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 Cα trace, various real-space refinement algorithms *etc.* They also describe two useful indictors 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 *CCP*4 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, 1999*b*).

(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, 1999*b*).

(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, 1997*a*). 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 (1997*a*). 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, 1994*b*, 1997*b*) (see below), to check if they are similar to (a part of) another protein whose structure is known (Kleywegt & Jones, 1994*b*).

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, 1994*b*, 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, 1996*a*). 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, 1997*a*). 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, 1996*a*). 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, 1992*a*; 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, 1994*a*). 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, 1994*b*, 1997*b*). 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 Cα 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, 1997*a*)], 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, 1994*b*). Although the search will be less sensitive in this case, if successful, it may shorten the initial model-building process considerably.

(3) *SPASM* (Kleywegt, 1999*a*). 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, 1997*b*; 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) *MOLEMAN*2. This is a general program for analysis and manipulation of molecules (in PDB-format files). It contains too

references