Journal of Computer-Aided Molecular Design, 7 (1993) 127–153
ESCOM

J-CAMD 194

SPROUT: A program for structure generation

Valerie Gillet[a], A. Peter Johnson[a,*], Pauline Mata[b], Sandor Sike[a] and Philip Williams[a]

[a]*School of Chemistry, University of Leeds, Leeds LS2 9JT, U.K.*
[b]*Departamento de Quimica, Fac Ciencias e Tecnologia, UNL, Monte de Caparica, Portugal*

Received 3 September 1992
Accepted 28 November 1992

Key words: Artificial intelligence; De novo design; Molecule design; Enzyme inhibitors

SUMMARY

SPROUT is a new computer program for constrained structure generation that is designed to generate molecules for a range of applications in molecular recognition. It uses artificial intelligence techniques to moderate the combinatorial explosion that is inherent in structure generation. The program is presented here for the design of enzyme inhibitors. Structure generation is divided into two phases: (i) primary structure generation to produce molecular graphs to fit the steric constraints; and (ii) secondary structure generation which is the process of introducing appropriate functionality to the graphs to produce molecules that satisfy the secondary constraints, e.g., electrostatics and hydrophobicity. Primary structure generation has been tested on two enzyme receptor sites; the *p*-amidino-phenyl-pyruvate binding site of trypsin and the acetyl pepstatin binding site of HIV-1 protease. The program successfully generates structures that resemble known substrates and, more importantly, the predictive power of the program has been demonstrated by its ability to suggest novel structures.

INTRODUCTION

Progress in molecule design has mainly been made in the fields of drug and pesticide research where there is intense interest in developing techniques in computer-aided molecular design. Since the development of geometric searching algorithms, molecular design programs are now available that are based on searching databases of 3D structures [1]. Examples of these programs include ALADDIN, CAVEAT and the DOCK program. The ALADDIN program [2] uses the concept of a pharmacophore (the 3D arrangement of functional groups required for a molecule to exert a particular biological effect) to search a database for structures that match the criteria. The CAVEAT program [3] uses vectors to identify molecular fragments that can be used as templates to hold conformationally flexible molecules in a specified conformation. The DOCK program [4] is based on the idea of molecular shape. A binding site cavity is taken and used to search a database for molecules which have complementary shape.

*To whom correspondence should be addressed.

Programs that are based on searching databases of 3D structures are limited in two main areas. Firstly, often a single conformation is stored for each structure and this is usually the one believed to be the lowest energy conformation. However, it is known that drugs often bind in conformations other than the lowest energy conformation and hence useful structures can be missed. The second limitation arises from the content of the databases. When the ALADDIN program was used to search different databases very little overlap was found between the results, implying that only a fraction of the structure space had been explored [5].

The design of structurally novel compounds remains a very difficult task. Lewis and Dean [6] first approached this problem using the concept of spacer skeletons. These are defined as topological artefacts that can model more than one structure at a time. Initially the spacer skeletons were restricted to ones that model planar ring systems. The first extension of this work into 3D uses a diamond lattice as a spacer skeleton [7]. This enables the design of acyclic structures, however, it is not a general solution since each atom in the lattice is sp^3 hybridized and all the torsion angles are staggered. Lewis [8] has recently described a new algorithm for creating diverse atom chains by solving a series of trigonometric equations within geometric constraints for a given set of atom types. This method is intended to form a component of a system for computer-aided drug design and allows fragments that have been placed at interaction sites to be bridged.

A number of other programs have been described for the design of novel molecules that use the knowledge of the 3D structure of the target enzyme [9–14]. Nishibata and Itai [9] describe a program called LEGEND for generating structures one atom at a time. The method is not exhaustive and involves the use of random numbers at each stage of the process, i.e., to select an anchor point for the first atom, to select a root atom for extending the structure each time a new atom is added, to determine the type of the atom and bond and to determine torsion angles. GROW [10] is a program for the generation of peptides by the connection of small molecular fragments. A large set of amino acid fragments are used as templates. Each amino acid is represented by several conformations. The search space is managed by a tree that is pruned according to a molecular mechanics-based energy function. The user specifies the number of successors to be included in the tree each time a peptide fragment is extended by one amino acid. The main limitation of this program is its restricted scope, i.e., to the design of peptide-like compounds. The LUDI program [11] first generates interaction sites within the receptor site. Molecular fragments are then fitted onto these sites and finally the fragments are connected by bridging fragments. The program BUILDER [12] combines database searching, structure generation algorithms and interactive graphics modelling to produce novel structures. Initially a database search is performed using the DOCK program to find structures that fit the site sterically. The retrieved structures are then superimposed within the site with the vertices from different molecules linked by virtual bonds to produce a molecular lattice. The user then specifies regions of interest, the appropriate parts of the lattice are displayed graphically and an attempt is made to join fragments by tracing paths through the lattice.

GENERAL DESCRIPTION

The aim of the SPROUT project is to build a general purpose program for the design of molecules appropriate to a wide range of applications, e.g., inhibitor design, the design of catalysts (particularly synthetic enzymes) and the design of agents for asymmetric synthesis. All of

these systems rely on molecular recognition where molecules interact because they are complementary to each other. This complementarity arises from the steric and electrostatic properties of the molecules, for example, an inhibitor is able to bind to an enzyme because it has complementary shape and electrostatic properties to a binding site on the receptor. SPROUT uses information about one molecule to constrain the design of other molecules with which it can interact. The constraints should be sufficient to determine the nature of the molecules. However, full structural information is not always available and sometimes must be inferred. For example, whereas there are a growing number of enzymes whose 3D structures are available to assist in the design of enzyme inhibitors, in other cases the structure of the enzyme receptor site can be inferred by overlaying sets of active and inactive compounds [15].

PROGRAM OVERVIEW

The major interactions involved in the association of molecules are steric, electrostatic (including hydrogen bonding), dispersion or van der Waals, and hydrophobic [16]. These interactions give rise to primary and secondary constraints for molecule design. The primary constraints are steric in nature and are dependent on the particular application and the information that is available, for example, in inhibitor design the 3D coordinates of the receptor may or may not be available. The secondary constraints arise from the electrostatic and hydrophobic properties of the known molecule.

Lewis and Dean used this divison of the constraints to divide structure generation into two phases: *primary structure generation* and *secondary structure generation*. SPROUT also consists of two phases. Primary structure generation is defined, in this context, as the process of generating a 3D molecular graph consistent with the primary constraints on the system, i.e., the shape of the receptor site. Secondary structure generation is the process of converting the molecular graph into a molecular structure, i.e., the vertices and edges of the graph are replaced by atoms and bonds appropriately to give the molecule the desired functionality. The secondary structure generation phase makes use of the secondary constraints, e.g., electrostatic and hydrophobic properties.

The primary constraints include the 3D shape of the receptor, which defines the volume the molecule must lie within. The volume is enclosed by a *boundary* which then restricts the shape of potential ligands. The volume is fixed, for example, in the case of an enzyme the receptor site is assumed to be rigid. Within this volume there are *interaction sites*. These are regions which if occupied by an atom of the ligand can lead to favourable interactions between the ligand and the receptor for example, electrostatic or hydrophobic interactions. If the interaction sites are sufficiently localised they form part of the primary constraints and are used to direct primary structure generation. This is the case for hydrogen bond interactions where it is possible to identify regions that are small enough to be occupied by a single atom. These localised interaction sites are called *target sites* and satisfying these interaction sites forms a requirement for primary structure generation. Other interactions, such as hydrophobic and van der Waals interactions, are less directional and lead to more diffuse interaction regions within the cavity. These interactions are the secondary constraints on structure generation.

During primary structure generation atoms are placed at each of the target sites and linked to produce molecules, without violating the volume specified by the boundary. This type of structure

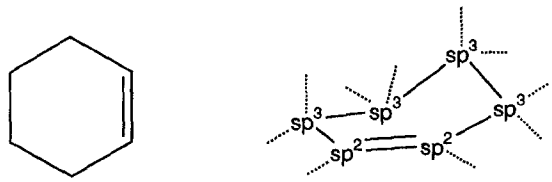

Fig. 1. The template representing the chair conformation of cyclohexane. The vertices of the template are labelled by the hybridisation state and the edges are appropriately labelled as single or double bonds.

generation is well known as a combinatiorial problem [17] where attempts at finding solutions rapidly produce a large number of intermediate structures. Therefore, methods of controlling this combinatorial explosion are required.

One way of reducing the number of intermediate structures is to use molecular fragments as building blocks. This allows larger distances to be spanned in a single step than if molecules are built one atom at a time. However, an enormous number of fragments is required to enable a wide range of molecules to be produced. Primary structure generation is mainly concerned with shape. This makes it possible to group molecular fragments by shape and connectivity so that initially structures are generated using a representative from each group. The representative fragment can be replaced by other fragments from the same group once structures have been found that match the steric constraints. *Templates* are used as representative fragments. Templates are 3D subgraphs where the vertices represent atoms and are labelled only by their hybridisation state (and not element type). The edges of the subgraphs represent bonds. The vertices are labelled as sp^3, sp^2 or sp and have tetrahedral, trigonal or planar geometry, respectively. The geometry defines the positions where new templates can be joined. The bonds are labelled as single, double, triple or aromatic and the distances between the vertices are the corresponding carbon-carbon bond lengths. The template representing cyclohexane is shown in Fig. 1. A number of molecular fragments can be produced from each template by replacing the vertices by any element that can adopt the appropriate hybridisation state (Fig. 2). Bond angles and bond lengths are also adjusted appropriately. This adjustment results in only minor differences in shape between a molecular fragment and its representative template.

The primary structure generation phase joins templates together to produce *skeletons*. (This process is also called *skeleton generation*). Each skeleton represents a number of molecules because each component template represents a number of molecular fragments. Each molecule which can be produced from a skeleton adopts approximately the same shape as the skeleton from which it is derived. Therefore, these molecules satisfy the primary constraints, i.e., they have the required shape. A skeleton which does not satisfy all the requirements is called a *partial skeleton*.

Skeleton generation begins by selecting a template and positioning it at one of the target sites thus satisfying one of the steric requirements. One vertex of the template is anchored at the centre of the target site. The template can be rotated about the anchoring vertex to occupy any position. A representative set of orientations is chosen and each orientation gives rise to a partial skeleton. New templates are added to build skeletons of increasing size. A skeleton is grown in the direction of the remaining target sites. A solution has been found when all the steric requirements are satisfied and no boundary violations have occurred. This process is illustrated in Fig. 3.

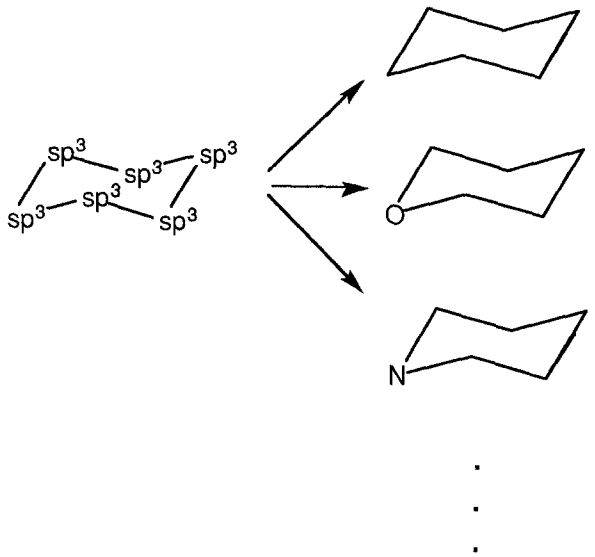

Fig. 2. Several molecular fragments can be produced from each template by replacing the vertices by elements that can adopt the same hybridisation state.

A description of the templates used in the program, the mechanism for joining templates and the use of template joining rules will be given in the subsequent paper in this series [18]. Templates are divided into cyclic and acyclic templates. Acyclic templates can be joined to cyclic or acyclic templates by forming a new bond between one vertex from each template. This type of join is called the *new bond* join. Rotation is possible about a new bond and so a number of conformations is produced. The current program produces the staggered conformations when two sp^3 atoms are joined by this method. Two cyclic templates can be joined by fusion, bridging, spiro joining or by forming a new bond between them. These joins are illustrated in Fig. 4.

The secondary structure generation phase fulfils the requirements made by the secondary constraints, by making atom substitutions to produce the required electrostatic and hydrophobic properties. This phase will be described in future papers.

Program summary

The main components of the program are identified as:

(1) Representation of the primary constraints, i.e., the steric constraints.

(2) Construction of a library of 3D molecular fragments or templates.

(3) Methods for joining templates into larger approximate structures which are consistent with the primary constraints.

(4) A strategy for controlling the combinatorial explosion inherent in structure generation.

(5) Representation of the secondary constraints, e.g., electrostatic and hydrophobic effects.

(6) Atom substitutions to convert the approximate structures into molecules that are consistent with the secondary constraints.

(7) Organising and evaluating the resulting molecules, e.g., by using conformational analysis and clustering techniques.

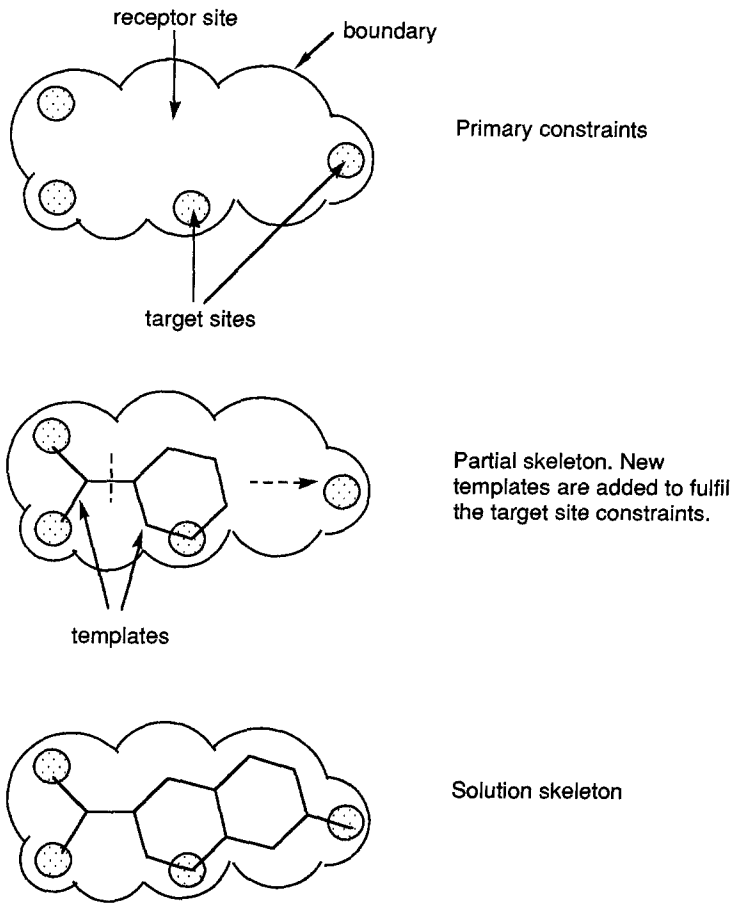

Fig. 3. Skeleton generation begins by placing a template at one of the target sites. New templates are added in the direction of the remaining target sites. A solution is found when all the target sites are satisfied and no boundary violations have occurred.

This paper describes the primary structure generation phase of the system together with some results. A summary of the approach is given in Fig. 5.

The program has been applied to the design of enzyme inhibitors where the constraints can be derived from an enzyme binding region. The primary constraints are a 3D boundary together with target sites within the cavity. These constraints are extracted from the X-ray data of enzymes that have been crystallised, from NMR experiments, or they are inferred by overlaying sets of active and inactive compouds to determine a pharmacophore.

PRIMARY CONSTRAINTS

Target sites

The target sites used as primary constraints are derived from localised regions of the cavity where an atom of the inhibitor must be placed to interact with the receptor. This interaction is

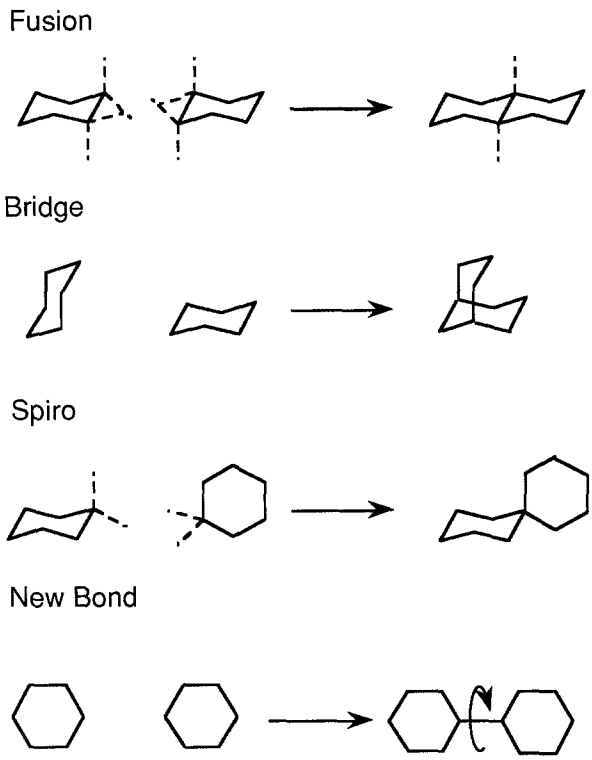

Fig. 4. Any two templates can be joined by forming a new bond between them. Two ring templates can also be spiro joined, fused or bridge joined.

usually a hydrogen bond. In skeleton generation, this cavity region is represented by part of the volume, called a target site, that must be occupied by a vertex. Accurate hydrogen bonding geometries can be found if the receptor is assumed to be rigid and the receptor hydrogen positions are known [11, 19–20]. In the present work, the target sites are represented by larger volumes to allow for the differences between skeletons and molecules mentioned previously. They are represented by spheres of fixed radius; a value of 0.5 Å is used for this radius in the present work. A target site becomes *satisfied* when exactly one vertex of the skeleton falls within its associated sphere.

Approximating the target sites to spheres allows some freedom in the position of a skeleton: the position of a partial skeleton can be altered without losing the correspondence between vertices and target sites. For example, if only one target site is satisfied the partial skeleton can be moved by any distance less than the radius of the target site sphere in any direction without losing the correspondence between the anchoring vertex and the target site. Moving a partial skeleton may allow a previously unsatisfied site to become satisfied. Whereas it is impractical to explore all the possible displacements available to a skeleton each time a new template is added it is worthwhile in some situations. These situations are identified by examining the relationship between a skeleton and the target sites each time a new template is added. A number of relationships can exist between a skeleton and a target site:

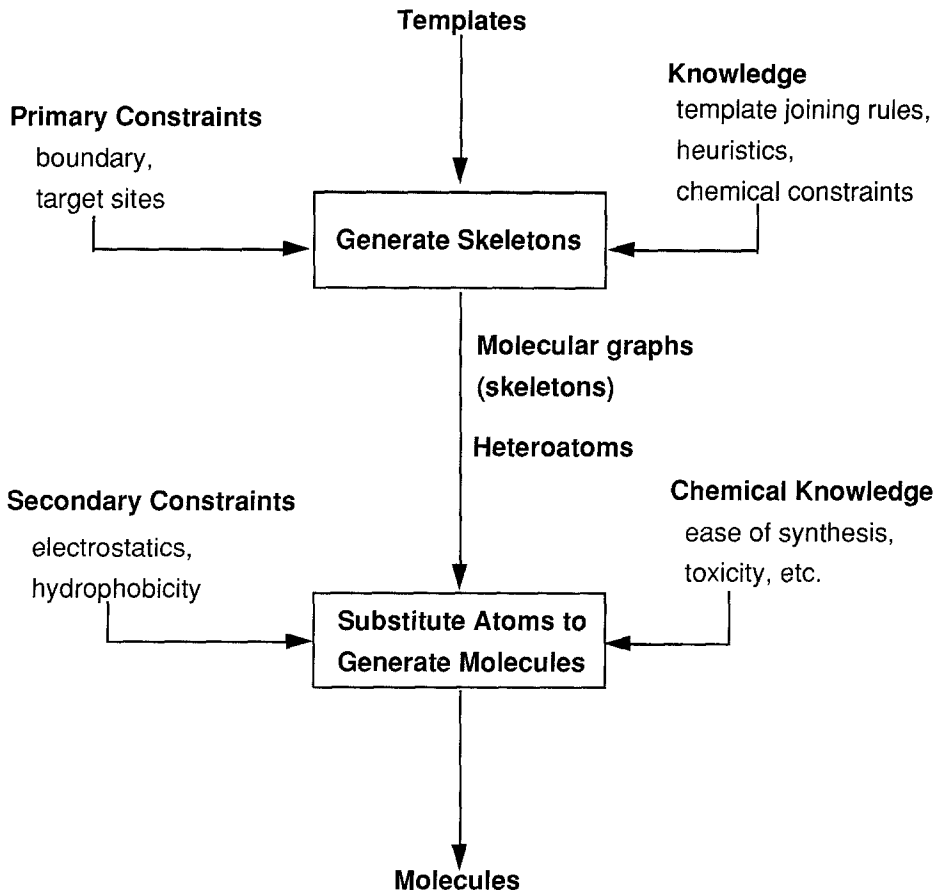

Fig. 5. An outline of the components required for structure generation.

(a) The skeleton satisfies the target site, i.e., one of its vertices lies inside the sphere representing the target site.

(b) The skeleton does not satisfy the target site but is close enough to it to prevent any successor from satisfying the target site. If a vertex is within half a bond length from the centre of the site it will prevent any successor from satisfying the site. If it is more than half a bond length away adding a new template to the skeleton can result in a new vertex being closer to the site or satisfying the site.

(c) The distance between the skeleton and the target site is too large for either (a) or (b) to apply. Templates are added until (a) or (b) occurs.

Cases (a) and (c) do not require any special handling. Case (b) can result in the loss of potential solutions. The procedure for avoiding this situation is as follows: an additional sphere is included around each target site of radius equal to half an average bond length. The area inside this sphere and outside the target site sphere is called the *close region*. Whenever a vertex is found in a close region then this skeleton is the best that can be achieved for that arrangement of templates. An

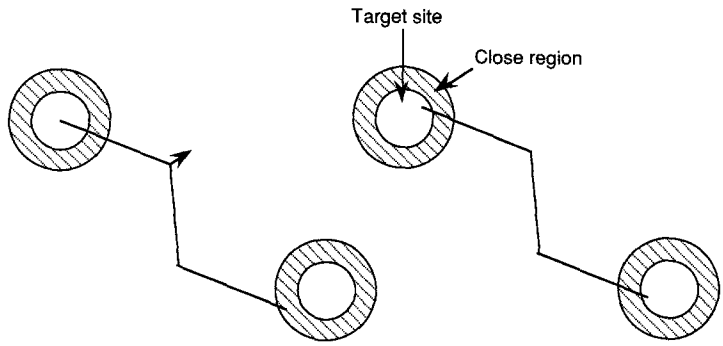

Fig. 6. When a skeleton does not satisfy a target site but does have a vertex within its close region an attempt is made to reposition the skeleton to satisfy the target site.

attempt is now made to adjust the position of the skeleton to satisfy this site. If this fails the skeleton is discarded.

Three types of adjustment are possible: displacement of the skeleton as a rigid structure, rotation of the skeleton as a rigid structure about an axis, and internal changes to the skeleton (changing bond lengths and/or torsion and bond angles). Any combination of these transformations could be applied to the skeleton. However, rotations are not considered because a small rotation can have a large effect on the position of the skeleton and it is difficult to maintain existing correspondences between target sites and vertices (and ensure that the boundary is not violated). Internal changes are not considered at this stage. Therefore, the problem is reduced to that of finding a displacement vector that will move the vertex in the close region into the target site sphere but still maintain all existing correspondences (Fig. 6).

The boundary

When the 3D coordinates of the atoms of the receptor are known the boundary of the receptor site can be derived from the solvent accessible surface [21]. This surface determines the closest possible position of the centres of the atoms of a neighbouring molecule and is appropriate since the vertices of the skeletons are points and not atoms. However, the boundary defined by the solvent-accessible surface must be modified before skeleton generation for the following reasons: (a) in the secondary structure generation phase the skeletons are converted to molecules which adopt a slightly different shape; (b) the skeletons produced should be independent of the target site selected to start skeleton generation. (Different methods of selecting the first target site are described later); (c) the displacement caused by adjustment around target sites must be considered, i.e., temporary boundary violations must be allowed. The boundary is adjusted to compensate for each of these effects.

Case (a). Skeletons are built from hydrocarbon fragments. Replacing some of the vertices by heteroatoms to produce molecules usually results in reduced bond lengths. Therefore, the molecules produced from a skeleton usually occupy a reduced volume compared to the hydrocarbon skeleton. This is compensated for by increasing the size of the cavity by reducing the radii of the spheres that represent the receptor atoms by 0.2 Å.

Case (b). As discussed above, a region is included around each target site to identify those vertices that are close to the site but do not satisfy it, i.e., they fall within the close region. If a vertex is placed in the close region an attempt is made to reposition the skeleton to move it into the target site. These regions must fall within the volume to avoid the results being dependent on the target site chosen to start skeleton generation. Thus, the close regions are added to the volume.

Case (c). A *margin* is introduced around the entire boundary to compensate for the displacements that can be applied to a skeleton. The width of the margin is equal to the maximum displacement during skeleton adjustment. The maximum skeleton displacement is equal to the radius of the target sites (0.5 Å). Thus, the radii of the spheres representing the receptor atoms are reduced by this amount.

The radii of the atoms used to define the boundary are therefore composed of four components as shown in Eq. 1:

$$r_i = \text{van der Waals radius}_i + \text{water radius} - 0.2 - \text{margin} \quad (1)$$

The surface of these spheres forms the boundary for skeleton generation.

Introducing the margin around the boundary leads to a larger cavity that the solvent-accessible surface (actually the solvent accessible surface less 0.2 Å). This can result in the generation of skeletons that are larger than the volume. During skeleton generation, if a partial skeleton will ultimately exceed the volume, it is not considered further. When a new template is added to a skeleton its vertices are allowed to fall within the margin. If this occurs an attempt is made to reposition the skeleton within the solvent accessible surface without losing any of the correspondences between vertices and target sites. If the skeleton cannot be repositioned it is invalid.

Orientations

The first template is positioned in the volume with one vertex anchored at the centre of a target site. The template can be rotated about this vertex to occupy any position. The actual position chosen for the template determines the skeletons that can be generated since this fixes the orientation of a skeleton in the volume. A representative set of orientations is chosen by finding an even distribution of points on the surface of a sphere that is centred at the centre of the target site. An angle, θ, is chosen and a set of unit vectors whose angles with respect to the z axis are a multiple of θ, are generated. When these vectors are rotated about the z axis each traces a circle on the surface of the sphere. The circumference of each circle is divided into a set of points separated by a fixed distance equal to $360/\theta$. Each point on the sphere leads to one orientation of the template. The number of points, i.e., the resolution, is configurable. The values of θ currently available in the program are 30, 15 and 7.5°, and they lead to 46, 176 and 654 points on the surface of the sphere, respectively. The orientations that do not violate the boundary constraints become partial skeletons.

Selecting the start site

The number of orientations chosen initially can determine the size of the search space. Some of the target sites can be more restrictive than others, e.g., they fall in a small concave region of the cavity, thus restricting the positions available to the template. Three different methods can be used to calculate the target site to use to start skeleton generation or it can be specified by the user:

(a) The first method considers the distances between the target sites. Firstly, the pair of sites that are separated by the smallest distance is chosen. Secondly, the site from this pair that is closest to the remaining sites is selected as the start site.

(b) The second method uses the distance between the target sites and estimates the number of templates required to reach a solution. The target site requiring the smallest number of templates is selected as the start site.

(c) The third method calculates the number of partial skeletons consisting of a single template for each target site and selects the target site with the smallest number. This method compares the breadth of the graphs at the first level and assumes that a narrower graph initially will lead to a smaller search graph.

The third method makes use of the boundary which can have a strong pruning effect. This is the preferred method when generating structures to fit a macromolecule of known 3D structure. If the boundary is less well defined then a method based on the distances between the target sites is preferred.

GRAPH SEARCHING IN SKELETON GENERATION

Skeleton generation is represented by a graph where each node of the graph represents a possible state of the system: the root node of the graph represents the initial state, the leaf or goal nodes represent solutions, and the other nodes (intermediate nodes) represent partial solutions. The initial state in skeleton generation is represented by the primary constraints, i.e., the boundary and target sites. A solution is a skeleton that satisfies all the target sites without violating the boundary. The intermediate nodes represent partial skeletons that have one vertex selected. This vertex is used to add new templates to the skeleton. A node is expanded into several nodes at the next level of the graph by selecting templates from the template library and joining them to the partial skeleton at the selected vertex. A number of new skeletons is produced, each one differing from its parent by the addition of a single template. After this node has been expanded it is replaced in the graph but with a different vertex selected. This is repeated until all the available vertices have been used. The search graph is illustrated in Fig. 7.

The search graph can be very large and so a control strategy is required so that paths through the graph from the root node to solution nodes can be found as efficiently as possible. The aim of the search algorithm is to find several varied solutions that satisfy the constraints, i.e., the search does not terminate when it has found the first solution. It is impractical to search the graph exhaustively and therefore it is important that the branches of the graph that lead to solutions are found as early as possible.

The search is directed by associating a cost or score with each of the nodes in the graph, i.e., it is an A algorithm [17]. The A algorithm is a best-first search method. It uses knowledge about the problem domain in the form of heuristics to decide which node in the graph to expand next. The cost of a node is given by a function, f; the node with lowest value of f is expanded at each stage of the search process. Thus, the search attempts to find the lowest cost paths between the start node and goal nodes. The function f is given by Eq. 2:

$$f(n) = h(n) + w_1\, g(n) \qquad (2)$$

where h(n) is the estimated cost of reaching a goal node from node n and g(n) is the cost of

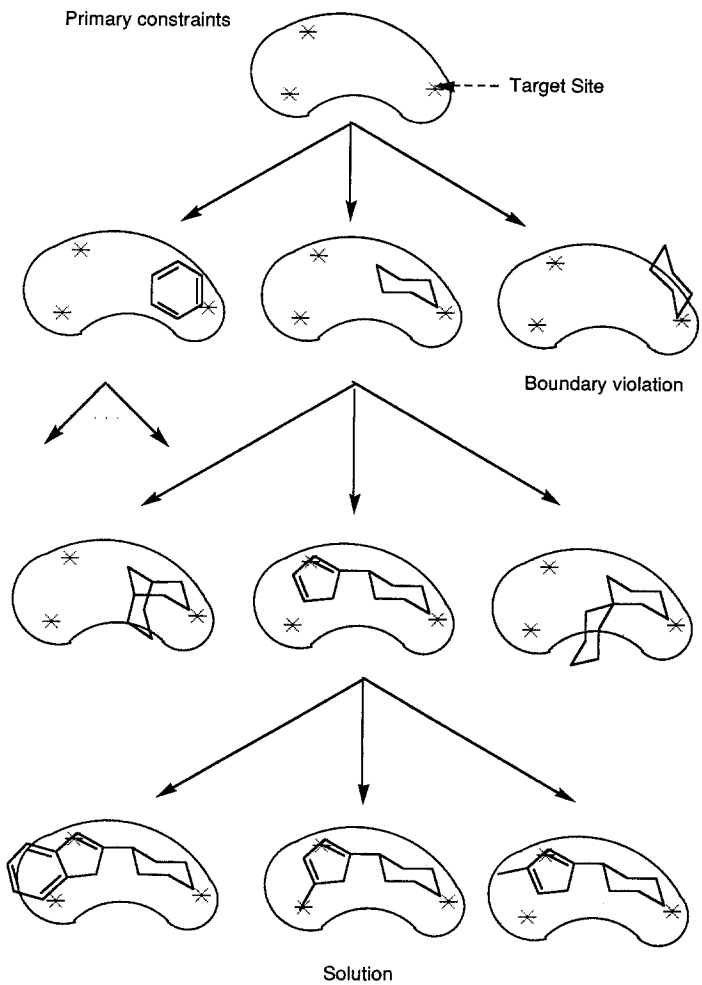

Fig. 7. The search graph for skeleton generation.

reaching node n from the root node. This function can also be used to simplify the graph by removing nodes with a cost above a given threshold value. However, rather than discarding the corresponding skeletons they are stored in a list and used later if required. Weighting one component relative to the other changes the search from a depth-first search, when h(n) dominates, to a breadth-first search, when g(n) dominates.

The cost of reaching a goal node, h(n), is estimated by Eq. 3:

$$h(n) = d_{min}(n) + w_2 (M - m(n)) \tag{3}$$

where $d_{min}(n)$ is the minimum distance between a vertex of the skeleton and the closest unsatisfied target site; M is the total number of target sites and m(n) is the number of satisfied target sites, thus (M − m(n)) is the number of unsatisfied sites. A skeleton having a large value of m(n) and a small value of $d_{min}(n)$ is assumed to be close to a solution. (M − m(n)) must be scaled relative to

the distance $d_{min}(n)$. The term w_2 (M − m(n)) estimates the distance that remains to be spanned to fulfill the constraints. The total number of templates required to satisfy all the target sites is estimated (see the later discussion of depth limits). w_2 is then equal to the number of templates required per target site multiplied by the average distance spanned by a template.

The component g(n) is included in the function f to avoid an extensive depth-first search of the graph. g(n) is the cost of reaching node n from the root node. The cost of reaching a node is measured by the distance spanned by the partial skeleton it represents. This can be estimated by the number of templates used to construct the partial skeleton if an average template size is determined. However, for a given pair of templates the distance that is spanned depends on how they are joined. For example, the partial skeleton produced by joining two benzene rings by a new bond spans 4.0–4.5 Å, whereas, fusing two benzene rings spans 2.0–2.5 Å. Therefore, g(n) is given by Eq. 4:

$$g(n) = n_{fuse}\, d_{fuse} + n_{spiro}\, d_{spiro} + n_{newb}\, d_{newb} + d_{first} \tag{4}$$

where n_{fuse}, n_{spiro} and n_{newb} represent the number of fuse joins, spiro joins and new bond joins, respectively; d_{fuse}, d_{spiro} and d_{newb} represent the average distances spanned by joining a new template to a partial skeleton by a fusion, a spiro join or a new bond; and d_{first} is the distance spanned by the first template. In the examples included in this paper $d_{fuse} = 2$, $d_{spiro} = 3$, $d_{newb} = 3$, and $d_{first} = 2$. Thus the different types of join are weighted differently to compensate for the differences in distances that can be spanned. d_{spiro} is set relatively large to disfavour this type of join.

Energy function

The cost of a node can also be estimated by considering the energy of the state represented by a node. Each node represents a partial skeleton bound to the receptor site and a number of energy terms must be considered. The interation involved in the association of molecules into a complex, e.g., the binding of a ligand to a receptor, can be summarised as follows (Eq. 5, [22]):

$$\Delta G = \Delta G_{(trans + rot)} + \Delta G_{(hyd)} + \Delta G_{(vdw)} + \Delta G_{(rotore)} + \Delta H_{(conf)} + \Delta G_{(i)} \tag{5}$$

Not all of these terms are appropriate for skeleton generation where the vertices are undefined according to element type and also where the aim is to compare one partial skeleton with another rather than to calculate the absolute energy of a skeleton. The skeletons are considered as hydrocarbons in the following calculations. $\Delta G_{(trans + rot)}$ represents the loss of translational and rotational free energy of the molecules when the complex is formed. This term is mainly dependent on the mass of the molecules. As the skeletons grow to fill the receptor site a relatively small range of masses will be produced and therefore the term is ignored. $\Delta G_{(hyd)}$ and $\Delta G_{(vdw)}$ are the hydrophobic binding and van der Waals interactions in the receptor site. The van der Waals energy will not differ greatly between different skeletons. The hydrophobic binding energy varies with the hydrocarbon surface area buried on binding. These terms cannot be modelled accurately for skeletons and are unlikely to be decisive at selecting between skeletons.

The remaining terms are accounted for during skeleton generation. $\Delta G_{(rotore)}$ represents the loss of internal rotations in the ligand due to binding. Each rotatable bond frozen out through binding results in an entropic penalty (of about 5–6 kJ mol^{-1} [22]. $\Delta H_{(conf)}$ is the energy required to reach

the bound conformation. This is composed of two terms: the conformational strain energy of each template and the van der Waals energy on joining templates. The conformational strain energy in a template is the difference between the given conformation and the minimum energy conformation of the template. This term is precalculated using the MM2 force field [23] and assumes that the templates are hydrocarbons. It includes the van der Waals interactions between atoms in the template. The van der Waals energy on joining templates is calculated using a 'soft' 6–12 potential that considers the interactions between every atom of the new template and every atom in the partial skeleton. The soft potential compensates for the fact that the skeletons are approximate representations of structures. The potential is based on that used by Hagler et al. [24]. Both these terms are calculated by assuming that the templates represent hydrocarbons. $\Delta G_{(i)}$ is the internal energy release by a group binding and includes the enthalpy of interaction, loss of solvation energy of the group and favourable entropy terms as solvent molecules are released from binding. This term depends on the nature of the interaction site (+ or −, h-bond donor, h-bond acceptor, hydrophobic). No functional groups are included explicitly in the skeleton and so this term has to be approximated, for example, -20 kJ mol^{-1} is used each time a target site is reached [21]. Thus the binding energy for node n is approximated to:

$$e(n) = \sum_{i=1}^{T} Si + \sum_{i=1}^{t} \sum_{j=1}^{s} Vij + xn_{rb} + ym \tag{6}$$

where S is the strain energy of a template, T is the number of templates; V is the increase in van der Waals energy on joining a new template, t is the number of vertices in the new template, and s is the number of vertices in the skeleton before the new template is joined; x is the entropic penalty per rotatable bond; n_{rb} is the number of rotatable bonds in a skeleton; y is the energy release on reaching a target site; and m(n) is the number of satisfied target sites.

The estimated binding energy of a skeleton increases as the skeleton increases in size; this can be due to additional van der Waals interactions, adding new rotatable bonds and including templates in conformations other than the global minimum. A breadth-first search is avoided by adding a penalty term each time a new template is joined to a skeleton. Thus, the energy-based scoring function has the form:

$$f_e(n) = e(n) + zT \tag{7}$$

where z is the penalty term. If z is set too large a depth-first search results. This is a parameter within the program and a representative value is -10 kJ mol^{-1}.

GRAPH PRUNING

In practice, many nodes of the graph cannot lead to solutions. Whenever a new template has been added to a skeleton, the new skeletons are evaluated to determine whether they should be removed or whether they can be added as new nodes in the search graph. The evaluation consists of some non-heuristic pruning methods. In addition, heuristics can be used to attempt to identify unsuccessful branches of the graph as early as possible.

Non-heuristic pruning methods

There are three non-heuristic pruning methods. Firstly, skeletons are removed from the graph if they violate the boundary and cannot be adjusted to fall entirely within the cavity.

Secondly, duplicate skeletons are removed. Each distinct skeleton is represented by a new node in the graph. It is possible that duplicate skeletons can be produced by different branches of the search graph. For example, two skeletons might be constructed from the same templates by adding the templates in a different order. Whenever a new skeleton is generated it is checked against all the skeletons in the graph. The skeleton is only added as a new node if it has not been generated previously. To be identical, two skeletons must also be oriented in the same way relative to the cavity. Two skeletons that are constructed from the same templates that have been joined in the same way but are oriented differently within the cavity, represent two independent nodes in the search graph.

Thirdly, a skeleton is removed if it cannot satisfy a particular binding constraint. This condition may arise when a vertex is close enough to a target site to prevent any other vertex of the skeleton from being closer but it does not fall within the target site. If the partial skeleton cannot be repositioned successfully it is removed from the search graph.

Heuristic pruning methods

A number of heuristics is also used to reduce the size of the graph. Their effect is to eliminate skeletons that are chemically improbable and skeletons that have unfavourable scores.

A skeleton is removed from the graph if joining in a new template results in unfavourable van der Waals interactions. The skeleton is treated as a hydrocarbon to calculate its van der Waals energy. The efficiency of the calculation is improved by considering only the interactions between vertices of the skeleton and the newly joined template. This is possible since each template alone represents a hydrocarbon fragment at an energy minimum.

Heuristics can be derived which are based on the shape of the boundary. Setting a limit on the number of templates that can be used to construct a skeleton can be a very effective way of pruning nodes from the search graph. This limit is called the *depth limit* since it is related to the depth of the node in the search graph. Any skeleton that has reached this limit but does not satisfy the primary constraints is removed from the graph. This limit can be set interactively or can be calculated automatically. It can be difficult for a user to estimate a reasonable limit particularly when the boundary has a complicated shape.

The algorithmic determination of depth limits is dependent on the idea of an average template size. This should be chosen cautiously since too large a value may result in solutions being missed, whereas too small a value makes the depth limits ineffective. Depth limits can be determined statically, i.e., before skeleton generation has begun, in which case the same limits apply to all skeletons. The limits can also be determined dynamically for each skeleton. This can be more accurate since the actual number of templates used in the skeleton is known rather than estimated. Depth limits can also be used between target sites as well as for the whole skeleton. In this case, a limit is determined by the number of templates required to span a given distance.

Restrictions imposed on the physical nature of the molecules to be generated can be used to prune the search graph. For example, a limit can be placed on the number of vertices in a solution skeleton (or atoms of a molecule), and any skeleton exceeding this limit is removed. Similarly, limits can be used to restrict the different ring sizes, e.g., three-membered rings can be excluded or can be restricted to one per molecule. Another potentially useful constraint is to specify the ratio of ring vertices to chain vertices. This constraint allows some restriction to be placed on the rigidity of the generated molecules.

MONTE CARLO APPROACH

Skeleton generation, as described so far, explores the conformational space of skeletons by including templates in a number of conformational minima and by sampling conformations around rotatable bonds. Templates are joined at idealised bond angles and bond lengths to produce structures that are at local energy minima in the gas or solution phase. For example, the template library contains all the staggered conformers of saturated chains and three staggered conformers are produced when two sp^3 hybridised atoms are joined by a new bond. A molecule which is bound to a receptor can be deformed away from its minimum energy conformation since it is the ligand–receptor complex that is at an energy minimum. Thus, the program may fail to find non-minimum energy solutions that are still capable of powerful binding. Decreasing the sampling interval allows a wider search of conformational space, however, this can result in a significantly larger search graph. The large size of the search graph for skeleton generation, i.e., the large number of partial skeletons on which the calculations must be performed, prevents a rigorous approach such as energy minimisation of the structures within the receptor site.

Metropolis Monte Carlo methods are frequently used to search the conformation space available to a molecule in the gas or solution phase [25,26]. They have also been applied to the automated docking of substrates to proteins [27,28] and in the GROW [10] program described in the Introduction. They begin with an initial conformation that is altered randomly by changing a bond length or torsion angle. If the new conformation is lower in energy the change is always accepted. If the new conformation is higher in energy the probability of accepting the change depends on the Boltzmann distribution between the two states. A Metropolis Monte Carlo simulation rapidly minimises the energy of poor structures and then searches the lower energy areas of conformational space. Global energy minima can be located by applying simulated annealing to the Metropolis method.

In structure generation, the aim is to search the conformational space available to the bound substrate and locate a low energy minimum on this surface. However, in skeleton generation the energy of the bound substrate must be approximated since the method is applied to skeletons with vertices that are undefined by element type. The energy of the bound skeleton, represented by node n, is approximated by the following function:

$$E_b(n) = E_{vdw}(n) + bx\,(n) + ym(n) \tag{8}$$

where E_{vdw} is the van der Waals energy of the skeleton; x is the number of vertices that violate the boundary and b is a penalty for each of these vertices; y is the energy release on reaching a target site; and m(n) is the number of satisfied target sites.

The Monte Carlo method modifies skeletons by rotating around bonds. If a skeleton violates the boundary or misses a target site then a random change in the conformation may improve the skeleton. A modification which results in a boundary violation or moves an atom from a target site will result in a significant rise in the value of the function and thus be unlikely to be accepted. The Monte Carlo method is applied each time a template is added to a skeleton. Because of the frequency of its application the number of iterations is kept relatively small, for example, up to 100 iterations. The number of iterations, the number of bonds rotated in each step and the temperature of the simulation are user definable.

RESULTS

Two enzymes have been used during validation of SPROUT. The objectives of the tests performed were:

(1) To generate molecular skeletons that are closely related to the known substrates. This demonstrates the validity of the program.

(2) To generate novel substrates. This demonstrates the predictive value of the program.

The aim in each case is to design inhibitors which are similar to known substrates and also to suggest novel structures. The effect of the different scoring functions on the ordering of the

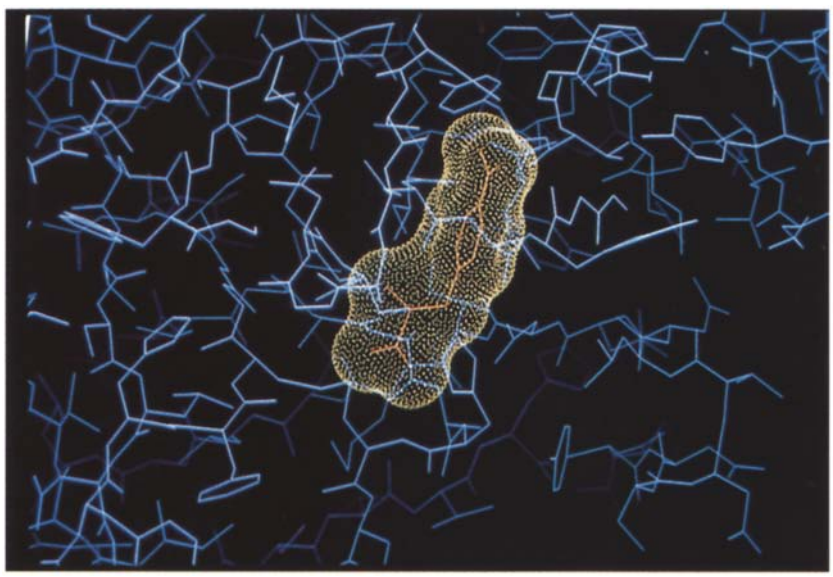

Fig. 8A. APPA bound to trypsin.

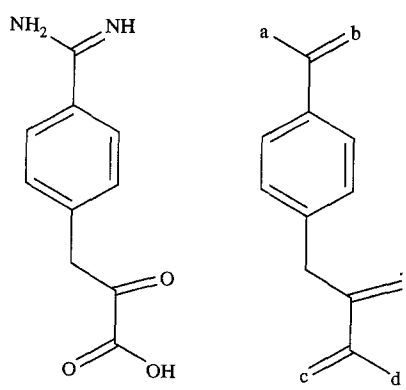

Fig. 8B. APPA and its skeleton. The atoms used as target sites are labelled a to d.

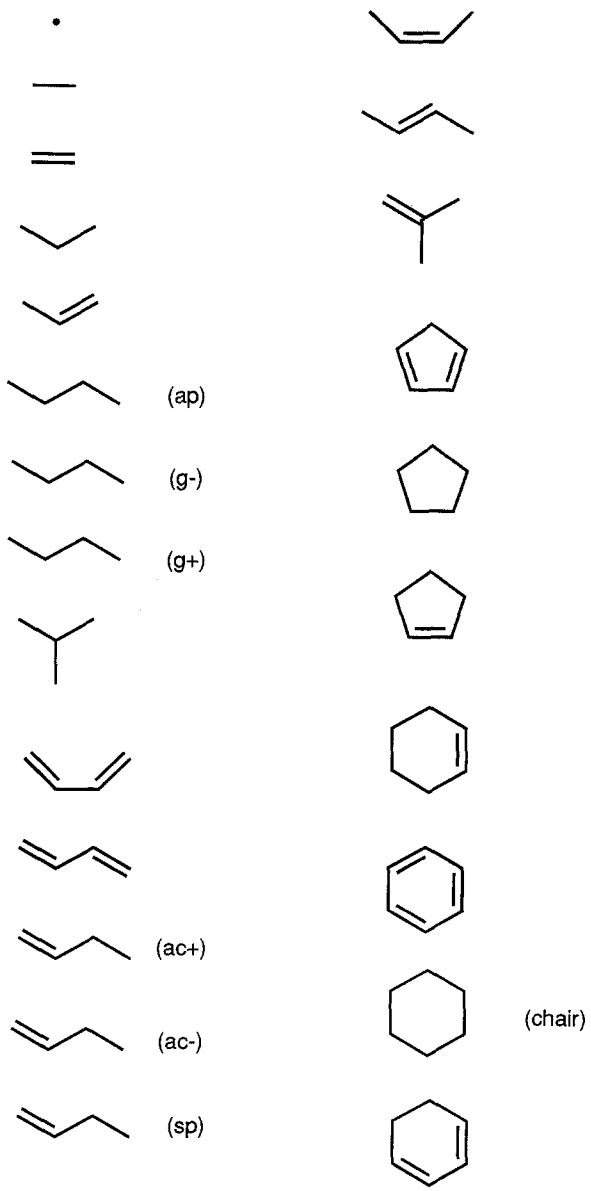

Fig. 9A. A template dictionary used to produce a variety of skeletons for the APPA binding site of trypsin.

solutions and the effect on the output caused by including the Monte Carlo procedure were investigated.

Trypsin

The program was applied to the enzyme trypsin complexed with 2-*p*-amidino-phenyl-pyruvate

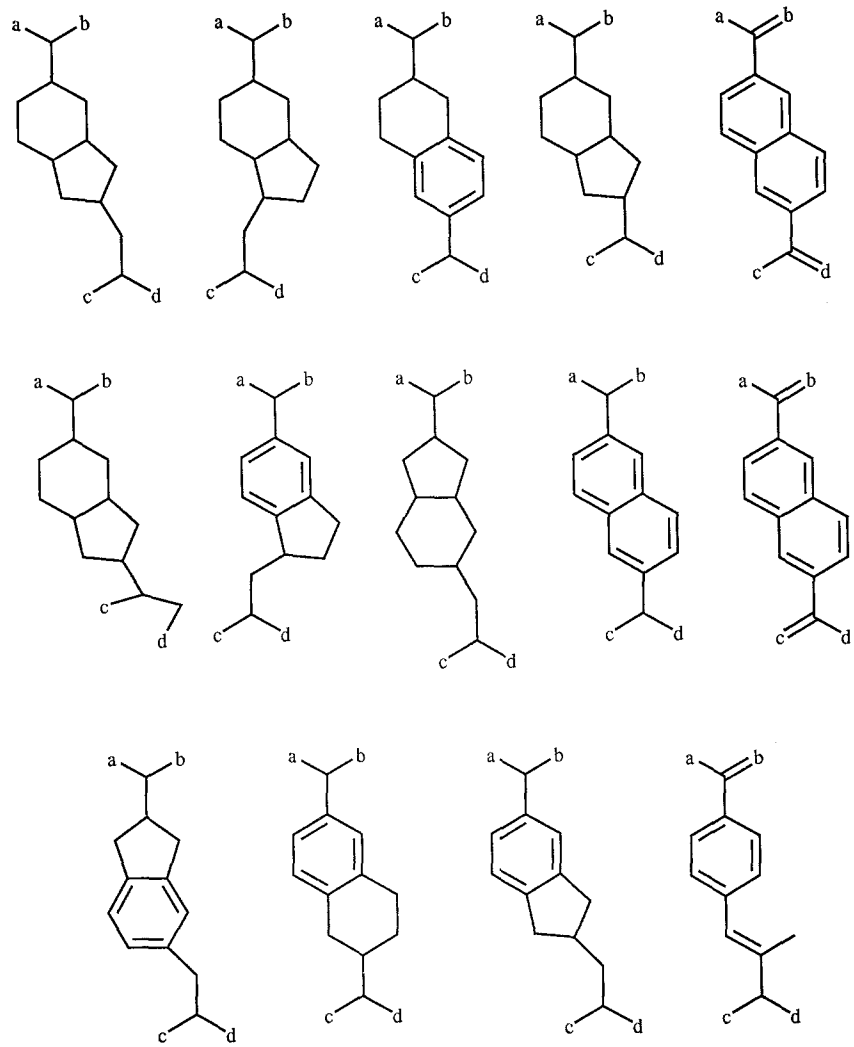

Fig. 9B. Skeletons generated to fit the APPA binding site of trypsin. The vertices that correspond to the target sites are labelled a to d. The bonds are not distinguished as single or double (except for the atomatic rings) — a variety of skeletons were found for each one shown.

(APPA) [29] (Brookhaven Protein Data Bank (PDB) entry 1TPP). The memory requirements of the program were reduced by restricting the atoms of the enzyme to those that surround the receptor site. All atoms within a sphere of 10 Å centred on APPA were included. APPA itself and the solvent molecules were removed. The coordinates were input to SPROUT as a PDB file, obtained using the program Quanta [30]. Target site coordinates were selected based on the atom positions of some of the atoms of APPA. APPA is shown bound to trypsin in Fig. 8A. APPA and its corresponding skeleton are shown in Fig. 8B and the atoms used to derive the target sites are labelled a to d. Several runs were carried out using these primary constraints. All the runs involved an exhaustive search of the graph.

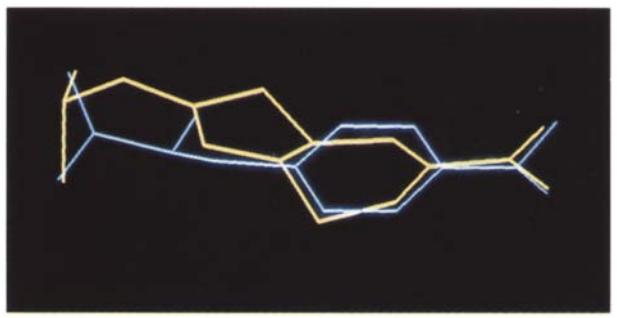

Fig. 9C. One skeleton (yellow) superimposed on APPA (blue).

Generating a variety of solutions

An initial run was carried out to investigate the variety of skeletons that can be produced for these constraints. The templates available for this run are shown in Fig. 9A. Some of the skeletons that were generated are shown in Fig. 9B. It can be seen that a wide variety of skeletons is produced. Some of these are very similar to the skeletons representing APPA and other known inhibitors whereas others reperesent novel structures that would be difficult to find using other computational methods. Figure 9C shows one of the skeletons superimposed on APPA.

Several runs were then performed to compare the different strategies that can be employed during skeleton generation.

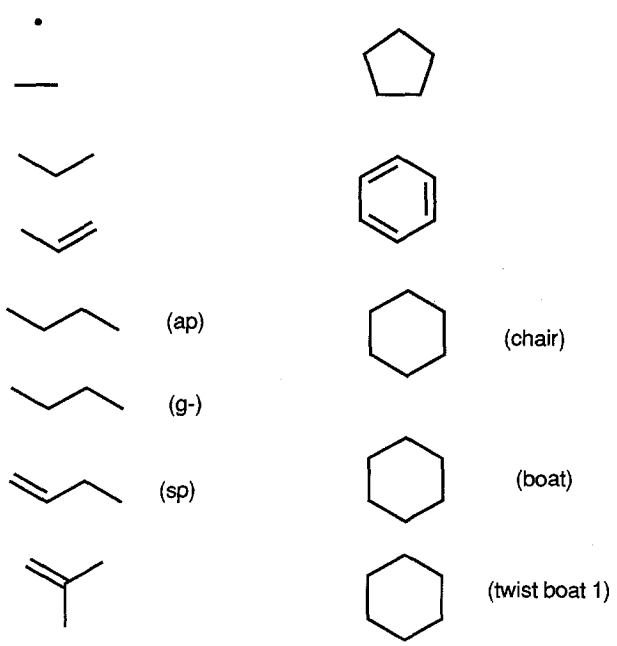

Fig. 10A. A template dictionary used to compare the performance of various parameters of the program.

Distance-based scoring function

A second run was performed using the scoring function based on distances, i.e., Eq. 2. No Monte Carlo iterations were included. The template dictionary included several five- and six-membered rings and several chain templates of 1–4 vertices, and is shown in Fig. 10A. The template at target sites A and B was fixed to the template that represents propenylene. This template could be oriented in several ways (the resolution was set to 15°). The run resulted in 39 skeletons. Some of the skeletons that include templates which represent six-membered rings are illustrated in Fig. 10B. The solutions containing the template that represents cyclohexane in the boat conformation were found earlier in the search than the skeletons containing the template that represents the more energetically favourable phenyl ring. This was because the distance-based scoring function was unable to distinguish between boats, twisted boats and planar rings as they all span a similar distance through space. Boat and twisted boat conformations of cyclohexane are relatively high-energy conformers, but the limited sampling of conformational space prevented solutions with the lower energy chair conformation from being generated.

Energy-based scoring function

The third run used the energy-based scoring function (f_e) (Eq. 7), to attempt to produce a better ordering of the solutions. The variables in Eq. 7 were set as follows: $x = 5$; $y = -25$; $z = -10$.

The same set of templates were available (Fig. 10A). The same solutions were generated as for the previous run but ordering the nodes of the graph by energy resulted in the low-energy solutions appearing earlier in the search. For example, the skeletons containing the planar rings were generated before the skeletons containing twisted boat and boat conformations. The results are illustrated in Fig. 11. This energy function attempts to estimate the energy difference between the bound conformation of the molecules represented by a skeleton and their unbound conformations in the gas phase. This energy difference is used to order skeletons: the skeleton which is closest in energy to its unbound conformation is processed first.

Monte Carlo procedure

The fourth run examined the effect of including the Monte Carlo routine within the method. In this experiment the torsion angles in the skeletons are varied randomly as the function in Eq. 8 is optimised (with $b = 150$; and $y = -30$).

Five iterations of the Monte Carlo procedure were performed each time a new template was added to a skeleton. The energy-based scoring function was used to order the nodes for expansion as in the previous run and the same set of templates was available. A number of solutions was produced many of which were similar to those obtained without the Monte Carlo procedure. In addition some new solutions were obtained which could not be generated previously. For example, Fig. 12 illustrates a skeleton that contains the chair conformation of the six-membered ring. Rotation has occurred about the 2–3 bond to produce a 1–4 dihedral angle of 78°. This dihedral angle cannot be produced using the current sampling interval used during template joining.

HIV protease

The second system to be tested was the enzyme HIV-1 protease complexed with acetyl pepstatin; Fig. 13A [31] (PDB entry 5HVP). The method for obtaining the primary constraints is as

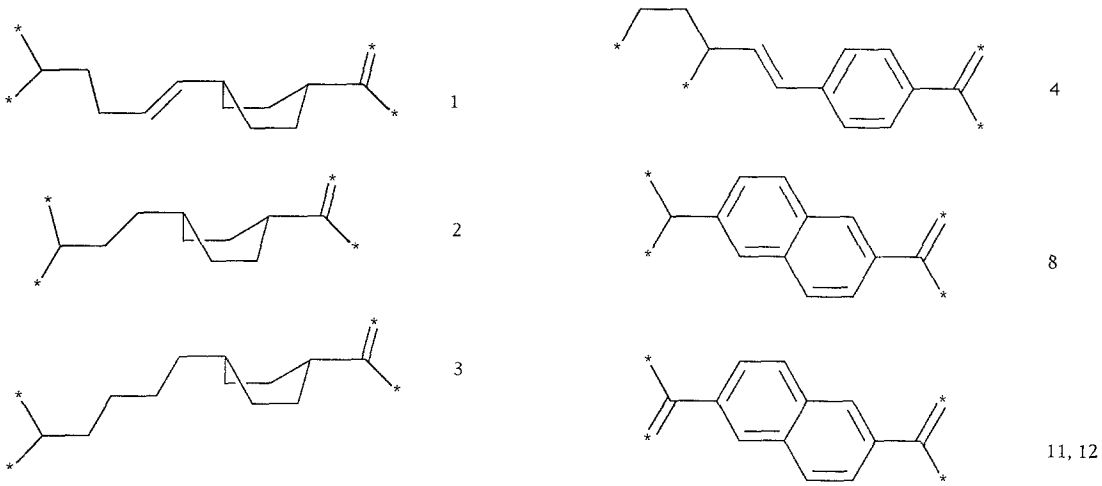

Fig. 10B. Some of the skeletons generated using the distance-based scoring function. The skeletons are numbered in the order in which they were found.

described for the APPA–trypsin complex. The first run included eight target sites and used the template dictionary shown in Fig. 10A. Acetyl pepstatin is shown in Fig. 13B and the atoms used as target sites are highlighted. The run used the energy-based scoring function (with x = 5; y = −25; and z = −20 in Eq. 7) to order the nodes and included five iterations of the Monte Carlo procedure (b = 150 and y = −30 in Eq. 8) each time a new template was added to a skeleton. A

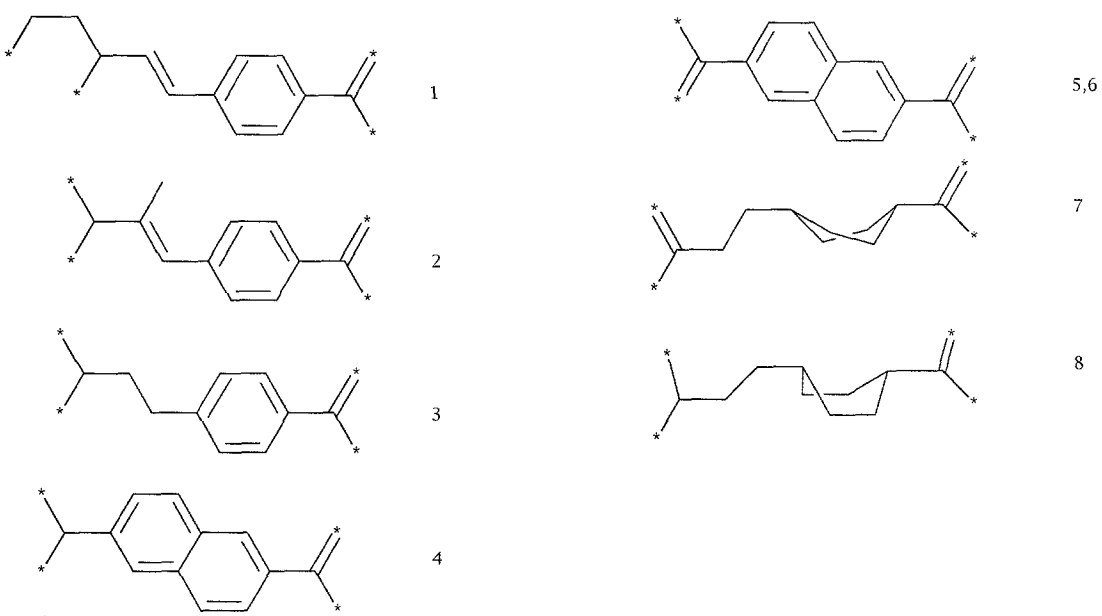

Fig. 11. Some of the skeletons generated using the energy-based scoring function. The skeletons are numbered in the order in which they were found. The twisted boat conformations appeared later. The energy-based scoring function resulted in the generation of the more energetically favourable skeletons first.

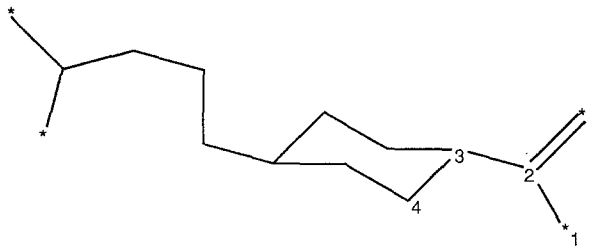

Fig. 12. Using the Monte Carlo routine resulted in the generation of skeletons containing the chair conformation of cyclohexane. The dihedral angle between atoms 1–4 is 78°. Without the Monte Carlo procedure only angles of 0, 60, 120°, etc. are permitted.

large number of skeletons was generated, one example is shown in Fig. 13C. In Fig. 13D this skeleton is superimposed on acetyl pepstatin. Two additional target sites were included in the second run to force the skeletons into some of the hydrophobic pockets of the receptor site. The target sites for this run are shown in Fig. 14A and a solution skeleton is shown in Fig. 14B. In Fig. 14C the skeleton is superimposed on acetyl pepstatin.

DISCUSSION

The trypsin example demonstrated both the validity of the approach and its predictive value. The example shows how the various parameters of the program can be altered and the effect that these can have on the order in which the skeletons are produced and the skeletons that can be generated. The energy-based scoring function can produce a better ordering of the solutions. However, in this case the search strategy tends towards a breadth-first search, hence this function can be less useful for very large search graphs. The distance-based scoring function results in a more depth-first search of the graph and this can be useful for finding solutions early when the graph is large. Including the Monte Carlo procedure allowed skeletons to be generated that could not be found by the stepped sampling of conformational space and therefore this is a useful procedure.

The HIV protease example highlighted some limitations of the approach. In the first run the program was aborted after 15 h cpu time on a VAX workstation (3100 Model 38) after 65 solution skeletons had been generated. All of these solutions reach the target sites and occupy a similar volume to acetyl pepstatin. However, many of these solutions are similar, i.e., they have many templates in common. This implies that the program had searched only a part of the graph and that if the program had been allowed to continue, an enormous number of solutions would have been found. This suggests that new strategies are required for receptor sites of this size, for example, moving from one branch of the graph to another unexplored part once some solutions have been found.

The target sites were initially developed to allow the hydrogen bonding constraints to be satisfied. The second test for HIV protease modelled some hydrophobic constraints using the same type of target sites. This was successful in forcing the skeletons to grow into the hydrophobic pockets since the new target sites were satisfied. However, these target sites, that are satisfied by a single vertex, are not appropriate for the more diffuse constraints such as hydrophobicity. Different types of target sites are required for representing these constraints.

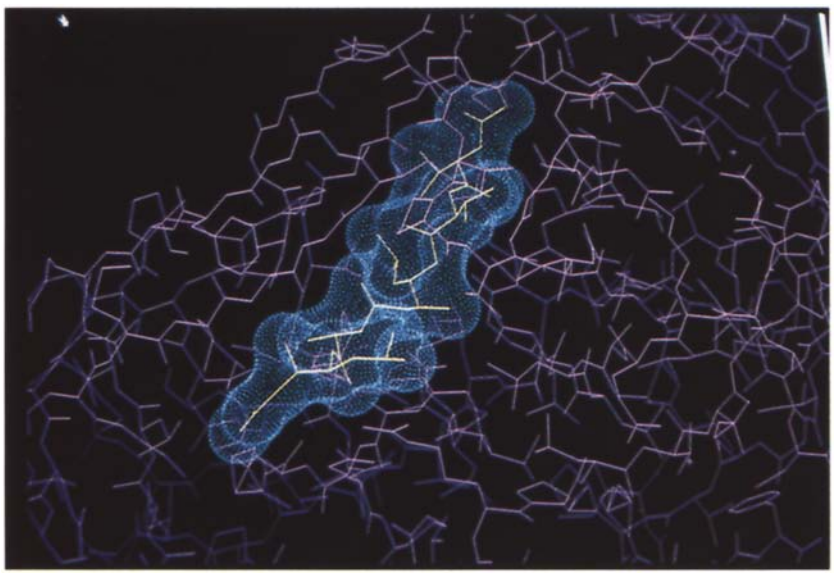

Fig. 13A. Acetyl pepstatin bound to HIV-1 protease.

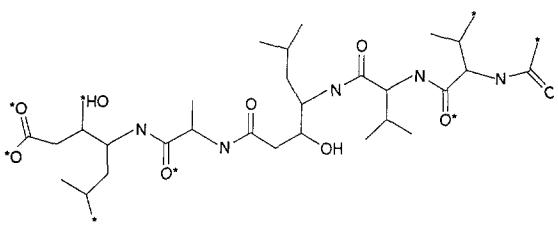

Fig. 13B. Acetyl pepstatin. The eight atoms used as target sites are highlighted by asterisks.

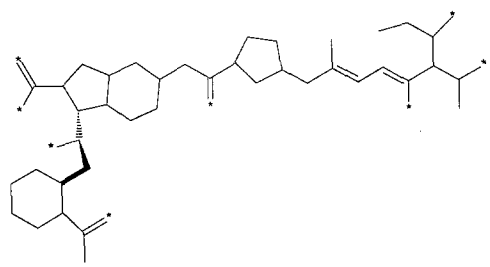

Fig. 13C. A solution skeleton generated for the acetyl pepstatin binding site of HIV protease using eight target sites. The atoms that correspond to the target sites are highlighted.

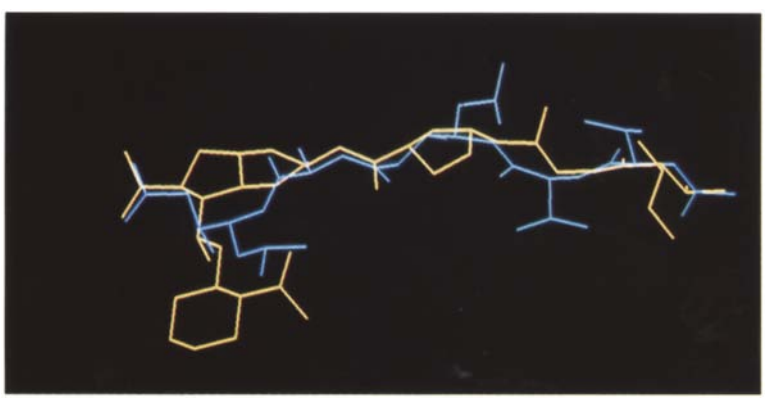

Fig. 13D. The skeleton (yellow) superimposed on acetyl pepstatin (blue).

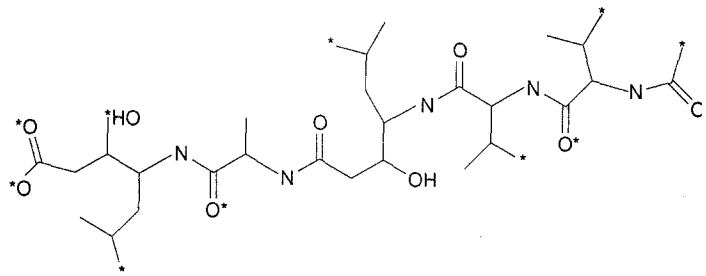

Fig. 14A. Acetyl pepstatin. Ten atoms are used as target sites: highlighted by the asterisks. Two target sites are included from the hydrophobic pockets of the binding site.

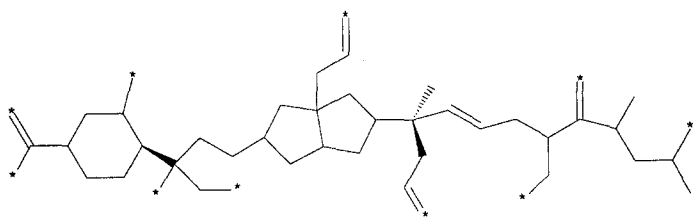

Fig. 14B. A solution skeleton generated for the acetyl pepstatin binding site of HIV protease using ten target sites. The atoms that correspond to the target sites are highlighted.

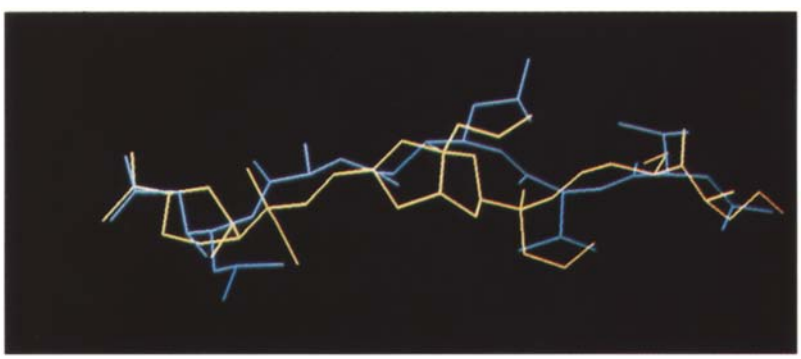

Fig. 14C. The skeleton (yellow) superimposed on acetyl pepstatin (blue).

CONCLUSIONS

SPROUT has been introduced in this paper for the design of structures to fit receptor sites of enzymes whose 3D coordinates are avilable. However, the program is designed to generate structures to fit a wider variety of constraints. In the examples shown the receptor coordinates are used to define a volume that limits the shape of the structures. A volume can be specified in other ways, for example, by a molecular surface produced by superimposing sets of molecules. The volume defines a boundary that can be useful in pruning the search graph. However, *the boundary is not essential* since structure generation is always directed towards the target sites. Thus the program can be applied equally well to situations where the boundary is less well defined, e.g., to

the output from a hypothesis program such as CATALYST [32]. We are currently developing a graphical interface which will allow modifications to be made to the boundary to handle flexible boundaries.

The variety of structures that can be generated is determined by the templates that are available. The program is designed for flexibility and the template library is not a fixed entity. It can be modified, either to restrict the templates available for a particular application or to add new templates into the library. Fragments where the atom types are differentiated, such as functional groups, can also be added to the library to allow functional groups to be positioned at target sites for more tightly constrained problems.

Of the programs described in the Introduction, SPROUT is most similar in concept to LEGEND, GROW and LUDI. However, it is broader in scope that any of these programs, both in the kinds of structures that can be generated and in the range of applications. LEGEND builds structures from single atoms giving rise to a much larger search space than when larger fragments are used as building blocks. A systematic search through the problem space is not feasible and a limited search is made based on the use of random numbers. The GROW program uses molecular fragments as building blocks but a large number of fragments is required since the atom types are differentiated. The size of the problem is managed by restricting the number of successors each time a new fragment is added. The fragments are derived from the amino acids and GROW is restricted to the generation of peptide-like structures. The LUDI program also uses molecular fragments. It positions fragments at each site and then attempts to generate structures by bridging the fragments. To explore the search space thoroughly using this method each fragment would need to be positioned in a number of orientations at each site. This would give rise to a large number of combinations of fragments to be linked.

SPROUT uses a number of techniques to moderate the combinatorial problem inherent in structure generation and thus is able to make a more thorough exploration of the search space. These techniques include the definition of templates, each of which represents a number of molecular fragments, and the use of heuristics. In the examples presented here target sites are selected manually based on the atom coordinates of bound ligands. This explores only a single binding mode for each receptor site. However, different methods can be used to define the target sites automatically, e.g., using GRID [33], HSITES [19] or GREEN [20]. We are currently investigating these methods as a way of exploring different binding modes for a given receptor site.

The results presented here demonstrate the potential of the method for generating novel structures which would be difficult to find by other methods. Some limitations of the method have been reported. In summary, our efforts are currently directed towards increasing the efficiency of the program so it can be applied more effectively to large receptor sites; implementing the secondary structure generation phase of the program; and the development of a flexible interface. Secondary structure generation will ensure that the target sites are satisfied by substructures that can exhibit the desired properties.

Persons interested in using the program are requested to contact the authors.

ACKNOWLEDGEMENTS

We thank Drs. Richard Simpson and Chris Marshall for helpful discussions. The work has been supported by The Maxwell Foundation.

REFERENCES

1 Martin, Y.C., J. Med. Chem., 35 (1992) 2145.
2 Martin, Y.C., J. Comput.-Aided Mol. Design, 3 (1989) 225.
3 Bartlett, P.A., Shea, G.T. and Telfer, S.J., In Roberts, S.M. (Ed.) Molecular Recognition: Chemical and Biological Problems, The Royal Society of Chemistry, London, 1989, pp. 182–196.
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.
5 Martin, Y.C., Tetrahedron Comput. Method, 3 (1990) 15.
6 Lewis, R.A. and Dean, P.M., Proc. R. Soc. London Ser. B, 236 (1989) 125.
7 Lewis, R.A., J. Comput.-Aided Mol. Design, 4 (1990) 205.
8 Lewis, R.A., J. Mol. Graphics, 10 (1992) 131.
9 Nishibata, Y. and Itai, A., Tetrahedron, 47 (1991) 8985.
10 Moon, J.J. and Howe, W.J., Proteins Struct. Funct. Genet., 11 (1991) 314.
11 Böhm, H.J., J. Comput.-Aided Mol. Design, 6 (1992) 61.
12 Lewis, R.A., Roe, D.C., Huang, C., Ferrin, T.E., Langridge, R. and Kuntz, I.D., J. Mol. Graphics, 10 (1992) 66.
13 Verlinde, C.L.M.J., Rudenko, G. and Hol, W.G.J., J. Comput.-Aided Mol. Design, 6 (1992) 131.
14 Wipke, W.T., Pitman, M.C., Koehler, R.T., Kislin, B.S. and Anderson, G.D., Abstracts of Papers, 202nd National Meeting of the American Chemical Society, New York, NY, Fall 1991, American Chemical Society, Washington, DC, 1991.
15 Doweyko, A.M., J. Med. Chem., 31 (1988) 1396.
16 Cohen, N.C., Blaney, J.M., Humblet, C., Gund, P. and Barry, D.C., J. Med. Chem., 33 (1990) 883.
17 Nilsson, N.J., Principles of Artificial Intelligence, Springer-Verlag, Berlin, 1982.
18 Gillet, V.J., Johnson, A.P., Mota, P. and Sike, S., in preparation.
19 Danziger, D.J. and Dean, P.M., Proc. R. Soc. London Ser. B, 236 (1989) 101.
20 Tomioka, N., Itai, A. and Iitaka, Y., J. Comput.-Aided Mol. Design, 1 (1987) 197.
21 Richards, F.M., Annu. Rev. Biophys. Bioeng., 6 (1977) 151.
22 Williams, D.H., Cox, J.P.L., Doig, A.J., Gardner, M., Gerhard, U., Kaye, P.T., Lal, A.R., Nicholls, I.A., Salter, C.J. and Mitchell, R.C., J. Am. Chem. Soc., 113 (1991) 7020.
23 Mohamadi, F., Richards, N.G.J., Guida, W.C., Liskamp, R., Lipton, M., Caufield, C., Chang, G., Hendrickson, T. and Still, W.C., J. Comput. Chem., 11 (1990) 440.
24 Hagler, A.T., Huler, E. and Lifson, S., J. Am. Chem. Soc., 96 (1974) 5319.
25 Howard, A.E. and Kollman, P.A., J. Med. Chem., 31 (1988) 1669.
26 Saunders, M., Houk, K.N., Wu, Y., Still, W.C., Lipton, M., Chang, G. and Guida, W.C., J. Am. Chem. Soc., 112 (1990) 1419.
27 Goodsell, D.S. and Olson, A.J., Proteins Struct. Funct. Genet., 8 (1990) 195.
28 Guida, W.C., Bohacek, R.S. and Erion, M.D., J. Comput. Chem., 13 (1992) 214.
29 Marquart, M., Walter, J., Deisenhofer, J., Bode, W. and Huber, R., Acta Crystallogr. Sect B., 39 (1983) 480.
30 QUANTA, Polygen Corp., Waltham, MA 02154, U.S.A.
31 Fitzgerald, P.M.D., McKeever, B.M., van Middlesworth, J.F., Springer, J.P., Heimbach, J.C., Leu, C.-T., Herber, W.K., Dixon, R.A.F. and Darke, P.L., J. Biol. Chem., 265 (1990) 14209.
32 Teig, S.L., In Collier, H., (Ed.) Proceedings of the Montreux 1992 International Chemical Conference, Infonortics Ltd., Calne, U.K., 1992, pp. 195–208.
33 Goodford, P.J., J. Med. Chem., 28 (1985) 849.