

[Submitted to "Crystallographic Methods" on 2003-07-07]

Quality control and validation

Gerard J. Kleywegt

Department of Cell and Molecular Biology, Uppsala University, Biomedical Centre, Box 596, SE-751 24 Uppsala, Sweden.

Degree: PhD

E-mail: gerard@xray.bmc.uu.se

Phone: +46 - 18 - 471 48 70

Fax: +46 - 18 - 53 69 71

Keywords: electron density; model bias; model building; protein structure; quality control; Ramachandran plot; refinement; structure; validation; X-ray crystallography

Running title: Quality control and validation

Spelling: British English

Abstract

This chapter discusses two important aspects of the structure-determination process that are related to the accuracy and reliability of the model under investigation. Quality control is defined as the analysis of an intermediate model to identify aspects of it that are unusual in some sense and that could therefore be due to errors in the model building or refinement process. Any such errors need to be fixed, if at all possible, prior to analysis and publication of the model. Validation is the process of assessing the reliability of the final model (or certain aspects of it, *e.g.*, the active site residues) that is about to be analysed, published, deposited and possibly used in follow-up studies.

1. Introduction

Before a model of a macromolecule can be analysed in the light of its biological function, one needs to have a thorough understanding of its reliability, both in terms of the overall structure and of its details [1-4]. The results of such an assessment should be reflected in the subsequent analysis. For instance, if a 3.5 Å structure is described, it will in general be inappropriate to discuss hydrogen bonds, let alone to list their lengths with a precision of 0.01 Å. To assist users of the model (biologists, enzymologists, medicinal chemists, *etc.*), the final published model should be validated, *i.e.*, its validity (reliability, accuracy, completeness) should be assessed and expressed quantitatively. A related task is quality control of intermediate models. This entails analysis of the model to identify any features that are unusual in some sense (*e.g.*, in terms of geometry, temperature factors, or fit to the density). Whereas validation is usually concerned with overall statistics (*i.e.*, pertaining to the entire model), quality control is typically performed at the level of individual residues and small molecules (ligands, ions, *etc.*). Nevertheless, the tools used are often the same. For instance, whereas the percentage of outlier residues in the Ramachandran plot conveys information about the global quality of a protein model, each of the individual outlier residues will be the object of scrutiny in a quality control exercise.

Quality indicators are statistics that are calculated from the model, the data, or both, and

that provide information about the quality of the model or the data, or of the fit of the model to the data. Assessment of the quality of the data is really a part of the data-processing stage (see chapter 19) and will not be considered further in this chapter. Quality indicators can be classified in various ways. For instance, calculation of some quality indicators requires only the model as input (*e.g.*, deviations from ideal geometry), whereas others require both the model and the data (*e.g.*, the R-value). Some indicators measure global properties (*e.g.*, the free R-value), whereas others are local (*e.g.*, side chain conformations). Finally, some indicators pertain to properties of the model that are “orthogonal” to the properties that have been refined (*e.g.*, the unrestrained torsion angles χ_1 and χ_2), whereas others reflect aspects of the model that have been used (explicitly or implicitly) in the refinement process (*e.g.*, bond lengths and angles). The former kind of quality indicators are sometimes called “strong” since they provide independent support for the reliability of the model, whereas the latter are “weak” and only verify that the refinement program has done its work properly [3]. Obviously, strong quality criteria are much more informative for validation purposes than weak ones (although outliers of weak criteria should be scrutinised carefully). For quality control purposes, the objects of interest are the individual residues and small molecules (ligands, solvent molecules, salt ions, cofactors, inhibitors, *etc.* [5]), and hence *strong* and *local* quality indicators are the most useful. For validation of the final model, on the other hand, *strong* and *global* indicators are the most useful. However, many local indicators also provide information about the global quality of the model through their averages, variances, Z-scores, fraction outliers, *etc.*

The purpose of quality control is to identify residues or small molecules that are unusual in some sense. It is important to realise that unusual model aspects (outliers) can be one of two things: either a genuine *feature* of the *structure*, or an *error* in the *model*. In many cases, the distinction can only be made when the experimental data (in the form of an electron-density map) is available, and this is precisely the task of the crystallographer in the quality control process. Any errors need to be fixed using the tools in the modelling program, a process that is known as rebuilding, and that usually proceeds concurrently with the quality control operation.

In this chapter, the choice of software tools and procedures is heavily, but inevitably,

biased by local customs and personal experience and preferences. However, alternatives to the programs and protocols are provided in the Notes section. The chapter covers quality control and validation from the point of view of a practising crystallographer who is refining or about to analyse and publish a structure. The emphasis is on quality control and validation of protein models, although many of the tools are equally applicable to nucleic acids and other molecules. The theory behind individual quality criteria (Ramachandran analysis, real-space fit scores, rotamer fits, *etc.*) is not discussed here. The reader is instead referred to an extensive compendium of quality criteria [3] and the references to the primary literature provided therein.

One aspect of validation that is beyond the scope of this chapter is validation by non-experts of models retrieved from public databases, using publicly available (and often web-based) tools. This topic has, however, been discussed elsewhere recently [6]. In addition, there is a web-based tutorial available on this subject (<http://xray.bmc.uu.se/gerard/embo2001/modval>). Other useful references include [2, 7-12].

2. Materials

For a list of related websites, see **Table 1**. These, and all other links in this chapter, have also been collected at a web page (<http://xray.bmc.uu.se/gerard/supmat/qualcont>). This web page also provides suitable example files for readers who wish to test the methods described here. For alternatives to some of the programs mentioned below, see the cited notes.

1. Atomic model. This is usually the product of the previous refinement cycle, but can also be obtained from elsewhere, *e.g.*, from colleagues or the Protein Data Bank (PDB) [13, 14]
2. Experimental structure factor data (amplitudes or intensities, experimental sigmas, test-set flags), either one's own, or obtained from elsewhere
3. CCP4 (Collaborative Computational Project Number 4) phase and map calculation software (see Note 1) [15]
4. WHAT IF (or WHAT_CHECK) validation software (see Note 2) [16]
5. MAPMAN software (see Note 3) [17]

6. OOPS2 software (see Note 4) [18]
7. O validation and model building software (see Note 5) [19]
8. Computer(s) to run the various software programs (see Note 6)

3. Methods

3.1. Electron-density maps

A model and two or more electron-density maps are the basic inputs to the validation (and model-building) software, and both are usually obtained as output from a cycle of refinement (see chapter 26). If the refinement program REFMAC [20] is used through the CCP4i interface [21], it can be instructed to produce two σ_A -weighted maps [22], one with coefficients $(2mF_o - DF_c, \sigma_{calc})$ and another with coefficients $(mF_o - DF_c, \sigma_{calc})$. The former shows positive density features for properly placed atoms that are already in the model and for strong features that have not yet been included (*e.g.*, missing loops, ligands, waters). The latter provides information about features that have not yet been included but should be, about features that have been included but should not be, and about atoms that need moving. As yet unmodeled features will show up with positive density values, whereas features that ought to be removed will show up as negative density. Finally, a combination of positive and negative density near an atom indicates that the atom should be moved in the direction of the positive density (but see Note 7).

An issue that tends to be confusing is that of the selection of the appropriate contour level for the various maps. The so-called σ -level of a map is simply the root-mean-square (RMS) value of the electron density in the entire unit cell (or asymmetric unit). Maps of the type $(2mF_o - DF_c, \sigma_{calc})$ are conventionally contoured at a level of “1 σ ”. In order to identify atoms that contain many electrons (sulfur, phosphorous, metals, chloride, *etc.*) it is also useful to contour this map at a higher level (*e.g.*, 4 σ) in a different colour. However, there are indications [23] that it may actually be a good idea to contour at different levels in different parts of the cell,

depending on the local average temperature factor of the model.

It is unfortunate that many crystallographers (and, indeed, referees) fail to recognise that the σ -level of $(mF_o - DF_c, \sigma_{calc})$ maps is a meaningless quantity. In the beginning of a refinement, when the model is incomplete and full of (random and other) errors, there is so much signal in such maps that a “2 σ ” feature could well be highly significant. However, toward the end of the refinement process, when the model is largely complete and correct, this type of map will be essentially flat (indeed, flatness of the $mF_o - DF_c$ difference map is a criterion for assessing the convergence of a refinement) and contain only random noise. In this extreme, even a “4 σ ” feature need not necessarily be significant. The notion that each and every “3 σ ” feature in an $mF_o - DF_c$ difference map must be significant is a fallacy, and engenders a deluge of false water molecules (*i.e.*, noise peaks misinterpreted as water). The best way to prevent this is to avoid the notion of σ -levels altogether, at least for $mF_o - DF_c$ difference maps. Instead, if all maps are calculated on an absolute scale (*i.e.*, in terms of electrons per cubic Ångström), the $(mF_o - DF_c, \sigma_{calc})$ map could be contoured at, for instance, one or two times the σ -level of the corresponding $(2mF_o - DF_c, \sigma_{calc})$ map (both positive and negative levels should be contoured). Any features visible at such levels would most likely be significant at any stage of the refinement procedure [23].

Apart from σ_A -weighted maps, many other types of maps can (and should) be used when appropriate. For instance, in the presence of non-crystallographic symmetry (NCS) or multiple, non-isomorphous crystal forms, averaged maps [24] can reveal features that may not be visible in the density of any of the individual subunits, or help resolve ambiguities in the tracing. If there is doubt about the tracing of a molecule, or when a poor molecular replacement model is rebuilt and one suspects model bias, various kinds of omit maps (regular, simulated annealing, or complete) can be used [25-27]. If an experimental map (MAD, MIR, SIRAS, *etc.*) is available, it should be consulted and checked at later stages as this map is guaranteed to be unbiased by the model that is being constructed. Similarly, an anomalous difference map may be useful to pinpoint or verify the location of anomalous scatterers in the model (*e.g.*, sulfur or selenium atoms).

3.2. Validation with WHAT IF

Validating a model with WHAT IF (or WHAT_CHECK; see Note 2) [16, 28-30] is simple: all that is needed is to start up the program and to type:

```
check ful model.pdb x y
```

where “*model.pdb*” needs to be replaced by the name of the PDB file that contains the current model. The program produces a number of files, but the most important one will be called “*pdbout.txt*”. This file contains an extensive report (in English) that should be checked carefully (see the guide for crystallographers at <http://xray.bmc.uu.se/usf/whatif.html>). However, the file can also be input to the program OOPS2 (see below), and this program will extract the information about individual residues that have been flagged by WHAT IF.

3.3. Validation with O

Assuming that the program O (version 9.0.0 or newer) is used both for calculating several residue-based quality indicators [31, 19] and for rebuilding [32, 33], a number of things need to be done (see Note 8):

1. Start up the program (without any arguments) and answer all questions by hitting the return-key until the graphics window appears. A new, empty O database has now been created. Save this database to a file, *e.g.*:

```
save model.o
```

2. Read in the current model, *e.g.* with the following command:

```
pdb_read model.pdb mod y y
```

This will read in the file “*model.pdb*”, name this model “*mod*” inside O, strip any hydrogen atoms and save any X-PLOR or CNS segment identifiers.

3. Select the molecule, define its cell constants and space group, and draw an all-atom model of it:

```
mol mod
```

```
sym_setup mod ; p212121
```

zone ; end

and subsequently centre the molecule on your screen using the mouse.

4. Define the maps that are to be used during rebuilding as well as their contouring levels (see Note 9). In this example, a $(2mF_o - DF_c, \sigma_{\text{calc}})$ map is used that is in a file called “*2mfo_dfc.map*”, that will be called “*2fofc*” inside O, and for which the symmetry operators for space group $P2_12_12_1$ should be used (see Note 10). Subsequently, the contour drawing radius for this map is set to 20 Å, the line type to solid, and two contour levels are defined: 1 σ to be drawn in blue and 4 σ in cyan. In a similar fashion the $(mF_o - DF_c, \sigma_{\text{calc}})$ map is defined (but see Note 11):

```
fm_file 2mfo_dfc.map 2fofc p212121  
fm_set 2fofc 20.0 solid 2 1.0 blue 4.0 cyan  
fm_file mfo_dfc.map fofc p212121  
fm_set fofc 20.0 solid 2 3.5 green -3.5 red
```

5. Calculate the pep-flip value for every amino-acid residue (a measure of how unusual the residue’s peptide plane orientation is), and save the resulting O data block in a text file (e.g., “*pepflip.o*”) for use with OOPS2 (see Note 12). Assuming that the first and the last amino-acid residues are called “A2” and “C135”, respectively, the following commands need to be issued:

```
pep_flip mod a2 c135  
write mod_residue_pepflip pepflip.o ;
```

6. Calculate the rotamer side chain fit score for every amino-acid residue (a measure of how unusual the side chain conformation of the residue is), and save the resulting O data block in a text file (e.g., “*rsc.o*”) for use with OOPS2:

```
refi_init mod  
refi_gen mod ;  
rsc mod a2 c135  
write mod_residue_rsc rsc.o ;
```

7. Calculate real-space fit values (a measure of how well each residue fits its surrounding

density) for all residues, ligands, water molecules *etc.* (for alternatives, see Note 13). This requires, besides the model, a ($2mF_o - DF_c$, σ_{calc}) map, a definition for every residue type of the atoms that should be included in the calculations (see Note 14), and values for some parameters (see Note 15). You can choose to calculate the real-space fit value as an R-value (“*rfact*” in the command below) or as a correlation coefficient (in that case, type “*cc*” instead):

```
rs_fit mod a2 c135 2fofc all rfact  
write mod_residue_rsfit rsfit.o ;  
stop
```

3.4. Validation and macro generation with OOPS2

OOPS2 (the successor of the slightly more cumbersome program OOPS [18]) is a program that has multiple purposes in the quality control process (see also its on-line manual at http://xray.bmc.uu.se/usf/oops2_man.html). First, given a model, it can carry out a few quality checks of its own (*e.g.*, Ramachandran plot analysis). Second, it can read and parse the results of validation tests carried out with other programs (most notably, WHAT IF, WHAT_CHECK and O). Most important, however, is the fact that its output facilitates a systematic (and educational !) approach to the quality control and rebuilding process. The output consists of a residue-by-residue critique of the model (both in a text file that can be edited during the subsequent manual model rebuilding session, and in an HTML file that can be posted on a website) and a set of macros for the program O. These macros are small files, one for each residue, that contain instructions for O to centre on a residue, possibly draw the local electron density, and to print a list of aspects of the residue that are unusual (*e.g.*, an unusual bond angle, unsatisfied hydrogen-bond donor, close contact, or outlier in the Ramachandran plot).

1. Before OOPS2 can be run, a subdirectory called “*oops*” needs to be created (if it does not exist yet), *e.g.*, using the Unix command:

```
mkdir oops
```

2. Now start up the program and answer the questions: do you want OOPS2 to print statistics and histograms, do you want it to generate some plot files, what is the name of your molecule in

O, and what is the name of your model's PDB file? OOPS2 will read your model, print some information about it and then present you with a menu of options (see **Table 2**). By issuing one of these commands, you include the corresponding quality check in the validation process. By issuing the same command again, the check will be excluded again. Most commands require some additional input, although the default values are usually appropriate. In the following, the most frequently used commands will be discussed briefly (more details can be found in the on-line manual at http://xray.bmc.uu.se/usf/oops2_man.html).

3. PEP: the program will prompt you for the name of the file that contains the O data block with the pep-flip values (e.g., "*pepflip.o*"). If you opted for statistics and histograms to be listed, you will be presented with information about the distribution of the pep-flip values. Finally, you are asked to supply a cut-off value - any residue whose pep-flip value exceeds the cut-off will be considered an outlier, and will thus be brought to your attention during the OOPS2-guided inspection or rebuilding of the model. For most models, a cut-off value of 2.5 Å is suitable. OOPS2 will list the number and percentage of outliers (the latter number could be included in a table with validation statistics). Once the protein model does not change too much any longer, the pep-flip check can be omitted, although it should be included again in the final rebuilding round.

4. RSC: provide the name of the data block with rotamer side chain fit values (e.g., "*rsc.o*") and a cut-off value (typically, 1.0 Å). As with the pep-flip check, once the protein model is essentially finished, this check can be omitted, although it should be carried out again in the final rebuilding round.

5. RAM: this command carries out a Ramachandran analysis using the definition of Kleywegt & Jones to define outliers [33]. Optionally, OOPS2 can also flag proline residues with unusual ϕ torsion angles, as well as any non-glycine residues with positive ψ values. Furthermore, residues that are in a left-handed helical conformation can be flagged.

6. WIF: provide the name of the file created by WHAT IF (normally, this file will be called "*pdbout.txt*"). OOPS2 will read this file and parse the results of more than two dozen of the WHAT IF quality checks.

7. RSR: the real-space R-values can be extracted either from a data block file created by O, or from a file created by MAPMAN (see Note 13). Select the appropriate option and provide the corresponding filename. You also need to supply a cut-off value - all residues whose real-space R-value exceeds this cut-off will be flagged by OOPS2. This cut-off value is usually chosen as one or two standard deviations above the average real-space R-value.
8. CCA: this command can be used if you calculated real-space correlation coefficients instead of R-values. The CCM and CCS commands can be used if you also performed the calculations separately for main chain and side chain atoms.
9. BFA: provide a lower and upper threshold for temperature factors (and a separate upper threshold for water molecules if you like). Residues that contain at least one atom with a temperature factor below the lower, or above the upper threshold will be flagged by OOPS2.
10. OCC: similar to the BFA command, in that any residues that contain atoms with unusual occupancies can be flagged by OOPS2. Since some or all atoms in residues with alternative conformations will have non-unit occupancies, using a lower cut-off value of, say, 0.99 is a simple way to get OOPS2 to flag all such residues and hence bring them to your attention during the rebuilding session (see Note 16).
11. PRE: the current model can be compared to (any) previous model. Residues that have undergone important changes (or that have newly been inserted or mutated) will be flagged by OOPS2. Changes in position, temperature factor, occupancy, as well as main chain and side chain torsion angles can all be taken into consideration.
12. When all the appropriate checks have been included, the GO command needs to be issued. Any user-defined criteria can be included at this stage (not normally used). You are also asked to provide a line of O commands that you want to execute for every residue that has been flagged by OOPS2 (see Note 17). Finally, you are asked three more questions about the O macros that OOPS2 will produce, but the default answers are almost always appropriate.

The program will now consider every residue in turn and check if it was flagged (*e.g.*, by WHAT IF, or by the real-space fit check, *etc.*). If so, a message will be printed and an O macro will be generated in the “oops” subdirectory (the macro has the same name as the residue). An

example of such a macro is shown in **Table 3**. Some of the other files that are created at this stage are:

- “*oops.omac*” - this is the main O macro that you need to execute when you start up O again and want to go on a journey along all residues that were flagged by OOPS2.
- “*mod_oops.html*” - this contains a residue-by-residue critique of the model in HTML format.
- “*mod_rebuild.notes*” - a similar critique, but as a simple text file (see **Table 4**). You may find it useful to edit this file as you rebuild your model so as to keep track of the changes you make to the model, any observations you make about the structure, *etc.*

3.5. Inspection of unusual model aspects

Inspection of the unusual aspects of the current model is a simple matter. First, start up O again, and provide the name of your previously created O database file (*e.g.*, “*model.o*”). Second, execute the main O macro created by OOPS2:

```
@oops.omac
```

This macro prints a lists of all the quality checks that were included, and then executes the macro it created for the first flagged residue. For this residue, it will execute the user-defined commands (*e.g.*, to draw the maps), and print a list of criteria that are violated or unusual for this residue. It is then up to you, the crystallographer, to inspect the model and the density and to decide if action is warranted, and if so, what kind of action. This is the process of model rebuilding, which falls outside the scope of this chapter. Unfortunately, even though model rebuilding is a very important part of the structure determination process, there is precious little literature addressing this issue [33].

Once inspection and possibly rebuilding of a residue is finished, the macro to proceed to the next flagged residue can be executed by clicking the appropriate command on the O user menu (this will be called something like “@oops/a3”, with “a3” being the name of the residue in question in this example).

3.6. Validation statistics

Validation of the final model is essentially an exercise in collecting quality-related statistics from a wide variety of sources. Here, a number of statistics and their usefulness for validation purposes are discussed. For an extensive discussion (including references to the primary literature) of quality indicators, see [3].

To assess the quality of the fit of the model to the data, both the conventional and the free R-value [34, 35] should be reported. In addition, some description of the real-space fit should be included, either qualitatively (*e.g.*, “poor or no density was observed for residue 12 to 21 as well as the side chains of residues 47, 98 and 132”) or, preferably, quantitatively. The average, standard deviation, and extremes of the real-space R-value or correlation coefficient can be reported in a table. However, plots of either quantity as a function of residue number are far superior in conveying the information. Such plots should be provided to users of the model as well as referees, even if they are not included in the actual manuscript.

To assess the quality of the model *per se*, results of strong global quality checks need to be reported. For proteins, these include the quality of the Ramachandran plot (including a reference to the definition that was used to derive the score, *e.g.* [30, 36-38]) and some measure of the “regularity” of the fold such as a profile score [39, 40] or the average DACA score [28]. In addition, the percentage of residues with unusual pep-flips or side chain conformations can be reported. The WHAT IF report file (“*pdbout.txt*”) contains a number of useful (strong and global) statistics under the heading “*structure Z-scores*”. If NCS is present, many statistics can be used to express the degree of similarity of the models [41] (pertaining to positions, torsion angles, and temperature factors) and even of their densities [42].

Other statistics that are often reported, but that do not necessarily convey much information about the correctness of a model, include RMS deviations from ideal values of geometric quantities (bond lengths, bond angles, *etc.* [43-45]), average temperature factors, number of atoms or refined parameters, coordinate error estimates, *etc.*

Finally, it should go without saying that if a model leads to a scientific publication, both the model and the experimental data should be deposited in the public structural database, the PDB [46].

4. Notes

1. Other programs that can be used to calculate electron-density maps include X-PLOR [47], CNS [48], TNT [49], XtalView/Xfit [50] and SHELX [51].
2. Many other programs exist that produce validation information, but we find that WHAT IF is the most comprehensive of these (for an annotated listing of its quality checks, see <http://www.cmbi.kun.nl/gv/pdbreport/checkhelp/>). A subset of WHAT IF that only contains the validation functionality is available free of charge under the name WHAT_CHECK, and this program can in principle be used instead of WHAT IF. Alternative programs include PROCHECK [36], DDQ [52], MolProbity [38], Verify3D [39, 40], ERRAT [53, 40], MOLEMAN2 [54], and NUCHECK (specifically for nucleic acid models) [55].
3. MAPMAN is used to convert electron density maps between different formats (*e.g.*, to convert CCP4 or CNS maps into O format). It can also be used to calculate residue-based real-space fit statistics (see Note 13).
4. OOPS2 (and its predecessor OOPS) generate and use residue-based quality information to generate a set of macros for O. When executed, these macros will take the crystallographer on a journey along all residues that are unusual in some sense or other.
5. Alternatives for O as a model building program include various derivatives of FRODO [56], XtalView/Xfit [50], and Quanta [57]. Note, however, that OOPS2 only works in conjunction with O.
6. Nowadays, almost all crystallographic software can be run on comparatively inexpensive personal computers running Unix-like operating systems such as Linux and Mac OS X, equipped with graphics cards to enable the use of interactive graphics programs such as O.
7. Positive and negative peaks in such maps may also have other causes. In general, positive density indicates that the model contains too little scattering matter locally. This could also be caused by the temperature factors being too high, or even misassignment of an atom type (*e.g.*,

oxygen instead of sulfur, or sodium instead of potassium).

8. For more information about the syntax, usage or purpose of specific O commands, please consult “A-to-Z of O” (http://xray.bmc.uu.se/alwyn/A-Z_of_O/A-Z_frameset.html).

9. In fact, you will probably want to execute these commands every time you start up O with your current database. A simple way to accomplish this is to use a text editor to create a small file in your directory which you call “*on_startup*” and which contains these commands (and any others you want to execute automatically when you start up O).

10. It is important to keep in mind if the map that you use comprises an asymmetric unit or unit cell, or whether it contains some other part of the cell (*e.g.*, cut out around the molecule). The former type of map is strongly preferred in O. In that case, you can provide the name of the space group, and O will use the appropriate symmetry operators to determine the density values outside the part of space that is covered by the map. Moreover, the σ -level of the map will be that of the unit cell. If you use maps that contain an arbitrary part of space, you should not provide the space group name, since map expansion may fail. In addition, the σ -level of such a map is not equal to that of the unit cell and hence contour levels need to be adjusted accordingly. For instance, if the RMS density level in the unit cell is $0.36 \text{ e}/\text{\AA}^3$, and that in a map carved out around the molecule is $0.30 \text{ e}/\text{\AA}^3$, then the latter must be contoured at a level of $0.36/0.30 = 1.2$ “ σ ” in order to portray the density at the unit-cell RMS level.

11. The contour levels that O uses are expressed in terms of the RMS density values in the corresponding map file. In order to draw the $(mF_o - DF_c, \sigma_{\text{calc}})$ map at a level that is equivalent to the σ -level of the $(2mF_o - DF_c, \sigma_{\text{calc}})$ map, we need to divide the absolute σ -level of the latter by that of the former. For instance, if the σ -level of the $(2mF_o - DF_c, \sigma_{\text{calc}})$ map is $0.344 \text{ e}/\text{\AA}^3$ and that of the $(mF_o - DF_c, \sigma_{\text{calc}})$ map is $0.097 \text{ e}/\text{\AA}^3$, then the proper contour level for the latter in O is $0.344/0.097$ which is $\sim 3.5 \sigma$ units. Note that the RMS (or σ) level of a map is printed by O when you open the map with the “*fm_file*” command (look for a line that says “Min, max, sigma”).

12. The steps involving the calculation of pep-flip, rotamer side chain fit and real-space fit values can also be carried out with the help of an O macro

(ftp://xray.bmc.uu.se/pub/gerard/omac/pre_oops.omac). Note that each of these commands automatically creates (or overwrites) a data block in O's database. The name of the data block will be the name of the molecule ("*mod*", in the example"), followed by the string "*_residue_*" and finally the name of the property (e.g., "*pepflip*").

13. Real-space fit calculations can also be carried out with the program MAPMAN (use its *RS_fit* command) and can then still be used with OOPS2 (see the on-line MAPMAN manual at http://xray.bmc.uu.se/usf/mapman_man.html). Note that this calculation requires two maps as input, namely a $(2mF_o - DF_c, \sigma_{calc})$ and an (F_c, σ_{calc}) map, and that both maps must cover the model. Other programs that calculate real-space fit values are CNS and SFCHECK [58].

14. The atoms that are to be included in the real-space fit calculations are defined in the major dictionary file for O, "*stereo_chem.odt*". If there are any entities in your model that are not yet in the dictionary file, they must be added to it. However, in future versions of O this will not be necessary any longer.

15. In order to calculate real-space fit values, O needs to compute a map based on the atomic model alone and compare that to the external map that you provide (see [59, 60] and references cited therein). There are two parameters in these formula that may need adjusting, call *A0* and *C*. The default values for these parameters (0.9 and 1.04, respectively) tend to work well if the resolution is $\sim 1.8 \text{ \AA}$. At different resolutions, these parameters can be optimised as follows:

- find a residue that fits the density very well and one that fits very poorly (by eye or using the default values of *A0* and *C* and calculating the real-space correlation coefficient).
- *A0* can usually be kept constant at ~ 0.9 .
- vary the value of *C* between 0.5 and 1.2 in steps of 0.05 and calculate the real-space correlation coefficient for the zones around the good and the bad residues (include two residues at both sides, e.g., if "*a69*" is your good residue, do "*rs_fit mod a67 a71*").
- the best value for *C* is that which gives the largest difference between the real-space fit values for the good and the bad residues.

Once you have obtained proper values for the parameters *A0* and *C* (for instance 0.9 and 0.8), you can set them to these values either by using the "*rsr_setup*" command in O or, quicker, by

typing:

```
db_set_dat .rsr_real 8 8 0.9
```

```
db_set_dat .rsr_real 7 7 0.8
```

16. To check the labelling, positional degeneracy and summed occupancies of atoms that occur in multiple conformations, you can use the program MOLEMAN2 (use the “*PDb SANity*” command). See the on-line manual for more details (http://xray.bmc.uu.se/usf/moleman2_man.html).

17. It is usually easiest to execute an O macro instead of typing all desired commands here. This macro is simply a text file that you create and that contains all those commands (*e.g.*, to draw the maps, to draw a sphere of residues, to save your database, to generate symmetry-related molecules, *etc.*). In that case your line of O commands is simply an “@”-sign followed by the name of your macro file (*e.g.*, “@*foreach.omac*”). This macro file could contain commands such as:

```
fn_draw fofc
```

```
fn_draw 2fofc
```

```
sym_sph mod ; 10
```

```
bell
```

```
save
```

Acknowledgements

The author would like to thank Emma Jakobsson for assistance with some refinement and map calculations, Sara Nystedt for acting as guinea-pig and trying out the methods described in this chapter, and Alwyn Jones, Emma Jakobsson and Sara Nystedt for useful comments on the manuscript.

The author is a Royal Swedish Academy of Sciences (KVA) Research Fellow, supported through a grant from the Knut and Alice Wallenberg Foundation. He is supported by KVA, Uppsala University, the Linnaeus Centre for Bioinformatics, and the Swedish Structural Biology Network (SBNet).

References

- [1] Brändén, C.I. and Jones, T.A. (1990). Between objectivity and subjectivity. *Nature* **343**, 687-689.
- [3] Kleywegt, G.J. (2000). Validation of protein crystal structures. *Acta Crystallogr.* **D56**, 249-265.
- [4] Davis, A.M., Teague, S.J. and Kleywegt, G.J. (2003). Application and limitations of X-ray crystallographic data in structure-based ligand and drug design. *Angew. Chem. Int. Ed.* **42**, 2718-2736.
- [2] Kleywegt, G.J. and Jones, T.A. (1995). Where freedom is given, liberties are taken. *Structure* **3**, 535-540.
- [5] Kleywegt, G.J., Henrick, K., Dodson, E.J. and van Aalten, D.M.F. (2003). Pound-wise but penny-foolish ? How well do micromolecules fare in macromolecular refinement ? Submitted.
- [6] Kleywegt, G.J. and Hansson, H. (2003). Retrieval and validation of structural information. In "Structural Proteomics", A. Edwards, M. Sundström & M. Norin, Eds., Marcel Dekker, New York. Submitted.
- [10] EU 3-D Validation Network (1998). Who checks the checkers ? Four validation tools applied to eight atomic resolution structures. *J. Mol. Biol.* **276**, 417-436.
- [9] Kleywegt, G.J. and Jones, T.A. (1995). Braille for pugilists. In "Making the Most of Your Model", W.N. Hunter, J.M. Thornton and S. Bailey, Eds., SERC Daresbury Laboratory, Warrington, 11-24.
- [11] Laskowski, R.A., MacArthur, M.W. and Thornton, J.M. (1998). Validation of protein models derived from experiment. *Curr. Opin. Struct. Biol.* **8**, 631-639.
- [7] MacArthur, M.W., Laskowski, R.A. and Thornton, J.M. (1994). Knowledge-based validation of protein structure coordinates derived by X-ray crystallography and NMR spectroscopy. *Curr. Opin. Struct. Biol.* **4**, 731-737.
- [8] Zou, J.Y. and Mowbray, S.L. (1994). An evaluation of the use of databases in protein

- structure refinement. *Acta Crystallogr.* **D50**, 237-249.
- [12] Laskowski, R.A. (2003). Structural quality assurance. *Structural Bioinformatics*, 273-303.
- [13] Bernstein, F.C., Koetzle, T.F., Williams, G.J.B., Meyer Jr, E.F., Brice, M.D., Rodgers, J.R., Kennard, O., Shimanouchi, T. and Tasumi, M. (1977). The Protein Data Bank: a computer-based archival file for macromolecular structures. *J. Mol. Biol.* **112**, 535-542.
- [14] Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N. and Bourne, P.E. (2000). The Protein Data Bank. *Nucleic Acids Res.* **28**, 235-242.
- [15] Collaborative Computational Project, Nr. 4 (1994). The *CCP4* suite: programs for protein crystallography. *Acta Crystallogr.* **D50**, 760-763.
- [16] Hooft, R.W.W., Vriend, G., Sander, C. and Abola, E.E. (1996). Errors in protein structures. *Nature* **381**, 272-272.
- [17] Kleywegt, G.J. and Jones, T.A. (1996). xdlMAPMAN and xdlDATAMAN - programs for reformatting, analysis and manipulation of biomacromolecular electron-density maps and reflection data sets. *Acta Crystallogr.* **D52**, 826-828.
- [18] Kleywegt, G.J. and Jones, T.A. (1996). Efficient rebuilding of protein structures. *Acta Crystallogr.* **D52**, 829-832.
- [19] Jones, T.A., Zou, J.Y., Cowan, S.W. and Kjeldgaard, M. (1991). Improved methods for building protein models in electron density maps and the location of errors in these models. *Acta Crystallogr.* **A47**, 110-119.
- [20] Murshudov, G.N., Vagin, A.A. and Dodson, E.J. (1997). Refinement of macromolecular structures by the maximum-likelihood method. *Acta Crystallogr.* **D53**, 240-255.
- [21] Potterton, E., Briggs, P., Turknburg, M. and Dodson, E. (2003). A graphical user interface to the CCP4 program suite. *Acta Crystallogr.* **D59**, 1131-1137.
- [22] Read, R.J. (1986). Improved Fourier coefficients for maps using phases from partial structures with errors. *Acta Crystallogr.* **A42**, 140-149.
- [23] Kleywegt, G.J., Harris, M.R., Zou, J.Y., Taylor, T.C., Wählby, A. and Jones, T.A. (2003). The Uppsala Electron Density Server (EDS) - a touch of reality. Submitted.

- [24] Kleywegt, G.J. and Read, R.J. (1997). Not your average density. *Structure* **5**, 1557-1569.
- [25] Bhat, T.N. (1988). Calculation of an OMIT map. *J. Appl. Crystallogr.* **21**, 279-281.
- [26] Hodel, A., Kim, S.H. and Brünger, A.T. (1992). Model bias in macromolecular crystal structures. *Acta Crystallogr.* **A48**, 851-858.
- [27] Vellieux, F.M.D. and Dijkstra, B.W. (1997). Computation of Bhat's OMIT map with different coefficients. *J. Appl. Crystallogr.* **30**, 396-399.
- [29] Hooft, R.W.W., Sander, C. and Vriend, G. (1996). Verification of protein structures: side-chain planarity. *J. Appl. Crystallogr.* **29**, 714-716.
- [30] Hooft, R.W.W., Sander, C. and Vriend, G. (1997). Objectively judging the quality of a protein structure from a Ramachandran plot. *Comput. Applic. Biosci.* **13**, 425-430.
- [28] Vriend, G. and Sander, C. (1993). Quality control of protein models: directional atomic contact analysis. *J. Appl. Crystallogr.* **26**, 47-60.
- [31] Kleywegt, G.J. and Jones, T.A. (1998). Databases in protein crystallography. *Acta Crystallogr.* **D54**, 1119-1131.
- [32] Jones, T.A. and Kjeldgaard, M. (1997). Electron density map interpretation. *Methods Enzymol.* **277**, 173-208.
- [33] Kleywegt, G.J. and Jones, T.A. (1997). Model-building and refinement practice. *Methods Enzymol.* **277**, 208-230.
- [34] Brünger, A.T. (1992). Free *R* value: a novel statistical quantity for assessing the accuracy of crystal structures. *Nature* **355**, 472-475.
- [35] Kleywegt, G.J. and Brünger, A.T. (1996). Checking your imagination: applications of the free *R* value. *Structure* **4**, 897-904.
- [36] Laskowski, R.A., MacArthur, M.W., Moss, D.S. and Thornton, J.M. (1993). PROCHECK: a program to check the stereochemical quality of protein structures. *J. Appl. Crystallogr.* **26**, 283-291.
- [37] Kleywegt, G.J. and Jones, T.A. (1996). Phi/Psi-chology: Ramachandran revisited. *Structure* **4**, 1395-1400.
- [38] Lovell, S.C., Davis, I.W., Arendall, W.B., 3rd, De Bakker, P.I., Word, J.M., Prisant, M.G.,

- Richardson, J.S. and Richardson, D.C. (2003). Structure validation by C α geometry: ϕ / ψ and C β deviation. *Proteins Struct. Funct. Genet.* **50**, 437-450.
- [39] Eisenberg, D., Lüthy, R. and Bowie, J.U. (1997). VERIFY3D: Assessment of protein models with three-dimensional profiles. *Methods Enzymol.* **277**, 396-404.
- [40] Dym, O., Eisenberg, D. and Yeates, T.O. (2001). Detection of errors in protein models. *International Tables for Crystallography. Volume F. Crystallography of Biological Macromolecules*, 520-525.
- [41] Kleywegt, G.J. (1996). Use of non-crystallographic symmetry in protein structure refinement. *Acta Crystallogr.* **D52**, 842-857.
- [42] Kleywegt, G.J. (1999). Experimental assessment of differences between related protein crystal structures. *Acta Crystallogr.* **D55**, 1878-1884.
- [47] Brünger, A.T., Kuriyan, J. and Karplus, M. (1987). Crystallographic *R* factor refinement by molecular dynamics. *Science* **235**, 458-460.
- [48] Brünger, A.T., Adams, P.D., Clore, G.M., DeLano, W.L., Gros, P., Grosse-Kunstleve, R.W., Jiang, J.S., Kuszewski, J., Nilges, M., Pannu, N.S., Read, R.J., Rice, L.M., Simonson, T. and Warren, G.L. (1998). *Crystallography & NMR System: a new software suite for macromolecular structure determination.* *Acta Crystallogr.* **D54**, 905-921.
- [49] Tronrud, D.E. (1997). The TNT refinement package. *Methods Enzymol.* **277**, 306-319.
- [50] McRee, D.E. (1999). XtalView/Xfit - a versatile program for manipulating atomic coordinates and electron density. *J. Struct. Biol.* **125**, 156-165.
- [51] Sheldrick, G.M. and Schneider, T.R. (1997). SHELXL: high-resolution refinement. *Methods Enzymol.* **277**, 319-344.
- [52] Van den Akker, F. and Hol, W.G.J. (1999). Difference density quality (DDQ): a method to assess the global and local correctness of macromolecular crystal structures. *Acta Crystallogr.* **D55**, 206-218.
- [53] Colovos, C. and Yeates, T.O. (1993). Verification of protein structures: patterns of nonbonded atomic interactions. *Prot. Sci.* **2**, 1511-1519.
- [54] Kleywegt, G.J., Zou, J.Y., Kjeldgaard, M. and Jones, T.A. (2002). Around O.

International Tables for Crystallography, Volume F, Crystallography of Biological Macromolecules, 353-356, 366-367.

- [55] Das, U., Chen, S., Fuxreiter, M., Vaguine, A.A., Richelle, J., Berman, H.M. and Wodak, S.J. (2001). Checking nucleic acid crystal structures. *Acta Crystallogr.* **D57**, 813-828.
- [56] Jones, T.A. (1978). A graphics model building and refinement system for macromolecules. *J. Appl. Crystallogr.* **11**, 268-272.
- [57] Oldfield, T.J. (2001). A number of real-space torsion-angle refinement techniques for proteins, nucleic acids, ligands and solvent. *Acta Crystallogr.* **D57**, 82-94.
- [58] Vaguine, A.A., Richelle, J. and Wodak, S.J. (1999). *SFCHECK*: a unified set of procedures for evaluating the quality of macromolecular structure-factor data and their agreement with the atomic model. *Acta Crystallogr.* **D55**, 191-205.
- [43] Engh, R.A. and Huber, R. (1991). Accurate bond and angle parameters for X-ray protein structure refinement. *Acta Crystallogr.* **A47**, 392-400.
- [44] Parkinson, G., Vojtechovsky, J., Clowney, L., Brünger, A.T. and Berman, H.M. (1996). New parameters for the refinement of nucleic acid-containing structures. *Acta Crystallogr.* **D52**, 57-64.
- [45] Engh, R.A. and Huber, R. (2001). Structure quality and target parameters. *International Tables for Crystallography. Volume F. Crystallography of Biological Macromolecules*, 382-392.
- [46] Jones, T.A., Kleywegt, G.J. and Brünger, A.T. (1996). Storing diffraction data. *Nature* **381**, 18-19.
- [59] Zou, J.Y. and Jones, T.A. (1996). Towards the automatic interpretation of macromolecular electron-density maps: qualitative and quantitative matching of protein sequence to map. *Acta Crystallogr.* **D52**, 833-841.
- [60] Jones, T.A. and Liljas, L. (1984). Crystallographic refinement of macromolecules having non-crystallographic symmetry. *Acta Crystallogr.* **A40**, 50-57.

Tables

Table 1. Links to websites from which the materials in section 2 can be obtained.

Resource:	URL:
CCP4	http://www.ccp4.ac.uk/main.html
O	http://xray.bmc.uu.se/~alwyn/o_related.html
PDB	http://www.pdb.org/
Uppsala Software Factory (OOPS2, MAPMAN, MOLEMAN2)	http://xray.bmc.uu.se/usf
WHAT_CHECK	http://www.cmbi.kun.nl/gv/whatcheck/
WHAT IF	http://www.cmbi.kun.nl/gv/whatif/

Table 2. Menu of options presented by OOPS2.

POSSIBLE COMMANDS :

PEP = include pep-flip

RSC = include rotamer

RAM = include Ramachandran

WIF = include WHAT IF diagnostics

RSR = include real-space R (all atoms)

CCA = include real-space CC (all)

CCM = include real-space CC (main)

CCS = include real-space CC (side)

BFA = include B-factors

OCC = include occupancies

MSK = include mask fit

DIS = include disulfides

CAC = include CA chirality

PLA = include peptide planarity

PRE = include previous model

GO = get going !

QUI = quit without doing anything

Table 3. Example of an O macro generated by OOPS2 ^a.

```
centre_atom MOD C126 CA
@foreach
print .....
print Residue ASP C126 [Loop or turn      ]
message OOPS - Residue ASP C126 [Loop or turn]
symbol oops_irc      114
print Bad RSC = 1.26
print Bad Phi-Psi = 91.55 -46.75
print NOTE - non-Gly positive PHI = 91.55
print Too high temperature factor = 32.86
print Unusual backbone torsions : 114 ASP ( 126 ) C Poor phi/psi
print Bumps : 113 ASP ( 125 ) C O -- 114 ASP ( 126 ) C CB 0.108 2.692
print Unusual backbone conformations : 114 ASP ( 126 ) C O
print .....
print Hit or type "@oops/c130" for next baddy
menu @oops/c130 on
menu @oops/c126 off
```

^a This is a macro for one particular residue (in this case “C126”). The macro instructs O to centre on this residue and to execute a line of user-defined O commands (in this case, the user has chosen to execute a macro: “@foreach”). Subsequently, the macro prints a lot of information about aspects of this residue that are unusual. Finally, it puts a new command on the O user menu - activating this command will take the user to the next flagged residue (in this case, residue “C130”).

Table 4. Example of some of the contents of the notebook file produced by OOPS2.

Created by OOPS2 V. 021121/1.2.5 at Tue Jul 1 23:41:13 2003 for gerard

Molecule MOD

OOPS has checked:

Pep-flip values; cutoff = 2.5

RS R-factor (all atoms); cutoff = 0.150 ; WATERS = 0.150

RSC values; cutoff = 1.

Too low temperature factors; cutoff = 5.

Too high temperature factors; cutoff = 30.000 ; WATERS = 40.000

Too low occupancies; cutoff = 0.99000001

Too high occupancies; cutoff = 1.

Phi-Psi angle combinations (Ramachandran)

WHAT IF diagnostics

OOPS - ASN A2 [Loop or turn]

Bad RS R-factor (all atoms) = 0.216

Too high temperature factor = 32.04

H/N/O side chain flips : 1 ASN (2) A

COMMENTS/ACTION -->

OOPS - PHE A3 [Loop or turn]

Bad RSC = 1.37

Bumps : 2 PHE (3) A N -- 44 GLN (45) A NE2 0.273 2.727

COMMENTS/ACTION -->

OOPS - ALA A4 [Loop or turn]

Unusual backbone torsions : 3 ALA (4) A omega poor

Unusual backbone conformations : 3 ALA (4) A 0

Unsatisfied H-bond donors : 3 ALA (4) A N

COMMENTS/ACTION -->