

17. MODEL BUILDING AND COMPUTER GRAPHICS

17.1. Around *O*

BY G. J. KLEYWEGT, J.-Y. ZOU, M. KJELDGAARD AND T. A. JONES

17.1.1. Introduction

The first protein structures to be solved were built as wire models. It was only at the end of the 1970s and the beginning of the 1980s that the necessary hardware and software became available to allow crystallographers to construct their models using computers. The first commercially available computer graphics systems were very expensive, and by today's standards rather primitive in their line-drawing capabilities. They were usually controlled by mini-computers loaded with 64k words of memory and removable disks capable of holding 1 Mbyte of data. The limited amount of addressable memory was a severe limitation in software production. Furthermore, each computer graphics system had its own graphics programming library that was totally incompatible with those of other systems. Despite these limitations, the benefits of using computer-graphics-based systems became apparent and they were fairly rapidly adopted by the laboratories that could afford them. The main benefits were not in the construction of the initial model, but rather as a tool in crystallographic refinement (Jones, 1978). Making small manual changes to a model being refined was difficult and time-consuming work, but rather easy to accomplish even with low-powered graphics systems.

The most widely used program of the early days, *Frodo*, was available on most of the contemporary computer graphics systems. A major step forward occurred with the development of laboratory-scale 32-bit computers with virtual memory operating systems. In particular, the first Digital Equipment VAX models rapidly became the machines of choice in crystallographic laboratories. This allowed the contemplation of real-time improvements in models under construction (Jones, 1982; Jones & Liljas, 1984). Soon afterwards, colour became available in commercial graphics systems. This was much more than a cosmetic enhancement, since colour could be used to convey information vital to the crystallographer, such as main-chain/side-chain status codes for skeletonized electron density (Jones & Thirup, 1986). Unfortunately, there was still no common graphics programming standard, and moving to a new graphics system was a major effort (Pflugrath *et al.*, 1984).

The next major advance in hardware occurred with the development of the workstation, combining the computer and graphics in one package. Although pioneered by Sun, the major player in the crystallographic community was a small Californian company, Silicon Graphics, which rapidly became large. Running the Unix operating system, workstations flourished, but still lacked a graphics environment that was portable between different hardware platforms. This changed when OpenGL was adopted as an industry standard. At the same time, prices stabilized and began to drop in terms of price/performance. Only in the late 1990s have price/performance indicators plummeted with the arrival of PC/graphics-board combinations capable of meeting the expectations of the current generation of crystallographers. The crystallographic workstation on every desk has finally arrived.

17.1.2. *O*

O was designed by Alwyn Jones to overcome some of the drawbacks associated with using *Frodo*. These problems had arisen because of the history of the program. In particular, *O* was designed

to use a general-purpose memory allocation system to store any kind of model-related data. This would allow the display of any number of molecules and the use of databases for modelling. Although the latter had been introduced in a *Frodo* variant (Jones & Thirup, 1986), *O* was designed to take the concept of database use further, some would say to the extreme. Furthermore, *O* was designed to make it easier for different developers to work on the code without interfering with each other. In the event, only Morten Kjeldgaard and Jin-Yu Zou made any developments with the program.

Much of the data used by *O* are kept in a memory allocation system, the *O* database. This database is used to save parameters used by the program (including such things as keywords), macromolecular coordinates and information derived from them (such as graphical objects). As such, the program has no built-in limitations concerning what can be saved and used. A set of coordinates can be downloaded from the Protein Data Bank (PDB) (Bernstein *et al.*, 1977) and stored as a series of vectors that describe the sequence, the residue names, the coordinates, the atom names, the unit cell *etc.* Some of these vectors contain residue-related data (*e.g.* the sequence), others contain atomic data (*e.g.* the atomic temperature factors), while yet others concern the molecule as a whole. The program therefore uses a strict naming convention in handling these data. Each molecule has a name, and the program then forces its own nomenclature for the standard atomic, residue and molecular properties. The user remains free to create new data outside *O*, bring them into the program by adopting the naming convention, and then make use of them to generate or manipulate graphical images. For example, a series of amino-acid sequences can be aligned with a computer program outside *O* and information on the degree of sequence conservation can be generated as a series of *O* data blocks. These can then be read into *O* and used to colour a C α trace of a model, for example.

The program also has a strong macro capability that can be used to configure quite complex interactive tasks. It can also be used by a programmer outside *O* to generate data and a series of instructions for later interactive use.

Similarly, data generated within *O* can be exported to *O*-aware programs, significantly reducing the complexity of some crystallographic calculations. For example, real-space averaging of electron-density maps requires, as a minimum, both a series of operators describing the noncrystallographic symmetry (NCS) and a mask. These can be generated from scratch in *O* and improved and used by *O*-aware programs without the crystallographer needing to be concerned about the myriad details of axis definitions, rotations and translations.

Plotting is carried out within the *O* system *via* an intermediate metafile. When a user creates objects for display within *O*, calls are made to a set of low-level routines that create the OpenGL instructions on the workstation. Some objects are described in their entirety within the *O* database, but others are not. Molecular objects fall into the former category, whereas electron-density maps fall into the latter. There are two sets of plot commands, therefore, that are appropriate for each class. To plot an object made from a molecule, the user merely issues a *plot_object* command, and the appropriate metafile is written out, complete with viewing data. To plot other things, the user activates the *plot_on* command and then starts creating objects. Every time a low-level graphics routine is

called, something gets written into the metafile. This is terminated with a *plot_off* command. The metafile contains much extraneous data, for example, instructions to the *O* pulldown menu system. However, it is built up from objects that are arranged in a hierarchy, where the highest-level object is called *disp_all*. Some objects, therefore, call instances of others, while other objects contain graphics instructions that define line start and end points, for example.

This metafile can be processed, and so far three different programs are available. *OPLOT* (written by Morten Kjeldgaard) generates PostScript output, carrying out a full traversal of the object hierarchy. The other two programs do not carry out such a traversal, but merely process the objects specified by the user. One (written by Mark Harris and Alwyn Jones) generates output suitable for input to the ray-tracing program *PovRay* (see <http://www.povray.org>). The third program (written by Martin Berg) generates VRML output suitable for web-based viewing.

O is in continuous development, and interested readers are encouraged to visit the various internet sites that we maintain. There they will find detailed descriptions of the *O* command set, as well as various introductory exercises for learning how to use the program. The following publications describe various aspects of *O*-related features and methods:

(1) Jones *et al.* (1991) provide an introduction to the *O* database, a description of a residue-based electron-density goodness-of-fit indicator, the use of databases to construct a poly-alanine from a $C\alpha$ trace, various real-space refinement algorithms *etc.* They also describe two useful indicators for detecting peptide-plane and side-chain errors that make use of comparisons with databases.

(2) Zou & Mowbray (1994) describe an evaluation of the use of databases in refinement.

(3) Zou & Jones (1996) describe their attempts towards finally automating the interpretation of electron-density maps. They also describe both qualitative and quantitative matching of the protein sequence to the map.

(4) Jones & Kjeldgaard (1997) review the different kinds of errors that can be introduced into a model and why these errors are made. They also describe some of the features of *O* and the steps needed in tracing and building a model (including the vital step of locating the sequence in the electron density).

(5) Mowbray *et al.* (1999) describe experiments aimed at evaluating the reproducibility of model building and discuss some of the more useful indicators of model error.

17.1.3. RAVE

RAVE is a suite of programs for electron-density map improvement and analysis, with a strong focus on averaging techniques (Kleywegt & Read, 1997). It is the successor of an older package ('*A*') (Jones, 1992), and at present it contains tools for single and multiple crystal form, single- and multiple-domain NCS averaging of electron-density maps, and the detection of structural units in such maps. The package works in conjunction with the *CCP4* suite of programs (Collaborative Computational Project, Number 4, 1994).

RAVE contains the following programs for density averaging involving one crystal form:

(1) *AVE* (Jones, 1992). This program carries out the averaging step and the expansion step (in which the averaged density is projected back into the entire unit cell or asymmetric unit).

(2) *COMA* (Kleywegt & Jones, 1999). This program uses the algorithm of Read (Vellieux *et al.*, 1995) to calculate local density correlation maps that can be used to delineate masks (molecular envelopes). It can also be used to validate structural differences between NCS-related molecules (Kleywegt, 1999b).

(3) *IMP* (Jones, 1992). This program can be used to optimize NCS operators relating two copies of a molecule (or domain) inside the same cell. The program adjusts the initial operator (*e.g.* obtained from heavy-atom positions) so as to maximize the correlation coefficient between the density inside the envelope and its NCS-related counterpart. The procedure can be controlled by the user or run in automatic mode, which usually gives satisfactory results.

(4) *NCS6D*. This program can be used to find NCS operators in cases where it is difficult to obtain them by other means. The program uses a set of *BONES* atoms (or a PDB file) and, for a large number of combinations of rotations and translations, calculates the correlation coefficient between the density around the atoms and that obtained after application of the rotation and translation. This approach was used, for instance, to find the operators in the case of maltoporin (Schirmer *et al.*, 1995).

(5) *COMDEM*. If a molecule contains multiple domains that have different NCS relationships, the individual domain densities can be averaged with *AVE* and subsequently combined with this program. *AVE* can then be used to expand the density back into the unit cell or asymmetric unit.

(6) *SPANCSI*. This program is useful when NCS-related molecules are known or suspected to have very different average temperature factors. One option is to analyse the similarities between NCS-related copies of the molecular density (variance, correlation coefficient, *R* factor). In addition, the program can carry out electron-density averaging and expansion, in which each copy of the density is scaled by its variance.

RAVE also contains tools for averaging between different crystal forms, namely:

(1) *MASKIT* (Kleywegt & Jones, 1999). This program calculates a local density correlation map from the density of two different crystals or crystal forms, using Read's algorithm (Vellieux *et al.*, 1995). This program can also be used to validate structural differences between related molecules, for which experimental electron density is available (Kleywegt, 1999b).

(2) *MAVE*. This program does the (skew) density averaging and expansion steps, but now separately because the density of the various crystal forms has to be averaged as well. This program also contains an option to improve operators that relate the position and orientation of the molecular envelope (mask) in one crystal form with those in other crystal forms.

(3) *COMDEM*. This program combines the individual (possibly averaged) densities from various crystal forms. The densities are scaled according to the number of molecules whose (averaged) density they represent as well as according to their variance.

(4) *CRAVE*. Since the book-keeping for multiple-crystal-form averaging can become rather complicated, this program can be used to generate one large C-shell script that will execute a user-defined number of cycles of multiple-crystal-form averaging.

More recently, *RAVE* has been expanded to include tools that can be of use in map interpretation:

(1) *ESSENS* (Kleywegt & Jones, 1997a). This program takes a (rigid) structural template (*e.g.* a penta-alanine helix or strand, or a ligand) and calculates how well it fits the density by doing an exhaustive rotational search for every grid point in the map. The resulting score map will reveal places in the map where the centre of gravity of the template fits the density well. The method is very effective for detecting secondary-structure elements (prior to human map interpretation), as discussed by Kleywegt & Jones (1997a). The *ESSENS* algorithm has also been implemented within *O* (Jones & Kleywegt, 2001).

(2) *SOLEX*. This program can be used to extract the best-fitting positions and orientations of a structural template as found in an *ESSENS* calculation. If the search used a template in helix or strand conformation, the program can also be used to combine short stretches of helix and strand into longer units. The results (helices

and/or strands of unknown connectivity and uncertain directionality) can be fed into another program, *DEJAVU* (Kleywegt & Jones, 1994b, 1997b) (see below), to check if they are similar to (a part of) another protein whose structure is known (Kleywegt & Jones, 1994b).

Finally, *RAVE* also contains three utility programs that can be used to manipulate three essential data structures encountered in averaging, map interpretation and refinement:

(1) *MAMA* (Kleywegt & Jones, 1994b, 1999). This program is used to generate, analyse and manipulate masks (molecular envelopes). It contains many of the tools described earlier by Jones (1992), but many new features have been added to it since. Masks can be generated from scratch, using a PDB file or *BONES* atoms, by ‘recycling’ another mask (*e.g.* from a different crystal form), or by combining several older masks. The quality of masks can be improved by filling voids, removing unconnected ‘droplets’, smoothing the surface, trimming regions that give rise to overlap through (non-)crystallographic symmetry and checking that all atoms in a model are covered by the mask (*e.g.* after changes or extensions to a model have been made).

(2) *MAPMAN* (Kleywegt & Jones, 1996a). This program was written for format conversion, analysis and manipulation of electron-density maps. Maps can be read and written in a variety of formats, including those used by *O*. Maps can be combined, scaled, peak-picked and subjected to ‘digital image filters’ (Kleywegt & Jones, 1997a). Several older stand-alone programs have been incorporated into *MAPMAN*, such as *MAPPAGE* and *BONES* (Jones & Thirup, 1986; the program previously used to skeletonize electron density for use with *Frodo* or *O*). Many statistics and types of histograms and plots (*e.g.* slices, or 2D and 1D projections) can be calculated or generated.

(3) *DATAMAN* (Kleywegt & Jones, 1996a). This program is used for simple format conversion, analysis and manipulation of reflection data sets [consisting of Miller indices, F , $\sigma(F)$, and possibly a cross-validation flag]. Data can be sorted, Laue symmetry can be applied, and data can be scaled by a temperature and a scale factor, re-indexed, and reduced in special cases where higher symmetry is present or suspected. The program contains a wide range of options to select ‘test set’ reflections that are to be set aside for cross-validation purposes (Brünger, 1992a; Kleywegt & Brünger, 1996). Many statistics and types of histograms and plots can be calculated or generated.

17.1.4. Structure analysis

A number of programs are available for the analysis of protein models. Most of these programs are interfaced with *O*, producing files (such as maps and *O* macros) that allow for quick and easy visualization and inspection of their results.

(1) *VOIDOO* (Kleywegt & Jones, 1994a). This program can be used to find cavities in macromolecular structures. The program will detect cavities, measure their volumes and produce files that can be used by *O* to visualize the cavities, any atoms inside them and protein residues surrounding them.

(2) *DEJAVU* (Kleywegt & Jones, 1994b, 1997b). This is a program for fold recognition. It uses an abstracted representation of protein structure, namely the coordinates of the start and end points of secondary-structure elements (SSEs). The user can define a motif of SSEs (*e.g.* a four-helix bundle that represents only one domain of a much larger molecule) and the program will search a database derived from the PDB (Bernstein *et al.*, 1977) to look for other protein structures that contain a similar structural motif. Alternatively, the program can take all SSEs of a structure into account and look for structures in the database that have as many SSEs as possible in an arrangement similar to the user’s structure. This

approach led to the discovery of the structural similarity between glutathione synthetase and D-alanine:D-alanine ligase (Fan *et al.*, 1994, 1995), and that between the N-terminal DNA-binding domain of the diphtheria toxin repressor and the C-terminal DNA-binding domain of catabolite gene activator protein (Qiu *et al.*, 1995). Although several other fold-recognition programs are available, *DEJAVU* has two finesses that distinguish it, namely, the fact that it does not require that atomic coordinates (*e.g.* of the $C\alpha$ atoms) be available, and the fact that the program can use SSEs even in cases where their directionality and/or connectivity is unknown. This situation typically occurs in early stages of the map interpretation process, when some SSEs can be discerned in the density [*e.g.* using *ESSENS* (Kleywegt & Jones, 1997a)], but their direction is still difficult to determine, and when their connections may still be uncertain. In that case, the start and end points (in either order) can simply be estimated by the crystallographer (or using the program *SOLEX*, see Section 17.1.3) and the set of SSEs can be used as input to *DEJAVU* to look for similar structures in the database (Kleywegt & Jones, 1994b). Although the search will be less sensitive in this case, if successful, it may shorten the initial model-building process considerably.

(3) *SPASM* (Kleywegt, 1999a). This program is related to *DEJAVU*, but operates at a more detailed level. It can be used to look for known structures that contain a user-defined motif, consisting of two or more protein residues. This program has been used, for instance, in the analysis of phosphoenolpyruvate carboxykinase (Matte *et al.*, 1996), where it revealed a striking similarity between this protein’s P-loop structure and the P-loops of adenylate kinase isozyme III, RecA protein and p21ras. We have also developed a program (*RIGOR*) that takes the opposite approach: using a database of pre-defined motifs (*e.g.* ligand and metal-binding sites, catalytic centres), this program will check if any of these also occur in the user’s protein model.

(4) *SBIN* (Kleywegt & Jones, 1998; Kleywegt, unpublished programs). *SBIN* is a suite of programs that can be used to derive *PROSITE* patterns (Bairoch & Bucher, 1994) or Gribskov-style sequence profiles (Gribskov *et al.*, 1987; Gribskov & Veretnik, 1996) from sets of superimposed protein models. These patterns and profiles in turn can be used to retrieve protein sequences from databases such as *SWISS-PROT* and *TrEMBL* (Bairoch & Apweiler, 1997) that may be related in structure and/or function.

17.1.5. Utilities

Many utility programs are available from Uppsala, most of them aimed at practising crystallographers. Some of these (*MAMA*, *MAPMAN*, *DATAMAN*) have been discussed in Section 17.1.3. A few others are discussed below.

(1) *LSQMAN* (Kleywegt & Jones, 1997b; Kleywegt, 1996). This is a program for analysing and manipulating multiple copies of a molecule or multiple molecules. It contains tools to superimpose molecules (including an option to find such superpositioning automatically), to improve the fit of two superimposed molecules in myriad ways, to calculate and plot r.m.s. distances and φ , ψ or χ_1 , χ_2 torsion-angle differences and circular variances (Allen & Johnson, 1991; Korn & Rose, 1994; Kleywegt, 1996) of different molecules, to generate multiple-model Ramachandran plots, to compare the solvent structure in two molecules, to find the ‘central’ molecule of an ensemble [defined as the molecule that has the smallest r.m.s.(r.m.s.d.), *i.e.* the r.m.s. value of its pairwise r.m.s.d.’s to each of the other molecules], and to align molecules in an ensemble to the ‘central’ molecule. The program can handle proteins, nucleic acids and other types of molecules.

(2) *MOLEMAN2*. This is a general program for analysis and manipulation of molecules (in PDB-format files). It contains too

17. MODEL BUILDING AND COMPUTER GRAPHICS

many options to list here, including tools that are of use when submitting structures to the PDB, several options that interface with *O*, *X-PLOR* (Brünger, 1992b), *CNS* (Brünger *et al.*, 1998) and *CCP4* (Collaborative Computational Project, Number 4, 1994), tools to analyse and manipulate temperature factors and occupancies, simple validation tools for protein models [e.g. Ramachandran (Kleywegt & Jones, 1996c) and C α -Ramachandran plots (Kleywegt, 1997)], and simple distance and sequence analysis options.

(3) *ODBMAN*. This is a program for analysis and manipulation of *O*-style data blocks, offering a superset of the functionality available in *O* itself for this purpose. Data blocks can be plotted, combined, generated *etc.*

(4) *OOPS2* (Kleywegt & Jones, 1996b). This program uses the current model and quality-related properties calculated in *O* or by *OOPS2* or external programs to produce a detailed account of the quality of the model and the individual residues, and to generate a set of macro files for *O* that will take the crystallographer to all the residues that are outliers for one or more of the quality criteria.

(5) *SEAMAN*. This program offers several tools to help the crystallographer generate search models for use in molecular-replacement exercises. It is designed to handle 'multiple search models' (e.g. an ensemble of NMR structures) as easily as single models.

(6) *SOD*. This program produces various types of file for use with *O*, based on aligned amino-acid sequences. For instance, it can generate data blocks and macro files that can be used to colour a protein model according to the degree of conservation or variability of each residue, or it can be used to generate an *O* macro file that will 'mutate' an existing protein model into a molecular-replacement probe for a related protein.

(7) *XPLO2D*. This program was written to perform various functions for users of the refinement program *X-PLOR* (Brünger, 1992b). Its most useful component is the option to generate topology and parameter files automatically [for use with *X-PLOR*, *CNS* (Brünger *et al.*, 1998), *TNT* (Tronrud *et al.*, 1987), or *O*] for hetero entities based on the coordinates of one or more copies of such an entity.

17.1.6. Other services

The following is an overview of internet-based services related to *O* and the associated programs.

Information. General information about *O* and the associated programs is available through three web sites: <http://xray.bmc.uu.se/alwyn/>, <http://imsb.au.dk/~mok/o/> and <http://xray.bmc.uu.se/usf/>. These sites provide access to manuals, release notes *etc.*

Software. Software can be downloaded from <ftp://xray.bmc.uu.se/pub/> or <ftp://kaktus.imsb.au.dk/pub/o/>. The initial installation of *O* is distributed differently (contact TAJ for details, e-mail: alwyn@xray.bmc.uu.se). All associated programs discussed in Sections 17.1.3, 17.1.4 and 17.1.5 are available free of charge to academic users (others may contact GJK for details, e-mail: gerard@xray.bmc.uu.se).

Educational material. Material is available from <http://xray.bmc.uu.se/usf/> for learning about or teaching the use of *O*, rebuilding protein models with *O*, map tracing with *O* and electron-density averaging with *RAVE*.

HIC-Up. HIC-Up (Hetero-compound Information Centre – Uppsala) is a web site (<http://xray.bmc.uu.se/hicup/>) concerned with the use of hetero entities in macromolecular structure determination. It contains a wealth of information regarding all hetero compounds encountered in the PDB, and has servers to generate geometric and other dictionaries for use with *O*, *X-PLOR*, *CNS* and *TNT*.

Miscellaneous. A list of literature references, reprints and preprints of technical papers *etc.* is available at <http://xray.bmc.uu.se/usf/>.

Servers. Several web-based servers are operated from Uppsala by Dr Tom Taylor. These include services for inspecting the electron density of published structures through a VRML interface and for generating Ramachandran (Kleywegt & Jones, 1996c) and C α -Ramachandran plots (Kleywegt, 1997). All servers can be accessed through <http://xray.bmc.uu.se/tom/>.

Acknowledgements

This work has been supported by the the Swedish Natural Sciences Research Council (NFR), Uppsala University, the EU-funded 3-D Validation Network, the Swedish Foundation for Strategic Research (SSF) and its Structural Biology Network (SBNNet).

within the molecular landscape (Taylor *et al.*, 1993). The challenge of bridging across the scales of size and complexity of the molecular world may lead us into the realm of virtual reality. Data from X-ray crystallography are being combined with data from large molecular complexes, characterized by electron microscopy. These data, in turn, can be integrated with those from optical confocal microscopy and other imaging techniques. With structures of molecules,

assemblies and distributions, as well as data on molecular inventories, we can start to piece together integrated pictures of cellular environments, but with full atomic modelling at the base (Fig. 17.2.5.1). Thus, while climbing around inside a protein molecule might not add much in the way of perceptual advantage, navigating through the molecular environment of a cell may prove to be instructive as well as inspirational.

References

17.1

- Allen, F. H. & Johnson, O. (1991). *Automated conformational analysis from crystallographic data. 4. Statistical descriptors for a distribution of torsion angles.* *Acta Cryst.* **B47**, 62–67.
- Bairoch, A. & Apweiler, R. (1997). *The SWISS-PROT protein sequence data bank and its supplement TrEMBL.* *Nucleic Acids Res.* **25**, 31–36.
- Bairoch, A. & Bucher, P. (1994). *PROSITE: recent developments.* *Nucleic Acids Res.* **22**, 3583–3589.
- Bernstein, F. C., Koetzle, T. F., Williams, G. J. B., Meyer, E. F. Jr, Brice, M. D., Rodgers, J. R., Kennard, O., Shimanouchi, T. & Tasumi, M. (1977). *The Protein Data Bank: a computer-based archival file for macromolecular structures.* *J. Mol. Biol.* **112**, 535–542.
- Brünger, A. T. (1992a). *Free R value: a novel statistical quantity for assessing the accuracy of crystal structures.* *Nature (London)*, **355**, 472–475.
- Brünger, A. T. (1992b). *X-PLOR: a system for crystallography and NMR.* Yale University, New Haven, CT, USA.
- 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. & Warren, G. L. (1998). *Crystallography & NMR System: a new software suite for macromolecular structure determination.* *Acta Cryst.* **D54**, 905–921.
- Collaborative Computational Project, Number 4 (1994). *The CCP4 suite: programs for protein crystallography.* *Acta Cryst.* **D50**, 760–763.
- Fan, C., Moews, P. C., Shi, Y., Walsh, C. T. & Knox, J. R. (1995). *A common fold for peptide synthetases cleaving ATP to ADP: glutathione synthetase and D-alanine:D-alanine ligase of Escherichia coli.* *Proc. Natl Acad. Sci. USA*, **92**, 1172–1176.
- Fan, C., Moews, P. C., Walsh, C. T. & Knox, J. R. (1994). *Vancomycin resistance: structure of D-alanine:D-alanine ligase at 2.3 Å resolution.* *Science*, **266**, 439–443.
- Gribskov, M., McLachlan, A. D. & Eisenberg, D. (1987). *Profile analysis: detection of distantly related proteins.* *Proc. Natl Acad. Sci. USA*, **84**, 4355–4358.
- Gribskov, M. & Veretnik, S. (1996). *Identification of sequence patterns with profile analysis.* *Methods Enzymol.* **266**, 198–212.
- Jones, T. A. (1978). *A graphics model building and refinement system for macromolecules.* *J. Appl. Cryst.* **11**, 268–272.
- Jones, T. A. (1982). *FRODO: a graphics fitting program for macromolecules.* In *Computational crystallography*, edited by D. Sayre, pp. 303–317. Oxford: Clarendon Press.
- Jones, T. A. (1992). *A, yaap, asap, @#*? A set of averaging programs.* In *Molecular replacement*, edited by E. J. Dodson, S. Glover & W. Wolf, pp. 91–105. Warrington: Daresbury Laboratory.
- Jones, T. A. & Kjeldgaard, M. (1997). *Electron density map interpretation.* *Methods Enzymol.* **277**, 173–208.
- Jones, T. A. & Kleywegt, G. J. (2001). *New tools for the interpretation of macromolecular electron-density maps.* In *preparation*.
- Jones, T. A. & Liljas, L. (1984). *Crystallographic refinement of macromolecules having noncrystallographic symmetry.* *Acta Cryst.* **A40**, 50–57.
- Jones, T. A. & Thirup, S. (1986). *Using known substructures in protein model building and crystallography.* *EMBO J.* **5**, 819–822.
- Jones, T. A., Zou, J.-Y., Cowan, S. W. & Kjeldgaard, M. (1991). *Improved methods for building protein models in electron density maps and the location of errors in these models.* *Acta Cryst.* **A47**, 110–119.
- Kleywegt, G. J. (1996). *Use of non-crystallographic symmetry in protein structure refinement.* *Acta Cryst.* **D52**, 842–857.
- Kleywegt, G. J. (1997). *Validation of protein models from C α coordinates alone.* *J. Mol. Biol.* **273**, 371–376.
- Kleywegt, G. J. (1999a). *Recognition of spatial motifs in protein structures.* *J. Mol. Biol.* **285**, 1887–1897.
- Kleywegt, G. J. (1999b). *Experimental assessment of differences between related protein crystal structures.* *Acta Cryst.* **D55**, 1878–1884.
- Kleywegt, G. J. & Brünger, A. T. (1996). *Checking your imagination: applications of the free R value.* *Structure*, **4**, 897–904.
- Kleywegt, G. J. & Jones, T. A. (1994a). *Detection, delineation, measurement and display of cavities in macromolecular structures.* *Acta Cryst.* **D50**, 178–185.
- Kleywegt, G. J. & Jones, T. A. (1994b). *Halloween . . . masks and bones.* In *From first map to final model*, edited by S. Bailey, R. Hubbard & D. A. Waller, pp. 59–66. Warrington: Daresbury Laboratory.
- Kleywegt, G. J. & Jones, T. A. (1996a). *xdlMAPMAN and xdlDATAMAN – programs for reformatting, analysis and manipulation of biomacromolecular electron-density maps and reflection data sets.* *Acta Cryst.* **D52**, 826–828.
- Kleywegt, G. J. & Jones, T. A. (1996b). *Efficient rebuilding of protein structures.* *Acta Cryst.* **D52**, 829–832.
- Kleywegt, G. J. & Jones, T. A. (1996c). *Phi/Psi-chology: Ramachandran revisited.* *Structure*, **4**, 1395–1400.
- Kleywegt, G. J. & Jones, T. A. (1997a). *Template convolution to enhance or detect structural features in macromolecular electron-density maps.* *Acta Cryst.* **D53**, 179–185.
- Kleywegt, G. J. & Jones, T. A. (1997b). *Detecting folding motifs and similarities in protein structures.* *Methods Enzymol.* **277**, 525–545.
- Kleywegt, G. J. & Jones, T. A. (1998). *Databases in protein crystallography.* *Acta Cryst.* **D54**, 1119–1131.
- Kleywegt, G. J. & Jones, T. A. (1999). *Software for handling macromolecular envelopes.* *Acta Cryst.* **D55**, 941–944.
- Kleywegt, G. J. & Read, R. J. (1997). *Not your average density.* *Structure*, **5**, 1557–1569.
- Korn, A. P. & Rose, D. R. (1994). *Torsion angle differences as a means of pinpointing local polypeptide chain trajectory changes for identical proteins in different conformational states.* *Protein Eng.* **7**, 961–967.
- Matte, A., Goldie, H., Sweet, R. M. & Delbaere, L. T. J. (1996). *Crystal structure of Escherichia coli phosphoenolpyruvate carboxykinase: a new structural family with the P-loop nucleoside triphosphate hydrolase fold.* *J. Mol. Biol.* **256**, 126–143.
- Mowbray, S. L., Helgstrand, C., Sigrell, J. A., Cameron, A. D. & Jones, T. A. (1999). *Errors and reproducibility in electron-density map interpretation.* *Acta Cryst.* **D55**, 1309–1319.
- Pflugrath, J. W., Saper, M. A. & Quiocho, F. A. (1984). *New generation graphics system for molecular modeling.* In *Methods and applications in crystallographic computing*, edited by S. R. Hall & T. Ashida, pp. 404–407. Oxford: Clarendon Press.
- Qiu, X., Verlinde, C. L. M. J., Zhang, S., Schmitt, M. P., Holmes, R. K. & Hol, W. G. J. (1995). *Three-dimensional structure of the*

17.1 (cont.)

- diphtheria toxin repressor in complex with divalent cation corepressors. *Structure*, **3**, 87–100.
- Schirmer, T., Keller, T. A., Wang, Y. F. & Rosenbusch, J. P. (1995). Structural basis for sugar translocation through maltoporin channels at 3.1 Å resolution. *Science*, **267**, 512–514.
- Tronrud, D. E., Ten Eyck, L. F. & Matthews, B. W. (1987). An efficient general-purpose least-squares refinement program for macromolecular structures. *Acta Cryst.* **A43**, 489–501.
- Vellieux, F. M. D. A. P., Hunt, J. F., Roy, S. & Read, R. J. (1995). DEMON/ANGEL: a suite of programs to carry out density modification. *J. Appl. Cryst.* **28**, 347–351.
- Zou, J. Y. & Jones, T. A. (1996). Towards the automatic interpretation of macromolecular electron-density maps: qualitative and quantitative matching of protein sequence to map. *Acta Cryst.* **D52**, 833–841.
- Zou, J.-Y. & Mowbray, S. L. (1994). An evaluation of the use of databases in protein structure refinement. *Acta Cryst.* **D50**, 237–249.

17.2

- Bailey, M., Schulten, K. & Johnson, J. (1998). The use of solid physical models for the study of macromolecular assembly. *Curr. Opin. Struct. Biol.* **8**, 202–208.
- Bajaj, C. L., Pascucci, V. & Schikore, D. R. (1996). Fast isocontouring for improved interactivity. In *Proceedings of the ACM SIGGRAPH/IEEE symposium on volume visualization*, pp. 39–46. San Francisco: ACM Press.
- Barry, C. D. & McAlister, J. P. (1982). High performance molecular graphics, a hardware review. In *Computational crystallography*, edited by D. Sayre, pp. 274–255. Oxford: Clarendon Press.
- Branden, C.-I., Jornvall, H., Eklund, H. & Fureugren, B. (1975). Alcohol dehydrogenase. In *The enzymes*, edited by P. Boyer, pp. 104–186. New York: Academic Press.
- Brooks, J. P. Jr, Ouh-Young, M., Batter, J. J. & Kilpatrick, P. J. (1990). Project GROPE – haptic displays for scientific visualization. *Comput. Graphics*, **24**, 177–186.
- Burns, M. (1954). *Automated fabrication: improving productivity in manufacturing*. New Jersey: Prentice Hall.
- Carson, M. (1991). RIBBONS 2.0. *J. Appl. Cryst.* **24**, 958–961.
- Connolly, M. L. (1983). Solvent-accessible surfaces of proteins and nucleic acids. *Science*, **221**, 709–713.
- Connolly, M. L. & Olson, A. J. (1985). GRANNY, a companion to GRAMPS for the real-time manipulation of macromolecular models. *Comput. Chem.* **9**, 1–6.
- Diamond, R. (1982). BILDER: an interactive graphics program for biopolymers. In *Computational crystallography*, edited by D. Sayre, pp. 318–325. Oxford: Clarendon Press.
- Drebein, R., Carpenter, L. & Hanrahan, P. (1988). Volume rendering. *Proc. ACM SIGGRAPH'88 (Atlanta, Georgia, 1–5 August 1988)*. In *Comput. Graphics Proc. Annu. Conf. Ser.* 1988 (1993), pp. 65–74. New York: ACM SIGGRAPH.
- Duncan, B. S. & Olson, A. J. (1993). Approximation and characterization of molecular surfaces. *Biopolymers*, **33**, 219–229.
- Duncan, B. S. & Olson, A. J. (1995). Approximation and visualization of large-scale motion of protein surfaces. *J. Mol. Graphics*, **13**, 250–257.
- Ferrin, T. E., Huang, C. C., Jarvis, L. E. & Langridge, R. (1988). The MIDAS display system. *J. Mol. Graphics*, **6**, 13–27.
- Goodford, P. J. (1985). A computational procedure for determining energetically favorable binding sites on biologically important molecules. *J. Med. Chem.* **28**, 849–857.
- Goodsell, D. S., Mian, I. S. & Olson, A. J. (1989). Rendering of volumetric data in molecular systems. *J. Mol. Graphics*, **7**, 41–47.
- Goodsell, D. S. & Olson, A. J. (1992). Molecular illustration in black and white. *J. Mol. Graphics*, **10**, 235–240.
- Gouraud, H. (1971). Continuous shading of curved surfaces. *IEEE Trans. Comput.* **20**, 623–628.
- Gwilliam, M. & Max, N. (1989). Atoms with shadows – an area-based algorithm for cast shadows on space-filling molecular models. *J. Mol. Graphics*, **7**, 54–59.
- Hessler, D. S., Young, S. J. & Ellisman, M. H. (1996). A flexible environment for visualization of three-dimensional biological structures. *J. Struct. Biol.* **116**, 113–119.
- Höhne, K. H., Bomans, M., Pommert, A., Reimer, M., Schiers, C., Tiede, U. & Wiebecke, G. (1989). 3D-visualization of tomographic volume data using the generalized voxel-model. Volume visualization workshop, Chapel Hill, NC. Department of Computer Science, University of North Carolina at Chapel Hill.
- Hubbard, R. E. (1986). HYDRA: current and future developments. In *Computer graphics and molecular modelling*, edited by R. Fletcher & M. Zoller, pp. 9–12. Cold Spring Harbor Press.
- Johnson, C. K. (1970). ORTEP: a Fortran thermal-ellipsoid plot program for crystal structure illustrations. Report ORNL 3794. Oak Ridge National Laboratory, Tennessee, USA.
- Jones, T. A. (1978). A graphics model building and refinement system for macromolecules. *J. Appl. Cryst.* **11**, 268–272.
- Kabsch, W., Mannherz, H. G., Suck, D., Pai, E. F. & Holmes, K. C. (1990). Atomic structure of the actin/DNAse I complex. *Nature (London)*, **347**, 37–44.
- Koradi, R., Billeter, M. & Wuthrich, K. (1996). MOLMOL: a program for display and analysis of macromolecular structures. *J. Mol. Graphics*, **14**, 51–55.
- Kraulis, P. J. (1991). MOLSCRIPT: a program to produce both detailed and schematic plots of protein structures. *J. Appl. Cryst.* **24**, 946–950.
- Kremer, J. R., Mastronarde, D. N. & McIntosh, J. R. (1996). Computer visualization of three-dimensional image data using IMOD. *J. Struct. Biol.* **116**, 71–76.
- Lauher, J. W. (1990). Chem-Ray: a molecular graphics program featuring an umbra and penumbra shadowing routine. *J. Mol. Graphics*, **8**, 34–38.
- Lee, B. & Richards, F. M. (1971). The interpretation of protein structures: estimation of static accessibility. *J. Mol. Biol.* **55**, 379–400.
- Levinthal, C. (1966). Molecular modeling by computer. *Sci. Am.* **214**, 42–52.
- Lorenson, W. E. & Kline, H. E. (1987). Marching cubes: a high resolution 3D surface construction algorithm. *Comput. Graphics*, **21**, 163–169.
- Max, N. (1983). SIGGRAPH'84 call for OmniMax films. *Comput. Graphics*, **17**, 73–76.
- Motherwell, W. D. S. & Clegg, W. (1978). PLUTO. Program for plotting molecular and crystal structures. University of Cambridge, England.
- Mujeeb, A., Kerwin, S. M., Kenyon, G. L. & James, T. L. (1993). Solution structure of a conserved DNA sequence from the HIV-1 genome: restrained molecular dynamics simulation with distance and torsion angle restraints derived from two-dimensional NMR spectra. *Biochemistry*, **32**, 13419–13431.
- Nichols, W. L., Rose, G. D., Ten Eyck, L. F. & Zimm, B. H. (1995). Rigid domains in proteins: an algorithmic approach to their identification. *Proteins Struct. Funct. Genet.* **23**, 38–45.
- O'Donnell, T. J. & Olson, A. J. (1981). GRAMPS – a graphics language interpreter for real-time, interactive three-dimensional picture editing and animation. *Comput. Graphics*, **15**, 133–142.
- Olson, A. J. (1981). Tomato bushy stunt virus. Film. Lawrence Berkeley Laboratory, University of California, Berkeley, California, USA.
- Olson, A. J. (1983). Computer graphics in biomolecular science. NICOGRAPH '83, pp. 332–356. Tokyo: Nihon Keizai Shimbun, Inc.
- Pepinsky, R. (1952). X-RAC and S-FAC: electronic analogue computers for X-ray analysis. In *Computing methods and the phase problem in X-ray crystal analysis*. Pennsylvania State College, USA.
- Phillips, D. C. (1970). *British biochemistry, past and present*, edited by T. W. Goodwin, pp. 11–28. Academic Press.
- Phong, B. T. (1975). Illumination for computer generated images. *Commun. ACM*, **18**, 311–317.