Journal of Computer-Aided Molecular Design, 7 (1993) 3–22
ESCOM

J-CAMD 185

FOUNDATION: A program to retrieve all possible structures containing a user-defined minimum number of matching query elements from three-dimensional databases

Chris M.W. Ho and Garland R. Marshall*

Center for Molecular Design, Washington University, St. Louis, MO 63130, U.S.A.

Received 26 May 1992
Accepted 5 September 1992

Key words: Drug design; Database searching; Clique algorithm; Foundation; Molecular graphics

SUMMARY

A program is described that searches three-dimensional, structural databases, given a user-defined query, in order to retrieve all structures that contain *any combination* of a user-specified *minimum number* of matching elements. Queries consist of three-dimensional coordinates of atoms and/or bonds. Numerous query constraints are described which allow the investigator to define the chemical nature of the desired structures as well as the environment within which these structures must reside. They include:
(1) Bonded vs. isolated atom distinction;
(2) Atom type designation;
(3) Definition of subsets with occupancy specification (>, =, < X atoms);
(4) RMS-fit;
(5) Active site volume accessibility of atoms linking query elements;
(6) Number, atom type, and cyclic structure constraints for atoms linking pharmacophoric elements;
(7) Automatic error boundary adjustment – *ad infinitum* constraint.
To illustrate the capabilities of this program, queries based on the crystal structure of a thermolysin–inhibitor complex were tested against a subset of the Cambridge Crystallographic Database. Several compounds were returned which satisfied various aspects of the query, including fitting within the active site. Combination of segments of compounds which satisfy partial queries should provide a method for generating unique compounds with affinity for sites of known three-dimensional structure.

INTRODUCTION

Most drugs in current use have been developed based on optimization of lead compounds, either isolated from the screening of natural products, or modified from the hormone or transmitter whose activity is being studied. Modern techniques offer increasing opportunities to define biochemically a logical therapeutic target, to clone its gene and express the product, and to

* To whom correspondence should be addressed.

determine the three dimensional (3D) structure, either by crystallography or through modern NMR spectroscopic techniques. The availability of the 3D structure of a therapeutic target presents a challenge to those proponents of rational drug design. The technology to design ligands for a given 3D receptor site is under rapid development, although still in its early stages.

The majority of drugs are non-covalent ligands designed to bind to specific enzymes and/or receptors in order to modulate their biological activity. As a medicinal chemist seeks to modify the potency of a particular drug, visualization of the system at an atomic level offers insight into the basis of molecular recognition and suggests possible chemical alterations. Even when studying the crystal structure of a particular drug–receptor complex, however, molecular modifications necessary to improve binding are not immediately obvious, although examples of the successful use of this approach exist, e.g. improvements in affinity for DHFR antagonists [two groups]. In fact, due to competitive pressures, a requirement for non-obvious alterations to allow patent protection is often imposed.

To aid in the process of designing drugs for known or hypothetical receptor sites, medicinal chemists have recognized the potential of searching 3D chemical databases. Several databases are well known, such as the Cambridge Crystallographic Database [1], which contains nearly 90 000 structures of small molecules. The crystal coordinates of proteins and other large macromolecules are deposited into the Brookhaven Protein Databank [2]. Non-crystallographic databases have been developed as well. One such example is the 3D database of structures from Chemical Abstracts generated using CONCORD [3], which contains nearly 500 000 entries.

The use of such databases is most applicable when the binding of a particular ligand and its receptor has been fully understood and a crystal structure of the complex is known [4]. The 3D orientation of the key regions present on the drug that are crucial for molecular recognition and binding is termed the pharmacophore [5,6]. One approach is to develop a novel chemical architecture (i.e. scaffolds) that positions these pharmacophoric groups, or their bioisosteres, in the correct 3D arrangement. It is hoped that the conformations present in crystallographic databases reflect low-energy conformers readily attainable in solution and in the receptor complex. The investigator then searches the 3D database using a query for fragments that contain the pharmacophoric functional groups in the proper 3D orientation. With these fragments as 'building blocks', completely novel structures may be constructed through assembly and pruning [7]. Receptor sites are complex, both in geometrical features as well as their potential energy fields, and many diverse compounds can bind to the same protein by occupying various combinations of subsites.

Gund conceived the first prototypic program designed to search molecules for matches to 3D pharmacophoric patterns [5]. This program, MOLPAT, performed atom-by-atom searches to verify the identity of interatomic distances between pattern and candidate structures. Although rigorous, such an approach was tedious, and required optimization. Lesk [8] devised a method that used the geometric attributes of the query to screen potential candidates. Similarly, Jakes and Willett proposed that screens based upon interatomic distances and atom types could considerably increase the search efficiency [9]. In addition, Jakes et al. showed that methods widely used in 2D structure retrieval could be applied to 3D searches to remove the vast majority of compounds prior to more rigorous comparisons [10]. This was validated in test searches against a subset of the Cambridge Structural Database. This concept was furthered by Sheridan et al., who included screens based on aromaticity, hybridization, connectivity, charge, position of lone pairs, and centers of mass of rings [4]. To contain this wealth of information, an inverted bit map was used. This allowed for highly efficient screening, as hundreds of thousands of compounds could be screened in minutes.

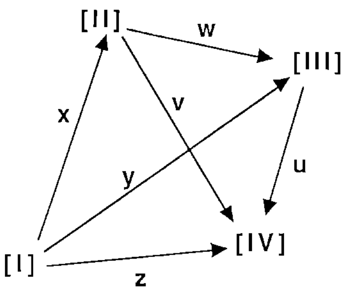
These methods have been incorporated into a number of current database searching systems. Programs such as CAVEAT [11], ALADDIN (Abbott) [12], 3DSEARCH (Lederle) [4], MACCS-3D [13], CHEM-X (Chemical Design Ltd.) [14] and others contain considerable functionality useful for such an approach. CAVEAT is designed to assist a chemist to identify cyclic structures that could serve as the foundation for novel compounds. In particular, it allows an investigator to rapidly search structural databases for compounds containing substituent bonds that satisfy a specific geometric relationship. ALADDIN, 3DSEARCH, MACCS-3D, and CHEM-X are similar in that geometric relationships between various user-defined atomic components can be rapidly searched to retrieve matching structures. Features have been included to allow the user to delineate molecular characteristics (atom type, bond angles, torsional constraints, etc.) to ensure the retrieval of relevant compounds. Additional constraints have been incorporated into 3DSEARCH and ALADDIN, including the consideration of retrieved ligand–receptor volume complementarity. Furthermore, CHEM-X performs a rule-based conformational search on each structure in the database in order to account for conformational flexibility. For a comprehensive review of 3D chemical database searching, see Martin et al. [15,16].

The feature that most programs have stressed is their search speed and efficiency. This has meant the use of preliminary screens, based on various criteria, to eliminate the vast majority of compounds prior to more rigorous pattern matching comparisons [15,16]. As such, structures lacking a particular query element are promptly screened out. This search strategy is indeed very quick and efficient; however, a result is that all retrieved compounds must contain every query component, at least, as defined in the preliminary screens and currently practised. As the number and complexity of the query elements increase, one would anticipate fewer true hits, but a corresponding rise in the number of near-misses. If such near-misses could be recovered, effective ligands may simply arise from slight conformational modifications to maximize receptor interactions. Furthermore, the retrieval and combinatorial assembly of numerous pharmacophore subcomponents would intuitively produce many, more-diverse structures than the quest for a 'magic bullet' compound incorporating the entire pharmacophore, i.e. all requirements of the query. This suggests an approach that would retrieve compounds containing any combination of a minimum number of matching pharmacophoric elements. We have developed a program, called FOUNDATION, which searches a 3D database of chemical structures for a user-defined query containing the coordinates of various atoms and/or bonds. FOUNDATION finds all possible structures that contain any combination of a user-specified minimum number of matching atoms and/or bonds. In addition, our program incorporates numerous user-defined constraints to retrieve the most useful compounds. With this capability, FOUNDATION greatly improves the effectiveness of the search process in terms of functionality, if not in speed, and allows exploration of useful concepts in design specification.

DISCUSSION OF ALGORITHMS

Representation of 3D structures

The first step in representing the 3D relationship of a group of atoms is to transform their



	I	II	III	IV
I	---	x	y	z
II	x	---	w	v
III	y	w	---	u
IV	z	v	u	---

Fig. 1. A sample query and its conversion into a distance matrix. Atoms are shown as roman numerals while distances are represented by lowercase letters.

description in Cartesian space to one in distance space. This is shown in Fig. 1. For any given query with N atoms, there are N(N−1)/2 independent descriptive distances. What results is a distance matrix detailing the interatomic distances as a function of atom pairs. Previous investigators have used distance space representations of geometric data since it is an orientation-independent description [4–12,15,16,22–28]. This is important as solutions to the query may reside in any frame of reference. The task then becomes one of finding structural fragments in the database whose distance descriptors match those of the query.

Graph theory applications

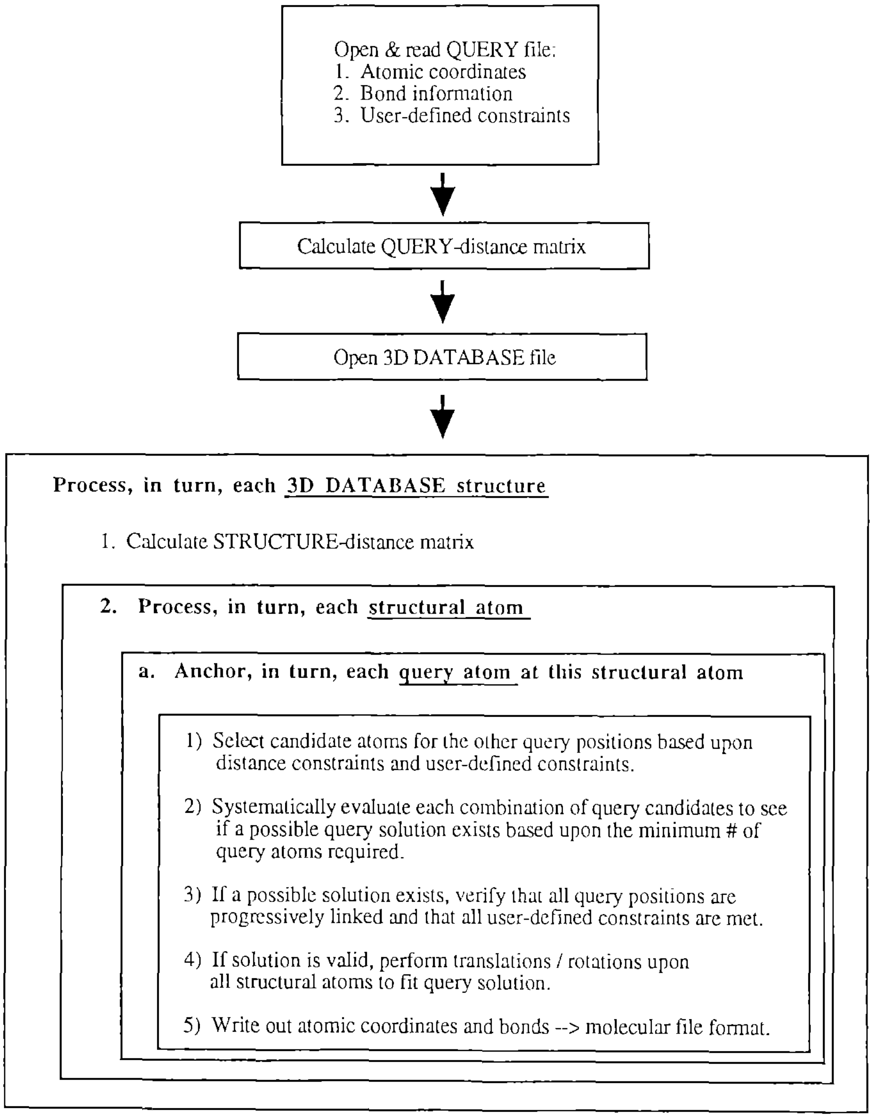
This task can be described using graph theory [17]. In graph theory, a graph is a structure comprised of nodes (vertices) connected by edges. A graph is completely connected when all nodes are connected to one another. A subgraph is any subset of a larger graph. The largest completely connected subgraph of any graph is called a clique. Thus, our query is a completely connected graph, as all interatomic distances are determined in the distance matrix. The task is then to search a structural database to find all cliques that contain at least a user-determined number of matching nodes.

There are many published clique-finding algorithms. Some of the more well-known procedures include those by Bierstone [18], Bonner [19], Gerhards and Lindenberg [20], and Bron and Kerbosch [21]. Computational chemists have adapted these algorithms or implemented similar ideas to facilitate searching for 3D structures within databases [22–28]. In contrast to the rapid screening approaches described above, clique-finding methods must contend with additional computational burden. That is, the numerous solutions generated from the combinatorial assortment of nodal subsets. However, the gain in using these methods is the ability to determine partial substructural matches.

I.D. Kuntz has applied similar principles in his geometric approach to determining binding modes of ligands within receptors [22]. In this and subsequent work [23,24], Kuntz detailed a method of reducing the volume contained within a receptor to a set of points in 3D space. These points represent the centers of logistically placed packing spheres. Similarly, a set of points is produced to represent the volume of a potential binding ligand. The published algorithm enables one to determine numerous binding geometries by matching subsets of receptor patterns with those of the ligand. Since partial matches are recovered, binding configurations must be verified to eliminate those containing explicit steric overlap.

Crandell and Smith published an algorithm to determine the maximal common 3D substructure (MCS) of a set of compounds [25]. Their algorithm entailed the 'growing' of cliques atom by atom and verifying commonality to all structures in the dataset. A clique-finding approach was necessary since any combination of atoms (nodes) could produce the MCS. Brint and Willett [26] later showed that the clique-finding algorithm of Bron and Kerbosch [21] was substantially faster than that of Crandell and Smith.

Kuhl et al. also considered the problem of predicting ligand–receptor binding modes [27]. They



Open & read QUERY file:
1. Atomic coordinates
2. Bond information
3. User-defined constraints

Calculate QUERY-distance matrix

Open 3D DATABASE file

Process, in turn, each 3D DATABASE structure

1. Calculate STRUCTURE-distance matrix

2. Process, in turn, each structural atom

a. Anchor, in turn, each query atom at this structural atom

1) Select candidate atoms for the other query positions based upon distance constraints and user-defined constraints.

2) Systematically evaluate each combination of query candidates to see if a possible query solution exists based upon the minimum # of query atoms required.

3) If a possible solution exists, verify that all query positions are progressively linked and that all user-defined constraints are met.

4) If solution is valid, perform translations / rotations upon all structural atoms to fit query solution.

5) Write out atomic coordinates and bonds --> molecular file format.

Fig. 2. Schematic outline of entire search process.

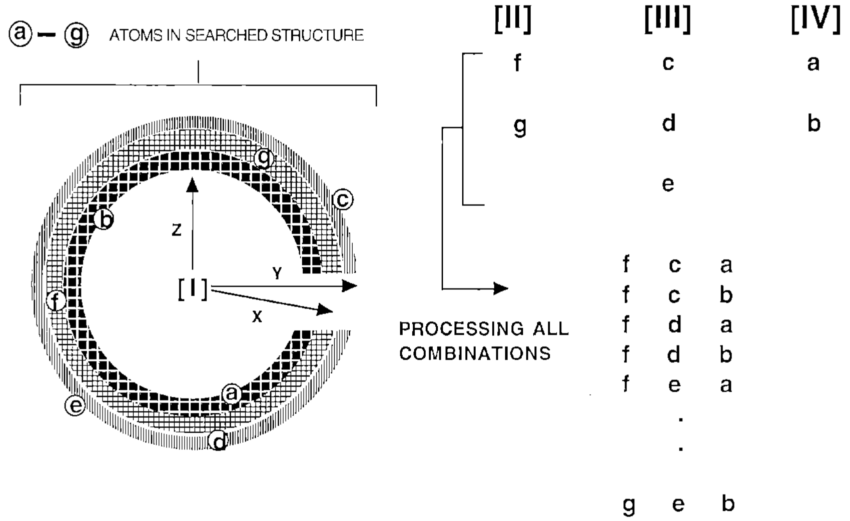


Fig. 3. Selection of candidate atoms for each query position [II]→[IV] by using distance filters x, y, and z, respectively.

described a similar scenario where a set of ligand points must be mapped onto a multitude of binding sites loci. However, they did not use methods to decompose the receptor or ligand volumes into simpler representations. In their study they used a non-recursive implementation of the Bron and Kerbosch clique-finding algorithm [21] to determine possible binding configurations.

More recently, Smellie et al. again used the Bron and Kerbosch clique-finding algorithm to perform fast drug–receptor mapping [28]. A novel approach was developed to determine quickly whether any of a given series of compounds could interact with an active site of interest. The implementation of the clique-finding algorithm discussed above was used [27]. However, specific filters were described that isolated the best ligand candidates and increased computational efficiency.

Clique-finding methodology

Our clique-finding method is explained in Figs. 2 and 3. It is a modified form of the geometric search procedure detailed by Jakes et al. [10]. In their paper, they discussed set reduction techniques. In effect, set reduction methods increase the 'signal-to-noise ratio' of the searched structure by highlighting those atoms whose relative geometry resembles the query while obscuring those that do not. What results is a list of candidate atoms that correspond to specific query elements. Combinations of the candidate atoms are then checked for equivalence to the query by comparison of all interatomic distances. In the past, this was done by brute force, but a more efficient depth-first search procedure was proposed by the authors [10]. In our implementation, we retrieve potential matches by systematically superimposing each query element upon every atom of the structure being processed. Candidates for the remaining query elements are collected using the interatomic lengths specified in the distance matrix as shown in Fig. 3. All candidate combinations are then evaluated for query equivalence.

Computational efficiency

Naive clique-finding algorithms belong to the class of problems known as non-deterministic polynomial-time, NP-complete [29]. Such problems exist in varied disciplines, are often of great practical value, and have attracted the attention of researchers for years. The common dilemma of all such problems is that no efficient solutions exist. Thus, one must computationally restrict the problem using assumptions to allow feasible solutions.

Well-chosen assumptions have enabled previous investigators to reduce the computational burden associated with the clique-finding solution. I.D. Kuntz, in his ligand–receptor docking procedures, eliminated considerable computational burden by considering ligands whose volume was similar to that of the receptor [22]. This reduced the total number of evaluated nodes during clique-searching. Furthermore, binding modes (cliques) with fewer than four ligand–receptor interactions (nodes) were rejected. In a similar fashion, Smellie et al. eliminated binding modes (cliques) containing fewer than five ligand–receptor interactions (nodes) in their algorithm to perform fast drug–receptor mapping [28]. Sets of degenerate cliques were excluded as well. Lastly, Kuhl et al. implemented constraints based upon ligand geometry in restricting their binding mode determination algorithm sufficiently to avoid NP-completeness [27].

In our approach, computational efficiency is critical. During the selection of query element candidates, user-specified error margins influence the number of atoms chosen. Because of the combinatorial nature of the clique-search procedure, the large number of candidate atoms per element translates into a never-ending process. To prevent this, a variety of user-specified constraints have been implemented. These constraints perform two major functions. First, by screening and removing those query candidates that fail to meet user specifications, a majority of candidate atoms are eliminated. This drastically reduces the number of combinations that must be checked. Second, because the algorithm is designed to retrieve all combinations of query atoms, an overabundance of structures may be recovered. User-defined query constraints act as a filter to ensure that the most relevant candidates for the system under study are returned.

Atom type constraints allow for the specification of particular atom types at each query element. All candidates not possessing the appropriate atom type(s) are removed.

Bonded atom constraints allow an investigator to search for bonded atom pairs wherever

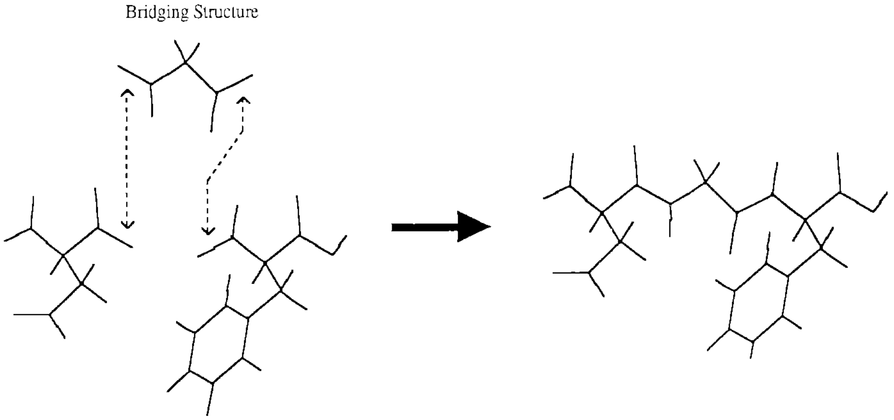


Fig. 4. An illustration of using bond constraints when bridging two separate structures.

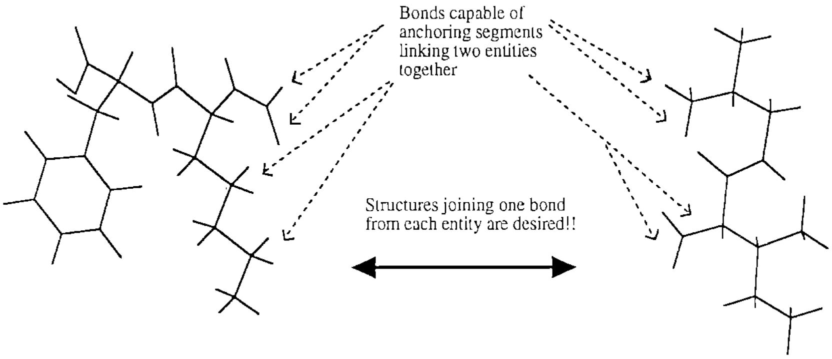

Bonds capable of anchoring segments linking two entities together
Structures joining one bond from each entity are desired!!

Fig. 5. Linking two entities which contain numerous possible anchoring bonds.

desired. This is necessary when bridging structures are sought between chemical groups. As shown in Fig. 4, bridging compounds must contain bonds that align with those in both entities. This constraint is implemented by scanning the candidates collected for each of the two bonded query elements. Any unpaired candidates are eliminated.

A subset constraint allows the user to specify groups of atoms upon which various occupancy requirements must be met. Although our program is designed to retrieve all combinations of the specified query elements, occasions may arise where a specific group of atoms must always be present for any retrieved structure to be useful. These atoms can be defined as a subset with the requirement that all be present for any structure to be considered as a hit. In other instances, one may have a group of atoms from which at least one or more must be present. The same atoms are again placed into a subset, but a requirement of (number of atoms present >0) is designated instead. We discuss two instances where this constraint is useful.

In linking two entities, FOUNDATION can be used to retrieve bridging structures, as shown in Fig. 5. However, because structures matching any combination of query elements are returned,

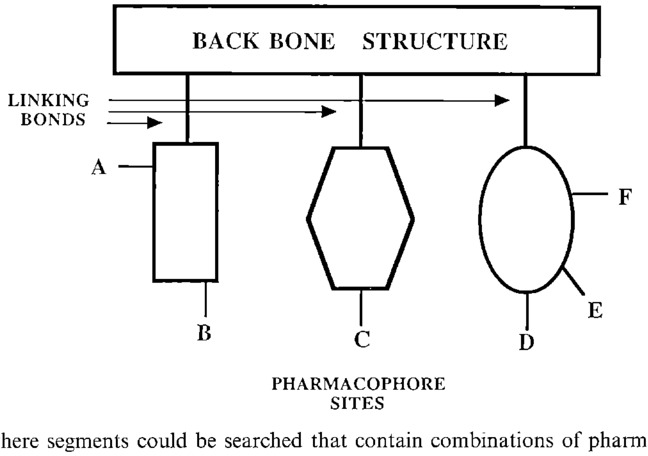


Fig. 6. A situation where segments could be searched that contain combinations of pharmacophoric sites [A→F] and branch from a common backbone structure.

compounds linking bonds within the same chemical entity may also be recovered. To retrieve useful structures, we define two subsets that contain the anchor atoms of each chemical entity. An occupancy requirement of one bond per subset guarantees that any structure found spans both entities.

In modifying peptide ligands, the backbone often makes hydrogen bonds with the receptor/enzyme and may, initially, best be left alone. What remains are the side chains that may contain several other pharmacophoric binding sites as well. This is shown schematically in Fig. 6. The best strategy in this case is to search for segments that contain any combination of the pharmacophoric sites A–F, yet retain bonds that link each segment to the backbone. We accomplish this by placing all three linking bonds into a subset and specifying an occupancy requirement of one bond in this subset. This ensures that all recovered segments can branch from the backbone.

An rms-fit constraint measured from the loci of the original query atoms is necessary due to structural ambiguity that may occur with the use of interatomic distance descriptors. Discernible differences in orientation due to small changes in atomic position may not alter interatomic distances significantly. Therefore, the routine BMFIT [30] is implemented after potential hits are found to orient and align each solution to the original query. An rms-fit coefficient is calculated to be certain that a solution structure matches the query. Furthermore, the use of an rms-fit constraint allows the recovery of specific stereoisomers since both a compound and its mirror-image are described by identical interatomic distances.

Another important constraint determines the structural complementarity that a query solution possesses with the ligand-accessible space in which it must reside. To represent the ligand-accessible space, we used a filler lattice of points [31] created by using a flood-fill algorithm [32] that conforms to the internal volume of the active site. Potential hits are aligned with the query following the rms-fit procedure above. A search is used to determine the atoms comprising the shortest connecting paths between all atoms matching the query. An atom-checking routine used by previous investigators is then used to establish whether these atoms reside within the cavity [12,33]. The user can specify a threshold percentage of atoms which must be clear for a structure to be accepted.

There are several reasons for considering the atoms comprising the shortest path:

First, we prevent the rejection of numerous structures that contain forbidden steric contacts even though the shortest path connecting the pharmacophoric elements may be clear. If these extraneous atoms were pruned or modified during the drug design process, these structures would be acceptable.

Second, we eliminate the problem of hits that are not 'continuous'. In any crystallographic database, for example, there are often structures that contain several partial unit cells, or include bound waters. Atoms matching the query may be found in these separate, distinct structures, resulting in hits that contain no atomic connection between pharmacophoric groups. These hits are thus eliminated.

Third, by isolating the connecting atoms, we can implement specifications that govern both the number and types of atoms found in these connecting regions. An excellent example is in the recovery of crystallographic structures containing dummy (Du) atoms. Because the dummy atom, usually a metal ion, is often coordinated to numerous other atoms, these structures appear as hits quite frequently. By specifying that all connecting atoms be non-Du atoms, we can easily filter out these anomalies. One might argue that it would be more efficient to simply eliminate these structures from the database altogether. However, since the Du atom is usually a small part of the entire structure, the remainder can still provide interesting and valid structures.

Finally, by simply checking for two or more different, unique paths between two pharmacophoric subsites, cyclic systems can be selected for. Since all the linking paths are accounted for, one can also specify how many different query elements should be involved in the ring system. This can be very useful when ligand stability is desired.

The final constraint acts to prevent FOUNDATION from calculating ad infinitum if the error limits allowed for each query element are too large. As mentioned above, the error margins designated for each query element determine the number of candidates found. If the user-defined constraints fail to screen out the majority of candidates for each query element, an enormous number of distance comparisons results. For example, if, on average, seven candidates are recovered for each item in a ten-atom query, the brute force algorithm must process approximately 7^{10} combinations of candidate atoms. This is nearly 300 000 000 cycles, and would effectively stall the execution of the program. One approach would be to reduce the error margins designated for each query element. However, this may drastically reduce the number of hits for the entire search. Instead, by using the ad infinitum constraint, we can maintain larger error margins for the query elements. Prior to the geometric search, the total number of iterations is calculated. If this number exceeds a predetermined limit, the error margins for that particular structure are reduced by a small percentage and candidates are again selected. This process continues until a satisfactory number of candidates is recovered.

Postprocessing and reporting results

Retrieved structures are written into a file to generate a query report. Pertinent data include the structure name, listing number, number of matching elements, and corresponding query and atom numbers. FOUNDATION can rank the assorted hits as a function of the number of matching atoms, rms-fit, and steric complementarity. Furthermore, since each compound is rotated to match the query configuration in calculating the rms-fit, all hits are saved in the proper orientation for superposition with the query.

Database storage

There are several aspects to the storage of vast amounts of structural information. Preprocessing the data allows one to store it in a form that is more efficiently used by the retrieval system, and usually decreases the search time dramatically. Since the FOUNDATION program is an approach designed for unanticipated queries and completeness rather than speed, any preprocessing of the data, such as inverting the database to provide keyed entries, is unnecessary. Consequently, only the minimum amount of information is needed. For each structure, the following data are required: structure name, number of atoms, number of bonds, {xyz} coordinates and type for each atom, and bonded pair list. Because such minimal information is needed, the FOUNDATION program can be interfaced with virtually any pre-existing database.

COMPUTATIONAL METHODS

Implementation

FOUNDATION is written in C and presently runs on the Silicon Graphics IRIS, SUN, and

TABLE 1
SEARCH QUERY #1 FOR THERMOLYSIN INHIBITOR

Database searched:	**cho_dbase**
Threshold number of query atoms	**= 4**
Number of structures skipped	**= 0**
Hit limit	**= 1000**

Atoms

1	−2.30530	2.10470	−3.45600	±0.150
2	−0.49630	−0.64430	0.03100	±0.150
3	0.93460	−6.35910	−1.36360	±0.150
4	2.66670	−2.22830	−0.73600	±0.150
5	0.66170	−3.22130	0.59000	±0.150
6	−0.52930	−4.83730	−1.28000	±0.150
7	−1.36050	−0.97290	0.43190	±0.150
8	−0.33380	−3.08190	0.56660	±0.150
9	0.45420	−6.63630	−2.13770	±0.150

Bonds
[1] query atoms: 2→7
[2] query atoms: 5→8
[3] query atoms: 3→9

Type: Atom # 7⇒	13								
Type: Atom # 8⇒	13								
Type: Atom # 9⇒	13								
Type: Atom # 1⇒	8	9							
Type: Atom # 4⇒	8	9							
Type: Atom # 6⇒	8	9							
Type: Atom # 2⇒	5	6	7	11	28	19	31	8	9
Type: Atom # 3⇒	5	6	7	11	28	19	31	8	9
Type: Atom # 5⇒	5	6	7	11	28	19	31	8	9

RMS-fit threshold set at 0.1500 Å.

Cavity file = tln.fil. Cavity inclusion dist. = 1.000000.
Specified fraction of path atoms in cavity = 0.750000.
Ad infinitum constraint set at 35000.000000 iterations.

Progress monitored every 200 structures.

E&S ESV machines. This program is compatible with SYBYL [34] molecular modeling software with MOL2 files as input and output, but can easily be modified to accept other molecular coordinate formats. Routines have been written to automate the packing of structures from both the Cambridge Structural Database [1] as well as fragments from the Brookhaven Protein Databank [2]. FOUNDATION can be licensed from Washington University (contact the Center for Molecular Design for licensing information).

Creation of structural database

As an example of its use, all structures in the example database were derived from the Cambridge Crystallographic Database. Structures deposited since 1980 in classes 01–59 were extracted and valences filled with hydrogen atoms using the SYBYL molecular modeling software. This amounted to approximately 30 000 compounds. The required data for each structure described above was written sequentially to an ASCII file. By limiting structures to a maximum of 250

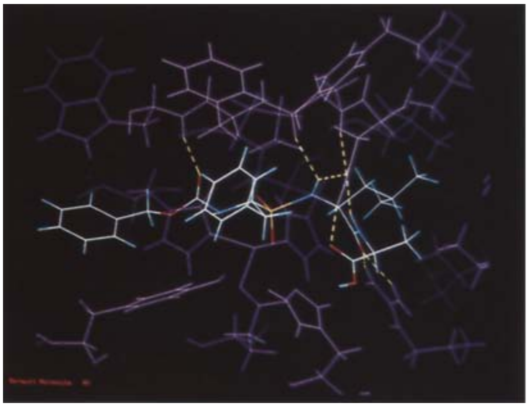


Fig. 7. Crystal structure of thermolysin with inhibitor bound within active site.

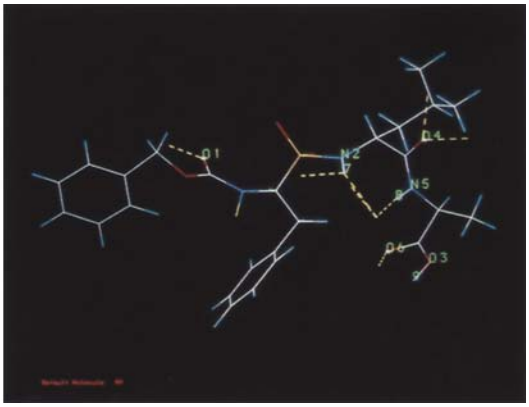


Fig. 8. Search query #1 for thermolysin inhibitor.

atoms, we were able to encode the data in byte form. As a result, the entire database required only 30 000 blocks of disk space.

Development of search queries

To demonstrate the FOUNDATION program, we developed two search queries using the X-ray structure of thermolysin shown in Fig. 7, co-crystallized with an inhibitor bound in the active site [35].

(a) Search query #1. In this query, we wished to retrieve structures linking any combination of the designated pharmacophoric groups. The inhibitor was extracted from the enzyme complex and all atoms forming inter-molecular hydrogen bonds were labeled as shown in Fig. 8. These nine atoms were isolated and transformed into the query listed in Table 1. Each atom was assigned an error margin of 0.15 Å. Bonds were defined between atoms 2 and 7, 5 and 8, and 3 and 9 to insure the retrieval of bonded atoms at these positions. The following atom type specifications were made: 7, 8, 9 → hydrogens (SYBYL atom id #13); 1, 4, and 6 → oxygens (SYBYL atom id #8–9); and 2, 3, and 5 → hydrogen bond donors (any nitrogen or oxygen). An rms-fit specification of 0.15 Å was defined. A file containing a filler-lattice delineating the extent of the enzyme-active site was used to ensure that all hits would fit within the active site cavity. This file, tln.fil, is shown visually in Fig. 9 as it resides within the thermolysin-active site. Recovered structures were required to have all atoms comprising the shortest atomic path between pharmacophoric loci within the active site. Finally, to prevent processing ad infinitum, a limit of 35 000 distance-checking iterations was imposed. Using this query, a search was conducted to retrieve structures containing a minimum of four matching atoms.

(b) Search query #1 – cyclic. To demonstrate the ability to retrieve structures that contain cyclic moieties in the regions connecting the query elements, we submitted a search query that was identical to that described above, with the added constraint that all structures contain at least one cyclic system linking the query elements.

(c) Search query #2. To further demonstrate the program's capabilities, a different search strategy was used to retrieve structures incorporating the same pharmacophoric sites as above. In

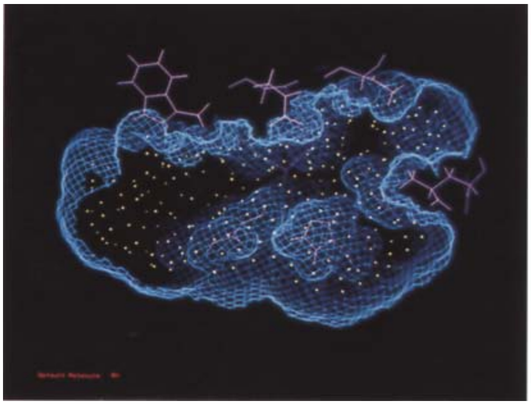


Fig. 9. Filler lattice used to delineate the ligand-accessible region of the thermolysin-active site.

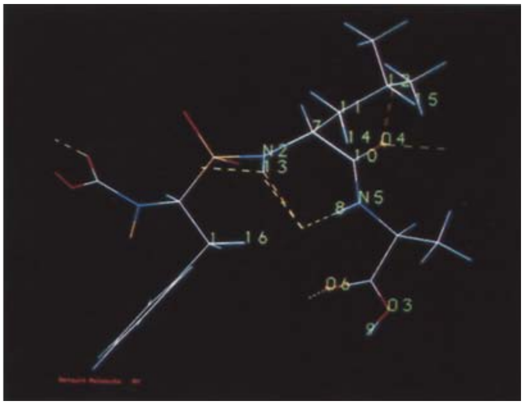


Fig. 10. Search query #2 for thermolysin inhibitor.

this query, we wished to preserve the integrity of the inhibitor 'backbone', using it to anchor the recovered segments. Accordingly, five more bonds, atoms 1–16, 2–7, 7–10, 11–14, and 12–15, were judged to be suitable sites to which segments could be attached. These were added to the previous query as shown in Fig. 10. Table 2 lists the resulting 17-atom query. Again, similar atom type designations for each pharmacophoric group were employed: atoms 8, 9, and 13 → hydrogen and atoms 2, 3, 4, 5, 6, and 17 → hydrogen bond donors. The same rms fit and cavity specifications were used as well. To insure that all recovered structures would bridge the backbone to the pharmacophoric groups, a subset was created that contained all of the attachment site atoms. A query restriction of one pair of atoms (1 bond) was implemented for this subset. Again, to prevent processing ad infinitum, a limit of 60 000 distance checking iterations was imposed. Using this query, a search was conducted to retrieve structures containing a minimum of four matching atoms.

RESULTS

Sixty structures were retrieved using search query #1, requiring approximately ten CPU hours of processing time to scan the entire database on an IRIS 4D/380. As shown in a listing of selected recovered structures (Table 3), numerous combinations of matching atoms were present. Probability dictates that as the number of matching atoms increases, the odds of retrieving such a structure decreases considerably. Accordingly, the vast majority of compounds were four-atom matches. In fact, only one five-atom match was retrieved, and no six-atom hits were present. Bonded atoms as well as appropriate atom types were recovered according to query specification. Furthermore, the atoms linking the pharmacophoric groups of every hit were contained within the thermolysin-active site. Figure 11 displays selected structures.

The cyclic variation of query #1 yielded a subset of the above-retrieved structures containing fifty hits. The high number of hits was surprising. Upon analysis, it was found that a majority of the cyclic crystal structures contained bound metals with numerous 'dummy bonds' to denote coordination. Such structures often turn up in cyclic queries since the coordinated metal provides

TABLE 2
SEARCH QUERY #2 FOR THERMOLYSIN INHIBITOR

Database searched:	**cho_dbase**
Threshold number of query atoms	**= 4**
Number of structures skipped	**= 0**
Hit limit	**= 1000**

Atoms

1	−1.53040	−1.98010	−2.66860	±0.150
2	−0.49240	−0.63410	−0.00560	±0.150
3	0.93460	−6.35910	−1.36360	±0.150
4	2.66670	−2.22830	0.73600	±0.150
5	0.66170	−3.22130	0.59000	±0.150
6	−0.52930	−4.83730	−1.28000	±0.150
7	0.70560	−0.80510	0.82840	±0.150
8	−0.33380	−3.08190	0.56660	±0.150
9	0.45420	−6.63630	−2.13770	±0.150
10	1.43660	−2.14710	0.66140	±0.150
11	0.40660	−0.66610	2.32640	±0.150
12	1.63760	−0.76210	3.19740	±0.150
13	−1.35660	−0.96270	0.39530	±0.150
14	−0.32200	−1.47200	2.52040	±0.150
15	2.23860	−1.65390	2.99440	±0.150
16	−1.48110	−2.45130	−1.67670	±0.150
17	−2.30530	2.10470	−3.45600	±0.150

Bonds	*Type:* Atom # 8⇒ 13
[1] query atoms: 1→16	*Type:* Atom # 9⇒ 13
[2] query atoms: 5→8	*Type:* Atom # 13⇒ 13
[3] query atoms: 3→9	*Type:* Atom # 2⇒ 5 6 7 11 28 19 8 9
[4] query atoms: 7→10	*Type:* Atom # 3⇒ 5 6 7 11 28 19 8 9
[5] query atoms: 7→12	*Type:* Atom # 4⇒ 5 6 7 11 28 19 8 9
[6] query atoms: 2→13	*Type:* Atom # 5⇒ 5 6 7 11 28 19 8 9
[7] query atoms: 12→15	*Type:* Atom # 6⇒ 5 6 7 11 28 19 8 9
[8] query atoms: 11→14	*Type:* Atom # 17⇒ 5 6 7 11 28 19 8 9

RMS-fit threshold set at 0.1500 Å.

Cavity file = tln. fil. Cavity inclusion dist. = 1.000000.
Specified fraction of path atoms in cavity = 0.750000.

Ad infinitum constraint set at 60000.000000 iterations.

Required = 2
1 7 10 11 12 14 15 16

Progress monitored every 200 structures.

multiple atomic pathways to link the pharmacophoric elements. The use of a constraint to exclude dummy atoms from the linking regions eliminated all but fourteen structures. Selected structures are also listed in Table 3 and displayed in Fig. 11.

A partial listing of the results of query #2 are likewise listed in Table 4. Twenty-two structures were found, requiring approximately fifteen CPU hours of processing time to scan the entire

TABLE 3
PARTIAL LISTING OF STRUCTURAL MATCHES – QUERY #1

Refcode	Structure #	# Match	[Query atom] Corresponding structure atom				
JAMFEI	2138	4	[1] 67	[2] 27	[6] 25	[7] 78	
BITZAF[a]	2545	4	[2] 10	[3] 12	[7] 13	[9] 15	
BIVSEE	2547	5	[2] 15	[5] 14	[6] 6	[7] 53	[8] 52
CINVUQ[a]	2724	4	[1] 12	[5] 61	[6] 60	[8] 102	
BENXEX[a]	7079	4	[3] 3	[4] 11	[6] 8	[9] 17	
GEYHOH[a]	23945	4	[1] 63	[4] 88	[5] 13	[8] 108	
CGDLLL10[a]	28958	4	[2] 61	[4] 68	[6] 72	[7] 137	
POCPAG[a]	29022	4	[4] 3	[5] 8	[6] 4	[8] 87	
GERCIP[a]	29602	4	[2] 5	[5] 13	[7] 31	[8] 42	
Selected cyclic structures retrieved							
BENXEX[a] (above)							
SADCOP[a]	23958	4	[2] 11	[3] 16	[7] 49	[9] 51	
COSGUM[a]	29252	4	[2] 28	[3] 30	[7] 55	[9] 59	
VEGROO[a]	29733	4	[2] 17	[4] 23	[6] 45	[7] 178	

[a] Denotes that this structure is displayed in Fig. 11.

database on an IRIS 4D/380. Again, numerous combinations of pharmacophore elements were found. However, in accordance with the subset query specification, one backbone-to-segment linking bond was present in every hit. Although a greater number of atoms were present in this query, a satisfactory search time was achieved with the additional subset constraints. Selected hits are shown in Fig. 12.

DISCUSSION

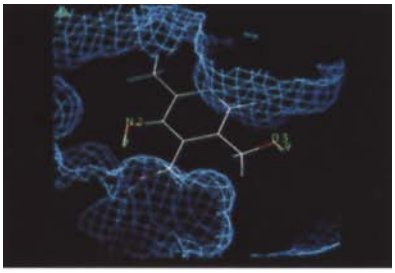
We have demonstrated that valid combinations of query elements can be retrieved using graph/clique-searching methods. The number of combinations varies according to the size of the query, extent of the constraints, and the number of matching elements requested. For the nine-

TABLE 4
PARTIAL LISTING OF STRUCTURAL MATCHES – QUERY #2

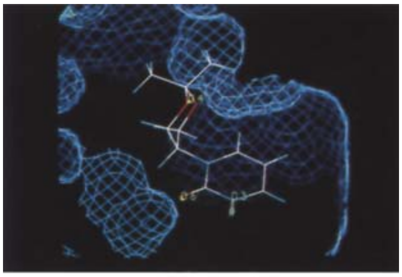
Refcode	Structure #	# Match	[Query atom] Corresponding structure atom			
COGREV[a]	1681	4	[2] 3	[4] 32	[7] 16	[10] 30
SEGZAF[a]	3507	4	[4] 17	[6] 9	[7] 14	[10] 15
GATVUS[a]	4726	4	[2] 13	[7] 10	[10] 6	[13] 21
DERREX[a]	6432	4	[4] 3	[6] 15	[11] 28	[14] 13
FUWVIC	9236	4	[4] 12	[11] 33	[14] 9	[17] 27
GEMZUT[a]	9351	4	[2] 1	[6] 30	[7] 2	[10] 3
BORDOB[a]	25277	4	[4] 6	[6] 13	[12] 31	[15] 9
BIKWOH10	25400	4	[4] 24	[12] 65	[15] 14	[17] 21

[a] Denotes that this structure is displayed in Fig. 12.

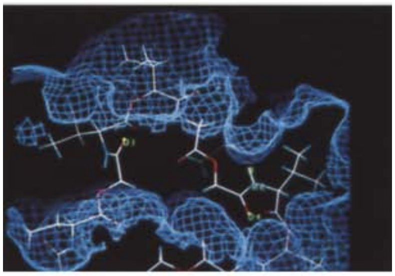
BITZAF



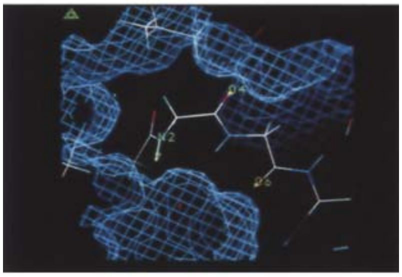
BENXEX



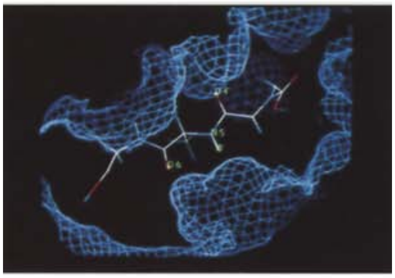
GEYHOH



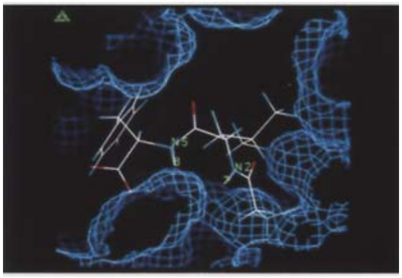
CGDLLLLO



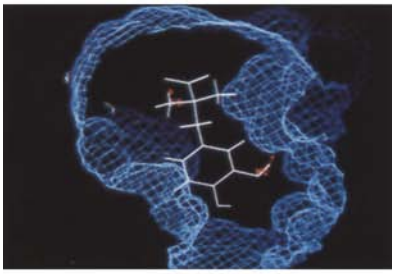
POCPAG



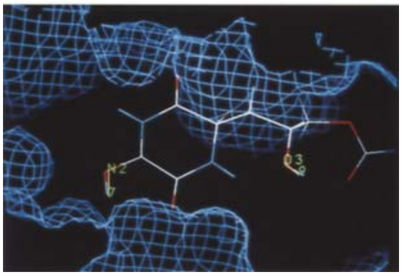
GERCIP



COSGUM



SADCOP



VEGROO

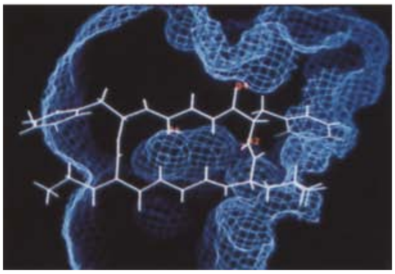
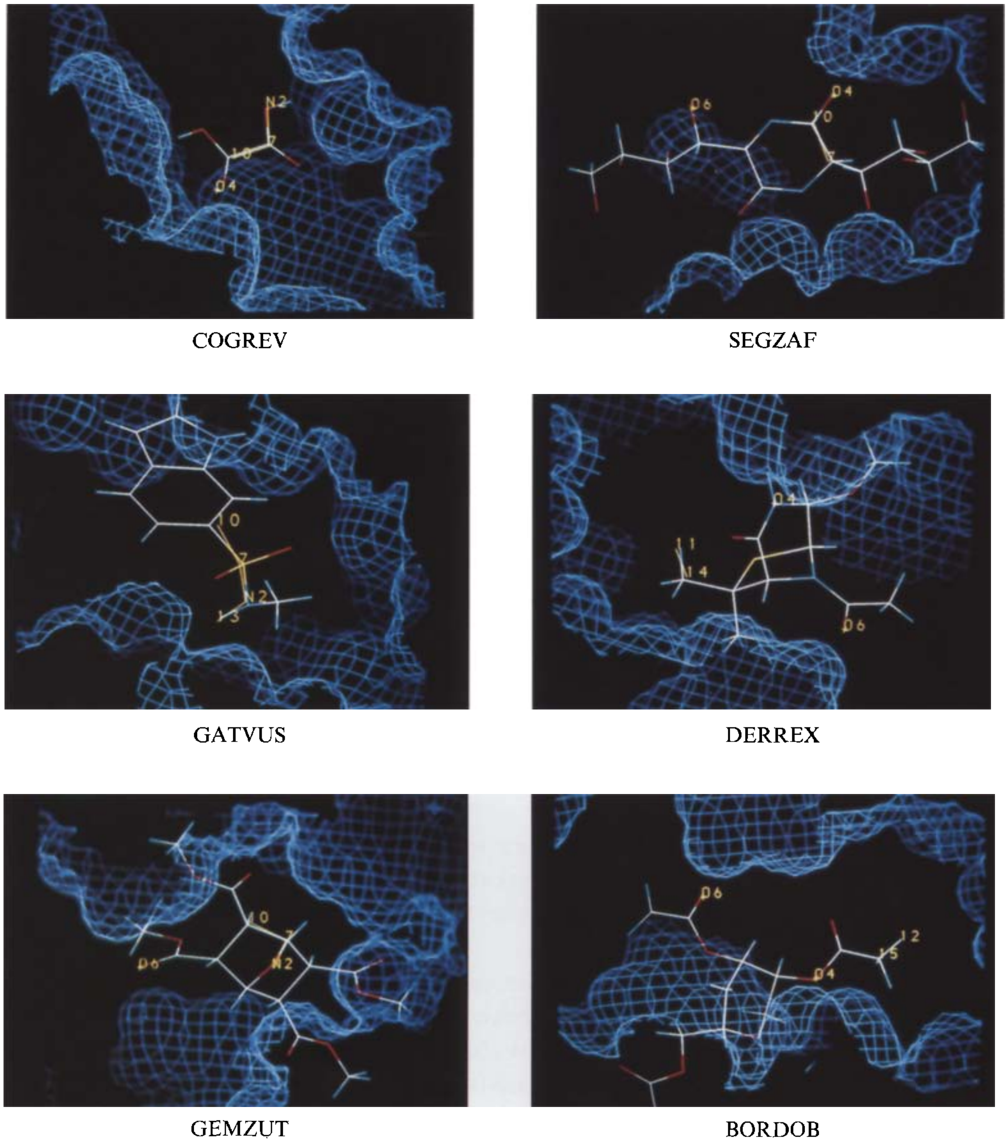


Fig. 11. Display of selected hits recovered using query #1.



COGREV

SEGZAF

GATVUS

DERREX

GEMZUT

BORDOB

Fig. 12. Display of selected hits recovered using query #2.

and sixteen-atom queries, far more than eight or nine different four-atom assortments would have been expected. However, bonded atom constraints allow only select groupings. Furthermore, our error margin of 0.15 Å is small and could easily have been doubled. This would increase both the number of matching elements as well as the number of returned hits.

The ability of FOUNDATION to retrieve partial hits is of great importance. As queries become more complex, the probability that a single compound will match all elements diminishes considerably. As such, a great many structures with slight imperfections would have been overlooked. Our program can retrieve and designate these compounds, allowing the user to edit the offending region(s).

The impetus to develop FOUNDATION coincides with an emerging trend in computer-aided molecular design. In recent publications, researchers have engineered binding ligands via substructural assembly [36–39]. In essence, candidates for various component parts of a ligand are sought. These candidates are selected on the basis of complementarity to local regions of the receptor-active site. They are then assembled through various means to produce novel structures. The combinatorial nature of this method allows for the generation of many compounds.

Bohm, in his program LUDI, uses this strategy in a three-step method to produce novel binding ligands [36]. Interaction sites are located within an active site by designating both hydrogen bonding and hydrophobic regions. Molecular fragments are then fitted within these interaction sites. Finally, bridge fragments are used to link subsites together.

Moon and Howe, in their program GROW, systematically build up peptide ligands within the binding site of a known 3D structure by piecing together low-energy molecular fragments [37]. At each iterative stage, energy evaluations determine the best candidates and their optimal location to be spliced onto the growing structure.

Lewis et al. [38], in their program BUILDER, use a design approach that focuses on chemical fragments that interact with key receptor groups. Database searching techniques [22] and structure generation algorithms [3] are used to retrieve candidate structures that complement the key regions of a binding site. A molecular lattice is then created which allows the program to determine feasible bonding patterns between the molecular fragments. An interactive graphics modeling environment is then used to allow the user to visualize and determine potential bridging configurations.

Verlinde et al. describe a protein structure-based linked-fragment approach for receptor-inhibitor design [39]. Again, molecular fragments are designed to bind within various subregions of a receptor. Bridging segments are then used to link these fragments into potential lead compounds. In their work, they use this method to design selective inhibitors of triosephosphate isomerase from *Trypanosoma brucei.*

FOUNDATION is ideally suited to complement these methodologies. Given a receptor with numerous interaction sites, our program can retrieve the component substructures necessary for recombination by any of the aforementioned techniques. In future work, attention must be focused upon the linking procedures as they are the most user-intensive. Chemical heuristics or expert systems may provide a ‘first-pass’ screening, eliminating the vast majority of inappropriate connections prior to human intervention.

In this regard, the rate-determining step in any molecular design process is clearly the user-analysis of the hits [12]. Although our program is slow by current standards, the additional information that it retrieves more than accounts for this shortcoming. To afford the investigator the most efficient usage of time, FOUNDATION searches independently as a background process, continually updating status files and informing the user of new structures. In this manner, initial hits can be evaluated to determine if appropriate compounds are being retrieved with current constraints.

The drawbacks to using 3D database searching programs as molecular tools have been discussed repeatedly by previous investigators [4,12]. Clearly, the most obvious weakness is that a single static structure is used to represent even the most flexible compounds. Thus, the induced fit of the ligand to the receptor site is not accommodated. The procedure outlined by Smellie et al. [28] offers the use of a distance range matrix as a representation of the conformational flexibility inherent in most molecules. Inclusion of such flexibility in FOUNDATION in its current form would dramatically increase the computational complexity. The use of the active site cavity as a constraint implies an additional assumption, that of a static cavity. In the case of thermolysin, where numerous crystal structures of different inhibitors bound to the active site are available, such an assumption may be tenable as the active site is remarkably stable despite ligand modification. Furthermore, the accuracy of any retrieval system is governed by the quality of the structures present within the database itself. As crystal data may contain misassigned atom types, improper bond designations, or other errors, frustration can result when initial query solutions are realized to be invalid structures.

It is hoped that through the use of crystallographic, or heuristically derived, structural retrieval systems, an appropriately constructed ligand may exhibit some semblance of its true, bound conformation. Clearly, determination of relatively rigid fragments which fit different subsites of the active site pocket are feasible with FOUNDATION. Additional iterations of the program to find appropriate scaffolds to link the fragments together should generate numerous novel structures for synthetic evaluation. Considering the difficulty in developing a ligand which possesses the desired pharmacophoric elements in a rigid framework and whose synthesis is chemically straightforward, the numerous candidate structures derived from known compounds can provide a stimulus to the natural creativity of the medicinal chemist.

CONCLUSIONS

We have developed a 3D, structural-database search system capable of retrieving all possible structures that contain any combination of a user-specified minimum number of matching atoms. Query specifications can contain any number of atoms or bonds provided that adequate constraints are given to reduce the computational load. These constraints include:

(1) Bonded vs. isolated atom distinction;
(2) Atom type designation;
(3) Definition of subsets with occupancy specification (>, =, < X atoms);
(4) RMS fit;
(5) Active site volume accessibility of atoms linking query elements;
(6) Number, atom type, and cyclic structure constraints for atoms linking pharmacophoric elements;
(7) Automatic error boundary adjustment – ad infinitum constraint.

While it is highly likely that the efficiency of this program can be improved, the functionality which it embodies would appear to be the minimum required as an efficient tool for molecular design using 3D databases, when a set of 3D requirements can be defined.

ACKNOWLEDGEMENTS

This work was supported by the National Institutes of Health (grant GM24483) and the Medical Scientist Training Program in the Division of Biology and Biomedical Sciences at Washington University (training grant GM07200).

REFERENCES

1 Allen, F.H., Kennard, O. and Taylor, R., Acc. Chem. Res., 16 (1983) 146.

2 Abola, E.E., Bernstein, F.C. and Koetzle, T.F., The Protein Data Bank, In Glaeser, P.S. (Ed.) The Role of Data in Scientific Progress, Elsevier, New York, 1985.

3 Rusinko, III, A., Skell, J.M., Balducci, R. and Pearlman, R.S., University of Texas at Austin, distributed by Tripos Associates Inc., St. Louis, MO, U.S.A.

4 Sheridan, R.P., Rusinko, III, A., Nilakantan, R. and Venkataraghavan, R., Proc. Natl. Acad. Sci. USA, 86 (1989) 8165.

5 Gund, P., Progr. Mol. Subcell. Biol., 5 (1977) 117.

6 Gund, P., Ann. Rep. Med. Chem., 14 (1979) 299.

7 Andrews, P.R., Lloyd, E.J., Martin, J.L. and Munro, S.L.A., J. Mol. Graphics, 4 (1986) 41.

8 Lesk, A.M., Commun. ACM., 22 (1979) 219.

9 Jakes, S.E. and Willett, P., J. Mol. Graphics, 4 (1986) 12.

10 Jakes, S.E., Watts, N., Willett, P., Bawden, D. and Fisher, J.D., J. Mol. Graphics, 5 (1987) 41.

11 Bartlett, P.A., Shea, G.T., Telfer, S.J. and Waterman, S., In Roberts, S.M. (Ed.) Molecular Recognition: Chemical and Biological Problems, Royal Society of London, 1989, pp. 182–196.

12 Van Drie, J.H., Weininger, D. and Martin, Y.C., J. Comput.-Aided Mol. Design, 3 (1989) 225.

13 Molecular Design Limited, San Leandro, CA., U.S.A.

14 Chemical Design Ltd., Oxford OX2 OJB, U.K.

15 Martin, Y.C., Bures, M.G. and Willett, P., In Lipkowitz, K.B. and Boyd, D.B. (Eds.) Reviews in Computational Chemistry, VCH Publishers New York, 1990, pp. 213–256.

16 Martin, Y.C., J. Med. Chem., 35 (1992) 2145.

17 Gibbons, A., Algorithmic Graph Theory, Cambridge University Press, Cambridge, 1988.

18 Bierstone, E., Cliques and generalized cliques in a finite linear graph, Unpublished report, University of Toronto.

19 Bonner, R.E., IBM J. Res. Develop., 8 (Jan. 1964) 22.

20 Gerhards, L. and Lindenberg, W., Computing, 27 (1981) 349.

21 Bron, C. and Kerbosch, J., Commun. ACM, 16 (1973) 575.

22 Kuntz, I.D., Blaney, J.M., Oatley, S.J., Langridge, R. and Ferrin, T.E., J. Mol. Biol., 161 (1982) 269.

23 DesJarlais, R.L., Sheridan, R.P., Seibel, G.L., Dixon, J.S., Kuntz, I.D. and Venkataraghavan, R., J. Med. Chem., 31 (1988) 722.

24 DesJarlais, R.L., Seibel, G.L., Kuntz, I.D., Montellano, P.O.D., Furth, P.S., Alvarez, J.C., DeCamp, D.L., Babe, L.M. and Craik, C.S., Proc. Natl. Acad. Sci., 87 (1990) 6644.

25 Crandell, C.W. and Smith, D.H., J. Chem. Inf. Comput. Sci., 23 (1983) 186.

26 Brint, A.T. and Willett, P., J. Mol. Graphics, 5 (1987) 49.

27 Kuhl, F.S., Crippen, G.M. and Friesen, D.K., J. Comput. Chem., 5 (1984) 24.

28 Smellie, A.S., Crippen, G.M. and Richards, W.G., J. Chem. Inf. Sci., 31 (1991) 386.

29 Cormen, T.H., Leiserson, C.E. and Rivest, R.L., Introduction to Algorithms, McGraw-Hill, St. Louis, MO, 1991.

30 Nyburg, S.C., Acta Crystallogr., B30 (1974) 251.

31 Ho, C.M.W. and Marshall, G.R., J. Comput.-Aided Mol. Design, 4 (1990) 337.

32 Foley, J.D. and Van Dam, A., Fundamentals of Interactive Computer Graphics, Addison-Wesley, Reading, MA, 1982.

33 Sheridan, R.P. and Venkataraghavan, R., J. Comput.-Aided Mol. Design, 1 (1987) 243.

34 Tripos Associates Inc., St. Louis, MO., U.S.A.

35 Holden, H.M., Tronrud, D.E., Monzingo, A.F., Weaver, I.H. and Matthews, B.W., Biochemistry, 26 (1987) 8542.

36 Bohm, H., J. Comput.-Aided Mol. Design, 6 (1992) 61.

37 Moon, J. and Howe, W.J., Proteins: Struct. Funct. Genet., 6 (1991) 314.

38 Lewis, R.A., Roe, D.C., Huang, C., Ferrin, T.E., Langridge, R. and Kuntz, I.D., J. Mol. Graphics, 10 (1992) 66.

39 Verlinde, C.L.M.J., Rudenko, G. and Hol, W.G.J., J. Comput.-Aided Mol. Design, 6 (1992) 131.