J-CAMD 053

Upperbound procedures for the identification of similar three-dimensional chemical structures

Andrew T. Brint and Peter Willett*

Department of Information Studies, University of Sheffield, Western Bank, Sheffield S10 2TN, U.K.

Received 5 March 1988
Accepted 11 July 1988

Key words: Chemical structure databases; Maximal common subgraph isomorphism algorithm; Molecular similarity; Upperbound algorithm

SUMMARY

This paper describes techniques for calculating the degree of similarity between an input query molecule and each of the molecules in a database of 3-D chemical structures. The inter-molecular similarity measure used is the number of atoms in the 3-D common substructure (CS) between the two molecules which are being compared. The identification of 3-D CSs is very demanding of computational resources, even when an efficient clique detection algorithm is used for this purpose. Two types of upperbound calculation are described which allow reductions in the number of exact CS searches which need to be carried out to identify those molecules from a database which are similar to a 3-D target molecule.

1. INTRODUCTION

The three-dimensional (3-D) structure of chemical compounds is of great importance in determining their biological activity and there is thus considerable interest in the development of molecular graphics systems for the storage and interactive analysis of 3-D chemical structure information [1–3]. The availability of large numbers of sets of 3-D molecular coordinates has led to the development of searching systems which go beyond the mere display of molecules by allowing the retrieval of those structures in a database which contain *pharmacophoric patterns*, user-defined patterns of atoms in 3-D space which are believed to be responsible for biological activity [4–9]; such systems are the 3-D analogues of the techniques which have been used for many years for *substructure searching* in databases of two-dimensional (2-D) chemical structure diagrams [10, 11].

Substructure searching, whether in 2-D or in 3-D, is an example of *partial match retrieval*, where the matching of a query against a file of structures results in a partition of the database into those

* To whom correspondence should be addressed.

molecules that contain the specified query substructure, and the remainder which do not. Such an approach is appropriate when the required substructure can be specified precisely and in detail. A quite different sort of search is required if there is a need to identify the structures in the database which are similar to an input query molecule, using some quantitative definition of intermolecular structural similarity. This is an example of *nearest neighbour*, or *best match*, *retrieval*. There has recently been interest in the use of similarity-based techniques with databases of 2-D chemical structures, the applications including both nearest neighbour searching [12–15] and a variety of quantitative structure–activity relationship studies [16–20]; a detailed review of this work is presented by Willett [21]. Most of these studies have used inter-molecular similarity measures which are based on the number of fragment substructures in common between a pair of molecules; while simple to calculate and effective in operation, such measures take no account of the topological relationships which exist between the fragments, and Johnson and his co-workers [17, 18] have recently suggested that better results might be obtained from the use of the *maximal common substructure* (MCS) as the similarity measure [22]. A 2-D chemical structure can be regarded as a labelled graph, in which the atoms and bonds of the structure correspond to the nodes and edges of a graph. A labelled graph representation, called a *connection table*, forms the basis for most modern chemical structure information systems [10, 11]. Given a pair of graphs, a maximal common subgraph corresponds to the largest common set of nodes and edges [22]; an MCS can thus be derived by using an appropriate maximal common subgraph isomorphism algorithm and there have been several reports of the use of such algorithms with 2-D structural data for chemical reaction indexing [23–25] and structure elucidation [26, 27]. More recently, this work had been extended to the identification of MCSs with 3-D chemical structures, the edges of the maximal common subgraph in this case corresponding to the inter-atomic distances (rather than to the bonds as in the case of 2-D structures).

Two main types of algorithm have been described for the identification of 3-D MCSs. Crandell and Smith discuss the use of a breadth-first search procedure in which individual atoms common to all of the structures under investigation are systematically extended, one atom at a time, until the resulting 3-D substructure cannot be extended any further [28]. An implementation of this algorithm on a microprocessor-based, multiprocessor system is described by Brint and Willett [29]. Golender and Rozenblit [30] and Kuhl et al. [31] have used algorithms which are based on the detection of *cliques* in a correspondence graph linking together pairs of molecules. Brint and Willett have extended the clique detection algorithm to allow it to be used for the comparison of any number of molecules, rather than just two, and have shown it to be generally more efficient in operation than the Crandell–Smith algorithm [32].

All of these studies have considered the comparison of only small numbers of structures but there does not seem to be any inherent reason why the technique should not be extended to entire databases of 3-D chemical structures to allow the identification of structures having a high degree of similarity with an input *target structure*. In this paper, we investigate the computational requirements of an experimental system which retrieves from a database all molecules having a 3-D substructure in common with the target which is greater than some user-defined size; thus, the inter-molecular similarity measure studied here is the size of the common substructure (CS). Section 2 describes the basic clique detecion algorithm which we have used for the identification of 3-D CSs. Section 3 describes an initial upperbound calculation which can be used to eliminate from detailed consideration structures in the database which cannot possibly have a CS with the target molecule larger than some user-defined size. This upperbound calculation uses descriptors assigned to the query and to database structures which represent pairs of atoms together with an associated distance range. A more precise upperbound calculation, based on the actual distances present in the target and database structures, is discussed in Section 4 and the paper closes with a summary of our findings.

2. CLIQUE DETECTION ALGORITHM

The clique detection algorithm which we have used in this work seems to have been first described by Levi [33]; it has been used both in computer vision research [34, 35] and in chemical information systems [9, 27, 30–32].

Given a pair of graphs, A and B, a *correspondence graph*, C, can be formed by the following process:

(1) Create the set of all pairs of nodes, one from each of the two graphs, such that the nodes of each pair are of the same type.

(2) Form the graph C whose nodes are the pairs from Step 1. Two nodes (A_I, B_X) and (A_J, B_Y) are marked as being connected in C if the values of the arcs from A_I to A_J and B_X to B_Y are the same.

Maximal common subgraphs then correspond to the *cliques* of the correspondence graph, where a clique is a subgraph in which every node is connected to every other node and which is not contained in any larger subgraph with this property. Thus, the identification of the MCS for a pair of 3-D chemical structures, where the nodes and arcs correspond to the atomic types and the interatomic distances, is equivalent to the identification of the largest clique in the correspondence graph linking together the two structures; in the present application, those cliques, and hence CSs, are required which are at least some given size. Clique detection has been studied extensively, and Brint and Willett [32] present the results of a comparative study of the efficiencies of several clique finding algorithms which can be used for 3-D CS identification. Their results suggest that the algorithm of Bron and Kerbosch [36] is the most generally useful for this application and we have accordingly used this algorithm for all of the work reported here. A detailed account of the application of clique finding algorithms to 3-D chemical structures is presented by Brint [37].

Clique detection is extremely demanding of computational resources, owing to the very large number of cliques in a correspondence graph. A further problem is that many of these will correspond to very small CSs, consisting of only a few atoms, and these are unlikely to be of any interest in a practical context. Accordingly, the computational requirements of CS-based similarity searching in a large database might be much reduced if some way could be found to rapidly eliminate from further consideration those molecules which have only a very small CS with the target structure; the following two sections describe upperbound procedures which can be used to eliminate many of these uninteresting structures without the need to use the detailed clique detection procedure.

3. UPPERBOUND CALCULATION BASED ON DISTANCE RANGES

Substructure searching, whether in 2-D or in 3-D, corresponds to the classical subgraph isomorphism problem which is known to be NP-complete and thus extremely demanding of computational resources. To increase the efficiency of searching, chemical substructure search systems use a two-stage retrieval mechanism [10, 11]. In the initial stage, queries and database structures are represented by lists of small fragment substructures called *screens*; a match at the screen level is a necessary, but not sufficient, condition for a match at the second stage, subgraph isomorphism level in which query and database atoms are mapped onto each other using a depth-first, backtracking search. Jakes and Willett have described an algorithmic selection procedure for the identification of inter-atomic distance screens which can be used to increase the efficiency of 3-D substructure searching [7]. We have used the screens resulting from this procedure as the basis for a rapid upperbound calculation which can be used to give an upperbound to the size of the substructures common to a database structure and a target molecule. The screens used are the AA screens of Jakes et al. [8], where an AA screen is one in which the atomic types of both of the atoms in a pair are specified; Jakes et al. also describe the use of AX screens, in which only one of the atomic types is specified so as to allow for the encoding of generalised substructure queries, but these are not used here.

The basic idea underlying the screen upperbound calculation is as follows: if some pair of atoms in the target molecule, T, matches some pair in a database structure, D, i.e., if the atoms are of the same types and if they are separated by the same distance (to within any allowed tolerance), then the screens corresponding to the pair must be set in the lists of screens representing both T and D. Thus, all pairs of atoms in D which can *possibly* be involved in a CS with T can be identified by determining the screens in common between T and D. Knowing these pairs of atoms, a *screen-based correspondence graph* can be generated; the clique detection algorithm can then be used to calculate an upperbound to the size of the substructure common to T and D. In detail, the procedure is as follows:

(1) An $NT \times NT$ adjacency matrix, M, is created where NT is the number of nodes, i.e., atoms, in the graph representing T.

(2) The inter-atomic distance between each pair of atoms, I and J, in T is calculated and the screen corresponding to this distance, say X, identified from the screen dictionary; the element M_{IJ} is set to one, or zero, depending upon whether X is present in, or absent from, the list of screens representing D.

(3) Once all of the $NT\,(NT - 1)/2$ distinct pairs of atoms in T have been processed in this way, the screen-based correspondence graph, M, is examined to find the size of the largest clique present in it; this size then represents an upperbound to the size of a substructure common to both T and D.

(4) If the upperbound value for the size of the CS is above a threshold, expressed as some minimum number of atoms, then the target and database structures are analysed in detail using the algorithm described previously in Section 2. Alternatively, if the upperbound value is not above the threshold, that molecule can be removed from further consideration since there is no possibility that it contains any CS which is sufficiently large to be of any practical interest.

To illustrate this process, consider a target molecule containing two oxygen atoms, separated by a distance of 6.7 Å, this corresponding to a screen which represents oxygen-oxygen separations in the range 6.34 – 6.85 Å. Then, if the screen list corresponding to some database structure, D, does not contain this screen, these two atoms cannot both be present in a common substructure. The two oxygens are therefore unconnected in the correspondence graph and cannot belong to the same clique. Alternatively, if the relevant oxygen-oxygen screen had been assigned to D, then the two oxygens could possibly be present in the same cliques.

The screen matching operation increases the efficiency of CS searching in another way, in addition to eliminating those database structures which have only small substructures in common with the target molecule. Specifically, for those molecules which do match at the screen level, only some of the constituent atoms need to be considered in the construction of the *exact correspondence graph*, i.e., that used in the second stage where the CS is calculated on the basis of exact distance matches (rather than screen matches). This is because the largest clique derived from the exact correspondence graph with contains some particular atom cannot be larger than the largest clique derived from the screen-based correspondence graph which contains that atom. Thus, only those database atoms which occur in screen-based cliques larger than the threshold need be considered in the second, more detailed comparison.

The efficiency of this upperbound procedure was tested using a sample of 999 structures from the Cambridge Crystallographic Database [38]. These structures had an average size of 20.3 non-hydrogen atoms; each of them was represented for search as a bit string containing 694 bits, each of which corresponded to the presence or absence of a particular AA screen. The screens were stored as a list of TRUE/FALSEs whilst the coordinates were stored separately because of their bulk. A tolerance of 0.15 Å was used when checking to see whether a distance in the target structure could be matched to one of the screens for a database structure. The cliques of the resulting screen-based correspondence graph were enumerated using the algorithm of Bron and Kerbosch; molecules matching the target were then passed on for the analysis of the exact correspondence graph. A problem with the clique detection approach is the size of the correspondence graph which may result when two structures are matched against each other. For a target and a database structure containing *NT* and *ND* atoms respectively, the exact correspondence graph contains up to NT^2ND^2 elements (although, as noted above, the screening stage may eliminate some of the nodes from consideration in the construction of this graph). Memory restrictions meant that it was not possible to process correspondence graphs containing more than 1000 nodes; however, no such graphs were encountered in practice during the experiments reported here. Full experimental details are given by Brint [37].

Three molecules, A, B and C in Fig. 1, were chosen from the database. Table 1 contains the experimental run times for CS searches using these as target structures. The first set of timings listed (in seconds of CPU time for FORTRAN 77 programs run on an IBM 3083 BX processor) are for the simple, brute force search in which the CSs are identified for each structure in turn without the use of any sort of upperbound calculation. The extended run times in the table demonstrate clearly the computational demands of CS identification when molecules of this sort are used as the target. When smaller molecules are used, the speed of clique detection increases drastically; e.g., a brute force search in which glycine was used as the target molecule required only 14.6 seconds of CPU time. The screen search, on the basis of which the upperbounds are calculated, is very rapid, requiring merely a scan of the screen lists to identify the number of screens in common between the target and each of the database structures. The main body of Table 1 describes the analysis of the exact correspondence graph to identify the cliques present. Each element of the table consists of three parts, these being the time taken to generate the cliques, the number of molecules actually containing a common substructure of the required size, and the number of molecules eliminated as a result of the analysis of the screen-based correspondence graph. Figures are listed for three minimal CS sizes, these being 4, 6 and 8 atoms. An analysis of the CSs identified showed that very many of them consisted solely of ring carbon atoms; it was hence decided to allow the minimum

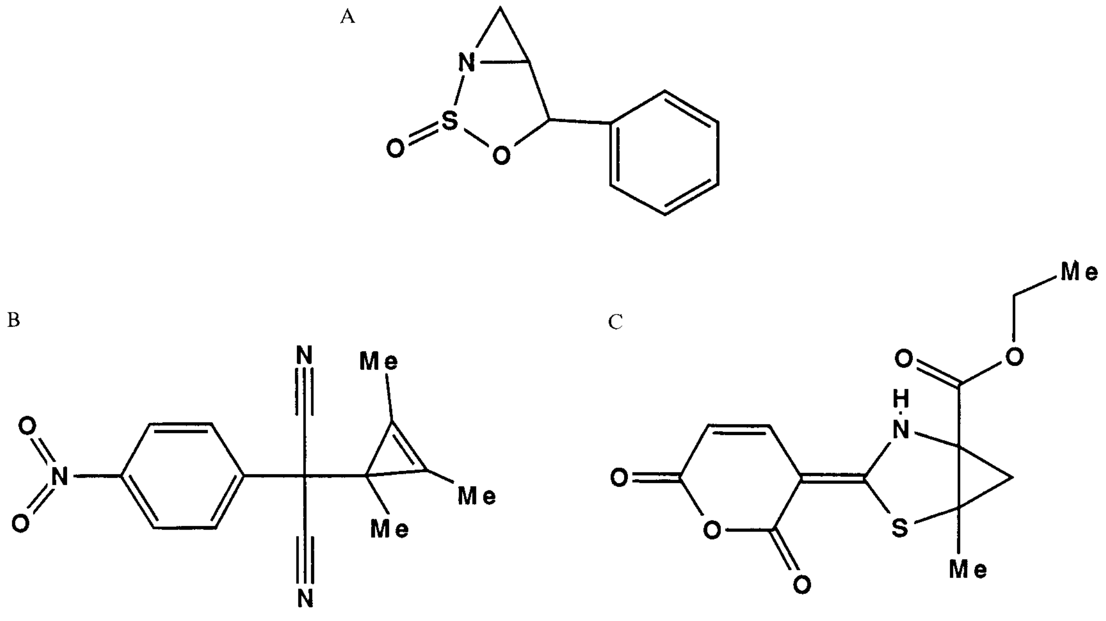

Fig. 1. Target structures used in the nearest neighbour searches.

TABLE 1

EFFECT OF VARIATIONS IN *H*, THE SMALLEST ACCEPTABLE NUMBER OF HETEROATOMS IN THE CS, AND IN THE NUMBER OF ATOMS IN THE SMALLEST ACCEPTABLE CS, ON THE EFFICIENCY OF SEARCHING*

	Minimum CS size	Target molecule								
			A			B			C	
No screening			240.5			701.5			543.8	
Screening			2.8			8.6			8.2	
H=0	4	239.9	839	24	700.1	905	19	540.4	896	11
	6	232.2	488	83	690.2	503	71	528.8	137	73
	8	218.0	31	272	672.3	67	161	502.4	2	222
H=1	4	200.8	284	102	532.4	437	117	484.8	381	76
	6	194.6	22	240	524.4	91	207	476.7	28	136
	8	176.8	7	410	506.3	24	321	449.4	1	278
H=2	4	105.0	19	473	303.4	107	475	365.1	117	281
	6	103.4	1	575	302.5	34	520	361.7	7	325
	8	97.5	1	680	298.6	22	571	347.4	1	426

*Each entry in the main body of the table contains the execution time followed by the number of structures actually containing a CS of the required size and the number of structures eliminated by the screening search.

acceptable number of heteroatoms to be specified. This number, H in Table 1, was set to be 0, 1 or 2. When $H=0$, it is clear that the upperbound calculation does little to reduce the computation, even with large CSs; however, the performance improves as H is increased. It should be noted that some of the increased screenout is due to the elimination of molecules which do not contain sufficient heteroatoms, a fact which could be identified very rapidly using a simple molecular formula check; however, Brint demonstrates that such a check would eliminate only a minority of the molecules eliminated from the clique detection stage by the upperbound calculation [37].

The figures in Table 1 point up the fact that the target molecules in any operational similarity-based retrieval system would need to be specified in quite restrictive terms if the user is not to be overwhelmed with a very large output; e.g., when $H=1$, a minimal CS size of 4 results in the retrieval of 28%, 44% and 38% of the file when A, B and C respectively are used as the target. The proliferation of small common 3-D substructures when molecules are compared has been noted in previous work [28, 32].

4. UPPERBOUND CALCULATION BASED ON EXACT DISTANCES

The figures in Table 1 show that the time spent in the comparison stage dominates that of the screening stage. Consequently, an extra, higher precision screening stage was added after the first screening stage (in a similar manner to the *distance search* component of the 3-D substructure searching system described by Jakes et al. [8]). As with the original screening stage, an $NT \times NT$ correspondence graph is used as the basis for the upperbound calculation; however, rather than setting M_{IJ} to one if the screens associated with some pair of target atoms, I and J, are present, this second comparison requires the equivalence of the actual distances in T and D (to within the allowed tolerance). The comparison is clearly much more precise than the screen-based comparison, since the matches there are based on the *possibility* of there being an exact match between the target and database distances. However, the first stage can be implemented very much more efficiently since only bit string operations are involved, and since there is no need to use the bulky coordinate data (which is stored on disc) to calculate the actual distances.

Table 2 gives the results of tests using this two-stage upperbound system with a minimal CS size of 6 atoms. The figures in the table represent the time taken for the analysis of this *distance-based correspondence graph*, the time taken for the analysis of the final, exact correspondence graph, and

TABLE 2
EFFECT OF VARIATIONS IN H, THE SMALLEST ACCEPTABLE NUMBER OF HETEROATOMS IN THE CS, ON THE EFFICIENCY OF SEARCHING WHEN BOTH UPPERBOUND CALCULATIONS ARE CARRIED OUT AND WHEN THE NUMBER OF ATOMS IN THE SMALLEST ACCEPTABLE CS IS 6*

H	A			B			C		
0	4.6	231.1	112	14.0	687.2	88	13.7	525.2	96
1	4.1	184.4	295	11.0	506.8	228	13.7	464.8	167
2	2.0	42.7	833	4.9	215.0	654	9.6	305.6	432

*Each entry in the table contains the execution times for the second screening and clique detection stages followed by the number of structures eliminated as a result of the two screening stages.

the number of structures eliminated prior to this final stage. The figures indicate that the second upperbound calculation can lead to a significant improvement when heteroatoms are specified as being required; however, a comparison with Table 1 shows that it has very little effect when cliques containing only carbons are allowed. We believe that the reason for this behaviour lies in the nature of the screens which are used. The screens have been designed to increase the efficiency of substructure searching and have been selected so as to occur approximately equifrequently in the database which is to be searched [7]. Since carbon atoms are the most frequent, screens containing pairs of carbons cover very much shorter distance ranges than do those for pairs of other atomic types; for example, there are 19 screens for the oxygen-oxygen pairs and 61 for the nitrogen-carbon pairs, but no less than 153 carbon-carbon screens. Hence, the screening system can predict more accurately whether a particular carbon-carbon distance is present, as opposed to an oxygen-oxygen distance, and thus the second-stage search is unlikely to bring about substantial increases in efficiency when there are few heteroatoms in the target molecule. Thus, the addition of an analysis of the distance-based correspondence graph after the initial, screen-based correspondence graph proved to be less effective than the use of the analogous distance search in our previous studies of 3-D substructure searching (where the pharmacophoric patterns which were used contained a significant fraction of heteroatoms, rather than being primarily carbon-based as with the target molecules used here) [8].

5. CONCLUSIONS

There are two major problems with the approach described in this paper, apart from the limitations inherent in the set of screens available to us. Firstly, the sheer numbers of structures which have substructures of non-trivial size in common with the target molecules; this can be overcome by demanding the retrieval only of those molecules having a very large CS and, ideally, by demanding that the CS should contain some minimal number of heteroatoms. If this cannot be done, then the run times will be high unless very small target molecules are employed. The second problem is the fact that the upperbounds calculated here are not sufficiently discriminating to eliminate many of the molecules that do not have CSs of the requisite size. This is illustrated by the figures in Table 3, which lists some of the upperbounds calculated following the second upperbound calculation, i.e., that based on the exact distances. It will be seen that the upperbound values, which are calculated for $H=0$, are much greater than the sizes of the actual common substructures; a good upperbound procedure is one in which the calculated upperbound value is only sligthly greater than the true value. In this respect, the upperbounds described here for 3-D similarity searching are much inferior to those which can be used for fragment-based similarity searching in databases of 2-D chemical structures [13]. The discrepancy between the calculated and actual CS sizes also means that it would not be of any real help to specify a very high upperbound value as a requirement for a clique to be considered in the exact correspondence graph; such a procedure might also result in the elimination of substructures which were of interest.

In this paper we have considered procedures for the searching of a target molecule against a database of molecules to identify those having 3-D substructures in common with it. This is a computationally demanding task if the target molecule is at all large and two upperbound calculations are described which can lessen the number of database structures which need to be considered. The experimental results show that these upperbounds can bring about substantial reductions in

TABLE 3
CALCULATED UPPERBOUNDS IN THE SECOND UPPERBOUND STAGE, ACTUAL CS SIZES AND NUMBERS OF MOLECULES WITH A CS OF THAT SIZE FOR SEARCHES WITH B AS THE TARGET MOLECULE

Calculated upperbound to CS	Actual CS	Number of molecules
16	4	9
16	5	15
16	6	21
16	7	9
16	8	7
16	>8	3
17	4	2
17	5	9
17	6	5
17	7	15
17	8	4
18	8	1
18	9	2

the computational requirements of CS detection if the CS is large or if it is possible to specify some mimimal number of heteroatoms which must be present if a CS is to be of interest. In other cases, the reduction in computation which can be achieved is quite small.

ACKNOWLEDGEMENTS

We thank David Bawden for assistance with the preparation of this paper and the Science and Engineering Research Council for funding.

REFERENCES

1 Morffew, A.J., J. Mol. Graph., 1 (1983) 17–23.
2 Vinter, J.G., Chem. Br., 21 (1) (1985) 32–38.
3 Cohen, N.C., Adv. Drug Res., 14 (1985) 41-145.
4 Gund, P., Prog. Mol. Sub-Cell. Biol., 5 (1977) 117–143.
5 Lesk, A.M., Commun. ACM, 22 (1979) 219–224.
6 Esaki, T., Fundamental Studies on Quantitative Drug Design, Ph.D. Dissertation, Nagoya University, Japan, 1983.
7 Jakes, S.E. and Willett, P., J. Mol. Graph., 4 (1986) 12–20.
8 Jakes, S.E., Watts, N., Willett, P., Bawden, D. and Fisher, J.D., J. Mol. Graph., 5 (1987) 41–48.
9 Brint, A.T. and Willett, P., J. Mol. Graph., 5 (1987) 49–56.
10 Ash, J.E., Chubb, P.A., Ward, S.E., Welford, S.M. and Willett, P., Communication, Storage and Retrieval of Chemical Information, Ellis Horwood, Chichester, 1985.
11 Willett, P., J. Chemometrics, 1 (1987) 139–155.
12 Carhart, R.E., Smith, D.H. and Venkataraghavan, R., J. Chem. Inf. Comput. Sci., 25 (1985) 64–73.
13 Willett, P., J. Chem. Inf. Comput. Sci., 23 (1983) 22–25.
14 Willett, P., Winterman, V. and Bawden, D., J. Chem. Inf. Comput. Sci., 26 (1986) 36–41.
15 Bawden, D., In Warr, W.E. (Ed.) Chemical Structures: The International Language of Chemistry, in press.

16 Willett, P. and Winterman, V., Quant. Struct. Act. Relationsh., 5 (1986) 18–25.

17 Johnson, M., In Alavi, Y., Chartrand, G., Lesniak, L., Lick, D.R. and Wall, C.E. (Eds.) Graph Theory and its Applications to Algorithms and Computer Science, John Wiley, New York, 1985, pp. 457–476.

18 Johnson, M., Naim, M., Nicholson, V. and Tsai, C.C., In Hadzi, D. and Jerman-Blazic, B. (Eds.) QSAR in Drug Design and Toxicology, Elsevier, Amsterdam, in press.

19 Wilkins, C.L. and Randic, M., Theor. Chim. Acta, 58 (1980) 45–68.

20 Broto, P., Moreau, G. and Vandycke, C., Eur. J. Med. Chem., 19 (1984) 66–70.

21 Willett, P., Similarity and Clustering in Chemical Information Systems, Research Studies Press, Letchworth, 1987.

22 McGregor, J.J., Software – Practice and Experience, 12 (1982) 23–34.

23 Bersohn, M. and Mackay, K., J. Chem. Inf. Comput. Sci., 19 (1979) 137–141.

24 McGregor, J.J. and Willett, P., J. Chem. Inf. Comput. Sci., 21 (1981) 137–140.

25 Willett, P. (Ed.), Modern Approaches to Chemical Reaction Searching, Gower, Aldershot, 1986.

26 Varkony, T.H., Shiloach, Y. and Smith, D.H., J. Chem. Inf. Comput. Sci., 19 (1979) 104–111.

27 Cone, M.M., Venkataraghavan, R. and McLafferty, F.W., J. Am. Chem. Soc., 99 (1977) 7668–7671.

28 Crandell, C.W. and Smith, D.H., J. Chem. Inf. Comput. Sci., 23 (1983) 186–197.

29 Brint, A.T. and Willett, P., J. Mol. Graph., 5 (1987) 200–207.

30 Golender, V. and Rozenblit, A., Logical and Combinatorial Algorithms for Drug Design, Research Studies Press, Letchworth, 1983.

31 Kuhl, F.S., Crippen, G.M. and Friesen, D.K., J. Comput. Chem., 5 (1984) 24–34.

32 Brint, A.T. and Willett, P., J. Chem. Inf. Comput. Sci., 27 (1987) 152–158.

33 Levi, G., Calcolo, 9 (1972) 341–352.

34 Barrow, H.G. and Burstall, R.M., Inf. Processing Lett., 4 (1976) 83–84.

35 Barrow, H.G. and Tenenbaum, J.M., Proc. IEEE, 69 (1981) 572–595.

36 Bron, C. and Kerbosch, J., Commun. ACM, 16 (1973) 31–42.

37 Brint, A.T., Matching Algorithms for Handling Three-Dimensional Molecular Coordinate Data, Ph.D. Dissertation, University of Sheffield, 1987.

38 Allen, F.H., Bellard, S., Brice, M.D., Cartwright, B.A., Doubleday, A., Higgs, H., Hummelink, T., Hummelink-Peters, B.G., Kennard, O., Motherwell, W.D.S., Rogers, J.R. and Watson, D.G., Acta Crystallogr., Sect. B 35 (1979) 2331–2339.