Journal of Computer-Aided Molecular Design, 3 (1989) 133–147
ESCOM

J-CAMD 062

Determination of clefts in receptor structures*

Richard A. Lewis

Department of Pharmacology, University of Cambridge, Hills Road, Cambridge CB2 2QD, U.K.

Received 21 November 1988
Accepted 22 January 1989

Key words: Automatic determination of clefts; Drug design; Binding sites; Dihydrofolate reductase; Trypsin; Voronoi tessellation

SUMMARY

The automatic determination of atoms which comprise a cleft in a receptor is of great importance in computer-aided drug design. X-ray studies of ligand/receptor pairs show that the ligand is often located in a cleft so that this structural feature will indicate a putative binding site. This information can be used in the design of new drugs by database searching and by automatic structure generation. The methods presented in this paper will find the complete accessible surface in a slice through a receptor and also all the clefts and dimples in this surface, using the properties of the Voronoi tessellation of the receptor. Clefts and binding sites can now be determined quickly and without observer bias.

INTRODUCTION

The shape of a biological receptor is basically similar to that of an irregular globule. There are pits, grooves, protuberances and clefts in the surface of this structure, together with small-scale irregularities at the atomic level. How then does the high specificity of a receptor for a particular ligand arise from such common features? The overall shape of a receptor may strongly influence the preliminary recognition and initial docking manoeuvres of the ligand via the disposition of electric fields [1,2] but differences in field structure alone will be insufficient for detailed discrimination at the stage of complex formation.

If a ligand molecule is surrounded in space by a number of site points in a unique geometric arrangement, then it can bind to these groups and form a strong complex with the host. The ligand may be partially enveloped by the receptor, and this can only happen in regions of negative curvature on the receptor surface, that is, in clefts, dimples, grooves and behind protuberances [3]; these features are illustrated in Fig. 1. X-ray studies of ligand/receptor pairs show that the ligand is often located in a cleft so that this structural feature will indicate a putative binding site. This infor-

*Prospective users of the program described herein are invited to contact the author.

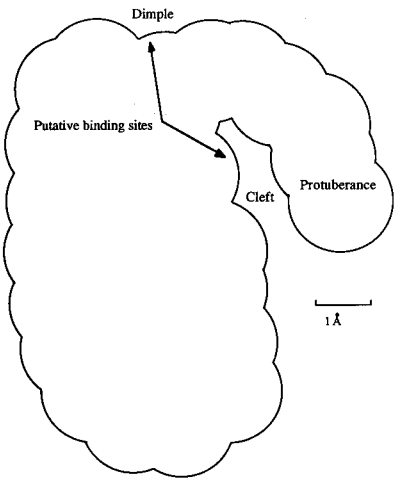

Fig. 1. The surface of a receptor showing a cleft, a dimple and a protuberance. These features are often associated with ligand binding sites.

mation can be used in the design of new drugs by database searching [4] and by automatic structure generation [5,6]. Hence methods for finding clefts are crucially important to the field of drug design. This paper describes an algorithm for finding clefts in the surface of a receptor.

The definition of an accessible surface

The van der Waals surface of a molecule is formed by union of the van der Waals surfaces of the constituent atoms. There will be some regions which cannot be reached by an approaching external atom due to invaginations in the molecular surface. This leads to the definition of two further molecular descriptors, the accessible surface and the contact surface [7,8], illustrated in Fig. 2. A probe sphere of set radius is rolled over the van der Waals surface of the molecule; the contact surface is the surface mapped out by that part of the probe sphere which actually touches the molecule. The accessible surface is swept out by the centre of the probe sphere. The shape of the accessible surface is very dependent on probe size: the large sphere, R2, will roll over small features such as dimples, whereas the small sphere, R1, picks out a cleft at X.

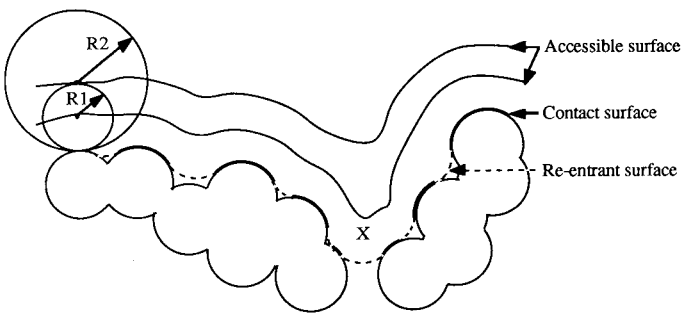

Fig. 2. The definition of the various types of surface found at an exposed section of a molecule.

The physical interpretation of an accessible surface is that it marks the closest points of approach of the nucleus of an external atom to the molecule. If the probe sphere is made the same size as the circumsphere of the solvent, say water, then the regions in which solvent may or may not be bound are highlighted. A corollary to this is that the atoms in the molecule whose van der Waals surface intersects the contact surface may interact with external atoms from a ligand; the other atoms are buried inside the body of the molecule and cannot participate in bonding.

The first computation and graphical realisation of accessible surfaces by Greer and Bush [9] has been followed by more sophisticated methods due to Connolly which give either an analytical formulation of the surface [10,11] or a triangular tessellation [12]. Connolly's algorithm allows the visualisation of the atoms and groups which can participate in ligand binding, but it does not automatically pick out the atoms which define clefts: this is left to the vagaries of a non-objective human interpreter. It is therefore desirable to seek methods to identify clefts automatically and hence putative binding sites from scratch.

The algorithm presented here rests on an application of the Voronoi tessellation. This technique has been used to study surface solvation sheaths of small molecules with respect to the solvent [13] and the solute [14]. Recent work [15] has suggested that the relative volumes of domains created by the tessellation can be used to identify clefts. The volume method is discussed and found to be complementary to this work.

METHODS

The aim of this algorithm is to locate automatically putative binding sites on a receptor surface. There are three basic subgoals to be solved: (i) surface determination in convex regions of the receptor; (ii) surface determination in concavities such as dimples (Fig. 1); and (iii) the identification of clefts which may encompass regions of positive and negative curvature. The method presented here uses the Voronoi tessellation to provide a convenient data structure for keeping track of complex geometric interrelationships and to decrease execution times.

The properties of a Voronoi tessellation

A Voronoi tessellation divides a given space into domains with the fundamental property that all points within a certain domain lie closer to the 'centre' of that domain than to any other centre. Consider a collection of n points P_i in 2-dimensional Euclidean space. This space is covered by n polygons, each of which contains just one of the points P_i. Let x be some other point in the polygon V_i. Then for all $x \in V_i$

$$\text{Distance}(P_i \rightarrow x) \leqslant \text{Distance}(P_j \rightarrow x) \qquad j = 1, 2, \ldots n \tag{1}$$

This gives a 'frog-spawn' pattern of polygons covering the plane as shown in Fig. 3. All the domains are convex and the partitioning is unique. This partition was first described by Voronoi [16] and investigated more thoroughly by Rogers [17]. Centres on the fringes of the tessellation may have domains which are unbounded or abnormally large; the procedure for dealing with the surface of a cluster is not formally defined and some arbitrary boundary conditions must be introduced. In the 2-dimensional case, a large square is placed around the region of the data points. Its

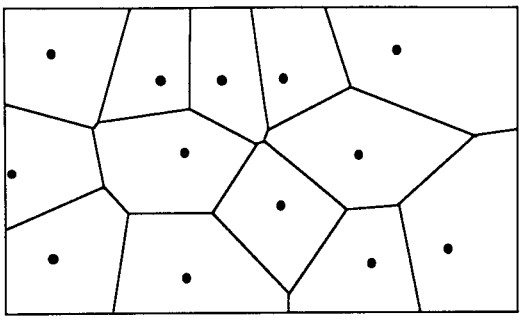

Fig. 3. The Voronoi tessellation of a set of random points performed within a rectangular boundary.

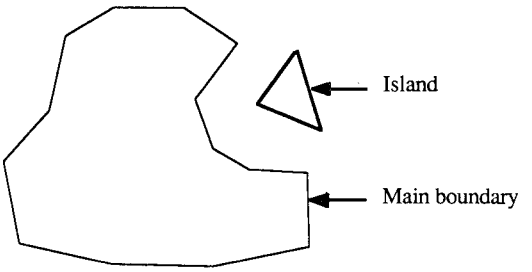

Fig. 4. The surface of a receptor as defined by a closed polygon through atomic centres. An island, separate from the main boundary, has formed; this is usually an artefact which is dependent on probe size.

size is not that important and has little effect on the running times of most tessellation algorithms. The problem of boundary conditions is dealt with more fully by Green and Sibson [18].

Construction of a Voronoi tessellation around a set of n points is a computationally complex problem, and several algorithms have been proposed in the literature [18–23]. The most efficient tessellation algorithm in 2 dimensions is that of Green and Sibson [18], which is about 5 times faster than the other routines [23]. It also has the advantage that it is not a once-and-for-all method; it can deal with changes in data once the tessellation has been computed. This is crucial to the surface-finding program, which involves the addition of dummy data points to the distribution. A summary of the Green and Sibson algorithm is given in the Appendix.

Automatic determination of clefts is a very difficult task and cannot be solved quickly. Simplifying assumptions have to be made for the task to be tractable. The 2-dimensional problem will be examined first, and the ways in which the general principles of this method can be extended into 3 dimensions will be discussed later. Consider a slice of finite thickness taken through a receptor. Any atomic surface which cuts the mid-plane of this slice may form part of the receptor surface within the slice. This will depend on the distance of the atom from the plane and the van der Waals radius of the atom. The van der Waals surface formed by the union of the circles from the projections onto the plane will be a good approximation to the true surface of the receptor in the plane, and to all planes within the slice if the thickness of the slice is small. The protein crystallographic data is input in Brookhaven format, and is windowed to exclude the atoms which do not impinge on the slice.

Assume for the time being that the surface has been computed and the identities of the atoms comprising the surface are known. As these atoms are approximately coplanar, their centres can be connected to form a closed polygon (Fig. 4) in the simplest case. This is called the main boundary. It is possible that there may be one or more other atoms set apart from the main body. These will form an island if the probe can move between these atoms and those in the main boundary. The formation of such islands is dependent on probe size and will be observed if a perpendicular protuberance on the receptor intersects with the slice.

The determination of a smooth convex surface

The simplest type of surface is one which is entirely convex, e.g., an ellipsoid. The surface can be approximated by a convex hull polygon; this has the property that any pair of vertices can only be joined by a line along an existing edge or by a line which cuts the polygon twice. The data is sequentially searched for the point with the lowest y-coordinate as this must form one of the vertices. The remaining points are rescanned and the angle θ between the line A-P_i and the X-axis is calculated. The point associated with the minimum value of θ corresponds to the next vertex (Fig. 5). The algorithm continues round, folding the surface like wrapping paper around a gift.

If the gift-wrapping algorithm is applied to a Voronoi tessellation, the scope and power of this method can be extended. The collection of points shown in Fig. 5 is tessellated within an unbounded area (Fig. 6). As before, the construction starts from point A. The program then sweeps round in an anticlockwise direction until the first neighbour C is found. The negative Y-axis or the vector-$\boldsymbol{j}$ is used for reference. As only the nearest neighbours to A share a common edge, and these edges have already been sorted into cyclic order on the value of θ, the search is trivially simple. The new reference vector is given by the line CA and the sweep proceeds round to find the next vertex, I. The process continues until the program returns to its starting point A.

The presence of boundary conditions does not affect this algorithm as the boundaries are made effectively infinite and Voronoi edges which correspond to the boundary conditions are given a special flag and are ignored in the search for the next vertex.

An algorithm for determining clefts and dimples

The convex hull of a cluster of points corresponds to the accessible surface of the cluster for a probe of infinite radius, that is, a straight line or plane. In the physical context of a protein surface, it is obvious that this is not a very meaningful or valid approximation to the 'true' accessible surface. Every atom has a van der Waals shell associated with it and the extent to which this shell is cut by the slicing plane varies from atom to atom. The atoms are therefore modelled as 'hard spheres' following the standard convention, and the appropriate radius of each sphere is calculated from the element type and position of each atom with respect to the plane. Next, the accessible

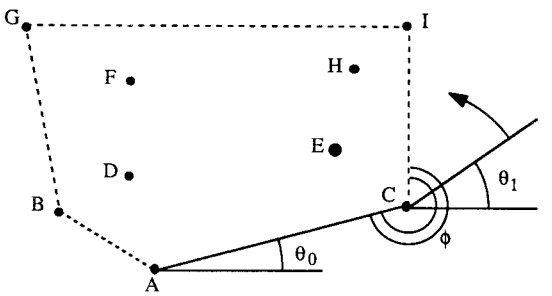

Fig. 5. The convex hull ACIGB is constructed side by side. AC is found first, and then a line from C is swept round anticlockwise until the next vertex, I, is encountered.

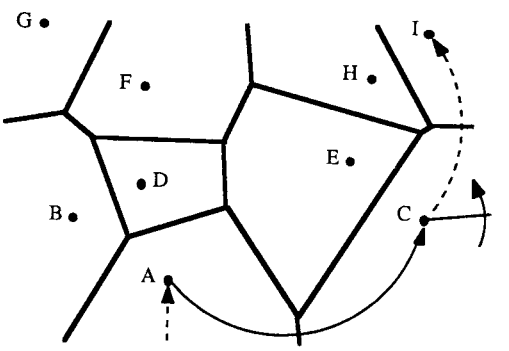

Fig. 6. The efficiency of the 'gift-wrapping' algorithm is improved by first constructing the Voronoi tessellation, as the next vertex of the hull must be one of the nearest neighbours.

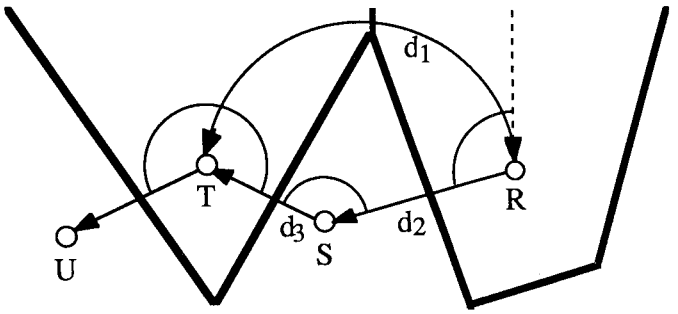

Fig. 7. Computation of the set of surface atoms guided by a Voronoi tessellation.

surface must be defined by declaring an arbitrary probe size, r_p. The condition for surface closure is:

$$d < \rho \quad \text{where} \quad \rho = 2 \cdot r_p + R_i + R_j \tag{2}$$

R_i and R_j are the sphere radii associated with atoms i and j respectively.

The Voronoi search is started as before except this time the distance, d_i, between the existing vertex, R, and the prospective new vertex, T, is calculated (Fig. 7). If $d_1 > \rho$, then the probe can just slip through between R and T and the surface is not defined by the edge RT. The next step is to find the first common neighbour to R and T, S, again sweeping anticlockwise. The distance d_2 is calculated and tested. If $d_2 < \rho$ then the surface is complete between R and S and the new vertex is S. The geometry of the construction means that T must be the next prospective vertex so no new search is needed. d_3 is now computed and it too passes the probe condition so that another section is complete. This method will deal with small dimples.

A new strategy must be used to deal with clefts. In the section of surface shown in Fig. 8, the surface between AB has been found and the next point to be tested is F. It is found that both the distance between B and F and the distance between B and the common neighbour E is too large to prevent the passage of the probe. A traffic island which can direct the search algorithm around the correct path BCDEF is required. This is achieved by the simple expedient of placing a dummy tile halfway between B and F using the Green and Sibson routine. A very small extra displacement is added to avoid problems with the new centre lying exactly on the old perpendicular bisector be-

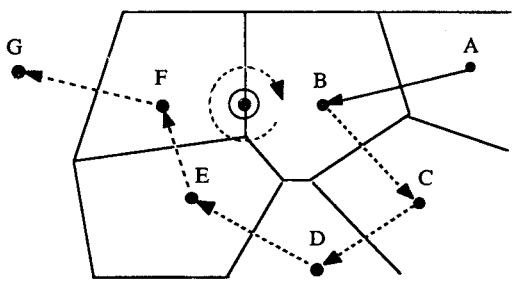

Fig. 8. The surface cannot be completed unless a 'traffic island' is added to push the algorithm around the path BCDEF.

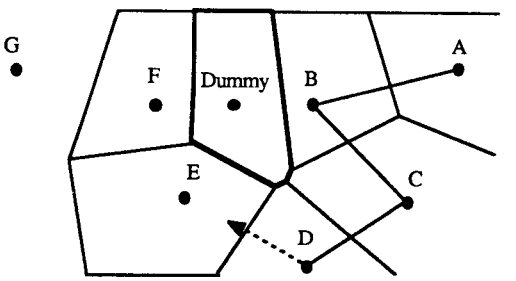

Fig. 9. The addition of a dummy tile means that the set of surface atoms can be quickly found, and also that the dimple in that region is clearly marked.

tween B and F. The edges due to this new polygon are given a special flag and the search moves directly on past them (Fig. 9) thus rapidly determining the new surface. A second advantage of this method is that the dummy polygon is left marking the cleft and the atoms which bound the cleft are the neighbours of this polygon.

If the cleft is a large feature then a 'clump' of contiguous tiles will be formed (Fig. 10). The surface search starts at A and looks at G; let $d_{AG} > 2 \cdot \rho$ so that there is no common neighbour and a dummy tile X1 is added forcing the search to B. If $d_{BF} > 2 \cdot \rho$ then another tile X2 must be added; gradually the whole channel is filled with dummy tiles. The number of tiles in the clump gives an indication of the size of the surface feature which they mark.

Islands can be isolated by the dummy tile method too. Consider the one-atom island F (Fig. 11). By definition, each island is separate from the main boundary so that $d_{iF} > \rho$ for i = A, B,...E. The search will always be trying to complete a section of the surface to F until $d_{iF} > 2 \cdot \rho$ and a dummy tile (Xn) is added to redirect it. The island will be left as an isolated atom contiguous to a dummy tile but which is not part of the main surface. It may even be completely surrounded by dummy tiles. The argument can be extended to multi-atom islands.

The FindSurf routine implements the local surface-finding algorithms discussed above, breaking down the whole surface into many smaller tasks which involve the determination of the surface between Voronoi neighbours. There are three cases of subtask: (i) the local surface between the two neighbours is complete and nothing further need be done; (ii) the local surface contains a small dimple defined by a common neighbour, which must be found and the two new sections of surface tested (Fig. 7); and (iii) the local surface contains a cleft, requiring a dummy tile to be added to the tessellation (Figs. 8 and 9). As discussed above, this may lead to the formation of an island so that the integrity of the surface around both neighbours must be verified (Fig. 11). All three subtasks end by returning the two new neighbours to be used in the next iteration. This process is continued until one of the termination criteria is met: these criteria are that either a user-

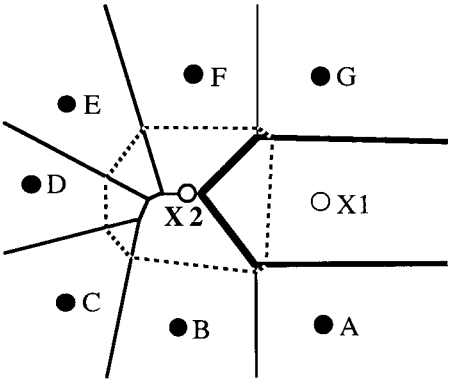

Fig. 10. The cleft described by atoms A to G is being marked with a clump of dummy tiles. Dummy X1 has been added and X2 is being inserted to direct the search away from BF and towards BC.

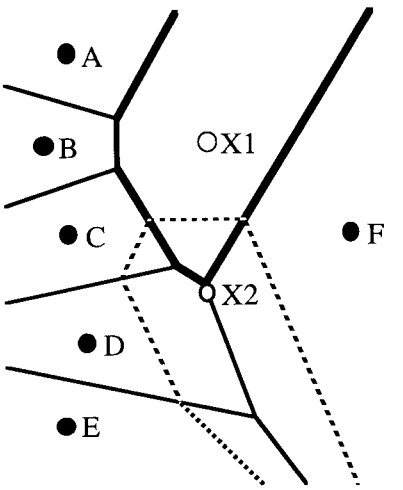

Fig. 11. The island tile F is separated from the main boundary by the addition of dummy tiles X1 and X2.

specified atom or the first atom on the surface list has been reached. The latter case signifies the total determination of the main boundary rather than an island, because small loops in the list (involving less than 7 atoms) are removed as soon as they are formed. These small loops are saved separately as islands. This routine is summarised in flowchart form in Fig. 12.

In this way, the surface of a slice through any protein can be determined and the clefts, dimples and islands readily distinguished. The algorithm will pick out both small and medium features of the receptor surface with the same size of probe sphere.

RESULTS

All the routines have been extensively tested on randomly generated data sets and have proved to be robust and efficient. For a typical run of 500 points, the tessellation takes 1.2 s of CPU time, and the determination of the surface of the slice takes 0.8 s of CPU time on an IBM 3084Q.

The output of the site-finding program can be obtained in two forms, either as a graphical display or as a series of variables suitable for printing or passing to other programs. The information produced from each run consists of: (i) a list of atoms which form the complete closed surface; (ii) lists of atoms which form each dimple and cleft in the surface; and (iii) a list of the atoms which form part of the largest cleft in the surface. Every set of atoms produced in categories (ii) and (iii) is a subset of the total surface. The set of atoms which make up a dimple is the set of neighbours of an isolated single dummy tile. Similarly, the atoms which make up a cleft come from neighbours of a clump of adjacent dummy tiles.

The program was tested by examining a slice constructed through the pteridine ring of methotrexate binding to the enzyme dihydrofolate reductase using the X-ray coordinates [24] contained in the Brookhaven database. Figure 13 shows the 'true' polygons in white, the dummy tiles in red and the set of surface atoms in green. The scale bar is 10 Å. There is a large cleft on the middle right-hand side of the surface. There are several small dimples all around the structure. When the structure of the pteridine rings of methotrexate is added in a close-up view of Fig. 13, it can be seen that the ligand is occupying this cleft picked out by program (Fig. 14). This time only the atoms which are neighbours of the cluster of dummy tiles have been included.

Similar results were obtained when a slice through the amidinophenylpyruvate (APPA) binding

FindSurf(i,j)

1. Find the distance d_{ij} between the current atom, i, and the neighbour, j.
2. Compute $\rho = 2r_p + R_i + R_j$.
3. Perform the probe test:
   (a) $d_{ij} < \rho$; the surface between i and j is complete. J is now the current atom.
   (b) $d_{ij} < 2\rho$; look for a common neighbour, k, then call FindSurf(i,k).
   (c) $d_{ij} > 2\rho$; there is a large gap between i and j. Insert a dummy tile and check for island formation. Redetermine the current atom and neighbour then call FindSurf(i',j').
4. Find the neighbour to the current atom.
5. Cycle back to step 1 until a termination condition, such as completion of the surface or incorporation of a certain atom into the surface, is met.

Fig. 12. The algorithm for FindSurf.

site of the enzyme trypsin was examined [25]. The cleft has been picked out by a large clump of dummy tiles (Fig. 15); there is also an island within the cleft, partially filling it. When APPA is placed in its correct orientation within the cleft, all atoms involved with the binding of the ligand make close contacts with the accessible surface but the remainder do not (Fig. 16). This shows that a good model for the surface of the binding site has been obtained.

DISCUSSION

The program to compute the surface of a slice through any protein and to determine the clefts and dimples in that surface has been written in Pascal. There are three basic tasks to be performed: the tessellation, the surface traversal and the collection of atoms which make up the clefts.

The data structure used was an array of records, each record being associated with one atom centre and carrying information about the element type, van der Waals radius, the Voronoi polygon, the Brookhaven database serial and sequence number, and other data required by the program. The record can be expanded to carry much more information if required by an individual user. The array is manipulated via an integer index vector to avoid the expense of copying long records.

Occasional difficulties with degenerate vertices have been experienced for tessellations of randomly distributed test points. A degeneracy occurs when four or more polygons meet at a single vertex. This is very unlikely with real molecular data but degeneracies can be removed in two ways: increase the precision of the calculation or give a small random displacement to one or more of the centres involved in the degeneracy. The latter is more general and satisfactory.

The main driving routine for the determination of the Voronoi surface is called FindSite; its control structure is given in Fig. 17. The surface within the slice is constructed in two arcs. Assume initially that the identities of two atoms on the surface, denoted *start* and *finish* are known. Then FindSite calls FindSurf to obtain first the surface between *start* and *finish* and then between *finish* and *start*. The full surface cannot be determined by a single call to FindSurf because of the possibility of islands. If, for instance, the atom *start* formed part of an island then only the surface around that island will be found. The use of two independent points to define the main boundary prevents the routines from cycling aimlessly around islands and guarantees the production of the principal surface. The atoms *start* and *finish* can be found either from the list of atoms which have one of the arbitrary boundary points as a Voronoi neighbour, or from the relative size of the Voronoi tile, as it has been observed that atoms with large tiles are very likely to be on the exterior of the tessellation [15].

The production of clefts from the main boundary is performed by the routine GetCleft. Once the list of surface atoms has been established, it can be split into two sets, S1, the set of atoms which border a dummy tile and S2, the set of those which do not. The surface list should be thought of as circular so that it can be traversed starting at any arbitrary point. We choose an atom which is an element of S2; the traversal is continued until a member of S1 is encountered; this marks the start of the first cleft, whose end is signalled by the next member of S2. The process is continued right around the surface to find all the clefts. This method allows for the possibility that the first atom in the list might be part of a cleft. The cleft comprising the largest number of atoms is retained and made into a closed polygon. All atoms which form part of an island are checked

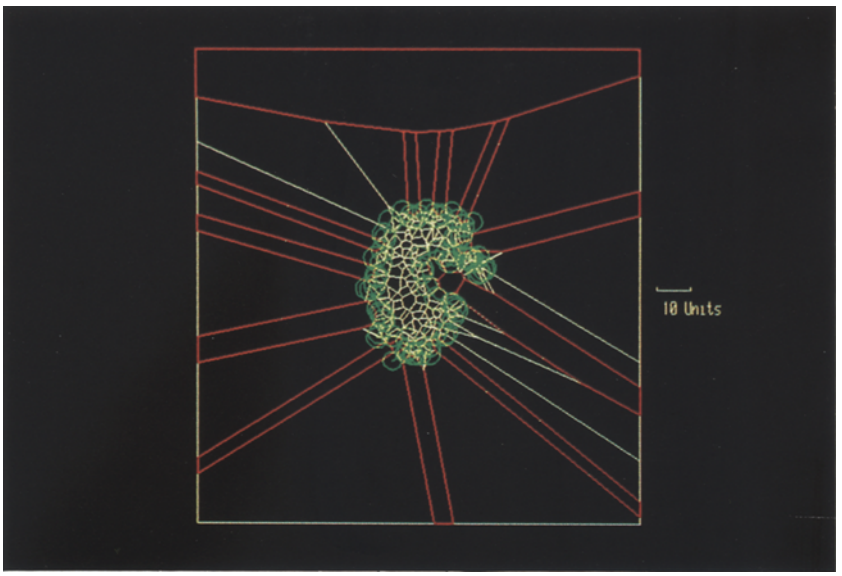

Fig. 13. The Voronoi tessellation through a 6-Å slice of dihydrofolate reductase. The scale bar is 10 Å long. All the dummy tiles (red) have been added. The atoms at the receptor surface are shown in green and the 'true' tiles in white. Notice that several dimples and one cleft have been found and marked by dummy tiles.

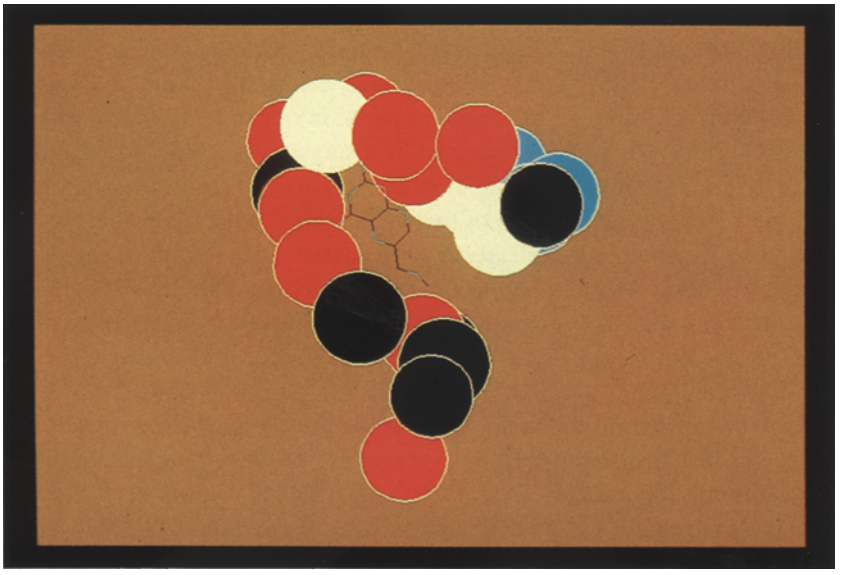

Fig. 14. A close-up of the region around the cleft in the surface of dihydrofolate reductase. Only the atoms which are neighbours of the cluster of dummy tiles are shown, together with the accessible surfaces they describe. They have been colour-coded according to atom type; water molecules are shown in white. These atoms form the binding site for the pteridine ring of methotrexate (black and blue bonds).

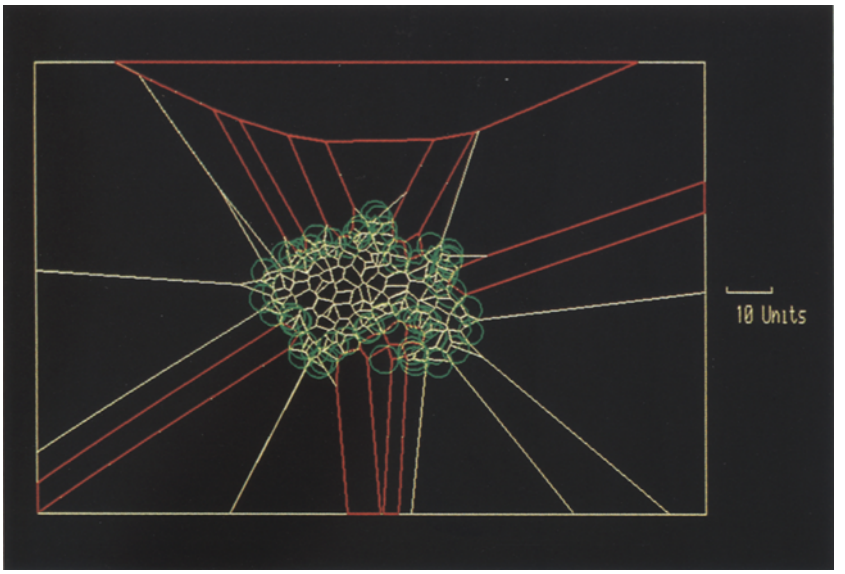

Fig. 15. The Voronoi tessellation through a 6-Å slice of trypsin. The scale bar is 10 Å long. All the dummy tiles (red) have been added. The atoms at the receptor surface are shown in green and the 'true' tiles in white. The dimples and one cleft are marked by dummy tiles. There is also a one-atom island just below the cleft.

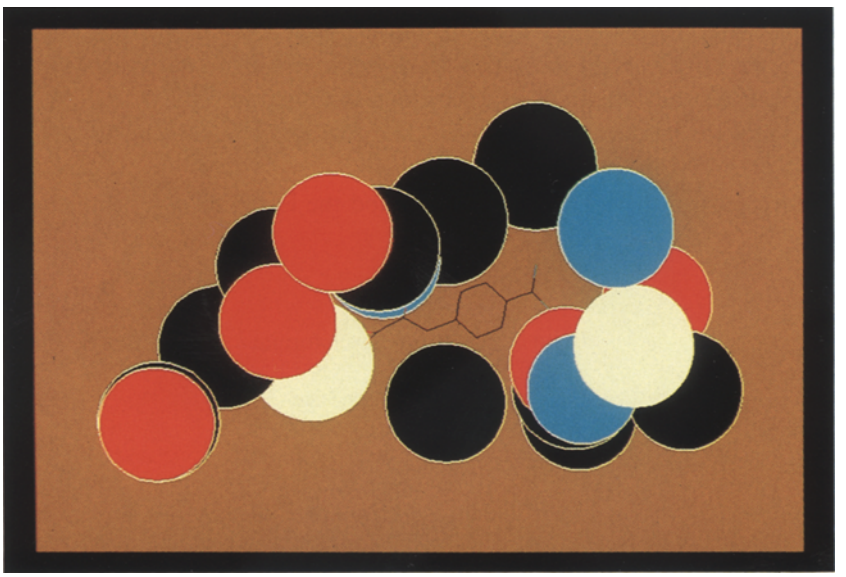

Fig. 16. A close-up of the region around the cleft in the surface of trypsin. Only the atoms which are neighbours of the cluster of dummy tiles are shown, together with the accessible surfaces they describe. The colour-coding is the same as for Fig. 14. These atoms form the binding site for APPA (black, red and blue bonds).

for inclusion within this polygon so that a true picture of the surface in the region of the cleft can be obtained.

The discussion so far has focussed on the application of these methods in two dimensions only. There are, in fact, many cases when the examination of only a slice through a receptor will be perfectly adequate. In drug design, it is found that many known ligands are planar or contain large planar sections [5]. It is therefore possible to perform more extensive calculations on the ligand/receptor pair if only the essential parts of the receptor surface, approximated by the binding cleft in the slice, are considered. However, if the plane of the binding site is not known, then the positioning of the slice will be arbitrary leading to an unsatisfactory situation. It would be better to look at the entire 3-dimensional surface and then to use this to define the orientation and thickness of the slice.

The most suitable method for performing this task is David's volume algorithm [15]. The Voronoi tessellation of all the atoms of the receptor is performed and the volumes of all the domains are examined statistically. These volumes fall into three fuzzy sets, *L(arge)*, *M(edium)* and *S(mall)*. The atoms which are elements of the *L* set are exterior atoms with large or unbounded domains; these atoms will comprise the bulk of the surface. Conversely, interior atoms will have small domains as they are submerged within the bulk of the receptor and hence have many close neighbours. The members of the *M* set will mark clefts and crypts within the receptor surface. They will have an intermediate volume because, as the channel must accommodate some external atoms, they are partially within the bulk and partially external. A channel will therefore be marked by a clump of contiguous domains from the *M* set. This algorithm provides an excellent method for the preliminary location of clefts within the receptor. Its main disadvantage is that volume criteria alone will not generate the entire accessible surface as defined by Eq. (2). Minor dimples within the cleft might fall on the wrong side of the cut-offs specified for the *M* set so that not all the atoms which could interact with a ligand in the cleft will be identified. All centres in the tessellation are given equal weight so that an atom with a small van der Waals shell might have a domain which occludes the domain of a larger atom, forcing it into the *S* set, whereas the reverse might be the case for the true accessible surface. These factors are taken into account in the FindSurf algorithm.

It can therefore be seen that David's algorithm and the FindSurf algorithm complement each

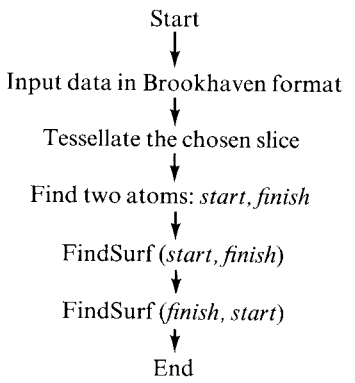

Fig. 17. The algorithm for FindSite.

other. The location of the largest cleft in the receptor is found by the former method and this information can then be used to define the orientation and thickness of the slice. If the cleft is particularly thick, then several overlapping slices may be employed. The combination of the two methods affords a completely automatic method for identifying first the cleft and then its constituent atoms. It is quite conceivable that there is more than one cleft; there could be a binding site for a coenzyme or some allosteric site. Each feature can be treated separately so that the receptor can be broken down into small sections of particular interest. This is of great value in automatic structure generation where most algorithms run in NP-time. It also allows the examination of secondary sites which a subjective human interpreter might have dismissed as unimportant.

CONCLUSION

The automatic determination of atoms which comprise a cleft in a receptor is of great importance in computer-aided drug design. The methods presented in this paper will find the complete accessible surface in a slice through a receptor and also all the clefts and dimples in this surface. Clefts and binding sites can now be determined quickly and without observer bias.

ACKNOWLEDGEMENTS

The author gratefully acknowledges the support of the M.R.C. in the form of a research studentship.

REFERENCES

1 Dean, P.M., Br. J. Pharmacol., 74 (1981) 47–60.
2 Sharpe, K., Fine, R. and Honig, B., Science, 236 (1987) 1460–1463.
3 Kuntz, I.D., Blaney, J.M., Oatley, S.J., Langridge, R. and Ferrin, T.E., J. Mol. Biol., 161 (1982) 269–288.
4 DesJarlais, R.L., Sheridan, R.P., Seibel, G.L., Dixon, J.S., Kuntz, I.D. and Venkataraghavan, R., J. Med. Chem., 31 (1988) 722–729.
5 Lewis, R.A. and Dean, P.M., Proc. R. Soc. London, Ser. B, 236 (1989) 125–140.
6 Lewis, R.A. and Dean, P.M., Proc. R. Soc. London, Ser. B, 236 (1989) 141–162.
7 Lee, B. and Richards, F.M., J. Mol. Biol., 55 (1971) 93–106.
8 Richards, F.M., Annu. Rev. Biophys. Bioeng., 6 (1976) 151–176.
9 Greer, J. and Bush, B.L., Proc. Nat. Acad. Sci. U.S.A. 75 (1978) 303–307.
10 Connolly, M.L., Science, 221 (1983) 709–713.
11 Connolly, M.L., J. Appl. Crystallogr., 16 (1983) 548–558.
12 Connolly, M.L., J. Appl. Crystallogr., 18 (1985) 499–505.
13 David, C.W., Comput. & Chem., 8 (1984) 225–226.
14 Bellagamba, V., Ercoli, R., Gamba, A. and Simonetta, M., J. Chem. Soc., Perkin Trans. 2, (1986) 1127–1132.
15 David, C.W., Comput. & Chem., 12 (1988) 207–208.
16 Voronoi, G.F., J. Reine Ang. Math., 134 (1908) 198–287.
17 Rogers, C.A., Packing and Covering, Cambridge Mathematical Tracts No. 54, Cambridge University Press, 1964.
18 Green, P.J. and Sibson, R., Comput. J., 21 (1978) 168–173.
19 Brostow, W., Dussault, J.-P. and Fox, B.L., J. Comput. Phys., 29 (1978) 81–92.
20 Bowyer, A., Comput. J., 24 (1981) 162–166.
21 Watson, D.F., Comput. J., 24 (1981) 167–172.
22 Tanemura, M., Ogawa, T. and Ogita, T., J. Comput. Phys., 51 (1983) 191–207.
23 Medvedev, N.N., J. Comput. Phys., 67 (1986) 223–229.

24 Bolin, J.T., Filman, D.J., Matthews, D.A., Hamlin, R.C. and Kraut, J., J. Biol. Chem., 257 (1982) 13650–13662.
25 Marquart, M., Walter, J., Deisenhofer, J., Bode, W. and Huber, R., Acta Crystallogr., Sect. B, 39 (1983) 480–490.

APPENDIX

The Green and Sibson method operates by adding a new centre P_N and creating a new Voronoi polygon, V_N, within the context of an existing tessellation. This is achieved by 'winning territory' from the prospective neighbours of P_N (Fig. 18). The nearest neighbour to P_N, K, is found and by determining which vertices of V_K are occluded, the next neighbour, L, is found. M is found by a similar procedure and V_L and V_N are modified. The edges and vertices of V_N are built up in clockwise order until the polygon is closed. There is still the possibility that V_N might be unbounded so that the device of creating dummy boundary points is utilised again. The efficiency of this algorithm in adding N points to an existing tessellation is $O(N^{3/2})$. It can be implemented to use memory space sparingly and is elegant in approach. Its principal disadvantage as a general procedure is that it is restricted to 2 dimensions only.

Routine I: Find the nearest neighbour to the new centre, P_N

The obvious solution is to calculate $d(P_i \rightarrow P_N)$, $i = 1, 2, \ldots N-1$ and find the minimum distance. In the absence of other a priori knowledge, this would be satisfactory as it is O(N) but there is a much more elegant method based on a random-directed walk which uses the information already present. Start at a centre J and move to a centre, K, which is nearer to N. K can be found quickly by only looking at the Voronoi neighbours of J. Repeat the process until no nearer centre can be found. The process is $O(N^{0.5})$ for a central starting position. A closer start will improve on this figure considerably.

Routine II: Traverse the vertices of the nearest neighbour, K

Step (i). Calculate the perpendicular bisector between N and K, H_{NK} and find the first vertex of

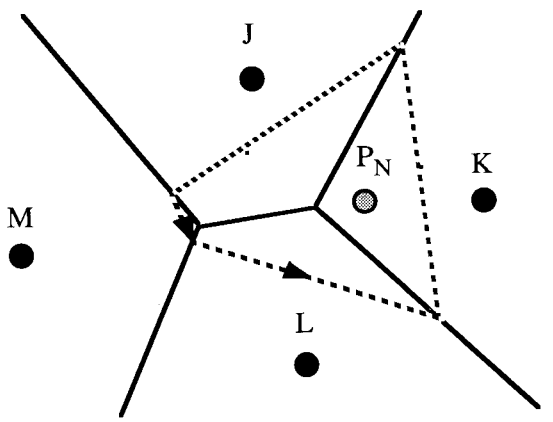

Fig. 18. The tile (dashed lines) around the new point P_N is built up by removing territory from existing tiles (solid lines).

the Voronoi polygon around K, V_K which is excluded. If necessary, move back until an included vertex is found. This vertex is associated with a new neighbour, K′.

Routine III: Create a new partition

Step (i). Find the vertex in $V_{K'}$, associated with K.

Step (ii). In a manner analogous to step II.(i), find the next neigbour K″.

Step (iii). Add edge $h_{NK'}$ to V_N and alter the edge list of $V_{K'}$ to include $h_{NK'}$ and to exclude any invalid edges. Recalculate the vertices of $V_{K'}$.

Step (iv). Set K″ to K′, and K′ to K and go to Step III.(i) until the routine returns to its starting centre.

Step (v). Calculate the vertices of V_N.