Acta Crystallographica Section D
Biological Crystallography
ISSN 0907-4449

Automated data processing on beamline FIP (BM30A) at ESRF

J.-L. Ferrer

IBS J.-P. Ebel CEA-CNRS, 41 Rue Jules Horowitz, 38027 Grenoble CEDEX 1, France

Correspondence e-mail: ferrer@lccp.ibs.fr

Automation of protein crystallography synchrotron beamlines is becoming necessary to face challenging structural genomics projects. In this context, a program has been developed that processes diffraction frames using popular software but analyzes statistics and makes choices the way crystallographers usually do. This program includes the classical peak search, indexing, integration, scaling and anomalous signal analysis. The result, comparable with that obtained by standard users, is rapidly available, providing the required information for a more efficient use of the beam time.

Received 22 June 2001
Accepted 13 August 2001

1. Introduction

Protein crystallography experiments can be highly automated: when performed on a given beamline, they use similar settings and involve the same steps. This automation effort, including robots for sample mounting and online data processing, has two sources of motivation. Firstly, it makes the experiment safer by reducing sample loss during the mounting process and preventing collisions of the diffractometer. Secondly, it is a way to improve the efficiency of the beamline. The predicted increase in requirement for synchrotron beamtime arising from structural genomics projects and the slow growth of available synchrotron beamlines may lead to a high demand on existing stations. Presently, on FIP, the French beamline for Investigation of Proteins (Ferrer *et al.*, in preparation), there have already been requests for twice the available beam time.

For these reasons, we have focused on automation of our experiment (beamline FIP-BM30A at the European Synchrotron Radiation Facility). So far, loop centering, data collection (including management of beam loss, reoptimization, wavelength changes in MAD experiments, chaining of several data collections and management of disk space) are fully automated. Study of a robot for sample mounting has also started. At the other end, we are developing a tool, named *ADP* (automated data processing; package available at the web address http://www.esrf.fr/exp_facilities/BM30a/User_guide/Data_processing/crystal.htm), for indexing, integration, scaling and analysis of data during the data collection without any intervention from the user. Despite the fact that this tool uses well known software for key operations, mostly *HKL* (Otwinowski & Minor, 1997), *MARHKL* (Klein, unpublished work), *CCP*4 (Collaborative Computational Project, Number 4, 1994) and *STRATEGY* (Ravelli *et al.*, 1997), it is based on a new approach in the way it automatically prepares all required scripts and makes all required decisions normally performed by users.

2. Generalities

ADP can be launched by the user or directly by the data-collection software *XNEMO* in a specific directory containing a parameter file (auto.par). This parameter file, created by *XNEMO*, specifies the file name, oscillation range and any required information to start processing. Only two parameters, the space group, assumed unknown and further evaluated by *ADP*, and the presence of anomalous signal, may need to be modified by the user to save time. During the processing, the description of the current step is documented in an html file with links to the command, log, statistics and graphics files created, including an anomalous difference Patterson map when anomalous signal is present.

3. Data-processing strategy

The first step of the processing is the peak search. This step is performed with the *MARPEAKS* program. During this process, the threshold of peak selection is iteratively reduced as long as the number of selected peaks is insufficient. When the peak search has succeeded, the log file is analyzed to evaluate the average spot size, which is used further for the determination of the box size in the inte-

Table 1
Comparison of a typical data set (3.8 Å resolution), processed by a crystallographer using *HKL* and *ADP*.

Each data processing is illustrated with statistics calculated by *SCALEPACK*. χ^2 is the weighted sum of $(I - \langle I \rangle)^2/\varepsilon^2$ corrected for the correlation between I and $\langle I \rangle$, where ε is the error model.

	User	*ADP*
Average I	3356.8	3281.8
Average $\sigma(I)$	285.7	259.9
Norm. χ^2	0.451	0.552
Linear R	0.125	0.128
Square R	0.083	0.090
% of reflections with $I/\sigma(I)$ less than		
0	6.9	6.1
1	23.1	21.4
2	34.4	32.6
3	42.5	40.5
5	53.6	51.5
10	69.3	68.0
20	84.1	83.4
>20	15.4	16.2
Total	99.5	99.6

gration process. This estimation of the box size is similar to that implemented in the *MOSFLM* program (Leslie, 1992).

The second step is the indexing, performed with *DENZO* (Otwinowski & Minor, 1997). The selected space group is the first one with a good score (currently, below a threshold estimated empirically: a more sophisticated analysis will be introduced in the future). If the user knows the space group, the space group guessed by *ADP* is provided for information but *ADP* uses that given by the user. The mosaicity is then estimated by iterative integration of the first frame, increasing the mosaicity parameter as long as the mosaicity histogram calculated by *DENZO* is truncated. This method is close to that currently implemented in *MOSFLM* for the mosaicity estimation: in this case, the mosaicity is iteratively increased as long as the integrated intensities of predicted frames does not reach a plateau (Leslie, 2001). After indexing, *STRATEGY* is run in order to calculate the completeness expected at the end of the data collection, as well as the frame number where the completeness will be high enough to run a first scaling.

The next step is the integration. This is performed frame by frame, waiting for the next one to be collected, decompressing the current frame if required and taking refined parameters from the previous frame to run *DENZO* on the current frame. If the next frame is not complete after a timeout of 10 min, integration is aborted and *ADP* switches on scaling, assuming data collection has been aborted. At the end of integration, χ^2 values are plotted for information.

Scaling is performed with *SCALEPACK* (Otwinowski & Minor, 1997) from the *HKL* package. Before starting, the data-collection log file is analyzed in order to avoid adding partials through data-collection interruptions. The scaling is iterated (with no merging of symetrical reflections if 'anomalous = yes' in the parameter file) until the number of rejections converges. The highest usable resolution is then evaluated as the limit where $I/\sigma(I) > 1$. A final scaling is run using this condition. Systematic extinctions are then analyzed along $h = k = 0$, $h = l = 0$ and $k = l = 0$ axes. This is used subsequently to guess the right space group, including screw axis, prior to the last scaling run.

When scaling is performed, the *SCALEPACK* reflection file is translated to an mtz file using the *SCALEPACK2MTZ* program from the *CCP*4 package (Collaborative Computational Project, Number 4, 1994). Intensities are then translated into structure factors using *TRUNCATE* (Collaborative Computational Project, Number 4, 1994) and assuming an average protein density for the calculation of the number of residues per asymmetric unit.

The last step in this automated data processing is the analysis of anomalous signal (if 'anomalous = yes' in the parameter file). This analysis is performed in different ways: (i) comparison of merged and unmerged statistics, (ii) the same comparison corrected by redundancy bias, (iii) the statistics of merging F^+ and F^-, as suggested in the *HKL* documentation. *SCALEIT* (Collaborative Computational Project, Number 4, 1994) is then run to evaluate the maximum acceptable anomalous differences and *FFT* and *NPO* (Collaborative Computational Project, Number 4, 1994) are used for the calculation of the Patterson anomalous map, including data to a resolution where $I/\sigma(I) > 2$.

All this processing is now fairly robust, with more than 90% of data sets properly processed and a few structures have now been solved by the MAD method after reduction of data with *ADP*. The source of most of the observed failures comes from wrong indexing arising from either very weak diffraction or poor frame quality (strong ice rings or smearing spots).

4. Conclusions

Using *ADP*, crystallographers can have their data processed within a few minutes after the end of their data collection. In this way, decisions for the next data collection can be faster and safer, leading to a more efficient use of beam time. The quality of this automated processing is sufficient for a quick evaluation of the data and is at least comparable to that carried out by a standard user (see Table 1). Perhaps the most important reason for using *ADP* is the rising number of projects expected in the structural genomics context. Automation of each step of a structure-resolution process, including the data reduction, will permit an increase in the number of projects per crystallographer. Presently, several MAD data sets collected on FIP and reduced with *ADP* are at the end of a successful phasing process.

I acknowledge the assistance provided by R. Kahn for writing this program and by C. Zubieta for correction of the manuscript.

References

Collaborative Computational Project, Number 4 (1994). *Acta Cryst.* D**50**, 760–763.

Leslie, A. G. W. (1992). *CCP4/ESF–EACMB Newsl. Protein Crystallogr.* **26**.

Leslie, A. G. W. (2001). *ESRF Newsl.* **35**, 36–38.

Otwinowski, Z. & Minor, W. (1997). *Methods Enzymol.* **276**, 307–326.

Ravelli, R. B. G., Sweet, R. M., Skinner, J. M., Duisenberg, A. J. M. & Kroon, J. (1997). *J. Appl. Cryst.* **30**, 551–554.