Pharmacophoric Pattern Matching in Files of Three-Dimensional Chemical Structures: Comparison of Conformational-Searching Algorithms for Flexible Searching

David E. Clark, Gareth Jones, and Peter Willett*

Department of Information Studies and Krebs Institute for Biomolecular Research, University of Sheffield, Western Bank, Sheffield S10 2TN, U.K.

Peter W. Kenny

Zeneca Pharmaceuticals, Mereside, Alderley Park, Macclesfield, Cheshire SK10 4TK, U.K.

Robert C. Glen

Wellcome Research Laboratories, Langley Park, Beckenham, Kent BR3 3BS, U.K.

Received May 7, 1993®

The conformational space of a flexible three-dimensional (3-D) molecule can be represented for searching purposes by a smoothed bounded-distance matrix. Such matrices provide an effective way of carrying out flexible searching, but search times can be much greater than with comparable rigid searches that take no account of conformational flexibility. The most time-consuming part of a flexible search is the final conformational search, and in this paper we compare the efficiency and the effectiveness of the distance geometry, systematic search, random search, genetic-algorithm, and directed-tweak methods for conformational searching. Experiments with sets of 1538 and 9886 flexible 3-D structures suggest that the most effective of these are the genetic-algorithm and directed-tweak methods, both of which are efficient enough to enable the searching of databases containing nontrivial numbers of flexible 3-D structures.

1. INTRODUCTION

Pharmacophoric-pattern searches of three-dimensional (3-D) databases are now routinely used in the search for novel active compounds.[1,2] Early 3-D searching systems represented structures by single conformations, but the use of such a *rigid search* is likely to fail to identify flexible structures that are able to adopt a conformation that contains the query pharmacophore. That this occurs in practice has been clearly demonstrated in recent work by Haraki *et al.*[3]

Various approaches have been suggested to alleviate this problem.[4-6] However, a general solution to the problem of conformational flexibility requires an explicit representation of the full range of conformations that a flexible molecule can assume, together with appropriate search algorithms for the chosen representation. We have recently proposed[7] that flexible molecules can be represented and searched using graph-theoretic methods analogous to those that are used in two-dimensional (2-D) and rigid 3-D substructure searching systems. For a flexible 3-D molecule, we suggest that each edge in a molecule should contain two real values, these being the lower and upper bounds to the interatomic distance that are obtained when a triangle-inequality bounds-smoothing procedure is carried out upon the molecule.[8] This bounded distance matrix representation of a conformationally-flexible molecule can be searched using a three-stage *flexible-searching* algorithm. Here, the *screening* and *geometric-searching* stages familiar from current, rigid-searching systems are used to define a (hopefully small) subset of a 3-D database that is then processed by a *conformational-searching* algorithm, which carries out a detailed examination of the hits resulting from the first two stages of the flexible search. Our initial experiments demonstrated the effectiveness of this three-stage algorithm for flexible searching,[7] and we have recently discussed ways in which the efficiency of searching might be increased.[9]

The conformational-searching algorithm that is used plays a central role in determining the efficiency (i.e., the computational requirements) and the effectiveness (i.e., the final number of hits) of a flexible search, since it is this stage that seeks specific molecular conformations which satisfy the geometric constraints of the pharmacophore model and which are of relatively low energy. Much effort has been invested in recent years in developing efficient algorithms for searching the conformational space of small- to medium-sized molecules,[10-12] and in this paper we present our experiences with five of them when they are used for 3-D searching. Specifically, we have implemented versions of the *distance-geometry*, *systematic-search*, *random-search*, *genetic*, and *directed-tweak* algorithms and then used them to search the outputs from the screening and geometric-searching stages of the flexible-searching system that we have described previously.[7,9]

2. IMPLEMENTATION OF CONFORMATIONAL SEARCH METHODS

2.1. Distance Geometry. The first set of experiments were conducted using the EMBED routine embodied in the well-known DGEOM program.[13] In this process, a set of random distances is selected from within the distance bounds of the structure in question, and the algorithm then seeks to embed these points from the high-dimensional distance space into 3-D Cartesian space. In general, the fit to the data which the EMBED algorithm produces is not perfectly accurate, so a subsequent numerical optimization procedure is employed to minimize an error function, the magnitude of which measures the deviations of the coordinates from the input distance bounds and chirality constraints.[14] The maximum deviations to be permitted in any generated conformation may be specified by the user *via* the MAXDIST and MAXCHI options for distance and chiral errors, respectively. An embedding is not always

[*] To whom all correspondence should be addressed.

[®] Abstract published in *Advance ACS Abstracts,* January 15, 1994.

0095-2338/94/1634-0197$04.50/0

possible at the first attempt because the points in distance space may not correspond to any realistic configuration in 3-D space. DGEOM accordingly allows the user to choose how many trials at embedding are to be made, using the NTRIALS option. The number of conformations to be generated for each structure is specified by the NSTRUCT option. The use of the CONSTRAINTS option enables constraints to be placed on distances in the conformations that are to be generated, thus ensuring that the geometric constraints identified in the subgraph isomorphisms that are returned by the geometric search are satisfied in the final hit conformation.

The DGEOM program was used to seek to produce a viable conformation from each of the structures that matched the query pattern in the geometric search based on smoothed distance matrices. In order to accomplish this, the output from the geometric-searching program was adapted for compatibility with the CONSTRAINTS option. The results that are detailed below were obtained using the default DGEOM setting of 0.5 Å^3 for MAXCHI, and a range of values for MAXDIST. Ten attempts were made to embed each structure under the overlap constraints (i.e., NTRIALS = 10), and only a single conformation was generated for each structure (i.e., NSTRUCT = 1).

2.2. Systematic Search. The second set of experiments used the deterministic, systematic-search algorithm that is embodied in the SYBYL molecular modeling package.[15] This CSEARCH program uses the constrained-search approach developed by Dammkohler *et al.*,[16] which provides an efficient mechanism for the imposition of distance constraints between pairs of atoms in the structure that is being analyzed, and also for the calculation of the energies of conformations as they are produced.

The output from the geometric searching program was modified so that it could be loaded into the SEARCH functionality of the SYBYL package. The parameters of a systematic search are specified by the SEARCH SETUP commands, and the conditions utilized in our experiments are detailed in Appendix 1. Once a search had been specified in this way, it was initiated by the DO_SEARCH command. Two measures were adopted in order to minimize the time requirements of the systematic searches. Firstly, searches were terminated as soon as the first hit conformation was discovered for a given structure. This was achieved by the command TAILOR SET SEARCH CONFORMATION_-LIMIT 1. Secondly, an arbitrary time limit of 4 CPU min was imposed for the processing of an individual molecule, i.e., of each individual hit from the geometric search. This figure was considered to be well in excess of the time that would be permitted in any praticable database searching system.

We have used two definitions of what constitutes the set of rotatable bonds in a molecule. In the *full definition*, all potentially-rotatable bonds within a structure were defined as rotatable. In the *reduced definition*, only those potentially-rotatable bonds lying between the pharmacophore atoms in a molecule were defined as rotatable. This definition considers only those bond rotations that affect the relative spatial positions of the pharmacophore atoms. In general, this definition results in a substantial reduction in the number of rotatable bonds specified for a given structure (and thus in the computational requirements); however, it can also result in structures with unfavorable van der Waals' interactions (unless a "clean-up" routine is used after a potential match has been identified). It should be noted that, in both definitions, all rings were defined as being conformationally inactive. While this is unrealistic for non-aromatic systems, it does reduce the conformational search problem to a more managable size.

2.3. Chang *et al.* have described a torsional Monte Carlo methodology for conformational searching and have demonstrated its efficacy in finding low-energy conformations of small- and medium-sized molecules.[17,18] A simplified version of this algorithm for a single structure is as follows:

(1) Calculate the initial energy.

(2) Open the rings and set the ring closure constraints.

(3) Select the torsion angles that are to be varied.

(4) Randomize the selected torsions.

(5) Check to see if the geometric constraints are satisfied. If not, then go to step 3.

(6) Close the rings.

(7) Perform constrained minimization.

(8) If the energy of final conformation lies within the user-specified window, then the structure is a hit. If not, then go to step 2, unless the maximum number of iterations has been reached, in which case the structure is not a hit.

This algorithm has two user-defined parameters. The first parameter is the number of iterations per structure, which controls how many attempts the algorithm makes to fit the structure to the geometric and energetic constraints. The second parameter is the energy window, which determines the maximum energy a conformation may be above the reference conformation stored in the database if the former is to be a hit.

As with the CSEARCH experiments described above, the geometric-searching program was adapted to produce a SYBYL Programming Language (SPL) script, which could then be loaded into SYBYL and used to invoke the random-searching program (which was written in SPL) with the correct geometric constraints for each structure.

2.4. Genetic Search. Genetic algorithms are a class of nondeterministic algorithms that provide good, though not necessarily optimal, solutions to combinatorial optimization problems at a low computational cost,[19,20] and there have been several recent reports of the use of GAs for conformational searching.[21-23] These studies suggest that GAs may be well-suited for this particular application area, and we have thus investigated their effectiveness and efficiency when used for the conformational search stage of a flexible search. The use of a genetic algorithm for measuring 2-D structural similarities has been reported recently.[24]

The genetic algorithm tackles the problem of conformational analysis by encoding each of the torsion angles corresponding to a rotatable bond in a molecule by a string of 8 bits. This corresponds to an angle increment of approximately 1.4° (i.e., 360°/256). Each conformation of the molecule is then represented by the concatenation of these strings to form a bit string of length 8*N*, where *N* is the number of rotatable bonds in the molecule. The algorithm begins by randomly generating an initial population of size POP_SIZE of these bit strings. Each of the strings is then evaluated using a *penalty function* (as defined below), and new bit strings are bred to replace the members of the population that have the largest associated penalty functions using a steady-state approach.[19] Since the probability of any given string being selected for breeding is derived from its penalty function (an approach that is commonly called *roulette-wheel selection*), it is likely that the average penalty function of the population will decrease over a number of generations until a solution, or the closest point to one, is found. The process of breeding usually takes place by means of the *crossover* operator. This involves two *parents* swopping a substring of randomly-chosen length to form two *children*, which have characteristics of both of the parents. The particular crossover mechanism used here, called one-point crossover, involves substrings that start or finish at one of the ends of each of the bit strings in the population. The other genetic operator used here is *mutation*, which causes at least one randomly-selected bit in a parent or child string to be switched to its opposite value. Mutation causes diversity to be maintained in the population, and this prevents the genetic algorithm becoming stuck in local minima which are not solutions. The basic genetic algorithm is summarized below (a more detailed account of its implementation for 3-D database searching is provided by Jones[25]):

(1) A set of reproduction operators (crossover and mutation in the work reported here) is chosen. Each operator is assiged a weight.

(2) An initial population is created randomly, and the penalty functions for its members determined.

(3) An operator is chosen using roulette-wheel selection based on the operator weights defined in step 1.

(4) The parents required by the operator (two for crossover, and one for mutation) are chosen using roulette-wheel selection based on the ranked values of their penalty functions.

(5) The chosen operator is applied and the child chromosomes produced. Their penalty functions are evaluated.

(6) The children with the smallest penalty functions replace the members of the population that have the largest associated penalty functions.

(7) If none of the termination conditions have been fulfilled, then go to step 3.

The penalty function that is used to evaluate each of the bit strings (and thus measure the extent to which each of the conformations is compatible with the input distance constraints) comprises two parts, as discussed below.

The first contribution to the penalty function is the *pharmacophoric-pattern match penalty*. This is calculated by comparing the distance between the pharmacophore atoms in the conformation coded for by each of the bit strings with the distance bounds specified by the input distance constraints. If the distance in the conformation lies within the required bounds, then no penalty is assigned. If, however, the distance is outside the bounds, then the modulus of the difference between the closer bound and the measured distance (Å) is calculated to yield the penalty value for that constraint. The total pharmacophoric-pattern match penalty is obtained by summing these deviations over all the input constraints. In some instances where the distance constraints are extremely tight, the inherent granularity in the possible torsion angles generated by the genetic algorithm can cause possible solutions to be "stepped over". A further parameter, TOLERANCE, was used to overcome this problem. The effect of this parameter is to relax each of the upper and lower bounds of the input constraints by TOLERANCE Å.

The second contribution to the penalty function is a *van der Waals' (vdW) energy penalty*. In seeking "hit" conformations, it is usually desirable that the solutions not only match the input distance constraints but are also of a reasonable energy. The addition of a vdW energy calculation enables conformations with bad steric clashes to be penalized, thus giving preference to conformations of low steric energy. The vdW energy for a given conformation was calculated using the 6–12 potential that is implemented in the General Purpose TRIPOS 5.2 Forcefield.[26] It should be noted that the vdW calculation forms a large part of the overall computational costs of the genetic algorithm and substantial increases in speed would be expected if a simple potential was used. Indeed, a simple overlap of vdW radii would suffice to produce reasonable conformations.

To calculate the vdW energy penalty, the genetic algorithm determines the energy difference, E_{Diff}, between the conformation in question and the reference conformation stored in the database. If the value of E_{Diff} exceeds that of VDW_ENERGY_WINDOW, a user-specified parameter, then the conformation is penalized by an amount given by (E_{Diff} − VDW_ENERGY_WINDOW) × VDW_ENERGY_WT, where VDW_ENERGY_WT is a user-specified parameter determining the weighting of the vdW energy penalty term relative to the pharmacophoric-pattern match penalty term. Thus, if VDW_ENERGY_WT is set to zero, then the energy term will never contribute to the penalty function and the conformation will be judged purely on the match to the distance constraints.

The genetic algorithm terminates when a solution is found, i.e., when the penalty function becomes zero; when the maximum number of operations (as specified by the parameter MAXOPS) is reached; or when the gradient of the curve of penalty function versus number of operations becomes equal to a user-specified value, i.e., the genetic algorithm will terminate if the penalty function does not decrease by an amount PENALTY_INC in OPS_INC operations.

The genetic algorithm was encoded in C and run using both the full and the reduced definitions of rotatable bonds, as with the systematic-search experiments.

2.5. Directed-Tweak Search. The algorithms we have discussed thus far were not originally developed specifically for flexible searching; conversely, the directed-tweak algorithm has been developed by Tripos Associates for the flexible-searching component of their SYBYL/3DB UNITY database package. In concept, this algorithm is similar to the *random-tweak* technique of Shenkin *et al.*[27,28] However, whereas random tweak was developed for use in modeling loops in macromolecular structures, directed tweak is a minimization procedure that is used for rapid, conformational searching of small-to-medium-sized molecules. The algorithm is described in detail by Hurst,[29] and we thus present here just a brief summary of the algorithm's main features.

For the purposes of 3-D database searching, directed tweak uses a pseudoenergy function which involves the sum of the squares of the deviations of the atom–atom distances in the structure from the input distance constraints as specified in the query pharmacophore. The algorithm proceeds toward a solution by seeking to minimize this function by altering the torsion angles of the structure. The derivatives of the pseudoenergy with respect to the torsion angles are simple and analytically-determined which makes the technique very fast in operation. By using these derivatives in any standard minimization procedure, it is possible to discover the set of torsion angles that corresponds to the lowest pseudoenergy. A hit is obtained, i.e., a molecule is identified as possessing the sought query pharmacophore, if the final pseudoenergy is within some user-defined limit above zero.

Minimization techniques are often beset by the problem of local minima on the potential energy hypersurface. However, directed tweak largely circumvents this difficulty by ignoring van der Waals' interactions until a satisfactory geometry is attained.[29] In this way, the terms in the pseudoenergy function are concerned only with the torsion angles and this results in a well-behaved energy surface with few local minima. Once a hit geometry has been found, the vdW interaction terms can be incorporated into the pseudoenergy function, which can

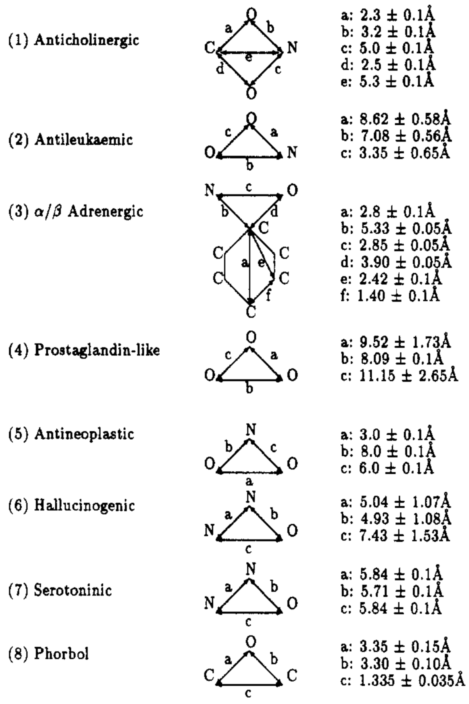

Figure 1. Query pharmacophoric patterns from the literature.

then be further minimized. In this manner, it is possible to produce a structure which not only satisfies the geometric constraints of the pharmacophore but which is also of a suitable energy.

3. EXPERIMENTAL DETAILS

Searches were run for each of the eight pharmacophoric patterns detailed in Figure 1. The search file for most of the experiments consisted of the set of 1538 bounded-distance representations of molecules from the POMONA 89 database that we have used previously[7,9,30] and that have been generated using Smellie's DG program.[31] A reviewer commented that the POMONA database contains many biologically-active molecules and that it might thus be possible to investigate the activities of the structures retrieved in response to each of the patterns in Figure 1. We have not done this to date, but it would be of interest to extend our studies to incorporate biological data in the future. The experiments reported in section 5 used a larger set of 9886 structures from the Chemical Abstracts Service database; here, the bounded-distance matrices were generated using the DIST_GEOMETRY functionality available in SYBYL version 6.0.[15]

A flexible molecule in one of these two data sets is represented by a bounded-distance matrix. This matrix is used for the assignment of the distance-range screens that are matched in the initial, screening stage of the flexible search and for the graph-matching operations involved in the subsequent geometric search, which is implemented using a modified version of Ullmann's subgraph isomorphism algorithm. A molecule undergoes the final conformational search if, and only if, it is a match for the query in both the screening search and the geometric search,[7] and thus the eights sets of matching structures (one for each of the query pharmacophores) from the geometric search formed the inputs to the various conformational-searching algorithms. Unless stated otherwise, all of the runs were carried out on an Evans and Sutherland ESV 3 Unix workstation, with programs written in C, Fortran, 77, or SPL. The numbers of hits in the screening and geometric searches for the eight query pharmacophores are listed in Table 1: in the following, the effectiveness of the various conformational-searching methods is described by the percentage of the hits in the geometric search for a query that were shown to contain the corresponding pharmacophore.

Table 1. Percentages of the Database That Were Hits in the Screen and Geometric Searches of the File of 1538 POMONA 89 Structures

query	screen search	geometric search
1	32.4	9.4
2	20.3	12.1
3	46.0	8.1
4	23.7	4.7
5	18.9	12.4
6	25.0	14.0
7	15.2	5.6
8	70.5	10.2
mean	31.5	9.6
median	24.4	9.8

Table 2. Percentage of Hits from the Geometric Search That Were Shown To Contain the Query Pharmacophore in the Distance-Geometry Embedding Experiments[a]

	percentage for given MAXDIST (Å)							
query	0.5		0.25		0.125		0.0625	
1	57.2	43.5	37.9	16.6	16.6	2.1	0.0	0.0
2	81.2	64.5	65.1	35.5	39.2	9.7	2.2	0.0
3	46.4	36.8	29.6	16.8	4.8	0.0	0.0	0.0
4	72.6	57.5	43.8	26.0	24.7	4.1	0.0	0.0
5	71.6	52.1	53.2	20.0	22.6	4.2	1.1	0.0
6	85.7	72.2	74.1	53.2	59.7	19.9	7.9	3.2
7	68.6	47.7	38.4	18.6	12.8	4.7	0.0	0.0
8	84.7	72.6	59.9	44.6	31.8	14.6	6.4	1.3
mean	71.0	55.9	50.3	28.9	26.5	7.4	2.2	0.8
median	72.1	54.8	48.5	23.0	23.7	4.5	0.6	0.0

[a] The first and second columns in each segment of the table are the percentage of hits after ten trials and after one trial, respectively.

Unless stated otherwise in what follows, only the first subgraph isomorphism detected in a structure during the geometric search was processed in the conformational search. We have shown previously that there may be some, or many, different subgraph isomorphisms for each structure,[7] and thus the numbers of hits listed in the tables of results below should be considered as underestimates of the true numbers of hits that would have been obtained if all of the isomorphisms had been processed. The investigation of all of the isomorphisms with all of the algorithms would have been totally infeasible because of the execution times that would have been required; as it was, the work that is summarized in this section extended over a period of more than a year.

4. RESULTS AND DISCUSSION

4.1. Distance Geometry. Table 2 shows the percentage of the hits from the geometric search for each of the eight queries that yielded a viable conformation to within the distance tolerance specified in the table. Each column displays the overall percentage of successful embeddings (i.e., in ≤10 trials), with the second figure representing the percentage that embedded successfully at the first attempt.

In considering these results, it is clear that the number of successful embeddings falls off as the permissible distance error (specified by the value of MAXDIST) is reduced. Indeed, with MAXDIST = 0.001 Å, no embeddings were achieved for any of the structures with any of the query patterns; i.e., the method was totally ineffectual if an exact match with the query was required. In principle, then, one can "tune" the number of hits obtained by altering the MAXDIST parameter, although it must always be remembered that the distance geometry algorithm does not adhere to a static bond-length and bond-angle formalism. It is thus often observed that in order to satisfy the distance constraints as nearly as possible, the distance-geometry algorithm may produce non-optimal bond lengths or angles or close Van der Waals' contacts that will, of course, result in high-energy structures.[32] This will be particularly true when the MAXDIST parameter is set to a high value, and it is observed that in excess of 70% of structures from the geometric search produce an embedding with this degree of tolerance. By reducing the permissible errors, fewer hits are obtained, but it is to be expected that the resulting structures will be of better quality and will be a better fit to the imposed constraints.

For the reasons outlined above, in most situations employing distance geometry for structure generation, a subsequent molecular mechanics minimization procedure is utilized to "clean up" the structures. This was not done in these experiments but would need to be implemented in an operational system using a rapid procedure, since a full minimization would be far too slow for use in a database-searching context.

It is clear from Table 2 that not every structure passed on by the geometric search actually yields a hit conformation. There are two conceivable reasons for the embedding process to fail. Firstly, the imposition of the overlap constraints upon a structure may lead to inconsistent distance bounds, which will be rejected by DGEOM. Alternatively, the structure may simply fail to embed satisfactorily after the 10 trials. In the latter case, it is possible that further attempts might produce a successful embedding, but this would obviously become prohibitive in terms of search time. For this reason, the enbedding figures in Table 2 should be considered as *minimum* values. Furthermore, it is possible that a structure which fails to embed under the "top-of-the-pile" isomorphism will embed under a subsequent one, if one exists. Finally, as might be anticipated, the more complex queries (i.e., queries 1 and 3) produce fewer successful embeddings than their simpler companions, which impose a smaller number of distance constraints upon the structure to be generated.

Detailed timing experiments were carried out for one of the queries (query 4) to investigate the time taken to produce an embedding with the varying values of MAXDIST. The fastest time to embed a single structure was about 10 s, while the slowest time for ten trials to be carried out unsuccessfully was as high as 347 s (these figures are CPU times on a VAX 8820 running the DGEOM program under the VMS operating system). Embedding can thus be extremely time-consuming, if not immediately successful. Even the fastest times are significantly slower than those obtained from other methods of conformational searching (as discussed further below), and we thus conclude that distance geometry is unsuitable as a means of executing a final conformational search in small- to medium-size molecules.

4.2. Systematic Search. The percentage of hits from the geometric search that can be demonstrated to contain the query pharmacophore in a systematic search will depend on the torsion increment that is used. Results are presented for increments of 10°, 20°, 30°, 40°, and 60° in Table 3 using both the full and the reduced definitions of flexibility.

In considering the figures in this table, it is important to recall that a maximum time of 240 CPU s was allowed for the systematic search of any individual structure. As the torsion increment is reduced, a larger and larger number of structures will not have been processed by the time that this limit is reached, with a consequent reduction in the number of hits that is achieved. At the same time, however, the decreased torsion angle allows a more detailed exploration of the conformational space that characterizes a particular molecule, with a consequent increase in the number of hits that is achieved. There are thus two conflicting factors at work here, with the maximum number of hits being obtained with an increment of 20°.

Table 3. Percentage of Hits from the Geometric Search That Were Shown To Contain the Query Pharmacophore in the Full-Definition and Reduced-Definition Systematic-Search Experiments

	percentage for given torsion increment									
	full definition					reduced definition				
query	60°	40°	30°	20°	10°	60°	40°	30°	20°	10°
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	23.7	31.2	33.3	33.3	30.1	21.0	34.4	36.0	38.2	36.6
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	16.4	15.1	19.2	15.1	9.6	15.1	16.4	21.9	17.8	13.7
5	3.2	10.5	13.7	21.6	21.6	2.1	10.5	15.3	25.3	26.8
6	22.7	29.2	31.0	30.1	30.1	25.5	30.1	32.9	35.7	33.3
7	7.0	12.8	16.3	17.4	17.4	5.7	17.4	19.8	24.4	22.1
8	0.6	4.5	8.9	11.5	12.7	0.6	4.5	8.3	10.8	11.5
mean	9.2	12.9	15.3	16.1	15.2	8.7	14.2	16.8	19.0	18.0
median	5.1	11.7	15.0	16.3	15.1	3.9	13.5	17.5	21.1	17.9

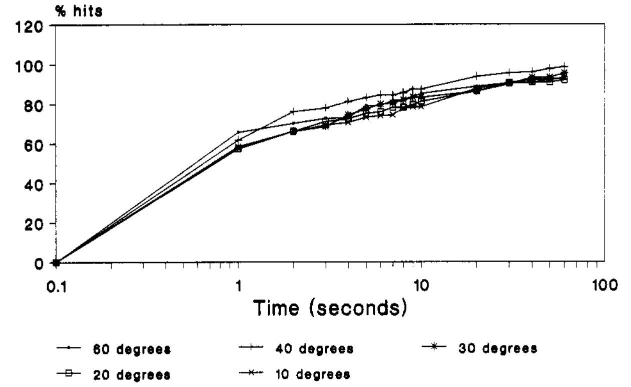

Figure 2. Effect of variations in the search time limit on the number of hits in the full-definition systematic-search experiments.

The hits from the full- and reduced-definition searches overlap but are not identical, since the two sets of experiments do not explore precisely the same conformational space. The full-definition searches explore a larger conformational space and are thus expected to identify a greater number of matches for the query pharmacophore than the reduced-definition searches. This is, in fact, what is observed with the 60° searches. When smaller torsion increments are used, the run times increase, with the result that an increasingly large fraction of the full-definition searches breach the 240-s limit; however, the reduced-definition searches are very much faster, since the number of conformations generated in a search is a power function of the number of rotatable bonds, and thus the reduced-definition searches yield the greater numbers of hits with the smaller torsion increments.

The number of hits that is obtained using a fixed torsion increment will depend on the time limit that is given to the systematic search procedure. The effect of variations in this limit are shown in Figures 2 and 3, which refer to the full and reduced definitions, respectively. Let N_{240} be the number of hits identified if the systematic search is run for the maximum amount of 240 s (on an ESV 3 workstation) and let N_t be the number of hits identified after t s. Then the figures illustrate the variation in 100 × (N_t/N_{240}) with t (so that all of the lines

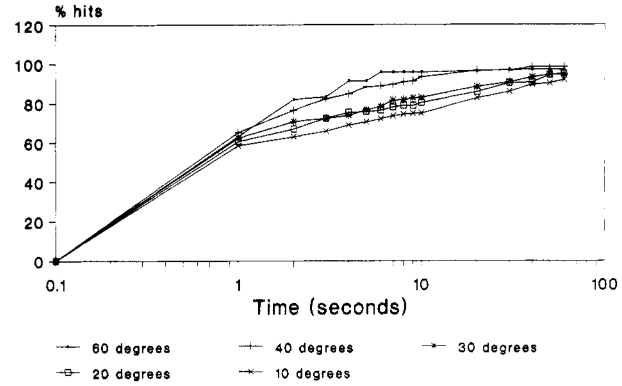

Figure 3. Effect of variations in the search time limit on the number of hits in the reduced-definition systematic-search experiments.

Table 4. Mean Percentage Reductions in the Number of Hits for Systematic Search Experiments That Were Allowed To Run for up to *t* s

t	full definition	reduced definition	*t*	full definition	reduced definition
0.5	34.0	43.8	10.0	87.1	85.4
1.0	61.8	61.8	30.0	91.3	92.1
5.0	77.6	80.5	60.0	94.3	95.3

Table 5. Percentage of Hits from the Geometric Search That Were Shown To Contain the Fourth Query Pharmacophore in the Random Searches

iterations	hits	iterations	hits	iterations	hits
100	2.7	1000	21.9	5000	21.9
500	11.0	2000	17.8	10000	31.5

tend toward 100% at 240 s). The mean results, when averaged over the five different angle increments, are summarized in Table 4. The results show that there is little to choose between the two definitions in terms of the rate of finding matches for a query pharmacophore. This suggests that the reason that the reduced-definition searches are observed to be faster overall than their fully-defined counterparts is that the searches which *fail* to yield a hit are dealt with more quickly in the former. A further deduction is that a search time limit of somewhere between 1 and 5 s could reasonably be imposed in an operational system. This would still guarantee that up to three-fourths of the possible hits (i.e., those that would be found if the algorithm was allowed to run for up to 240 CPU s) would be found, while eliminating much of the time wasted on searches that do not yield a hit or that take a long time to find one. This time limit could, of course, be placed at the user's discretion, giving a flexibility not possible with systems that rely on the preselection of representative conformations.

4.3. Random Search. An initial set of experiments was conducted using query 4 with a 50 kcal/mol energy window, the full definition of rotatable bonds, and various numbers of iterations. The results obtained are presented in Table 5.

It will be seen that, as expected, an increase in the number of iterations is generally accompanied by an increase in the percentage of hits; the anomalous 2000-iteration result may be explained by the random nature of the algorithm; i.e., if the experiments were to be repeated several times, it would be expected that the 2000-iteration run would produce more hits than the 1000-iteration run. It will be seen that the percentage of hits found with the largest number of iterations far exceeds the percentage obtained with a systematic search, even allowing for the fact that this approach allows us to include flexible rings (which was not done in the systematic-search experiments). We thus conclude that random search provides an effective means of carrying out the conformational search. Our SPL implementation was extremely slow, taking about 1 CPU s/iteration, which makes it totally infeasible for use in a database-searching context. However, the success (in terms of percentage hits) encouraged us to undertake the genetic algorithm experiments described below (since the genetic algorithm approach to conformational searching may be considered, in qualitative terms at least, as a directed form of random search).

Table 6. Percentage of Hits from the Geometric Search That Were Shown To Contain the Query Pharmacophore and CPU Times per Structure in the Full- and Reduced-Definition Genetic-Algorithm Experiments[a]

	full definition			reduced definition		
query	all hits	hits/run	*t*	all hits	hits/run	*t*
1	0.0	0.0	3.17	0.0	0.0	1.84
2	43.5	38.8	2.45	43.5	39.7	1.47
3	0.0	0.0	3.42	0.0	0.0	2.26
4	32.9	27.1	3.12	32.9	28.9	2.14
5	34.2	15.2	3.25	37.4	20.2	2.22
6	37.0	34.6	1.63	37.5	35.3	1.03
7	25.6	7.7	4.27	32.6	13.0	3.00
8	15.3	11.8	1.39	16.6	13.8	0.75
mean	23.6	16.9	2.84	25.1	20.1	1.84
median	29.3	13.5	3.14	32.8	17.0	1.99

[a] *t* is the mean time per structure in CPU seconds.

4.4. Genetic Search. A genetic algorithm is inherently stochastic in nature, and thus each search was repeated 10 times, and the reported results are the mean values when averaged over each such set of runs. The search time per structure is that for a single run.

The first set of experiments was run with the following parameter values: MAXOPS = 1000, POP_SIZE = 35, OPS_INC = 100, PENALTY_INC = 0.1, VDW_ENERGY_WT = 0.02, VDW_ENERGY_WINDOW = 50, and TOLERANCE = 0.0. The results are tabulated in Table 6, from which it will be seen that the reduced definition of rotatable bonds yields superior results to the full definition both in terms of the number of hits found and in the time taken to analyze a given structure. There are two reasons for this. The restriction of the rotatable bonds to those lying between the pharmacophore contact points decreases the volume of conformational space that needs to be searched while still permitting the retrieval of all of the possible hits. This statement holds as long as a generous energy window is permitted to allow for bad vdW contacts, which may occur due to the overlap of side chains which remain fixed under the reduced definition. In the full definition, conversely, such groups are allowed to relax by torsional motion and bad contacts may thus be relieved. It would appear from the above results that the energy window of 50 kcal/mol is sufficient to allow at least as many hits to be retrieved by the reduced as by the full definition and with a time-saving of at least 30%. The second point is that the reduction in the number of rotatable bonds reduces the length of the bit string to be manipulated and thus reduces the time for a given number of operations to be performed (though the time for the bit-string manipulations is small compared to that for the energy function).

In both sets of searches, except for queries 5 and 7, the number of hits obtained in a single run is a significant percentage of the total number of unique hits found in the complete set of 10 runs. This suggests, that apart from these two examples, the genetic algorithm samples the solution space effectively and consistently each time that it is invoked. A

Table 7. Percentage of Hits from the Geometric Search That Were Shown To Contain the Query Pharmacophore and CPU Times per Structure in the Full- and Reduced-Definition Genetic-Algorithm Experiments without the vdW Energy Penalty[a]

	full definition			reduced definition		
query	all hits	hits/run	*t*	all hits	hits/run	*t*
1	0	0.0	1.41	0	0.0	0.90
2	44.1	42.0	0.96	44.1	43.1	0.65
3	0	0.0	1.47	0	0.0	1.00
4	32.9	28.9	1.24	32.9	28.6	0.92
5	40.5	20.4	1.43	41.1	24.2	0.98
6	37.0	36.0	0.72	37.0	36.2	0.50
7	33.7	12.3	1.75	32.6	16.0	1.24
8	15.3	13.1	0.70	15.3	14.1	0.41
mean	25.4	19.1	1.21	25.4	20.3	0.83
median	33.3	16.8	1.33	32.8	20.1	0.91

[a] *t* is the mean time per structure in CPU seconds.

possible explanation for the behavior observed with queries 5 and 7 is presented later in this section.

The next set of experiments was run with exactly the same parameter values, with the sole exception of VDW_ENERGY_WT, which was set to 0.0, i.e., without the vdW energy penalty. The results are presented in Table 7. The reduced-definition searches again, in general, yield superior results to the full-definition searches, in that the number of unique hits found is approximately the same for both sets of searches, but the reduced-definition searches yield a greater number of hits in an individual run in a shorter period of time.

Comparing these results to those from the first set of experiments (which included the van der Waals' energy penalty term), it can be seen that omitting the vdW penalty leads to a small increase in the number of hits found since the energy criterion need no longer be satisfied by a given conformation and to a substantial decrease in the search times. This decrease is observed because less computation is required if the vdW energy penalty is omitted and because the removal of this energy constraint increases the probability that any given conformation will be a hit.

It will be clear from the description of the genetic algorithm in section 2.4 that there are may parameters involved and that the precise values of these may affect the results of any GA-based procedure. The values used thus far were obtained after initial experimentation suggested that they gave acceptable results. The eight query pharmacophores in Figure 1 were also searched using population sizes of 70 and 140 chromosomes (rather than 35 as in the experiments reported thus far) to determine whether a larger population would result in an increase in the number of hits. Although it might be anticipated that increasing the population size would lead to an increase in the number of hits, this is not observed in these experiments. This is because an increase in the population size also increases the number of operations required by the genetic algorithm to converge toward a solution. Thus, since the MAX_OPS parameter was held constant at 1000, the genetic algorithm converged in fewer instances; i.e., fewer hits were found compared to the earlier experiments with a population size of 35. Increases in MAX_OPS to 4000 and then to 10 000 did not increase the number of hits. It was also observed that increasing the population size led, as would intuitively be expected, to an increase in the time taken to analyze a given structure. For example, the mean time per structure (when averaged over the eight query pharmacophores) for the reduced-definition searches when the vdW energy penalty was used were 1.84, 2.19, and 2.41 CPU s for POP_SIZE = 35, 70, and 140, respectively. The eight pharmacophores were also searched using 20 runs per query, rather than 10 runs as previously, to determine whether the latter figure was large enough to provide an adequate sampling of the solution space. This change was found to have little or no effect on the numbers of hits that were obtained.

Table 8. Percentage of Hits from the Geometric Search That Were Shown To Contain the Query Pharmacophore in the Full- and Reduced-Definition Geometric-Algorithm Experiments with the vdW Energy Penalty for the Two Tightly-Defined Queries[a]

	full definition			reduced definition		
query	all hits	hits/run	*t*	all hits	hits/run	*t*
5	39.5	33.6	12.80	42.5	35.6	8.04
7	33.7	28.1	18.14	33.7	29.2	11.33

[a] The experiments used a value of 10 000 for MAX_OPS, of 1000 for OPS_INC, and of 35 for POP_SIZE. *t* is the mean time per structure in CPU seconds.

Table 9. Percentage of Hits from the Geometric Search That Were Shown To Contain the Query Pharmacophore and CPU Times per Structure in the Directed-Tweak Experiments[a]

query	hits	*t*
1	0.0	0.13
2	44.6	0.19
3	0.0	0.13
4	30.1	0.40
5	38.4	0.34
6	38.4	0.17
7	34.9	0.38
8	15.3	0.20
mean	25.2	0.24
median	32.5	0.20

[a] *t* is the mean time per structure in CPU seconds.

We thus conclude that the algorithm is fairly robust with respect to variations in the various parameters, which is a useful characteristic in that appropriate defaults should provide an adequate level of performance without the need for a user to have to specify several values before a search can be carried out successfully. That said, we have noted previously that the numbers of hits per run for queries 5 and 7 are significantly less than the total number of unique hits found, whereas the number of hits per run is far closer to the total hits found for the other queries. This suggests that in the cases of queries 5 and 7, the genetic-algorithm results vary considerably from run to run in terms of the structures found to be hits. It was postulated that the reason for this might be that these queries are both very tightly constrained and thus that solutions would tend to correspond to very small "pockets" of the conformational space. If this is so, it would seem reasonable that the genetic algorithm may have difficulty in locating the majority of them in a single run. Accordingly, these two searches were rerun with MAX_OPS set to 10 000 and OPS_INC to 1000 (rather than the standard defaults of 1000 and 100, respectively). The results from these experiments are shown in Table 8, where it will be seen that the mean number of hits per run is now very much closer to the total number of unique hits than was previously the case. This reduction in variance is, however, accompanied by a substantial increase in the search time per structure (owing to the 10-fold increase in the threshold number of operations).

4.5. Directed-Tweak Search. The results for the directed-tweak searches are given in Table 9, which shows that this algorithm yields excellent results both in terms of the number of hits found and the time taken to find them. On both scores, this approach would appear to be superior to all of the other algorithms that we have studied here, with only the GA offering

Table 10. Percentages of the Database That Were Hits in the Screen and Geometric Searches of the File of 9886 CAS Structures

query	screen search	geometric search
1	50.5	3.9
2	31.6	6.3
3	71.7	2.6
4	28.0	2.0
5	31.6	7.3
6	55.5	11.4
7	34.1	4.5
8	50.4	13.5
mean	44.1	6.4
median	42.1	5.4

Table 11. Percentage of Hits from the Geometric Search That Were Shown To Contain the Query Pharmacophore and CPU Times per Structure in the Genetic-Algorithm and Directed-Tweak Experiments with the Larger Database[a]

	genetic algorithm			directed tweak	
query	all hits	hits/run	*t*	hits	*t*
1	0.0	0.0	0.77	0.0	0.39
2	28.7	27.1	0.67	29.2	0.56
3	0.0	0.0	0.49	0.0	0.39
4	45.4	39.4	0.97	42.8	0.97
5	18.0	10.3	0.77	17.8	0.71
6	43.6	43.0	0.40	46.8	0.39
7	19.3	9.9	0.83	19.3	0.82
8	0.0	0.0	0.32	0.0	0.35
mean	19.4	16.2	0.65	19.5	0.57
median	18.7	10.1	0.72	18.6	0.48

[a] *t* is the mean time per structure in CPU seconds.

a substantial degree of competition on both accounts. That said, it should be noted that the directed-tweak algorithm used here is a commercial product that represents a considerable investment of resources, whereas the genetic algorithm is a research program that emphasises experimental flexibility, rather than the optimization of the code.

5. COMPARISON OF THE GENETIC-ALGORITHM AND DIRECTED-TWEAK METHODS

The experiments reported in the previous section suggest that the genetic-algorithm and directed-tweak methods are the most appropriate (in terms of both effectiveness and efficiency) for searching databases of flexible 3-D structures. However, the need to carry out a detailed comparison of the various methods has meant that our experiments have been restricted to a small database containing only 1538 structures. We have hence further investigated the genetic-algorithm and directed-tweak methods using our standard set of eight query pharmacophores on a larger file of 9886 CONCORD structures from the Chemical Abstracts Service database. The bounded-distance matrices here were generated using the DIST_GEOMETRY functionality available in SYBYL Version 6.0,[15] and these were submitted to the screen-search and geometric-search routines that have been described previously.[7] The results of these searches are given in Table 10, which lists the percentage of the database that was eliminated from further consideration. The sets of hits resulting from the geometric searches were then processed by the directed-tweak and genetic algorithms, the results of which are detailed in Table 11. The genetic algorithm here used the standard search parameters (POP_SIZE = 35, MAX_OPS = 1000, OPS_INC = 100, and PENALTY_INC = 0.1) without a vdW energy penalty, and both algorithms used the reduced definition of flexible bonds.

The results in Table 11 generally support those that were obtained with the smaller data set. Thus, the directed-tweak searches identify more hits than the single-run genetic-algorithm searches, although the numbers of hits are much more comparable if the genetic algorithm is run 10 times. However, while the numbers of hits may be similar, there are differences in the actual structures that are retrieved. For example, 119 of the 140 distinct hits for the sixth query pharmacophore were retrieved by both types of search; a further 11 hits were identified only in the genetic-algorithm search and a further 10 hits only in the directed-tweak search. In addition to being more effective than the genetic-algorithm searches, the directed-tweak searches are also more efficient. However, the difference in the times per structure here is noticeably less than in the experiments reported in section 4: this is due, at least in part, to improvements that were made to the coding of the genetic algorithm subsequent to the carrying-out of the experiments in section 4.4; it is likely that the times recorded here for the genetic algorithm could be further reduced, which would make it more competitive than currently appears to be the case.

There is again a large difference between the numbers of hits in the 1-run and 10-run genetic-algorithm searches for the fifth and seventh queries, but much less of a difference for the other queries. The mean and median numbers of hits are less here than in previous tables since there are now three queries that produced no hits in the conformational search, rather than just two queries as previously.

6. CONCLUSIONS

There is currently very considerable interest in the development of methods for searching databases of flexible 3-D structures. We have shown previously[7] that this can be accomplished by a three-stage procedure, in which suitably-modified versions of the screening and geometric-searching algorithms used for conventional rigid searching are complemented by a final conformational search. There are many conformational-searching methods available, and this paper has presented a detailed comparative evaluation of the merits of five such methods when they are used to search for eight query pharmacophores in two files of flexible molecules. Our experiments suggest that the distance–geometry approach is too slow to merit consideration, even if we neglect the need to clean-up the structures that result from the use of this method. Systematic search results in the retrieval of a fair number of high-quality hits but is again unacceptably slow unless a time limit is applied to the search; acceptable numbers of hits were obtained in our experiments with a threshold of 5 CPU s/structure. Our implementation of the random-search algorithm was also far too slow for practical use, but the large numbers of hits that were obtained suggested the use of a genetic approach: this has proved to be both an effective and an efficient means of conformational searching (though the precise level of performance that is achieved depends on the parameter values that are used). The best results with the 1538-compound data set were obtained with the directed-tweak algorithm, which retrieved a slightly larger number of hits than the genetic algorithm and which was noticeably faster. These findings were confirmed in a second set of experiments that used a larger file of 9886 flexible molecules.

We thus conclude that directed tweak is the algorithm of choice for the conformational-searching stage of a flexible search. That said, it should be noted that our experiments have used pharmacophoric patterns that involve only interatomic distance constraints. Other types of query constraint, e.g., valence or torsion angles, or included or excluded volumes, are easily encompassed by the genetic algorithm, since they require merely the specification of an appropriate penalty function, with the actual search algorithm being completely unaffected. However, the inclusion of nondistance constraints in a pharmacophoric pattern would require modifications to the directed-tweak algorithm to allow that pattern to be searched for in a database. A further potential advantage of the genetic algorithm is that it is very readily parallelized, and substantial increases in search speeds would be expected if it were to be implemented on a parallel processor; we hope to investigate this characteristic of genetic algorithms in the near future.

Despite the large number of experiments that we have carried out, there are still several areas that merit further investigation. The first, and most obvious one, is to determine the extent to which the availability of flexible searching does, indeed, result in an increase in the number of biologically-active molecules that are retrieved in 3-D database searches, when compared with the comparable rigid searches. Initial work in this area has been reported by Haraki *et al.*,[3] using the ChemDBS-3D system in which only a limited number of low-energy conformers are explored during a flexible search. Their studies need to be extended using the approach advocated here, in which the full conformational space of a flexible molecule is explored at search time. Secondly, comparative searches need to be carried out using a more varied range of pharmacophoric patterns than have been studied in our experiments (as mentioned above). Thirdly, there is an urgent need for the development of methods to handle the large numbers of hits that can be expected from flexible 3-D searches. We have noted previously that the conformational-searching algorithms here have been applied only to the first isomorphism for each molecule passing the geometric search; even so, large numbers of hits were obtained. Some limited experiments were carried out in which *all* of the isomorphisms were tested, and we found that *ca.* 60% of the structures that matched the query pattern in the geometric search proved to be hits in the final conformational search. If this figure is approximately correct, a search for a typical pharmacophore in a corporate database might well result in the retrieval of several thousands of hit structures, with a consequent need for postprocessing mechanisms that could help a searcher to assimilate the output file.

Thus, while the work reported here and previously[7,9] has demonstrated that it is possible to explore fully the conformational space of flexible molecules in 3-D database searches, many problems still need to be addressed before such searches can be carried out on the routine basis that characterizes 2-D substructure searching systems.

ACKNOWLEDGMENT

We thank Zeneca Pharmaceuticals, Wellcome Research Laboratories, and the Science and Engineering Research Council for the award of CASE studentships to D.E.C. and G.J. and Tripos Associates for the provision of hardware and software. We are grateful to Tad Hurst for the provision of the random-search and directed-tweak codes and for many valuable discussions on the subject of conformational flexibility. This paper is a contribution from the Krebs Institute for Biomolecular Research, which is a designated Centre for Molecular Recognition Studies under the Molecular Recognition Initiative of the Science and Engineering Research Council.

APPENDIX 1

The SEARCH SETUP parameters in SYBYL were specified as follows:

Cleaning Up CONCORD Structures. All of the database structures were subjected to a single energy minimization using the MAXIMIN2 facility, which serves to lessen any close nonbonded contacts in the CONCORD structure.

Definition of Rotatable Bonds. This was accomplished by means of the command ROTATABLE_BOND DEFINE bond_expr. Two sets of experiments were carried out: full definition and reduced definition, as defined in section 2.2. The full definition was specified by using "*" as the bond_expr option; while the reduced definition was specified by a bond_expr of the form (*a*:*b*), where *a* and *b* are the atom numbers of pharmacophore atoms of interest.

Definition of Search Angle Increments. The scan parameter of the conformational search is controlled by the command ANGLES INCREMENT step_size; experiments were carried out with step_size set to 60°, 40°, 30°, 20°, or 10°.

Definition of Distance Constraints. The geometric requirements of the pharmacophore were imposed upon the candidate structure using the command CONSTRAINING_DISTANCES DEFINE atom1 atom2 min_dist max_dist, where the latter values were the matching ranges produced by the geometric search program.

Energy Calculations. The energies of the conformations produced during the search were calculated using the command ENERGY ENERGY max_energy no_electrostatics. No energy limit was imposed in the initial experiments.

Remaining Parameters. All other settings were left as the program defaults; i.e., the reference conformation from which all angle increments are to initiate was "zeroed" (all rotatable bond torsion angles set to 0°), no bump checking was performed, and the vdW scaling factors were as determined by the program.

REFERENCES AND NOTES

(1) Bures, M. G.; Black-Schaefer, C.; Gardner, G. The Discovery of Novel Auxin Transport Inhibitors by Molecular Modeling and Three-Dimensional Pattern Analysis. *J. Comput.-Aided Mol. Des.* **1991**, *5*, 323–334.

(2) Martin, Y. C. 3D Database Searching in Drug Design. *J. Med. Chem.* **1992**, *35*, 2145–2154.

(3) Haraki, K. S.; Sheridan, R. P.; Venkataraghavan, R.; Dunn, D. A.; McCulloch, D. Looking for Pharmacophores in 3-D Databases: Does Conformational Searching Improve the Yield of Actives? *Tetrahedron Comput. Methodol.* **1990**, *3*, 565–573.

(4) Christie, B. D.; Henry, D. R.; Wipke, W. T.; Moock, T. E. Database Structure and Searching in MACCS-3D. *Tetrahedron Comput. Methodol.* **1990**, *3*, 653–664.

(5) Murrall, N. W.; Davies, E. K. Conformational Freedom in 3-D Databases. 1. Techniques. *J. Chem. Inf. Comput. Sci.* **1990**, *30*, 312–316.

(6) Güner, O. F.; Henry, D. R.; Pearlman, R. S. Use of Flexible Queries for Searching Conformationally Flexible Molecules in Databases of Three-Dimensional Structures. *J. Chem. Inf. Comput. Sci.* **1992**, *32*, 101–109.

(7) Clark, D. E.; Willett, P.; Kenny, P. W. Pharmacophoric Pattern Matching in Files of Three-Dimensional Chemical Structures: Use of Bounded Distance Matrices for the Representation and Searching of Conformationally-Flexible Molecules. *J. Mol. Graphics* **1992**, *10*, 194–204.

(8) Havel, T. F.; Kuntz, I. D.; Crippen, G. M. The Theory and Practice of Distance Geometry. *Bull. Math. Biol.* **1983**, *45*, 665–720.

(9) Clark, D. E.; Willett, P.; Kenny, P. W. Pharmacophoric Pattern Matching in Files of Three-Dimensional Chemical Structures: Implementation of Flexible Searching. *J. Mol. Graphics* **1993**, *11*, 146–156.

(10) Burt, S. K.; Greer, J. Search Strategies for Determining Bioactive Conformers of Peptides and Small Molecules. *Annu. Rep. Med. Chem.* **1988**, *23*, 285–294.

(11) Howard, A. E.; Kollman, P. A. An Analysis of Current Methodologies for Conformational Searching of Complex Molecules. *J. Med. Chem.* **1988**, *31*, 1669–1675.

(12) Leach, A. R. A Survey of Methods for Searching the Conformational Space of Small and Medium-Sized Molecules. In *Reviews in Computational Chemistry II*; Lipkowitz, K. B., Boyd, D. B., Eds.; VCH: New York, 1991; pp 1–55.

(13) Blaney, J. M.; Crippen, G. M.; Dearing, A.; Dixon, J. S. *DGEOM: Distance Geometry.* Quantum Chemistry Program Exchange program number 590, Department of Chemistry, Indiana University: Bloomington, IN.

(14) Crippen, G. M.; Havel, T. F. *Distance Geometry and Molecular Conformation*; Research Studies Press: Letchworth, U.K., 1988.

(15) SYBYL. Tripos Associates Inc., St Louis, MO.

(16) Dammkoehler, R. A.; Karasek, S. F.; Shands, E. F. B.; Marshall, G. R. Constrained Search of Conformational Hyperspace. *J. Comput.-Aided Mol. Des.* **1989**, *3*, 3–21.

(17) Chang, G.; Guida, W. C.; Still, W. C. An Internal Coordinate Monte Carlo Method for Searching Conformational Space. *J. Am. Chem. Soc.* **1989**, *111*, 4379–4386.

(18) Saunders, M.; Houk, K. N.; Wu, Y.-D.; Still, W. C.; Lipton, M.; Chang, G.; Guida, W. C. Conformations of Cycloheptadecane: A Comparison of Methods for Conformational Searching. *J. Am. Chem. Soc.* **1990**, *112*, 1419–1427.

(19) Davis, L., Ed. *Handbook of Genetic Algorithms*; Van Nostrand Reinhold: New York, 1991.

(20) Goldberg, D. E. *Genetic Algorithms in Search, Optimization and Machine Learning*; Addison-Wesley: Wokingham, MA, 1989.

(21) Blommers, M. J. J.; Lucasius, C. B.; Kateman, G.; Kaptein, R. Conformational Analysis of a Dinucleotide Photodimer with the Aid of the Genetic Algorithm. *Biopolymers* **1992**, *32*, 45–52.

(22) Legrand, S.; Merz, K. The Application of the Genetic Algorithm to Conformational Search. *FASEB J.* **1992**, *6*, A132.

(23) Payne, A. W. R.; Glen, R. C. Molecular Recognition Using a Binary Genetic Search Algorithm. *J. Mol. Graphics* **1993**, *11*, 74–91.

(24) Fontain, E. Application of Genetic Algorithms in the Field of Constitutional Similarity. *J. Chem. Inf. Comput. Sci.* **1992**, *32*, 748–752.

(25) Jones, G. *The Use of Genetic Algorithms in Chemical Information Systems*. Ph.D. Thesis, University of Sheffield. Manuscript in preparation.

(26) Clark, M.; Cramer, R.D., III; Van Opdenbosch, N. Validation of the General Purpose TRIPOS 5.2 Force Field. *J. Comput. Chem.* **1989**, *8*, 982–1012.

(27) Shenkin, P. S.; Yarmush, D. L.; Fine, R. M.; Wang, H.; Levinthal, C. Predicting Antibody Hypervariable Loop Conformation. I. Ensembles of Random Conformations for Ringlike Structures. *Biopolymers* **1987**, *26*, 2053–2085.

(28) Fine, R. M.; Wang, H.; Shenkin, P. S.; Yarmush, D. L.; Levinthal, C. Predicting Antibody Hypervariable Loop Conformation. II. Minimisation and Molecular Dynamics Studies of MCPC603 from Many Randomly-Generated Loop Conformations. *Proteins: Struct. Funct. and Genet.* **1986**, *1*, 342–362.

(29) Hurst, T. Flexible 3–D Searching: the Directed Tweak Technique. Paper presented at the Third International Conference on Chemical Structures, Noordwijkerhout, The Netherlands, June 6–10, 1993.

(30) Clark, D. E.; Willett, P.; Kenny, P. W. Pharmacophoric Pattern Matching in Files of Three-Dimensional Chemical Structures: Use of Smoothed Bounded Distances for Incompletely-Specified Query Patterns. *J. Mol. Graphics* **1991**, *9*, 157–160.

(31) Smellie, A. S. *Distance Geometry: New Methods and Applications.* Ph.D. Thesis, University of Oxford, 1989.

(32) Kuntz, I. D.; Thomason, J. F.; Oshiro, C. M. Distance Geometry. *Methods Enzymol.* **1989**, *177*, 159–204.