Decision Making in xia2

G. Winter,[a] C.M.C. Lobley[a] and S.M. Prince[b]

[a]Diamond Light Source, Harwell Science and Innovation Campus, Oxfordshire, UK, and [b]Faculty of Life Sciences, Manchester Interdisciplinary Biocentre, Manchester, UK

(Received 0 XXXXXXX 0000; accepted 0 XXXXXXX 0000)

automation; **data reduction**; **expert system**

Abstract

xia2 is an expert system for the automated reduction of macromolecular crystallography (MX) data, employing well trusted existing software. The system can process a full MX data set from images to intensities and structure factor amplitudes with no user input. To achieve this many decisions are made, the details of which are described in this manuscript. In addition it is critical to have a flexible framework which can support the testing of hypotheses and allow feedback from later stages in the analysis to the points where earlier decisions were made. The essence of this framework is also presented here. While the decision making protocols described were developed for *xia2*, the are equally applicable to interactive data reduction.

1. Introduction

Careful reduction of data is key to the success of the diffraction experiment [1]. This is typically an interactive process, with the crystallographer making use of prior knowledge about the project and data reduction tools with which they are familiar. Over recent years, however, the increase in throughput of macromolecular crystallography (MX) beamlines has made it difficult to interactively process all data while close to the experiment, giving rise to a need for automated data analysis software. This need inspired the development of *xia2* [13].

A key part of any automated analysis must be the implementation of the decision making process. In interactive processing the decisions are based on the user's experience and advice from progrm authors. These decisions include selection of images for the initial characterisation of the sample; testing indexing solutions; selection of processing parameters; handling of resolution limits; selection of pointgroup and selection of a suitable model for scaling. In the development of *xia2* it was necessary to take a systematic approach to building up this expertise, and a study was undertaken using data from the Joint Centre for Structural Genomics[1] (JCSG.) The outcomes of this investigation will be described here, along with the expert system *framework* in which the decision making was embedded to develop the final data reduction tool, *xia2*.

1.1. Workflow

Taken from the top level, the workflow of data reduction for macromolecular crystallography (MX) can be considered as three phases: (i) characterisation and (ii) integration of data from individual sweeps, followed by (iii) scaling and merging of all sweeps taken from a given crystal. Decisions made at earlier stages have implications for subsequent stages, and information from the later stages may contradict those earlier decisions. Any system to automate data reduction must therefore be sufficiently flexible to manage this situation gracefully, considering all decisions made as hypotheses that are subsequently tested. The system must also express the workflow being performed to give the structure into which to embed the decision making expertise.

1.2. Blueprint

For any given choice there may be a set of optimum parameters for a specific example data set. There may however also be a set of default parameters which can be shown to generally work well with limited knowledge of the problem in hand, for example a general “class” of data sets. These *guidelines* may be determined from a systematic analysis of a number of data sets, which should ideally be free of artefacts, e.g. split spots, sample misalignment and ice rings. This criterion highlighted the value of data from structural genomics programs, where the published data are well characterised (i.e. the structure has been solved) and have been collected in a consistent manner. This has the benefit that the decision making may focus on the crystallograhic choices, without needing to work around problems caused by poor sample quality or

[1] http://www.jcsg.org

inappropriate choices in data collection.

Finally, it is important to recognise that software for MX is constantly evolving, and that new packages will become available which ideally should be incorporated into the system. As such, some emphasis on abstraction of steps in the workflow is also helpful, to allow existing components within the system to be replaced. Currently *xia2* includes support for two main integration packages, Mosflm [10] and XDS [8] which are accessed as the 2D and 3D pipelines respectively, reflecting the approach taken to profile fitting. These are used in combination with Scala [2], XSCALE and more recently Aimless (Evans and Murshudov, this volume) as well as other tools from the CCP4 suite to deliver the final result. In addition, Labelit [11] and CCTBX [5] are also extensively used in the analysis.

2. Decision Making for Data Reduction: Characterisation

As discussed above, the crystallographic workflow is considered within *xia2* in three phases - characterisation, integration and scaling. Within each of these there will be decisions which need to be made as well as decisions as to how to handle passing information around the system, in particular from later stages to earlier.

Here, the decisions *within* each phase will be considered for each of the software packages used by *xia2*, including special cases and advice to the user about *xia2* options to use where appropriate. The objective of the characterisation stage is to determine for a given sweep of images:

- A list of possible Bravais lattice options and appropriate cell constants for each.
- Refined / updated values for the beam centre and distance.
- A lattice / cell proposal, based on an analysis of all of the possible options.

In addition, any crystal orientation matrices calculated should also be available. For a given program, the choices to make are: the selection of images to use for indexing, the thresholds to use for indexing, selection of the “best” solution and analysis to ensure that the selected solution meets the acceptance criteria.

2.1. Labelit and Mosflm

Labelit and Mosflm share the same underlying one-dimensional FFT indexing algorithm [12] though the implementation in Labelit allows for an additional search to refine the direct beam centre over a wider range. As such, the behaviour of the programs in indexing is similar, requiring only one analysis for the selection of images - though the analysis of solutions will depend on the program being used as they are scored differently. The authors of both Labelit and Mosflm recommend the use of two images for indexing, spaced by $\sim 90^\circ$ in rotation (subsequently ϕ for brevity) giving a wide coverage of reciprocal space over which to determine the basis vectors.

To optimise any process a scoring scheme is needed. Here the objective is to determine the selection of images which generally give the most accurate indexing solution. As the minimum root mean square (R.M.S.) deviation between observed and predicted spot centres is a target of refinement and the ideal absolute values of the unit cell constants are poorly defined (at least at this stage) these represent poor metrics. The metric penalty however, defined as the deviation from the constraints for each Bravais lattice [4] is appropriate as this is a test for internal consistency and is also relevent for automatic strategy systems such as EDNA [6]. With this in mind, the use of few images may be desirable.

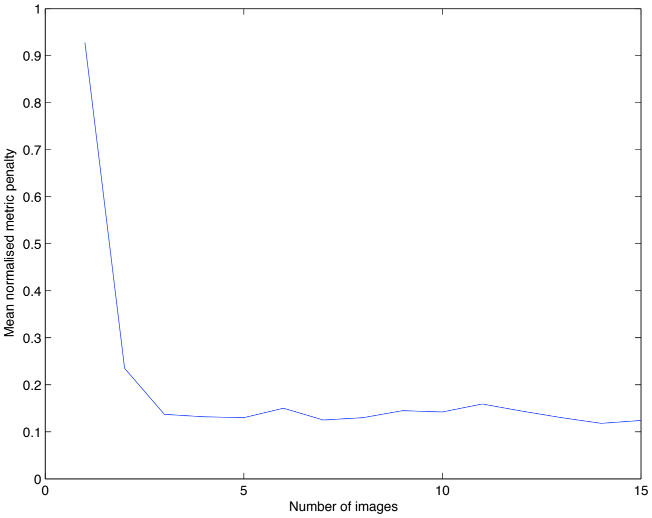

Fig. 1. Mean normalised metric penalty as a function of number of images used for characterisation with Labelit, where smaller values indicate more accurate solutions. The step from two images to three, a factor of around 1.7, is much larger than any subsequent change.

Data were taken from 86 sweeps from the JCSG archive, where the following four criteria were met: the lattice was not pseudosymmetric, nor triclinic, autoindexing with a single image gave the "correct" result and at least 90° of data were available. The resulting sweeps had resolution limits in the range 3.5Å to 1.2Å̇First, the number of images to use for indexing was considered. Figure 2.1 shows the mean normalized metric penalty, calculated as follows:

- all of the metric penalties for a given sweep were scaled over the range $0 - 1$
- then this normalized value averaged across all sweeps for a given number of images.

for one to 15 images spread across the range $0 - 90°$, where smaller values indicate a more accurate solution. Clearly the use of two images (~ 0.24) gives a substantially more accurate result than the use of a single frame (~ 0.92), confirming the advice from the Mosflm and Labelit authors. However a further improvement may be observed from using three images (to ~ 0.14), after which no further improvement is clear.

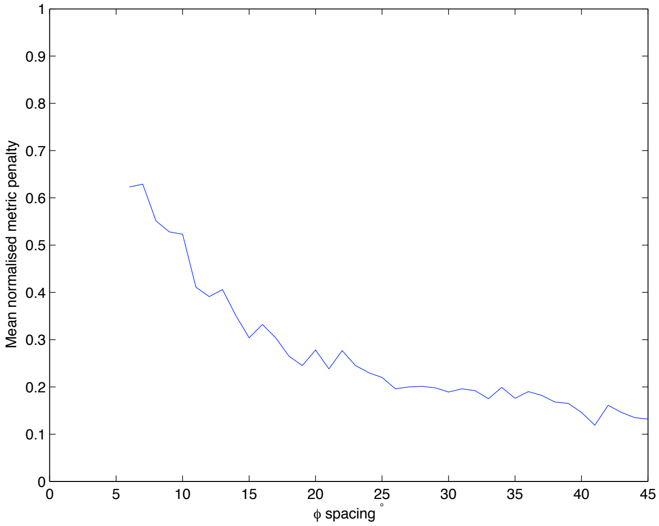

Fig. 2. Normalised metric penalty for indexing from three images, up to a maximum spacing of 45°.

Following a similar procedure it was found (Figure 2.1) that a spacing of more than $\sim 20-30^\circ$ generally gave the most accurate solution. Extending this analysis to sweeps of up to 180° confirmed this result (not shown.)

This result may be considered as follows: the one-dimensional FFT indexing procedure employed in Mosflm and Labelit computes the Fourier transform of the projection of the observed peaks gathered from all images in ~ 7000 reciprocal space directions. Those directions which show the strongest signal are then considered as possible basis vectors. Using three images spaced by $\sim 45^\circ$ ensures that every reciprocal space direction is likely to be well sampled. This will make it more likely that a fundamental basis vector, rather than some linear combination of basis vectors, will be found giving a more accurate result.

When using Labelit, the proposed solution is chosen to be the highest symmetry Bravais lattice / cell combination highlighted in the output. The Bravais lattice / cell combinations for all other lattices are also recorded, where the cell is chosen to be the one with the lowest metric penalty where multiple options exist. In usual operation Mosflm makes a selection from the possible solutions which satisfies similar constraints, which is taken by *xia2*. Once again, all possible options are saved. It is important to note that this selection of the highest symmetry plausible solution is a reasonable approach as it will be challenged in subsequent analysis steps.

2.2. XDS

Unlike Mosflm and Labelit which assume the ϕ centroid is the centre of the image, the indexing in XDS uses calculated ϕ centroids as well as positions on the image of reflections, and therefore operates most reliably when given one or more *wedges* of images. Indeed, it is perfectly possible to index with peaks taken from every image in the sweep, a process which may be desirable in some circumstances. The indexing algorithm is however less robust to errors in the direct beam position than the procedure in Labelit. The usual procedure in *xia2* is therefore to first index with Labelit and repeat this indexing with XDS using the refined beam centre and selected cell and symmetry. However this behaviour is under the user's control *via* command-line options which are listed in the *xia2* manual.

Where Mosflm and Labelit make explicit assumptions about the experimental geometry, XDS allows a general rotation axis and beam vector and will typically refine these during the indexing step. Therefore, if the intention is to use XDS for integration it is necessary to repeat the indexing of the data if Labelit or Mosflm were used for the initial characterisation. Determination of the optimum selection of images for indexing with XDS is therefore necessary.

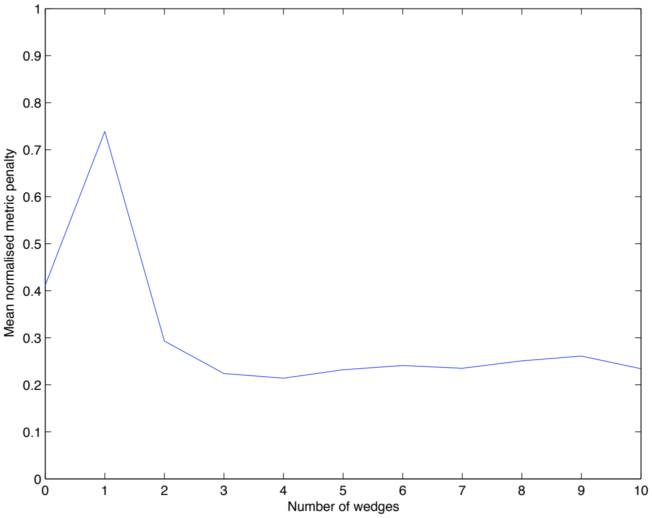

Fig. 3. Mean normalised metric penalty for the correct autoindexing solution from XDS, as a function of the number of 5° wedges used. The 0 point corresponds to the use of all images. From these results it is clear that the use of three wedges, rather than the full data set, works well.

By anology with Labelit, data were indexed with a triclinic basis from all images and from one to ten 5° wedges, with the resulting triclinic cell used to compute the metric penalty for the correct lattice using tools from CCTBX. As may be seen from Figure 2.2, use of a single 5° wedge gave the poorest results followed by the use of all images. The use of two and three wedges gave further improvement, with no improvement observed subsequently. Following from this, various wedge sizes were tested and no substantial trends, beyond using at least two frames, were found though a slight benefit was observed for using $\sim 5^\circ$ wedges (not shown.) Finally the ideal spacing was found to be $\sim 20 - 30^\circ$ (also not shown.)

In the case that XDS is used for the initial indexing, it is necessary to select a proposed solution from the 44 lattice characters output by XDS. In the analysis no solutions with a penalty higher than 40 were found to be correct, so with no guidance *xia2* will select the highest symmetry solution with a penalty lower than this threshold. If however the symmetry has been given by the user this limit is raised to 200. As with Labelit, in cases of multiple cell possibilities for a given lattice the one with the lowest penalty is chosen.

3. Decision Making for Data Reduction: Integration

The objective of the integration step is to:

- Accurately measure the intensities of the reflections.
- Validate the results of characterisation, i.e. test whether the lattice assigned in the characerisation step is appropriate.
- Provide a refined model for the experimental geometry and crystal lattice.

While these objectives have been listed in order of decreasing importance to the structure solution process, the second is perhaps most significant the development of an expert system. When processing interactively for example using iMosflm it is straightforward to see if the indexing solution is appropriate from a comparison of the observed and predicted reflection positions. When the analysis is performed with nothing in the way of visual feedback, as is the case inside *xia2*, an equivalent assessment is needed. This assessment depends on the program used, and will be discussed shortly.

3.1. Mosflm

A typical interactive integration session with Mosflm will start with indexing followed by the refinement of the cell, once the lattice has been chosen. At this stage the integration may be performed either through the GUI or from a script. However, by working through this process a great deal of information about the data has been learned by the program, for example spot profle parameters from the spot search prior to indexing. In automating this analysis, particularly when alternative programs may be used for some of the steps, some effort must be made to reproduce the program state. This, coupled with the cell refinement step, may be characterised as preparation for integration. The decision making will therefore be separated into preparation for integration and integration proper.

The preparation for integration is primarily to set up Moslfm into a suitable state for performing integration by setting the appropriate parameters. This will require the selection of images for the cell refinement and the configuration of the program state. To determine rules for the selection of images a similar process was followed to the selection of images for indexing with XDS, allowing for an additional constraint of the use of 30 frames or fewer in cell refinement, with a similar conclusion. In essence, the cell refinement was performed in P1 for the sweeps used earlier, and the resulting cell constants scored *via* metric penalty, resulting in the use of three small wedges of data spaced by ideally 45°. To robustly prepare the program state for cell refinement however it is necessary to first gather the reflection profile parameters that would usually be obtained during indexing. To achieve this an indexing step is performed, where the spot parameters and an estimate of the resolution limit are retained from the program output but the other results ignored. It was found that applying this rather conservative resolution limit during the cell refinement gave rise to much more reliable results.

To approximate the visual assessment described earlier, the cell refinement is performed with the proposed lattice from characterisation and with a triclinic basis, with the orientation matrix from characterisation transformed appropriately. When the R.M.S. deviations between the observed and predicted spot positions are compared in a pairwise manner (i.e. as a function of frame number and refinement cycle) between the proposed and triclinic lattices a ratio may be calculated. It was found that in all cases when the lattice had been correctly assigned this ratio was less than 1.5. However, in cases where the lattice had pseudosymmetry (i.e. monoclinic with $\beta \sim 90°$) the ratio exceeded this value, and 1.5 is therefore used as a cutoff. If in analysis the ratio exceeds this value the indexing solution is assumed to be incorrect. If this is the case that solution may be eliminated from consideration and the next lower symmetry lattice considered, a process which takes place within the characterisation. Other tests, such as comparing the deviation of the refined triclinic cell constants to those satisfying the metric constraints for the lattice symmetry as a function of the estimated standard deviation of those constraints was found to be unreliable. Once the preparation is complete there are a number of choices to be made for the integration proper:

- Whether to fix the cell constants during integration.
- Whether to perform the integration with the lattice constraints applied, as recommended by the Mosflm authors, or with a triclinic basis.
- Whether to apply a resolution limit during integration, as opposed to integrating across the entire active area.

Wheras the previous choices have been assessed in terms of the accuracy of the resulting cell constants, here the metric will need to relate to the accuracy of the observations, best observed by scaling the data and considering the merging statistics. Provided that the extent of the data are unchanged (i.e. same image range, same resolution limits) R_{merge} will be a reliable indicator of accuracy. To make this assessment, Pointless [2] and Scala were used, the latter with a “standard” scaling model: smoothed scaing on rotation with 5° intervals, secondary absorption correction and smoothed B factor correction with 20° intervals, the default in CCP4i. Protocols were determined for integration to assess each of the above choices, though a comparison with the procedures recommended by the program authors. In summary, it was found that fixing the cell constants during integration was helpful however applying a resolution cutoff and instead integrating across the full face of the detector made little difference, as did applying the Bravais lattice constraints. The conclusion was therefore to integrate across the active area of the detector, fixing the cell constants and applying the lattice constraints, though in hindsight there are arguments against the latter and developments are underway to perform all processing in with a triclinic basis.

3.2. XDS

Where Mosflm is typically run through the graphical user interface iMosflm, XDS is run on the command-line with a plain text input file, making it ideal for usage within an automated system. The advised use of XDS is to perform all processing with a triclinic basis, then apply the Bravais lattice constraints at the final step (i.e. in the postrefinement and scaling.) However it is possible, but by no means mandatory, to "recycle" parameters from the later stages of processing and rerun the integration. The aim here is to determine the choices which will generally give rise to the best quality results, ideally at minimal computational cost. As with the analysis of integration strategies using Mosflm, R_{merge} will be used as a metric for the quality of the data reduction. The four choices are:

- Whether to enforce the Bravais lattice constraints.
- Whether to recycle the reflection profile parameters.
- Whether to recycle the orientation matrix and experimental geometry model.
- Whether to recycle all results of postrefinement including local detector distortions.

These may be compared with the straightforward "XDS default" protocol described above. Using the same data as were used for Mosflm, it was clear that any of these changes to the strategy were "polishing" the data, with an average change in R_{merge} of around 2% of the value. The only marked improvement was found from recycling the reflection profile parameters, which define the grid for the transformation from the image to the Ewald sphere. To recycle these parameters it is necessary to reintegrate the full data set.

In processing the data with Mosflm the selection of Bravais lattice was tested *via* postrefinement. In XDS, global refinement is performed after integration allowing the lattice to be tested in a similar manner, once again with allowing a ratio of no more than 1.5 in the R.M.S. deviations. Currently in *xia2* the integration is performed with the Bravais constraints applied, for historical reasons, though performing the integration with a triclinic basis is planned. The same global refinement may then be performed.

4. Decision Making for Data Reduction: Scaling

While characterisation and integration operate on sweeps in isolation, scaling must consider all of the data at once. As such, it is at this point that all of the processed data are brought together and tested for consistency, both within individual sweeps (i.e. that the symmetry in intensity data is consistent with the Bravais lattice) and between sweeps (i.e. that the Bravais lattices are consistent and that the data are indexed in a consistent manner.) This requires careful management of derived information as well as the possibiity for feedback to earlier processing steps. To assist with this the scaling step is split into three phases - the preparation, the scaling itself and then post-processing.

Following the use of Mosflm and XDS, Scala (and now Aimless) and XSCALE, respectively, are natural for the main scaling step. However a number of other programs from CCP4 and elsewhere (including Cad, Truncate and Pointless) are used to help in this process.

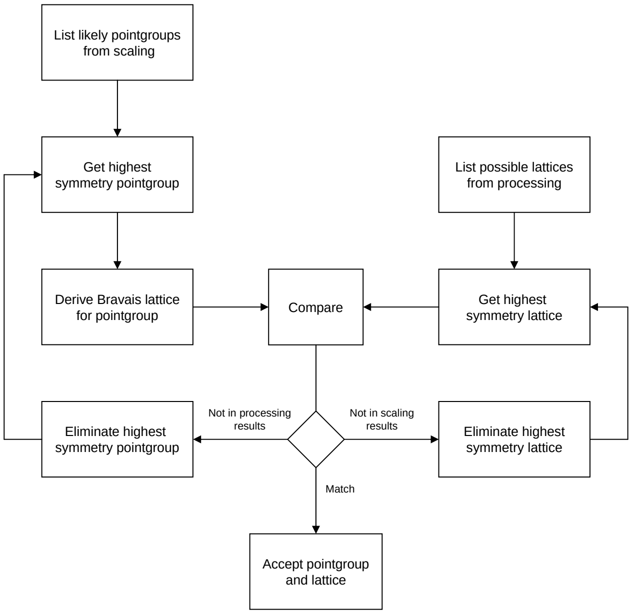
List likely pointgroups from scaling
Get highest symmetry pointgroup
List possible lattices from processing
Derive Bravais lattice for pointgroup
Compare
Get highest symmetry lattice
Not in processing results
Not in scaling results
Eliminate highest symmetry pointgroup
Eliminate highest symmetry lattice
Match
Accept pointgroup and lattice

Fig. 4. Procedure to determine the crystal point group and Bravais lattice consistent with the the diffraction data, taking input from the indexing and analysis with Pointless.

In the preparation phase the data from integration are tested for internal consistency both within and between sweeps. The first test is to determine whether the Bravais lattice used for processing is consistent with the apparent pointgroup of the data (Figure 4.1.) This test is performed by comparing the highest allowed Bravais lattice from processing with the most likely result from the analysis of intensities with Pointless. If the two are consistent (for example, pointgroup P422 with lattice tP) the test is passed immediately. If the Bravais lattice corresponding to the apparent pointgroup has higher symmetry than the allowed lattice, as may occur when a non-crystallographic symmetry axis is closely aligned with the unit cell axes, that pointgroup is ignored and the next most likely tested. If however the bravais lattice corresponding to the most likely pointgroup has *lower* symmetry than the Bravais lattice used for processing the lattice used for processing is eliminated from consideration and the data are reprocessed with the correct lattice. An example of this would be a monoclinic lattice with β angle indistinguishable from 90°.

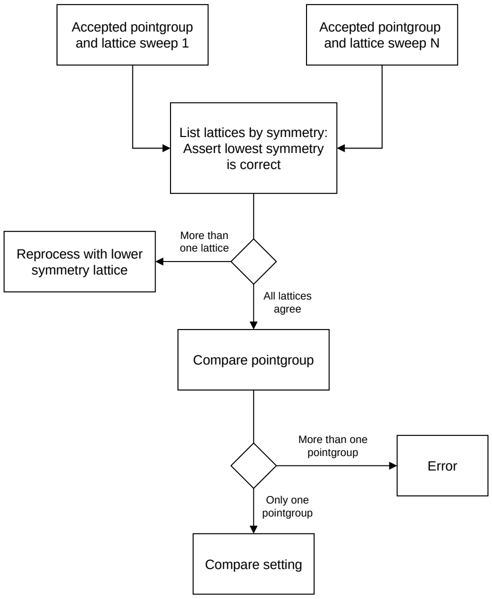
Accepted pointgroup and lattice sweep 1
Accepted pointgroup and lattice sweep N
List lattices by symmetry: Assert lowest symmetry is correct
More than one lattice
Reprocess with lower symmetry lattice
All lattices agree
Compare pointgroup
More than one pointgroup
Error
Only one pointgroup
Compare setting

Fig. 5. Procedure for combining pointgroup information from all sweeps, following on from 4.1, and assuming that the lowest symmetry lattice is correct.

Once this test has been passed for all sweeps, the data will have been processed with a lattice consistent with the pointgroup symmetry of the data. However, it may be the case (for example from a low and high resolution pass) that the conclusions for one sweep differ from those from another. In that case (Figure 4.1) the lowest symmetry Bravais lattice is assumed for all sweeps. If at this stage the pointgroup conclusions remain inconsistent an error is raised as it is likely that something has gone wrong - for example an inconsistent data set has been included in the processing.

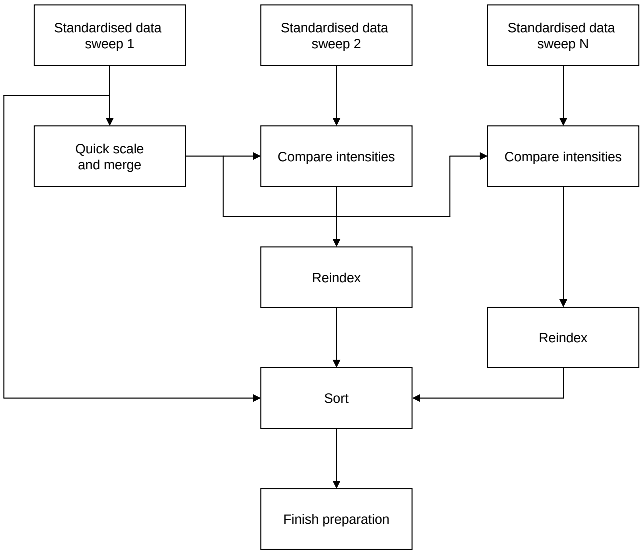

Fig. 6. Procedure to establish uniform indexing of all sweeps, following on from 4.1, which allows for intrinsic ambiguity in the origin choice where the lattice symmetry is higher than the pointgroup symmetry.

Finally, it is necessary to ensure that the data are consistently indexed to cope with cases where there may be indexing ambiguity, using Pointless (Figure 4.1.) At this stage the most likely spacegroup is also determined from the systematic absences, primarily for the benefit of the downstream sructure factor amplitudes, though often convenient for the program user. When sorting and scaling the data are put into the sweep order in which they were collected by appropriate renumbering of the batches.

4.2. Scaling with Scala

Though the scaling protocol recommended in the Scala documentation (smoothed scales over 5° spacing in rotation, absorption correction with 6 orders of spherical harmonics, $B-$factor correction smoothed over 20° spacing in rotation) works well in many cases there are examples when better results may be achieved by, for example, including the "tails" correction or excluding the decay. As such it is worthwhile to perform a search to decide the most appropriate model for scaling, which is complicted by the need for a protocol for decision making. It was found that using Scala with typical data sets there is essentially no risk of over fitting the scaling corrections, so the resulting R_{merge} was found to be a reasonable metric as the resolution limits and range of data considered are unchanged. However, this was found to give an excessive bias towards corrections for the high resolution data so the low resolution R_{merge} is also taken into account. Finally, in some cases the combination of scaling corrections may also converge badly or not at all, which is taken to be an indication of a poor scaling model. Therefore, the resulting protocol uses: scaling with only smoothed scale factors over 5° spacing in rotation to give a reference convergence rate. Then seven other protocols are tested: with and without decay, absorption and partiality correction, with the model which gives the best merging residuals while converging in less than twice the number of cycles of the simple run selected.

In terms of the parameterisation of the scaling, it was found that for typical data adjusting the rotation range for the scale and B-factors and the number of spherical harmonics used from the defaults as suggested in the documentation was of little benefit. Historically, scaling in *xia2* included an iterative process to determine appropriate correction factors for the error estimates. This procedure is now performed by Scala and Aimless automatically.

4.3. Scaling with XSCALE

Where scaling with Scala applies a parameterised model to the data in order to minimise the differences between symmetry related observations, scaling in XSCALE (and the XDS CORRECT step) apply corrections in terms of arrays of correction values which are indexed appropriately for decay, absorption and detector modulation. In this scaling, the principle is to remove correlation of the intensity values with image number, resolution and detector position, respectively. The choice of corrections to apply is determined by the user.

In the XDS CORRECT step these corrections are applied to each sweep in isolation. The same corrections are applied in XSCALE to all of the data simultaneously. It was found that performing the scaling twice gave no improvement but did effectively double the number of parameters used, so the choice made in *xia2* is to apply all of the scaling corrections in XSCALE and apply the minimum in the XDS CORRECT step. It was found that the application of all scaling factors (i.e. correcting for decay, detector modulation and absorption) systematically gives the lowest merging residual so this is the default choice in *xia2*, though a user option is available to control this choice. In scaling, all of the sweeps corresponding to each wavelenth are scaled together but not merged, and all wavelengths are scaled simultaneously.

After the initial scaling, resolution limits are calculated following the procedures set out below, which are then used for subsequent XSCALE runs. During scaling XSCALE suggests a list of reflections which do not appear to belong to the data set as a whole - “aliens” - these are subsequently excluded in an iterative manner.

After scaling the unmerged data are reformatted into their original sweep structure using tools from CCP4 and merged with Scala or Aimless, as this gives an easily interpreted report on data quality and also a route into producing MTZ files for the subsequent post processing, discussed below.

4.4. Resolution Limit Calculation

Early in the development of *xia2* the resolution limits were determined from an analysis of the output of Scala, which proved to be unreliable. Resolution limits are now calculated directly from the scaled intensity data themselves, using routines developed using the CCTBX toolbox, allowing the same procedures to be used for data from Scala and XSCALE, giving more consistent results.

The resolution limit calculations themselves use the merged and unmerged $\frac{I}{\sigma_I}$ values, the completeness and the R_{merge} as possible criteria for determining a cutoff, with the merged $\frac{I}{\sigma_I} > 2$ and unmerged $\frac{I}{\sigma_I} > 1$ by default. The cutoff applied to the unmerged data was found to be helpful when considering data with very high multiplicity - though the multiplicity provides a boost to the apparent $\frac{I}{\sigma_I}$, the resulting measurements were found in some cases to be distributed more like a normal distrbution than an exponential (i.e. Wilson) distribution as would be expected, as judged by the E^4 plot from Truncate (not shown.) This is assumed to be a result of a poor estimation of the errors for weak reflections, giving rise to systematic effects in merging. The internal correlation coefficients $CC\frac{1}{2}$ and CC^* have since been proposed as a more robust choice for resolution limit calculation (Evans & Murshudov, this volume and [9]) which will be investigated shortly.

4.5. Post Processing

From a certain perspective scaled and merged intensities are the end product of the data integration and scaling. It is however helpful to perform a small number of subsequent analysis tasks to prepare the data for downstream analysis:

- Calculation of structure factor amplitudes from intensities.
- Determination of an average unit cell.

The former of these uses the “truncate” procedure [3] where negative intensities are corrected to give small positive values. However, to best perform this correction it is necessary to remove systematically absent reflections from the data set, hence the spacegroup identification performed earlier during the preparation step. At this stage it is also helpful to get the overall scale for the amplitudes correct - if the amino acid sequence is provided, the number of molecules in the asymmetric unit is estimated [7] to give an estimate of the total number of atoms and hence the appropriate scale factor to apply. In the absence of this information it is assumed that the unit cell contains 50% solvent, with the remaining space filled with “average” protein.

While the determination of the average unit cell is not a fundamental part of the analysis, it is nevertheless helpful for downstream calculations. This is computed in *xia2* as an average from the sweeps, weighted by the number of reflections in each sweep, hence giving a bias towards higher resolution data sets.

5. Implementing Automated Data Reduction

In the previous section the decision making protocols for the data analysis were considered. These are not however sufficient to allow automated data analysis, as they need to be embedded in a framework which encodes the overall strategy for data analysis. In this section the structure of *xia2* will be described with an emphasis on the interaction between the analysis and the data.

5.1. Data Management

As an expert system can only make decisions based on the information available, careful management of data is critical. Most data familiar to macromolecular crystallographers define static information, for example coordinate files and reflection data. Within *xia2* however the knowledge at any stage is a *hypothesis* and hence subject to change following subsequent analysis. As such it is important to track the provenance of any derived information, such that subsequent invalidation of a result (for example, elimination of an indexing solution) will automatically invalidate any subsequent conclusions. This *dynamic* information must be treated carefully to ensure that results are updated when necessary.

The main information provided to the system is the raw diffraction data, with suitable metadata (i.e. image headers) to describe the experiment. In the majority of cases this information will be sufficient to build a useful model of the experiment.

Within *xia2* the raw data are structured in terms of sweeps of diffraction data, which belong to wavelengths (which are merged to a single MTZ data set in the output) which in turn belong to crystals. The crystals are contained within projects, but this is not currently considered in the analysis. Crystals are also the fundamental unit of data for scaling. All of these data structures (project, crystal, wavelength, sweep) map directly onto objects within *xia2* and initially contain only static data - that is, the information derived from image headers or from user input. However, starting from this input information it is possible to perform analysis as described earlier, and draw conclusions. Although it would be possible to store the results of this analysis in these data objects, it would be necessary to ensure that they are kept up-to-date which would generate a substantial amount of book-keeping. The alternative used inside *xia2*, is to maintain a link to the *source* of this information, which is essentially the analysis task which gave that result. To make this link in a general way it is necessary to have a standard interface to the analysis tasks.

5.2. Expert System Interfaces

As there are a small number of high level steps in the data analysis process but a number of possible program options for each, it is possible to define an abstract interface to describe each step, which can then be implemented using the existing software and the decision making protocols described above. The addition of an abstract interface also gives the option of centralising standard decision making steps, for example the management of solutions from indexing. As an example, both Mosflm and Labelit may be used to index a diffraction pattern based in a small number of images from a sweep, so both may present an *Indexer* interface. The detail of how the indexing is performed and how the results are interpreted will be program specific and implemented in *MosflmIndexer* and *LabelitIndexer*, respectively. The application of a standard interface however allows a standard two-way link to a sweep to be made, passing in generic information needed for indexing, receiving generic results from the analysis.

While this approach adds an extra degree of separation between the data and the analysis programs, it allows for generic aspects of the analysis to be centralised and the freedom to extend, including new software as it is developed.

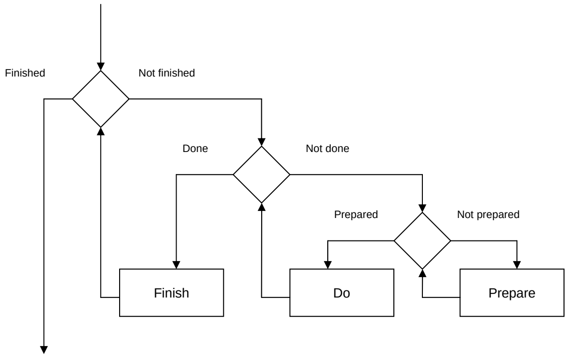

Fig. 7. General flow of expert system interfaces, showing how the prepare, do and finish functions are used. Decisions are diamonds, processing tasks as rectangles.

Within each interface (*Indexer* and *Integrater*, linked to sweeps and *Scaler* linked to the crystal) there are three standard phases - prepare, do and finish, arranged in a loop (Figure 5.2.) The status of each phase is managed internally and each phase will be performed until it is completed or an error occurs. Once the finish phase is completed any results which have been derived are assumed valid, however in requests for information the interface will check that this is the case, and if not will repeat the necessary steps. If some input information is changed this may invalidate the internal state, ensuring that next time results are requested they are recalculated taking into account this new information. While this structure is complex, the real benefit is felt through the the data management hierarchy.

5.3. Linking Data Structures and Interfaces

The benefit of having standard interfaces to all of the analysis steps, and linking those analysis steps to the data hierarchy, is that the source of a particular piece of information is clear. For example, if a sweep is asked for the unit cell and Bravais lattice it may delegate this request to it's indexer. If the data have been indexed already and the result has not been invalidated it may be returned immediately. If no indexing has been performed this may be done implicitly, in order to deliver the result. If that result is subsequently eliminated, the next request for the unit cell will give the next solution, perhaps repeating the indexing as necessary to refine this result.

There are two main outcomes of this. First, in scaling preparation, the intensities are requested from integraters which in turn request the cell and lattice information from the indexer. If in pointgroup analysis this lattice is shown to be incorrect it may be eliminated and new values requested for the cell, lattice and intensities. If necessary the full integration of the data set will be performed implicitly, giving the opportunity for sophisticated feedback without excessive code complexity. The looping structure in each interface will also ensure that this reprocessing will be performed optimally. Secondly, the full processing of the data, from indexing through to scaling of all data from a crystal, may be performed implicitly by requesting the location of the final scaled and merged intensity data from the crystal. To this end, the "main program" inside *xia2* is in effect a print statement, with all of the processing performed as a side-effect of providing the information to show in the output.

6. Example: Transketolase

With real data it becomes possible to illustrate the process of data analysis using *xia2* and to clearly identify the information provided to guide users through the decisions made. The data used here were collected at 100K at Diamond Light Source, beamline

I04-1, with a Pilatus 2M detector. The data set consisted of 1800 frames measured with 0.1° width and 0.1s exposure time. The protein crystals were of *Lactobacillus salivarius* UC188 transketolase (E.C. 2.2.1.1, Tkt), a ubiquitous enzyme responsible for the transfer of a dihydroxyethyl moiety from a ketose sugar to an aldose sugar in the pentose phosphate pathway of metabolism. Crystals were grown from protein in complex with the cofactor, thiamine pyrophosphate (TPP) and a divalent Mg^{2+} ion from magnesium chloride. These data were collected as part of a high throughput structure determination project being carried out as a collaboration between Diamond Light Source and the Oxford Protein Production Facility (paper in preparation.) As such it provides an ideal case study, since automated data integration running on the beamline as the data are collected provides an essential part of the structure determination pipeline.

The analysis of this data proceeded in a straightforward fashion from the command-line `xia2 -3d -project P100TRANSK -crystal OPPF4501 /path/to/data`. That is, to search in the directory `/path/to/data` for diffraction images, assign the project and crystal names and to use XDS for integration. The data were first indexed with a hexagonal primitive lattice, and integrated with this basis. In postrefinement, the R.M.S. deviation ratios for this lattice when compared with the triclinic were 1.02 and 1.07 for the positional and angular deviations respectively, showing that the Bravais lattice choice was consistent with the data, however the integration was repeated with an updated model for the reflection profile parameters and refined orientation. In the second pass of integration the measurements were reported to have a higher signal to noise. Pointgroup analysis with Pointless indicated $P321$ consistent with the lattice choice, and the spacegroup was assigned as one of $P3_121$ or $P3_221$ based on the systematic absence analysis. After the first round of scaling the resolution was estimated to be 2.3Å, after which the scaling was repeated and several rounds of outlier rejection performed. Finally the merging was performed with Scala, giving the statistics shown in Table 1.

Table 1. *Data collection and merging statistics for OPPF4501.*

Space group	$P3_121$ or $P3_221$		
High resolution limit	2.32	10.38	2.32
Low resolution limit	53.79	53.79	2.38
Completeness	99.9	99.2	100.0
Multiplicity	9.7	8.6	10.2
I/sigma	15.6	44.6	3.6
Rmerge	0.124	0.030	0.691
Rmeas(I)	0.131	0.032	0.727
Rpim(I)	0.042	0.011	0.225
Wilson B factor	35.438		
Total observations	270815	3301	20426
Total unique	27958	385	2000

Subsequent analysis showed a substantial gap between the R_{work} and R_{free} (~ 0.16 and ~ 0.24 respectively) in refinement. Investigation of this gap included a full repeat of the structure solution and refinement in $P1$, with the data reprocessed by simply adding `-spacegroup P1` to the command-line above. Analysis of this structure with the Zanuda server (http://www.ysbl.york.ac.uk/YSBLPrograms/index.jsp) showed that the conclusions reached by *xia2* were appropriate and the assigned symmetry and statistics valid.

7. Discussion

The field of macromolecular crystallography includes a number of experts, each with their small field of highly specialised knowledge. With the increasing demand for high resolution macromolecular structures, throughput has increased and the number of macromolecular crystallographers has increased, while the number of experts has remained relatively stable. This has created an increase in demand for “expert” software which can perform analysis on behalf of the users, to interpret the data and provide results for subsequent analysis. *xia2* provides a platform for analysis and decision making and makes extensive use of pre-existing software. By encoding decisions as hypotheses to be tested as analysis proceeds, *xia2* has the flexibility to use data from all steps in the analysis. The architecture also allows extension to include new software as an when it becomes available to adapt to the demands of crystallograhers and keep pace with new developments.

While the decision making protocols described above were developed to be embedded in *xia2*, they apply equally in interactive processing. As such these may be useful to a new student of crystallography wishing to learn more about the analysis of diffraction data.

8. Acknowledgements

The authors would like to thank Martin Walsh, the I04-1 team at Diamond and the OPPF for the example included in the manuscript, as well as the JCSG, the Division of Structural Biology at the Wellcome Trust Centre for Human Genetics, and numerous users for providing the test data which was used in developing this software. We would also like to acknowledge staff and users at Diamond Light Source for their input and feedback. The authors would like to thank Harry Powell and Andrew Leslie for help with making the best use of Mosflm, Phil Evans for help with Scala, Aimless and Pointless including specific changes to those programs, Wolfgang Kabsch and Kay Diederichs for help with XDS/XSCALE. Development work for xia2 is currently supported by Diamond Light Source Ltd., and was initially supported by the BBSRC through the e-Science pilot project e-HTPX, the EU Framework 6 through BioXHit and CCP4.

References

[1] Zbigniew Dauter. Data-collection strategies. *Acta Crystallographica Section D*, 55(10):1703–1717, Oct 1999.

[2] Philip Evans. Scaling and assessment of data quality. *Acta Crystallographica Section D*, 62(1):72–82, Jan 2006.

[3] S. French and K. Wilson. On the treatment of negative intensity observations. *Acta Crystallographica Section A*, 34(4):517–525, Jul 1978.

[4] R. W. Grosse-Kunstleve, N. K. Sauter, and P. D. Adams. Numerically stable algorithms for the computation of reduced unit cells. *Acta Crystallographica Section A*, 60(1):1–6, Jan 2004.

[5] Ralf W. Grosse-Kunstleve, Nicholas K. Sauter, Nigel W. Moriarty, and Paul D. Adams. The *Computational Crystallography Toolbox*: crystallographic algorithms in a reusable software framework. *Journal of Applied Crystallography*, 35(1):126–136, Feb 2002.

[6] Marie-Françoise Incardona, Gleb P. Bourenkov, Karl Levik, Romeu A. Pieritz, Alexander N. Popov, and Olof Svensson. *EDNA*: a framework for plugin-based applications applied to X-ray experiment online data analysis. *Journal of Synchrotron Radiation*, 16(6):872–879, Nov 2009.

[7] Kantardjieff K.A. and Rupp B. Matthews coefficient probabilities: Improved estimates for unit cell contents of proteins, dna, and protein-nucleic acid complex crystals. *Protein Science*, 12:1865 – 1871, 2003.

[8] Wolfgang Kabsch. *XDS*. *Acta Crystallographica Section D*, 66(2):125–132, Feb 2010.

[9] P. Andrew Karplus and Kay Diederichs. Linking crystallographic model and data quality. *Science*, 336(6084):1030–1033, 2012.

[10] AGW Leslie. Recent changes to the MOSFLM package for processing film and image plate data. *Joint CCP4 and ESFEACMB Newsletter on Protein Crystallography*, 26, 1992.

[11] Nicholas K. Sauter, Ralf W. Grosse-Kunstleve, and Paul D. Adams. Robust indexing for automatic data collection. *Journal of Applied Crystallography*, 37(3):399–409, Jun 2004.

[12] I. Steller, R. Bolotovsky, and M. G. Rossmann. An Algorithm for Automatic Indexing of Oscillation Images using Fourier Analysis. *Journal of Applied Crystallography*, 30(6):1036–1040, Dec 1997.

[13] G. Winter. *xia2*: an expert system for macromolecular crystallography data reduction. *Journal of Applied Crystallography*, 43(1):186–190, Feb 2010.

Synopsis

The basis for decision making in the program *xia2* is described, also giving guidelines which can be applied in interactive data processing.