Acta Cryst. (1994). D**50**, 178–185

Detection, Delineation, Measurement and Display of Cavities in Macromolecular Structures

BY GERARD J. KLEYWEGT AND T. ALWYN JONES*

Department of Molecular Biology, Biomedical Centre, University of Uppsala, Box 590, *S*-751 24 *Uppsala, Sweden*

(*Received* 1 *June* 1993; *accepted* 14 *October* 1993)

Abstract

A computer program, *VOIDOO*, is described which can be employed in the study of cavities such as they occur in macromolecular structures (in particular, in proteins). The program can be used to detect unknown cavities or to delineate known cavities, either of which may be connected to the outside of the molecule or molecular assembly under study. Optionally, output files can be requested that contain a description of the shape of the cavity which can be displayed by the crystallographic modelling program *O*. Additionally, *VOIDOO* can be used to calculate the volume of a molecule and to create a file containing data pertaining to the surface of the molecule which can also be displayed using *O*. Examples of the use of *VOIDOO* are given for *P*2 myelin protein, cellular retinol-binding protein and cellobiohydrolase II. Finally, operational definitions to discern different types of cavity are introduced and guidelines for assessing the accuracy and improving the comparability of cavity calculations are given.

Introduction

Most substrate–enzyme and ligand–receptor interactions are known to take place inside clefts or cavities that exist in proteins. Within such cavities, a tailor-made micro-environment has been created which chemically favours reactions and interactions between the two species. The study of cavities may give an insight into the mechanism of such interactions and might help in the design of novel ligands, substrates or inhibitors (for example, drugs).

Studying cavities inevitably involves the use of interactive computer graphics techniques. However, simple stick or ball-and-stick or even space-filling Corey–Pauling–Koltun (CPK) representations of protein structure are not ideally suited to this purpose (Ho & Marshall, 1990). Molecular surface displays (Connolly, 1983*a*,*b*, 1985*a*,*b*, 1993) are a step in the right direction since they show the space that a protein occupies, but better still are semi-transparent renderings of the surfaces of the cavities themselves. Such displays facilitate simultaneous analysis of the cavity and its occupant (ligand, substrate, inhibitor) as well as of the protein residues which surround it (and whose side chains are, therefore, expected to interact with the cavity's denizen).

In this paper, the generic term 'cavity' is used to denote any part of three-dimensional (3D) space which is mostly, but not necessarily completely, inside a protein and is not occupied by protein atoms. In order to discern different types of cavity, the following operational definitions are used (see Fig. 1): a void is a cavity which is completely surrounded by the protein and is therefore closed off from the 'outside world'; an invagination is a cavity which is connected to the outside world, but which would be closed off if the atomic radii were to be increased (invaginations may have more than one connection to the outside world); a pocket is a cavity which is connected to the outside world and which cannot be closed off by increasing the atomic radii. In other words, when the atomic radii are gradually increased, an invagination will become a void before it disappears whereas a pocket will not. It is important to realize that the definition of the extent of a cavity is arbitrary (and therefore subjective) in those cases where the cavity is in touch with the outside world.

Testing whether or not a point in space is occupied by a protein atom is usually accomplished by thinking of atoms as 'hard spheres' that have a (van der Waals) radius, which often includes any H atoms attached to the atom ('united-atom approach') (Richards, 1974). Since cavities are only of interest if they can accommodate at least a particular atom, molecule or ion, the accessible surface, rather than the van der Waals surface, is used to delineate the space occupied by the protein. The accessible surface is that described by the centre of a probe of an appropriate radius (typically, a water molecule is used with a radius between 1.4 and 1.6 Å) when it is rolled over the protein's van der Waals surface (Richards, 1974). Note that an invagination may turn into a void if a probe radius is applied.

* Author to whom correspondence should be addressed.

There have been several reports in the literature of algorithms to detect and/or delineate and/or measure and/or visualize cavities in protein structures. Voorintholt, Kosters, Vegter, Vriend & Hol (1989) describe a simple method for visualizing cavities. They map a protein onto a 3D grid and assign every grid point a value, depending on whether it is inside the protein, in between the van der Waals and the accessible surface or outside the accessible surface. By contour plotting this map at appropriate levels one obtains a view of the cavities in the structure. However, their program does not attempt to actually detect, delineate or measure the cavities.

Ho & Marshall (1990) report on a program which fills cavities. However, the user has to supply a 'seed point' (*i.e.* a point in 3D space of which it is known that it is inside a cavity) since the software is unable to detect cavities on its own. Additionally, if the cavity is in touch with the outside world, a distance constraint has to be used in order to prevent the cavity from 'spilling out' into the great wide open.

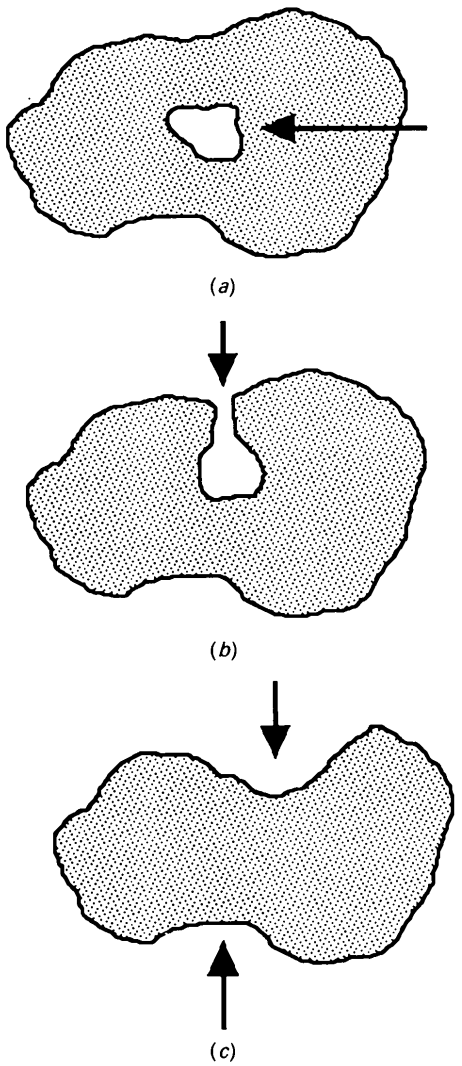

Fig. 1. Classification of cavity types using operational definitions (see text): (*a*) a void, (*b*) an invagination and (*c*) two pockets.

Delaney (1992) uses so-called cellular logic techniques derived from image processing (namely, expansion and contraction of a 3D 'image') in order to detect and delineate cavities in a protein structure which has been mapped onto a 3D logical grid. This approach is very elegant in that it provides a way of visualizing cavities which are not closed off from the outside world (*i.e.* pockets). However, the assertion that this method provides an objective way to define the extent of a cavity is not justified, since the result still depends on the grid spacing and on the orientation of the molecule. Despite its promise, the program seems unfinished, since it does not further analyse the cavities (it only produces a set of dots which can be displayed with graphics software).

Recently, Levitt & Banaszak (1992) published a program called *POCKET* which performs detection, volume measurement and graphics modelling of cavities. However, it uses a large probe radius (typically 3 Å) which is fixed for all atom types, as well as rather coarse grids. In addition, cavities must be closed off, given the radius of the probe, or the program will not detect them.

This paper is the first in a series describing new stand-alone programs that operate around and with *O* (Jones, Zou, Cowan & Kjeldgaard, 1991). Here a program called *VOIDOO* is described which detects and delineates cavities and measures their volume. *VOIDOO* uses a mechanism called 'atomic fattening' in order to be able to detect not only voids, but also invaginations. For every cavity which is detected, output files are created which can be displayed with the macromolecular and crystallographic modelling program *O* (Jones *et al.*, 1991). Cavity surfaces may be rendered as chicken-wire contour models or as semi-transparent surfaces. The built-in modelling tools of *O* can then be used to investigate residues which line the cavity of interest (these residues can be displayed by executing an *O* macro file which is automatically generated by *VOIDOO* for each separate cavity). In addition, *VOIDOO* can be used to produce files for the display of molecular surfaces and to calculate molecular volumes. *VOIDOO* was designed for, but is not limited to, the study of cavities in protein structures. Moreover, the program is fast enough to be run interactively; a typical run for an average-size protein (100–300 residues) requires between two and ten minutes of CPU time on a Silicon Graphics Indigo/Elan R3000 workstation, depending on the choice of parameters. The required CPU time increases approximately linearly with the number of grid points.

Program description

Input

The input to *VOIDOO* consists of three parts. First, a dictionary file has to be provided which contains information about the van der Waals radii of the various atom types as well as a list of residue types which are considered to be part of the molecule of interest (one usually would exclude solvent molecules, salt ions and ligands from this list). Our standard protein-related dictionary contains the van der Waals radii used in *AMBER* (Weiner *et al.*, 1984). They may be set for atomic elements (*e.g.* one may associate with every C atom a radius of 1.85 Å), as well as for specific (pseudo-) atom types [*e.g.* one could associate a radius of 2.0 Å with every CG1 atom of valine residues; Protein Data Bank (PDB) naming conventions are used]. Second, the structure of interest is read from a PDB file. There is no inherent limit to the number of molecules that can be read, so that it is a trivial exercise to investigate cavities which exist in and between two or more molecules [for instance, this enables investigation of the space in between (non-)crystallographically related molecules or other molecular assemblies]. The third part of the input consists of values for the program's main parameters (initial grid spacing, probe radius, *etc.*). When used interactively, the program prompts for all input, and supplies sensible default values.

Cavity detection

The first step of the cavity-detection algorithm is to map the molecule onto a 3D grid with a spacing of, typically, 0.5 to 1.0 Å. Initially, all grid points are assigned a value of zero. Subsequently, every point of the grid whose distance to the nearest atom is less than the sum of the van der Waals radius of that atom and the probe radius, is assigned a value of one. In order to exclude the outside world, all grid points on the faces of the grid are set to zero and subsequently the whole outside world is 'zapped'. Zapping entails finding a point with a certain value (in this case, zero) and setting the value of this point to something else (for example, to 2) and repeating this operation for all its neighbour points which also have a value of zero *etc.*, until no more points are changed. This method is also known as the 'flood-fill algorithm' (Foley & van Dam, 1983).

At this stage, all grid points which are inside a closed cavity still have a value of zero. If the user was looking for a specific cavity, *VOIDOO* checks if this cavity has been detected and, if so, it will go on to delineate the cavity and to measure its volume. Otherwise, the program counts the number of cavities which are larger than a user-specified size and a process called 'atomic fattening' is invoked. This entails multiplying the van der Waals radii of all atoms by a certain factor (the fattening factor, typically 1.05 to 1.2) and repeating the mapping of the protein and the zapping of the outside world. The effect of repeated atomic fattening is to close off

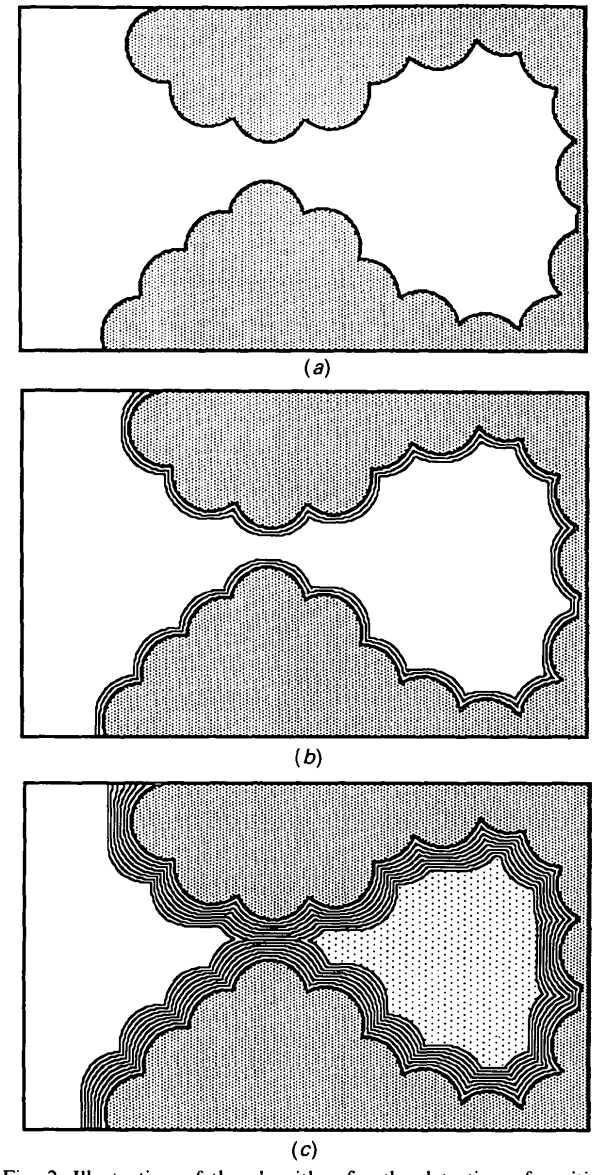

Fig. 2. Illustration of the algorithm for the detection of cavities that are connected to the 'outside world' when all atoms have their normal van der Waals radii, perhaps augmented by a probe radius. (*a*) All atoms have their standard van der Waals radii; the cavity is not completely 'embraced' by the molecule. (*b*) The van der Waals radii of all atoms are repeatedly multiplied by some factor, thereby slowly closing off the entrance to the cavity. (*c*) At a certain stage, the cavity is completely sealed off, in other words, isolated from the outside world. The extent of the cavity can now easily be delineated by the program and its shape and volume can subsequently be determined more precisely.

cavities which are in contact with the outside world (see Fig. 2) and to separate cavities which are connected to one another through small channels. After a number of atomic fattening cycles, the number of cavities usually drops since the atoms become so large that they begin to fill what were previously cavities. If the user was looking for a specific cavity, the program terminates the atomic fattening process as soon as that particular cavity has been found. Otherwise, *VOIDOO* either terminates it when the number of cavities begins to decline, or it repeats it a fixed number of times (*e.g.* ten cycles) and selects the fattening factor which yielded the largest number of cavities.

It should be noted that this algorithm will only detect voids and invaginations; it will not pick pockets (no pun intended). Pockets may be visualized with the help of another program, *MAMA* (GJK & TAJ, to be published), which was developed for creating, editing and manipulating molecular envelopes ('masks') that are used in real-space electron-density averaging (Jones, 1992; GJK & TAJ, to be published). This program contains tools for expanding and contracting masks as well as for logically combining pairs of masks ('AND', 'OR' *etc.*) which enable emulation of Delaney's cavity-detection methods (Delaney, 1992).

Cavity delineation and volume calculation

Any cavity detected is delineated and measured as follows (see also Fig. 3): first, the extent of the cavity is determined from the grid onto which the molecule was mapped. Then a correction is added on all sides consisting of the sum of the probe radius and the product of the largest van der Waals radius and the atomic fattening factor. This part of space is then mapped onto a second grid after resetting the atomic fattening factor to one. Since the program now focuses on only a small part of the whole molecule, the second grid can be made much finer. However, this is not done at first. Initially, the protein is mapped again and all cavity points are zapped and counted (and, from this, the volume of the cavity is calculated). Then, the grid spacing is reduced by a certain user-specified factor (grid-shrink factor, typically 0.75 to 0.95) and the mapping, zapping and counting process is repeated. This continues until convergence is reached (determined by the difference between two subsequent cavity-volume measurements, both absolute and relative; typical criteria are 0.1–0.5 Å^3 and 0.1–0.5%), or until a certain number of cycles has been carried out (typically, between 10 and 25). The average volume and its standard deviation are calculated from all measurements, the centre-of-gravity of the cavity is computed (in case one wants to repeat the calculations for this specific cavity with different parameters) and a file containing a 'mould' of the cavity is generated on a separate, user-defined grid. This mould can subsequently be displayed by *O*, for example as a chicken-wire contour, as a semi-transparent surface or as a dot surface. To aid the user in examining the cavities, *VOIDOO* prints a list of all protein atoms which line a certain cavity as well as all non-protein atoms that are contained inside the cavity. In addition, for each cavity an *O* macro file is generated which, when executed from within *O*, will automatically display all residues inside and around the cavity.

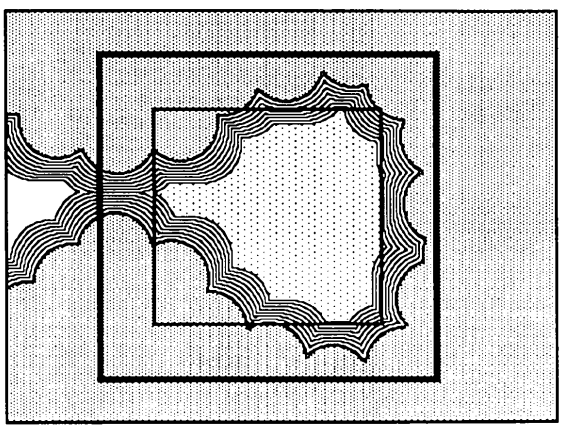

Fig. 3. Determination of the extent of a cavity. First, the minimum and maximum values of the x, y and z coordinates of the cavity points are determined (narrow-lined box). Since the atoms had been artifically 'fattened' by the cavity-detection algorithm, a margin consisting of the sum of the probe radius and the product of the maximum value of the standard van der Waals radii and the fattening factor employed, is added on all sides (broad-lined box). The resulting box is subsequently mapped onto an increasingly finer grid in order to determine the cavity's extent and volume more precisely.

Molecular volume calculations

As an 'encore' to the cavity detection, delineation and measurement capabilities of *VOIDOO*, an option to compute the volume of a molecule has also been implemented. The algorithm is similar to that for the measurement of cavity volumes in that the molecule is mapped onto increasingly finer grids and the volume is obtained by simply counting the number of grid points inside the protein and by multiplying this number by the volume that corresponds to one voxel (volume element). Note that this is not an analytical method.

Display of cavities and molecular surfaces

The plot files generated by *VOIDOO* can, depending on the chosen format, be converted into the standard electron-density map format ('DSN6') for *FRODO* and *O* with the auxiliary program 'map-

page', or read into *O* directly. As any map, cavities and molecular surfaces can be displayed as chicken-wire models. In addition, the Silicon Graphics version of *O* has been modified by Dr Michael Waldherr-Teschner (Silicon Graphics Computer Systems AG, Riehen, Switzerland) to enable the display of semi-transparent surfaces.

Accuracy, precision and standardization

By repeating the volume calculations of cavities and molecules on increasingly finer grids, one source of error (and cause of irreproducibility) has been eliminated. The standard deviation of the average volume is nevertheless more an indicator of precision than of accuracy. This can be demonstrated quite easily for the case of molecular-volume calculations. To this end, the volume of a set of 11 non-intersecting spheres with a radius of 1.0 Å, the 'molecular volume' of which can be analytically computed to be 46.08 Å^3, was calculated with *VOIDOO*. Starting on a grid with a spacing of 1.0 Å, 20 cycles of volume calculations were executed using a grid-shrink factor of 0.9. The results are shown in Fig. 4, where the dotted line indicates the analytical value of the volume. This figure shows that the calculated volume converges slowly towards its real value, but only when the grid spacing is less than approximately 0.3 Å does one obtain values which are different by less than ~0.5%. Also, the average value after 20 cycles happens to be quite close to the analytically computed value, namely 46.15 Å^3 (standard deviation 1.3 Å^3).

The accuracy of cavity volumes is intrinsically much harder to assess. The numerical value that is obtained depends on (obviously) the probe radius, but also on the orientation of the molecule (since it is mapped onto a discrete grid). Moreover, for invaginations, the definition of the cavity extent (and, hence, its volume) is dependent on the probe radius and the orientation as well. In addition, molecular speleologists are often plagued by the 'can-of-worms problem'. This problem is illustrated in Fig. 5: since the atoms are fattened during cavity detection and a

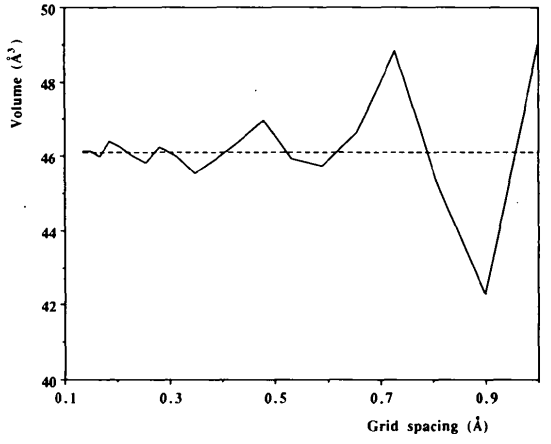

Fig. 4. Assessing the accuracy of molecular-volume calculations. A set of 11 non-intersecting spheres with a radius of 1.0 Å was used. The starting grid spacing was 1.0 Å and the grid-shrink factor was 0.9. 20 cycles of volume calculations were carried out and the volumes obtained have been plotted as a function of the grid spacing. The dotted line indicates the true (analytically computed) value of the volume, 46.08 Å^3.

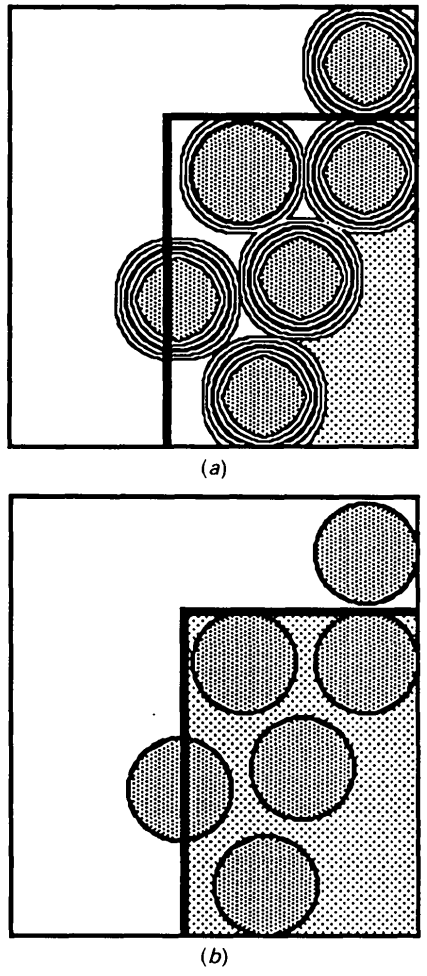

Fig. 5. Illustration of the 'can-of-worms' phenomenon. If the atoms needed to be fattened in order to detect a cavity, this indicates that the atoms of the molecule do not seal off the cavity with their standard van der Waals spheres (plus the probe radius, if it was applied). (*a*) Shows a case in point where a cavity close to the molecule's surface was detected after repeated 'fattening' of the atoms. The cavity-bounding box (broad lines) now includes a small part of the 'outside world'. (*b*) Shows what happens if the cavity is subsequently mapped onto increasingly finer grids: the atoms have assumed their standard van der Waals radii again, leaving small channels between the cavity and the outside world which the program considers to be part of the actual cavity. The best remedy is to apply the probe radius during cavity delineation and measurement as well and to repeat the calculations using several randomly rotated copies of the molecule.

safety margin is applied in their delineation, the program is bound to include small, dendritic channels between the atoms of the molecule which, intuitively, would not be considered to be part of the cavity. In the worst case, these channels lead into the outside world. One way to circumvent this problem which is often effective, is to apply the probe radius also while delineating and measuring cavities. An additional benefit of this approach is that one obtains cavities which are 'negative moulds' of the protein's (solvent-) accessible surface, rather than of its van der Waals surface; in other words, the cavity delineates the space available to the centre of a probe sphere. In analogy to the names given to molecular surfaces, a cavity which was delineated with a probe is called an accessible cavity, and one which was delineated without any probe a van der Waals cavity.

In order to have an independent check on the precision of the cavity volumes, a third cavity-delineation option was implemented which emulates the method used by Connolly's *MS* program (Connolly 1983*a*,*b*, 1985*a*,*b*). In this approach, the cavity volume is defined as the volume swept out by a probe rolling over the molecule's surface. *VOIDOO* creates such cavities by first delineating the accessible cavity and then adding all points around it which are not occupied by protein atoms and which have a distance to the surface of the accessible cavity which is less than the probe radius. This option was tested on the phage T4 lysozyme mutant L99A (Eriksson *et al.*, 1992); using a probe radius of 1.2 Å, the cavity volume obtained with *MS* is 149.5 Å^3. Using the same coordinates, atomic radii and probe radius, *VOIDOO* finds a volume of 150 (10) Å^3 (ten calculations using grid spacings between 0.295 and 0.114 Å; see Fig. 6). Since *MS* uses an analytical algorithm to compute the volume occupied by the probe spheres, this result indicates that the errors in the precision due to the use of discrete grids are small and are indeed averaged out by repeating the calculations on increasingly finer grids. In practice, however, we prefer to calculate accessible cavity volumes rather than *MS*-style 'probe-occupied' volumes since: (*a*) probe-occupied cavities can only be reliably used if the cavity is closed when the atoms have their normal van der Waals radii; (*b*) accessible cavities show the volume of space available to the centre of the probe sphere, which is more meaningful when stick models of ligands *etc.* are used; (*c*) probe-occupied cavities must be calculated on fine grids in order to yield reliable results, which increases the amount of CPU time needed for the calculations.

In order to obtain an estimate of the accuracy of cavity volumes, it is recommended that one repeats the *VOIDOO* calculations, for example ten times, with exactly the same input parameters, but with randomly rotated copies of the molecule(s). In this fashion random errors are averaged out. The average value thus obtained as well as its standard deviation should then be reported, together with some details of the calculations (probe radius, grid spacing, atomic fattening factor, grid-shrink factor, number of volume calculation cycles and number of randomly oriented structures used). At the very least, this will enable others to repeat the calculations. Moreover, this procedure yields values which are comparable (as opposed to apples and oranges) in case one investigates different, but related molecules (for example, wild-type and mutant structures). Naturally, systematic errors cannot be compensated for with this method. Such errors are due to the use of the 'hard-spheres' model of a molecule, to the use of the 'united-atom' approach and to the use of one, static structure of a molecule.

Software details

VOIDOO has been written in Fortran-77; it runs on Silicon Graphics Iris and Indigo as well as DEC Alpha/OSF1 and Evans & Sutherland ESV workstations. Visualization of cavities and molecular surfaces requires *O*, which runs on a variety of workstations although semi-transparent surfaces are currently only supported in the Silicon Graphics version of this program. For details regarding the availability of *VOIDOO* and/or *O*, contact TAJ (e-mail: 'alwyn@xray.bmc.uu.se').

Applications

P2 myelin protein

P2 myelin protein is a member of a family of hydrophobic ligand-binding proteins. It occurs in peripheral nervous system myelin and probably functions as part of a lipid transport and storage system in myelinating Schwann cells. Its structure was solved and refined in our laboratory (Jones, Bergfors, Sedzik & Unge, 1988; Cowan, Newcomer & Jones, 1993). The ligand, a fatty acid, is bound inside an invagination (Fig. 7) between the two β-sheets. The effectiveness of the method of using multiple, randomly oriented copies of a molecule to assess the accuracy of the cavity and molecular-volume calculations was tested using this protein; the results have been collected in Tables 1 and 2. They demonstrate the following points: (*a*) the values obtained for the molecular volumes are hardly affected by changes in the orientation of the molecule. Since the standard deviations for individual calculations (which convey information regarding the sensitivity to different grid spacings) are larger than that of the whole set (pertaining to the sensitivity to the orientation), it suffices to calculate volumes only once (for the non-rotated molecule); (*b*) the cavity volumes are quite

sensitive to the orientation of the molecule, in particular in the case of the van der Waals cavity where the volumes differ by as much as 20%; (*c*) the standard deviations of individual cavity volumes are not a good measure of accuracy. In this case, one would report the volume of protein as 13890 (10) Å^3, that of the fatty acid as 335.1 (0.2) Å^3, that of the accessible cavity as 272 (6) Å^3 and that of the van der Waals cavity as 1220 (80) Å^3.

CRBP

Cellular retinol-binding protein (CRBP) is structurally and functionally related to *P*2 myelin protein; its structure was determined and refined in our laboratory (Cowan *et al.*, 1993). The retinol ligand is bound inside a void (when using a 1.4 Å probe radius) rather than an invagination (Fig. 8). On using the random-orientation method, one finds that the volume of the accessible cavity is virtually independent of the orientation of the molecule. Ten such calculations (data not shown) yield an average volume of 174.5 (0.5) Å^3, with standard deviations of individual runs varying between 0.3 and 0.8 Å^3. This observation can easily be explained: since the cavity is a void, it is closed off when the atoms have their normal van der Waals radii. Therefore, there is no orientation-dependent can-of-worms effect which ensures that the volume calculation only suffers from errors due to the different grid spacings (as is the case for molecular-volume calculations).

The volume of the protein is 14440 (20) Å^3, that of the retinol 337 (1) Å^3 and that of the van der Waals cavity 1350 (100) Å^3.

*CBH*II

Cellobiohydrolase II (CBHII) is an enzyme with exoglucanase activity, *i.e.* it is able to break down cellulose by sequentially cleaving off terminal cellobiose disaccharides. The structure of this enzyme has been solved and refined in our laboratory (Rouvinen, Bergfors, Teeri, Knowles & Jones, 1990; Rouvien, Knowles & Jones, 1994). The substrate binds in a tunnel-shaped invagination (Fig. 9) which penetrates the protein and which is open at both ends. The

Fig. 6. View of the probe-occupied benzene-containing cavity inside the structure of the phage T4 lysozyme mutant L99A (Eriksson *et al.*, 1992). The cavity is shown in red and a CPK model of the benzene molecule in blue. The mutated residue Ala99 has been drawn with wide bonds.

Fig. 7. Clipped view of *P*2 myelin protein, the accessible cavity (red) and a stick model of the fatty acid ligand. The residues of *P*2 have been coloured from red to blue going from the N terminus to the C terminus.

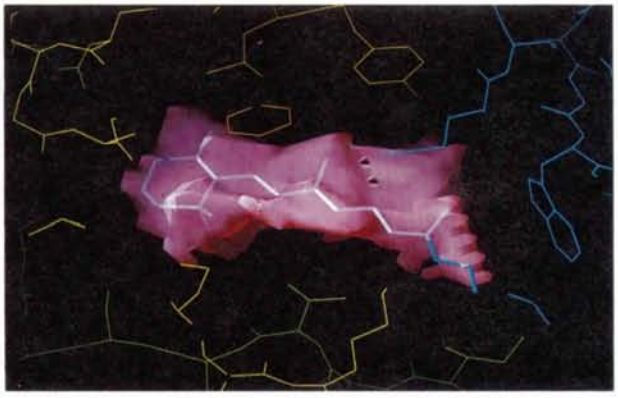

Fig. 8. Close-up of the accessible cavity (red) inside the structure of CRBP with a stick model of the retinol ligand. The residues of CRBP have been coloured from red to blue going from the N terminus to the C terminus.

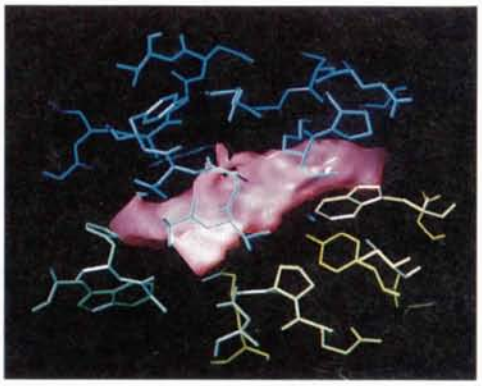

Fig. 9. Accessible cavity (red) in CBHII and the residues that surround it. The residues of CBHII have been coloured from red to blue going from the N terminus to the C terminus.

Table 1. *Results of cavity calculations for P2 myelin protein*

The refined structure of *P2* myelin protein was used (Cowan *et al.*, 1993). Nine copies of the molecule were generated by applying randomly selected rotations around the *x*, *y* and *z* axes. Cavities were detected on a 0.5 Å grid, using a probe radius of 1.4 Å and an atomic fattening factor of 1.1. In all cases except model 8, a fattening factor of 1.464 yielded the fatty acid cavity (for model 8 it was 1.331, *i.e.* one cycle earlier). The cavities were subsequently delineated and measured both with applying the probe (yielding the accessible cavity) and without (van der Waals cavity). Ten volume calculation cycles were carried out, using a grid-shrink factor of 0.9; only the accessible cavity volume of model 4 converged in fewer cycles (namely four; convergence criteria were 0.01 Å^3 and 0.01%). The average volume of the accessible cavity is 272 Å^3 with a standard deviation of 6 Å^3; for the van der Waals cavity these values are 1218 and 83 Å^3, respectively. For each model, the following have been tabulated; the rotation angles used to generate the structure, the volume (*V*), its standard deviation (*S*) and the CPU time it took to perform calculations for the accessible and the van der Waals cavities, respectively. CPU times were measured on a Silicon Graphics Indigo/Elan R3000 workstation and they include the complete cavity detection, delineation and volume measurement as well as the time needed to produce a plot file of the cavity on a 0.5 Å grid.

	Rotation angles ( )			Accessible cavity			van der Waals cavity		
Model	*x*	*y*	*z*	*V* (Å^3)	*S* (Å^3)	CPU (s)	*V* (Å^3)	*S* (Å^3)	CPU (s)
P2 default	0.0	0.0	0.0	265	0.5	527	1315	18	565
1	112.5	141.9	283.9	274	0.7	566	1091	19	612
2	329.3	306.1	223.0	266	0.8	519	1262	10	526
3	285.7	343.3	137.4	280	1.2	575	1239	28	688
4	116.4	101.2	195.7	275	1.5	555	1328	23	547
5	25.2	7.7	7.3	267	0.7	442	1166	22	460
6	196.9	328.7	257.1	279	0.6	480	1150	11	523
7	213.9	33.8	170.5	263	0.5	527	1218	8	561
8	305.0	344.6	249.1	275	0.9	507	1291	37	541
9	97.6	218.6	94.8	271	0.9	443	1121	17	479

Table 2. *Results of molecular volume calculations on P2 myelin protein and its ligand*

The refined structure of *P2* myelin protein was used (Cowan *et al.*, 1993). Nine copies of the molecule were generated by applying randomly selected rotations around the *x*, *y* and *z* axes (Table 1). All protein atoms were assigned their default *AMBER* van der Waals radii; for the fatty acid ligand a radius of 1.96 Å was used for all C atoms and one of 1.60 Å for all O atoms. Volumes were computed starting on a 0.5 Å grid for the ligand and a 1.0 Å grid for the protein. A maximum of 25 volume calculation cycles were carried out, using a grid-shrink factor of 0.9 for the ligand and 0.95 for the protein; however, all volume calculations converged in fewer than 25 cycles (convergence criteria were 0.01 Å^3 and 0.01%). The average protein volume is 13893 Å^3 with a standard deviation of 7 Å^3; the average ligand volume is 335.1 Å^3 with a standard deviation of 0.2 Å^3.

	Protein volume			Ligand volume		
Model	*V* (Å^3)	*S* (Å^3)	*N*	*V* (Å^3)	*S* (Å^3)	*N*
P2 default	13900	28	8	335.1	0.6	7
1	13900	14	7	335.1	0.4	10
2	13890	13	23	335.2	0.3	15
3	13890	18	18	335.2	0.8	7
4	13900	16	18	335.1	0.3	15
5	13900	17	19	335.5	0.5	11
6	13880	27	5	335.2	0.7	17
7	13890	20	13	335.0	0.9	5
8	13890	17	19	334.8	0.7	4
9	13890	15	23	335.2	0.5	18

volume of the accessible cavity is 250 (20) Å^3, which is about one-third of that of the van der Waals cavity [750 (100) Å^3].

This work was supported by the Netherlands Organization for Scientific Research (NWO) through a postdoctoral fellowship to GJK and by the Swedish Natural Science Research Council. We are grateful to Dr Michael Waldherr-Teschner (Silicon Graphics Computer Systems AG, Riehen, Switzerland) for implementing semi-transparent surface rendering in *O*, and to Frau Dr Astrid Elisabeth Sauer-Eriksson for providing us with the coordinates of the L99A lysozyme mutant and the details of the cavity calculations performed on this protein with *MS*.

Note added in proof: To address the question 'how many solvent molecules could fit inside a cavity?', we have written a small companion program called *FLOOD*. This program attempts to pack as many as possible solvent spheres in a solvent-accessible cavity which has been delineated by *VOIDOO*. *FLOOD* stores these solvent positions in a PDB file and it creates an *O* macro to display them. For example, using CRBP we find that the ligand-binding cavity contains enough space to house at least 26 water molecules if they are packed tightly.

References

CONNOLLY, M. L. (1983*a*). *Science*, **221**, 709–713.
CONNOLLY, M. L. (1983*b*). *J. Appl. Cryst.* **16**, 548–558.
CONNOLLY, M. L. (1985*a*). *J. Am. Chem. Soc.* **107**, 1118-1124.
CONNOLLY, M. L. (1985*b*). *J. Appl. Cryst.* **18**, 499–505.
CONNOLLY, M. L. (1993). *J. Mol. Graphics*, **11**, 139–141.
COWAN, S. W., NEWCOMER, M. E. & JONES, T. A. (1993). *J. Mol. Biol.* **230**, 1225–1246.
DELANEY, J. S. (1992). *J. Mol. Graphics*, **10**, 174–177.
ERIKSSON, A. E., BAASE, W. A., ZHANG, X. J., HEINZ, D. W., BLABER, M., BALDWIN, E. P. & MATTHEWS, B. W. (1992). *Science*, **255**, 178–183.
FOLEY, J. D. & VAN DAM, A. (1983). *Fundamentals of Interactive Computer Graphics*. Reading, Massachusetts: Addison-Wesley.
HO, C. M. W. & MARSHALL, G. R. (1990). *J. Comput. Aided Mol. Des.* **4**, 337–354.
JONES, T. A. (1992). *a,yaap,asap,(a #*? A Set of Averaging Programs. Molecular Replacement*, edited by E. J. DODSON, S. GLOVER & W. WOLF, pp. 91–105. Warrington: SERC Daresbury Laboratory.
JONES, T. A., BERGFORS, T., SEDZIK, J. & UNGE, T. (1988). *EMBO J.* **7**, 1597 1604.
JONES, T. A., ZOU, J. Y., COWAN, S. W. & KJELDGAARD, M. (1991). *Acta Cryst.* A**47**, 110–119.
LEVITT, D. G. & BANASZAK, L. J. (1992). *J. Mol. Graphics*, **10**, 229–234.
RICHARDS, F. M. (1974). *J. Mol. Biol.* **82**, 1–14.
ROUVINEN, J., BERGFORS, T., TEERI, T., KNOWLES, J. & JONES, T. A. (1990). *Science*, **249**, 380–386.
ROUVINEN, J., KNOWLES, J. & JONES, T. A. (1994). In preparation.
VOORINTHOLT, R., KOSTERS, M. T., VEGTER, G., VRIEND, G. & HOL, W. G. J. (1989). *J. Mol. Graphics*, **7**, 243–245.
WEINER, S. J., KOLLMAN, P. A., CASE, D. A., SINGH, U. C., GHIO, C., ALAGONA, G., PROFETA, S. & WEINER, P. (1984). *J. Am. Chem. Soc.* **106**, 765–784.