrelevant data runs, press Select in the blue box and then Remove Set in the Set Controls panel. The red check buttons within each blue data run block offer control whether the data is scheduled to be integrated and scaled. For initial indexing you need only one frame from one data run so perhaps deselect the others - in particular the Project/Crystal framework will often populate the Data window with data from all samples for that project. Also when processing data after collecting test oscillations, make sure you deselect those test frames since they add nothing to the data.

If the data directory does not show up automatically, navigate to it via the Directory Tree navigation panel at center-left, click to select the directory and press the >> button next to New Raw Data Dir. Select the processing subdirectory similarly and/or create your own. Data collection parameters for the selected data set are shown at the bottom of the tab.

To change the expected number of frames in a data set, select it using the SELECT button, and click the EDIT SETS button (center-right). Normally the correct number is found automatically, but you can use this to exclude ranges of frames at the start or end of a set from processing.

Indexing your diffraction pattern is a necessary precursor to strategy calculation - symmetry axis orientations need to be known. Go to the **Index tab** - verify that the data frame(s) listed are the ones you want to index via the panel at top left - adjust via the Data Tab. The text panel at top-right will contain the information for indexed/refined unit cell parameters etc. The buttons below-right control the workflow within Index. The check boxes below-left control what parameters you want to refine to improve predicted and observed spot positions, and a few things about spot size.

Click the DISPLAY button to view any frame within the run selected at top left. Click PEAK SEARCH to pick spots on the frame. Unlike XDS and MOSFLM, Denzo auto-indexes from a single frame. Most of the time this works, but potentially less precise than extracting the information from a pair (or range) or orthogonal images.

You can also do Peak Search on images that are not the first one in the data run, by entering the frame number in the box next to the peak search button. For really problematic indexings I try every 10th frame to get an indexing that I believe to be correct and work from there. Denzo is smart enough to take into account the offset between frame N and frame 1 when integrating the data.

It is possible to add spots from more than one contiguous set of images. But you cannot do it via the Peak Search button in HKL3000. First, use DISPLAY to bring up the first image in XDISP. Press the Peak Sear button in XDISP itself (lower left in button stack). Then middle-mouse click on the Frame button at top right in XDISP to load the next image, which will also peak search and the peaks added to the list. Repeat for a few more frames. This way, the peaks will be accumulated off multiple frames - and it also appears to improve the mosaicity estimate during initial auto-indexing.

Once you have got enough data spots selected press the INDEX button in the Index tab.

Denzo presents a table of possible indexings in the "Bravais Lattice Table". The first one to pay attention to is primitive triclinic all the way at the bottom - this is the inherent spot arrangement on the frame and the largest value here corresponds to your smallest inter-spot spacing. It is a *Primitive* (P) unit cell, which means that there are none of the systematic absences present in C,F,I lattices that lead to a large proportion of the reflections being systematically absent. On the lines above primitive triclinic, Denzo permutes and potentially distorts that unit cell to obey the symmetry-mandated requirements for each subsequent Bravais lattice, with highest symmetry toward the top. The percentage value is the degree of distortion required. Values in green required relatively little distortion and those in red require more distortion and are unlikely to be correct. Of the pairs of cell dimensions the top one is the permuted cell and the bottom one is that cell after it is "clamped" to the symmetry requirements. For primitive tetragonal, which is what the unit cell is for lysozyme as an example, the requirements are a=b and $\alpha=\beta=\gamma=90°$.

For weak auto-indexing results it is best to refine the auto-indexing result in primitive triclinic first and then re-open the table using the Bravais Lattice button in the Index tab. Success can be influenced by the parameters for Index Peaks Limit and Sigma Cutoff Index in the first line of the Controls panel within Index.

Be aware that the Bravais lattice table is responding only to the *location* of diffraction spots in the frame and not their intensity. While the highest symmetry choice is often the best one – for tetragonal lysozyme the point group is 422 with a primitive lattice so primitive tetragonal is indeed the correct choice - a minority of crystals show pseudo-symmetry and the correct assignment of point group and unit cell dimensions only becomes obvious during scaling. At which point you must revisit the Bravais Lattice table and re-integrate the data.

To refine the initial indexing result (e.g. in Primitive Triclinic) you select Fit Basic, refine a few times, then Fit All refine a few times more and then revisit the Bravais Lattice table for your best guess at the actual crystal symmetry/lattice system.

For integration you basically only need to produce a good correlation between predicted and observed diffraction maxima - some changes in Bravais lattice can be accommodated in scaling even if you have integrated it as the lowest symmetry primitive triclinic. However, it often stabilizes refinement to have the correct Bravais lattice assigned, and it avoids you having to think about complicated transformations (and applying them) in Scalepack later on.

To optimize the predicted vs observed maxima you need to refine the initial auto-indexing parameters. The boxes toggled in the Refinement Options panel at the left show what parameters are being refined the next time you press the Refine button. Some eigenvalue-filtering hopes to reduce the correlations between parameters but the idea here is to Fit Basic until convergence, add in Distance and perhaps Fit All with Mosaicity unchecked, followed by Fit Basic with Mosaicity checked, followed by Fit All again. Mosaicity estimates are sometimes problematic for crystals with streaky spots, and if the refined mosaicity climbs too high it will likely cause problems with spot overlaps (spots whose centroids are too close to each other on the frame).

For weak data you would probably do well to decrease the sigma cutoff for data in refinement, as given in the Refinement box in the Control panel to the right.

If you are unhappy with the spot predictions and wish to discard the current parameters you will need to press the Abort Refinement button to go back and do peak searching or indexing again.

Once you've started refining parameters, XDISP changes to display the predictions of spot centroids based on the refined parameter values - toggle this using the Update Pred button to see if the predictions (yellow, green, red) match the actual spot locations. Yellow centroid markers are *partial* reflections - reflections that are clipped at one or both ends of their profile during the frame and whose diffraction intensity is spread over multiple frames. Green reflections are *full* reflections that complete their entire passage through diffraction condition during the frame. You only see a significant number of fulls on thicker frames, particularly at higher resolutions. Reflections with centroids that are red have issues - highly variable background, overloads (unlikely with this detector), spots that are too close to each other ("overlaps").

If the predicted spot centroids match the observed ones well enough you can then move to the next tab

- **Strategy** - to calculate the runs necessary to collect a complete data set. Again, you will have to use existing images to assess the likely limit of diffraction based on your current exposure time, although of course you can adjust this during the data processing step.

Strategy

You will be using Rigaku's strategy program for the Pilatus detector rather than HKL3K's own strategy program since the former is aware of the multi-axis goniostat. Strategy makes use of the unit cell, space group and crystal orientation that you have determined during auto-indexing. As such, unless you really know what your space group is then the strategy is really an educated guess that must be verified during scaling.

After auto-indexing, click on the **Strategy** tab and change the space group as necessary - inherently you are making a decision on point group rather than actual space group at this stage. By default Denzo will opt for the lowest-symmetry space group/point group within the Bravais Lattice that you selected. In the case of tetragonal lysozyme it is space group *P4* in point group 4 (Laue group 4/m). Tetragonal lysozyme is actually space group *P43212* in point group 422 (Laue 4/mmm) which affects the strategy since 422 is higher symmetry than 4. If you do not know what your space group is, stick to the default. The worst that can happen is that you collect more data than you need.

Click Rigaku Strategy to launch the program. Click Yes I Want To Make Changes. In the sidebar are options for completeness (recommendation: 98% or greater), redundancy (recommendation: 3 or greater), resolution etc. Click Calculate Strategy to get the program to come up with something.

Data Processing

In a perfect world a data strategy program would guarantee that your data collection will result in complete, redundant data without you having to check. As with synchrotron data collections we *strongly* advocate checking the progress of your data collection in real time, and with the Pilatus system it is a lot closer to real time than the RAXIS system. Crystals on the home source do not show damage anywhere near as fast as at a synchrotron (at least a few days in the beam) but incomplete data is useless for most tasks and there is no excuse for collecting it.

Denzo will pick up goniostat values from the frame header, so if your crystal remains in the same position on the goniometer head during the different runs you should be able to use the same relative indexing - this is particularly important in point groups 3, 32, 4, 6 to avoid the relative indexing problem. Otherwise integrate each “sweep” separately, and subsequently scale the individual sweeps together.

Go back to the **Data tab** and select the runs you want to include in data processing, likely excluding the test images you took when assessing data quality or assessing a strategy. Index the data and refine the parameters using the **Index Tab** in exactly the same way as outlined above.

Check again the correspondence between predicted and observed spot centroids. Yellow and green reflection centroids are partials and fulls, respectively. Red reflections are ones with a problem: overloads; overlaps (spots that are too close); spots with badly behaved backgrounds (too close to module boundaries etc). If that looks OK be sure to check the spot integration parameter on the lower left of the Index tab under the Integration Box panel. Denzo, like other refinement programs, uses learnt profile fitting to provide a better estimate of weak data by estimating weak spot profiles by generating empirical spot profiles from nearby strong spots. The Profile Fitting Radius sometimes needs to be increased for particularly weak or anisotropic data or Denzo will refuse to integrate weak data with insufficent nearby strong spots to estimate the profile from. Only strong spots are used to construct spot profiles.

To get a look at the spot size, first click on Zoom Wind button in XDISP to open the zoom window, then middle-mouse click somewhere on the master image to center the view in the zoom window. Click on Zoom In a few times to zoom in. Click on Int Box to show the integration box. The square is the background box that Denzo fits a least-squares plane through to subtract the background from the spot prior to integration. It is OK for those boxes to overlap - that's controlled by the Box Size parameter in the Index tab. The box needs to be several times the size of the spot to get a reliable background estimate. The yellow circles are the spot centroid positions (yellow=partial) and inside these are two concentric circles around the spot. The central one is controlled by the Spot Size parameter in the Index tab. *If you change these parameters you have to do one cycle of Refinement to update the view in XDISP*. Caution: if you make the spot sizes too large, adjacent spots will overlap each other and Denzo will reject these spots as "overlaps". The spot centroids show up as red in this case.

If you have got the parameters refined so that the predicted spot centroids match reality, you have selected the correct Bravais lattice, and you have tweaked the spot size parameters as necessary you can start integration. However if your crystal does not diffract to the edge of the detector there is not much point integrating fresh air - you can limit the resolution for integration via the Resolution panel in the top left corner. Conversely, if you have really strong data you could integrate into the Half-Corner or Corner of the image, although you would get far better data coverage if you moved the detector closer (if possible) or moved the detector further out in 2θ. You will need to do one cycle of refinement, again, if you change the resolution limits.

Go to the **Integration Tab** to start integration. Generally speaking most of the parameters that affect integration are actually on the Index tab. But if you particularly want to fix Mosaicity during integration (e.g. for smeared spots or crystals with other types of large mosaicity issues) then this is the place to select it - in the Controls panel at top left.

Check that the data runs listed in the table at top left are all the ones you want to integrate, then press Integrate. If this is not the first time you have integrated the data you will be prompted if you want to overwrite the ".x" files or create a new subdirectory. I nearly always do the former. Denzo writes a file with extension ".x" that contains the spots that are integrated on a particular image, as well as the current integration and crystal parameters. Scalepack then reads all these .x files to scale and merge the data.

During integration Denzo will happily wait for frames that are being collected - just make sure it is not hanging up at the end of data collection waiting for frames that will never arrive. Denzo will display the position residuals (chi^2) for detector X and Y between predicted and observed spot positions. Something less than 1.5 is desired here. The "Integration Information" panel at lower right will show you positional chi^2, unit cell parameters etc during integration. It is not all that uncommon for one or more parameters to drift during integration since the initial indexing was done of one frame and the parameters are refined *locally* off one or a very small number of frames - they are not estimated over a large angular range. Nevertheless you should not see mosaicity or distance wander very much (although mosaicity can be anisotropic). If they do you might consider fixing them during integration.

Most of what goes on in the integration tab is not all that exciting in terms of monitoring data quality, but you still want to make sure those residuals are small.

Scaling

Scaling your data is where you start to find out what your intrinsic data quality is, so we need to define this: you want data that is *complete*, that is of high precision, and has a reasonable redundancy (aka multiplicity) so that the statistics are reliably estimated and the scaling procedure is stable.

Inherently a number of semi-empirical parameters are refined to minimize the intensity differences between observed reflections that should have the same intensities based on symmetry. This requires an expectation of what the error *should be* based on the strength of the data and of what the likely point group symmetry is.

Symmetry: In the example of tetragonal lysozyme the primitive tetragonal Bravais lattice can accommodate two different point groups, 4 and 422. The former has a single four-fold symmetry axis and the latter also has two-fold symmetry axes perpendicular to the four-fold. Alternatively speaking: point group 4 has eight symmetry-equivalent reflections in a complete sphere of data and 422 has sixteen of them (assuming Friedel's Law holds - no anomalous scattering). By picking a space group that is compatible with the Bravais lattice you intrinsically assign a point group to the data: space group P43 is in point group 4 and space group P43212 is in point group 422.

Friedel's Law: Friedel's Law applies in the case where anomalous scattering contributions are very small. It introduces a center of symmetry in the data, so that reflections *(h,k,l)* and *(-h,-k,-l)* have the same intensity. Elements C, H, N, O have minimal anomalous scattering at CuKα wavelength (1.54Å) but S, I, Lanthanides do not. For some well-diffracting proteins you can actually use the very small sulphur anomalous scattering signal to determine the structure but for weakly diffracting proteins you can mostly ignore the presence of this signal.

Scaling: Scalepack is the program that does the scaling. There are some behind-the-scenes analytical corrections for Lorentz and polarization factors, air absorption etc but then Scalepack refines a per-frame scale factor (k) and a per-frame B-factor (B) as a primary method of reducing systematic differences in the data - i.e. the differences in measured intensities of reflections that *should be indentical based on point group symmetry*. As k,B are expected to be smoothly varying they are often restrained during scaling. If you turn on absorption correction it will additionally refine an empirical absorption correction parameterized as spherical harmonics. The per-frame correction factors model things that vary primarily with image number, of which primary absorption and scattering effects due to the variation of the volume of the crystal in the beam are the main one, and a B-factor term that potentially models the impact of radiation damage on the crystal (fall-off in scattering with respect to resolution). In reality things are not quite that clean, since anisotropic scattering will turn up in part in the per-frame B-factor, and longer-term variations in e.g. detector response or beam intensity get folded into the per-frame scale factor. An absorption correction that is additional to this "k+B" scaling also serves to reduce errors and is doubtless analogous to the existing methods in XSCALE(XDS)/

SCALA/AIMLESS. In any event, all scaling methods involve exploiting the redundancy in your data as a method for estimating the quality of the scaling result is going to improve as the redundancy of your data improves. The estimates of the mean intensity of the *unique* reflections also improves with increased redundancy.

Quality Estimates: R_merge is a simple statistic that reflects the proportional deviation of reflections that should have the same intensity based on symmetry or when measured multiple times. Its synonym is R_sym. R_merge is systematically under-estimated for data with low redundancy and approaches its "true" value only in the case of high redundancy, which is inevitably a higher R_merge. This presents the ironic situation where low-redundancy R_merges are lower than high-redundancy R_merges for the same data, despite the expectation that the high-redundancy data has better-estimated intensity averages. To deal with this, R_pim was introduced, which is the precision indicating merging R factor. It describes the precision of the averaged intensity measurements and provides the standard error of the mean. In addition, R_meas or R_rim is used, which is the redundancy-independent merging R factor and gives the precision of individual intensity measurements independent of multiplicity. The *useful* high resolution cutoff is a matter of active debate but probably data can be considered useful up to R_meas values possibly above 80% but the statistic using CC1/2 is more useful. CC1/2 is the correlation coefficient between data randomly split between two halves of a data set. Current rules-of-thumb suggest that one should include data with CC1/2 higher than 0.5 Here again, however, CC1/2 is not going to be accurately estimated if you have no redundancy in your data.

Postrefinement: Under the Global Refinement section of the scale tab there is a variety of options that fall under the general heading of postrefinement (so named because it occurs *after* the data integration step, but "global" is perhaps a more informative name). Recall that during indexing and during integration the refinement of parameters - particularly unit cell dimensions - the refinement is quite *local* and does not consider the whole of diffraction space. Parameter refinement may quite easily fall into false local minima during minimization. Postrefinement - activated if you select anything above the bottom option - considers the location of diffraction maxima on all the frames at once, so in particular it should provide much more accurate unit cell dimensions for downstream use. It also provides more accurate estimates of the Lorentz correction which can be large for low resolution reflections that lie near the rotation axis. Usually you want to do some form of postrefinement even if it is not immediately obvious, which of the options is most appropriate.

The actual process of scaling is easier to comprehend than the more theoretical discussion above. Select the **Scale tab**. Make sure the data runs you want to merge are included in the "Pending Sets" list. Click on the toggles for Scale Restrain, B Restrain, Absorption Correction (Low), Write Rejection File and over on the right Small Slippage Imperfect Goniostat. Leave Use Rejections on Next Run unchecked. Pick the low resolution cutoff as 25Å since the beam stop is fairly small on this new system. If you've done any prior scaling click the "Delete Reject File" button and then at the top left of the button array in Controls, click Scale Sets.

What you have done is scale all the data runs lumped together, minimizing the discrepancies between symmetry-related reflections. It will take a few seconds to do this and then the GUI will parse the Scalepack log file and populate the graphs in the upper selection of the scale tab with a summary of the Scalepack log file. You can view the actual file (sometimes easier) using "Show Log File". But the second table from the top - "Global Statistics" - is a pretty good summary.

By default Index (and therefore Integrate) will select the lowest-symmetry point group consistent with the Bravais lattice you selected in the Index step. Assuming you do not have pseudo-symmetry or twinning this is a good choice for the initial scaling run (which would have been done in P4 in this case) and then you can explore how changing the space group changes the scaling statistics. Changing the point group from 4 to 422 might change them a lot (e.g. from space group P4 to P422) but changing between P4 to P41 will only make a minimal difference - same point group and P41 differs from P4 only by having some *(0,0,l)* reflections *systematically absent*. P41 differs from P42 in the pattern of systematic absences and P43 has the same pattern as P41 because they are enantiomorphic space groups where the only difference is the direction of the screw axis. So while chosing space groups be aware that different space group choices *within* a point group basically have the same scaling statistics. **The best way to tell the actual space group is after the scaling step when the structure is solved**.

Since you can solve the structure via S-SAD using this data I should have clicked on the "Anomalous" button in the middle column so it would output *(+h,+k,+l)* and *(-h,-k,-l)* without merging them.

Scroll down in the graphical data tables to see some statistics with resolution or by frame number. At left here is the scale factor and B-factor by frame, which you expect to move around smoothly but perhaps jump a little at the boundaries between data runs. At the bottom there is a button for Exclude Frames to allow you to omit some especially troublesome frames, but you can also go back to the Data tab and deselect the Scale checkbox for any runs you want to delete in their entirety. At right is the all-important completeness vs resolution table - if you want to make any real use of your data you want it to be > 85% but there is usually little excuse for it being anything < 95% unless it is in space group P1. The color-coding is by redundancy and the explanation for the scheme is via the Explanation button, as it is for most graphs.

Scalepack's error model expects that the Chi^2 should be 1.0 for data whose sigmas are estimated correctly. Inherently this is comparing what you would *expect* R_merge to be vs what the observed I/σ(I) actually is, and changing the scale of the σ(I) until the Chi^2 is 1.0. You do this via the Error Scale Factor just above the button stack - if Chi^2 is >1 increase error scale factor by 0.1 at a time until Chi^2~1. And reduce Error Scale Factor if Chi^2 is <1. You can also adjust the error scales in resolution shells via the Adjust Error Model button. Programs like AIMLESS and XSCALE do all this for you, so SCALEPACK is needlessly manual in the adjustments. If you are doing this step you **do** want to turn on Use rejections on next run to get the best estimate of final scaling statistics, but since this excludes outlier reflections from scaling, and the outlier criteria are based on σ(I) you need to Delete Reject File and do 2-3 cycles of scaling if you change Error Scale Factor significantly. You can get obsessive about adjusting the error model so that Chi^2~1 but the most important thing is to get it relatively close.

The other table is for R_merge and R_pim vs resolution. Best to click Show Log File and look at the last table on the log file to see what CC1/2 and R_pim are in the outermost shell and adjust the resolution cutoff accordingly.

The final steps are: adjust your high resolution cutoff; adjust error scale factor to make Chi^2~1; check your best-guess for space group; Delete Reject File and do 3-5 cycles of scaling to converge on the final output; go solve your structure.

HKL3000 is not an Expert System

The HKL3000 approach needs a substantial amount of manual intervention to optimize the results of data processing. A number of programs such as Mosflm, XDS and Xia2 allow for autoprocessing when run using default values with some shell scripts making decisions about the best possible parameters for data processing.

HKL3000 allows structure solution using a workflow for SHELX (HKL2MAP), CCP4 (CCP4I2) and PHENIX via their respective GUIs. Of course, if available, each of the listed program packages can be used on its own for the purpose of structure solution and refinement. The needed files for export are **output.sca** and **scale.log** from the processing directory.