Pergamon

Tetrahedron 57 (2001) 6035–6042

TETRAHEDRON

GASP: a computer program to generate automatically polycyclic structures

René Barone,[a,*] Rémi Barone,[b] Michel Arbelot[a] and Michel Chanon[a]

[a]*Laboratoire AM3-case 561, UMR CNRS 6009, Faculte des Sciences St Jérôme, Av. Escadrille Normandie-Niemen, 13397 Marseille Cedex 20, France*

[b]*ESIL/ES2I, Faculté des Sciences de Luminy, Marseille, France*

Received 15 December 2000; revised 12 March 2001; accepted 17 May 2001

Abstract—We present a new program to generate a virtual library of structures. GASP (Génération Automatique de Structures Polycycliques) answers to a question like: generate all structures with 3 rings of size 4, 6 and 8 for example. We present the results obtained for 'isomers' of several structures (steroid, taxane and triquinane skeletons). For steroid skeleton (4 rings of size 6, 6, 6 and 5) 988 structures were generated. Among them 48 are composed of fused rings. Comparison with the Beilstein database showed that among these 48 structures 21 are new.

1. Introduction

Combinatorial chemistry has become an important incentive tool for the generation of new structures.[1] Millions of molecules can be made in a matter of months with the hope to discover new leads. Nevertheless these approaches generate new structures by combinations of substituents, the core of the structures being known: thiazole, amides, amines, etc. The novelty originates from the way to assemble all the possible substituents owing to the automation of the process.

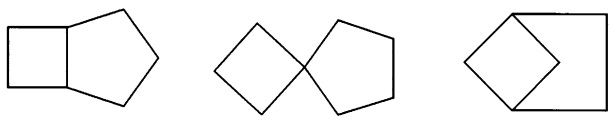
In the field of computer chemistry, programs have been written which generate libraries of virtual structures by a method similar to combinatorial chemistry.[2] These structures can then be evaluated to select those which seem interesting.[3] Works have also been developed in the field of de novo generation of molecules.[4] Programs have been written which generate isomers of a given gross formula[5] and, in a related field, simulation of reactions have been developed.[6] Recently G. Brinkmann et al. published a program which generates polycyclic chains with arbitrary ring size.[7] This prompted us to publish our work in this field.

We designed a new approach for the generation of structures. The aim of this program is to answer the following problem: generate all skeletons with a given number of rings whose size is determined. For example: generate all skeletons with three rings, one of size 5 and two of size 6. All rings (fused, bridged or spiro) are generated. In this paper we describe this program called GASP (Génération Automatique de Structures Polycycliques).

2. Program

We developed a simple method to write this program. GASP works like the chemist who draws a ring, then adds another ring, and so on: GASP starts with a ring, i.e. its connectivity table, then it adds the second ring in all possible positions, then, on each generated structure it adds the third ring, and so on. An example will be used to show the different steps of the program: the generation of structures with 3 rings of size 4,5,6.

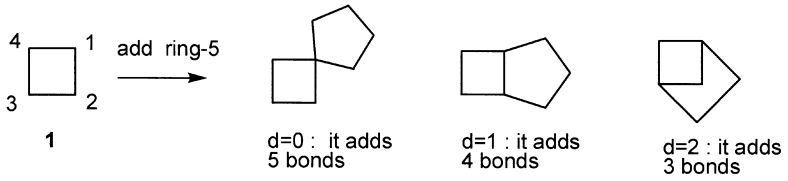
The first step is to calculate the set of combinations, i.e. the order in which the rings will be generated. GASP can start with a ring of size 4 (in the paper sentence like 'ring of size 4' will be summarised by ring-4), then it generates the ring-5 and then the ring-6, but other combinations are possible: 4,6,5—5,4,6—5,6,4—6,4,5 and 6,5,4. However some combinations lead to identical structures: if the program starts with a ring-4 then it adds the ring-5 or the contrary (ring-5 then ring-4) it will generate the same structures (Scheme 1).



Scheme 1.

Keywords: polycyclic structures; combinatorial chemistry; isomers; structure generator; steroids.
* Corresponding author. Fax: +33-4-91-28-82-34; e-mail: rene.barone@am3.u-3mrs.fr

0040–4020/01/$ - see front matter
PII: S0040-4020(01)00562-2



Scheme 2.

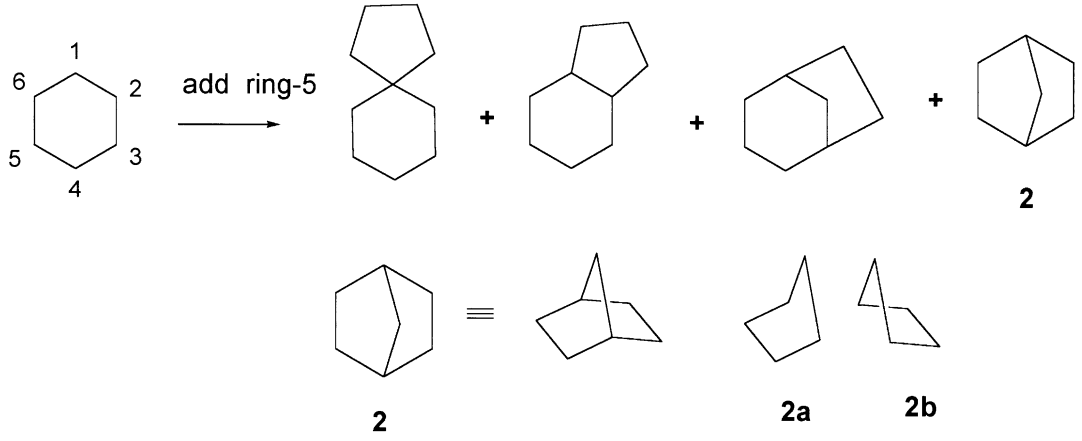
So the number of combinations is divided by two, and, for this example, only combinations 4,5,6—4,6,5 and 5,6,4 will be generated.

The second problem is the construction of the structures, i.e. their connectivity tables. Starting with a given ring, the program has to add a new ring. To do that it determines the set of atom pairs in the structure. For example for a ring-4 (Scheme 2).

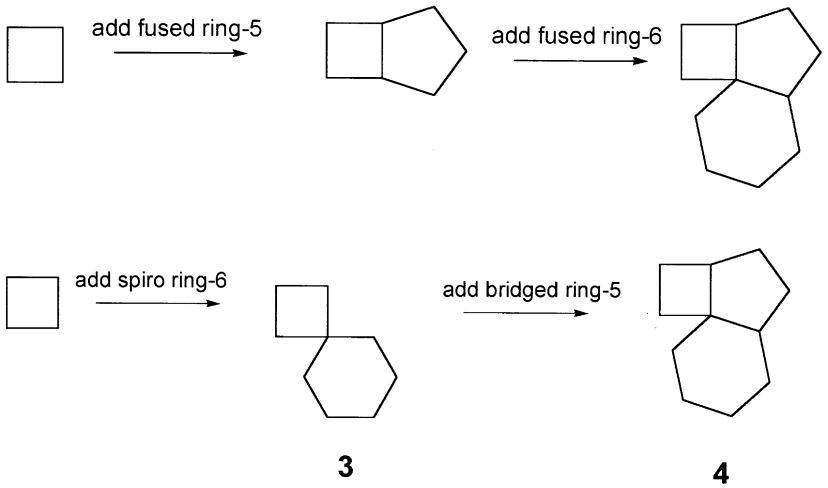
The pairs are: (1,1), (1,2), (1,3), (1,4), (2,2), (2,3), (2,4), (3,3), (3,4) and (4,4). The distance, i.e. the number of bonds, between two atoms of each pair is calculated. If the distance (d) is equal to 0, the new ring is added on the same atom which generates a spiro structure. If d=1, the structure is fused and when d>1 the structure is bridged. The distance allows to calculate the number of bonds which are necessary to build the new ring. If the program has to generate a ring of size S on a given structure, the number of bonds that it has to add is equal to $S-d$ as it can be seen in Scheme 2.

A particular case can be met: for example we want to generate all structures having two rings, one of size 6 and one of size 5. Having the ring-6, the program generates, in a second step, a ring-5, leading to spiro, fused and bridged structures, as indicated above (Scheme 3).

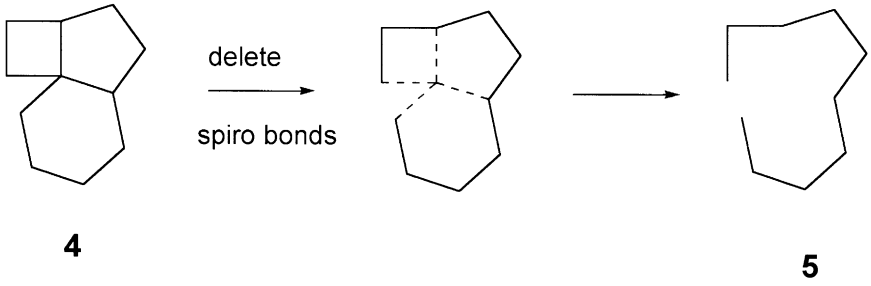
But when it adds two bonds on the pair (1,4) it generates structure **2** (Scheme 3). For bridged structures only the two smallest rings are counted. In structure **2** there are two rings of size 5 (**2a**, **2b**) and the six membered ring is not counted.



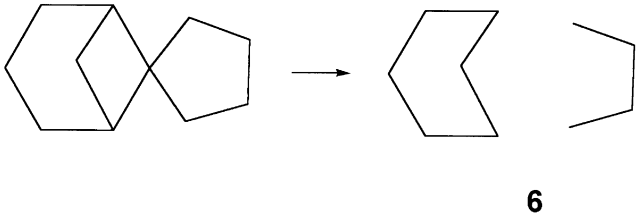
Scheme 3.



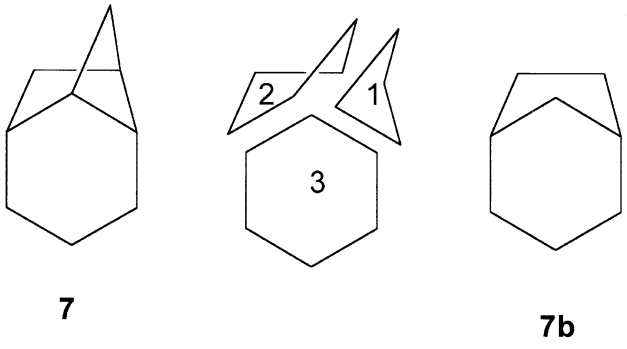
Scheme 4.



Scheme 5.



Scheme 6.



Scheme 7.

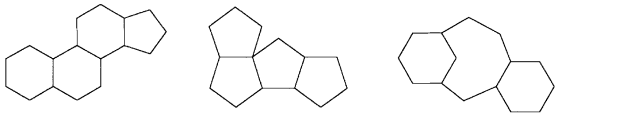
This new structure is not formed by a ring-6 and a ring-5 and does not satisfy to the condition. So it is necessary, for each generated structure, to verify if the wanted rings are present.

It is also necessary to avoid the generation of identical structures. For example when the program adds a new ring, in the spiro case, on structure **1** (Scheme 2), this ring can be added on atoms 1, 2, 3 or 4. But these atoms are equivalent and the generated structures would be identical.

So, each structure is canonized, according to Moreau's algorithm,[8] in order to search for equivalent atoms. When a new

```
Input of the number of rings (NbRing) and their size.
Generation of all combinations (NbComb) (for example : 4,5,6 – 5,6,4, etc.)
FOR I = 1 to NbComb
 Generation of the first ring : partial structure 1 (P.S.1) NbPS = 1
 FOR J = 2 to NbRing
   K=0
   DO
    K=K+1
    calculate all pairs (NbPairs) in P.S.K
    For N=1 TO NbPairs
     On P.S.K add ring J -> generate a  structure.
     Verify if the desired rings are present
     IF J = NbRing then
       A final structure is generated
       IF it is new THEN it is saved on disk
     ELSE
       A Partial Structure is generated
       IF this P.S. is new THEN it is saved (NbPS =NbPS +1)
     END IF
    NEXT N
   LOOP while K<= NbPS
 NEXT  J
 NEXT  I
 Calculate relationships of  each structure
 Display all or selected structures
```

Scheme 8.



Scheme 9.

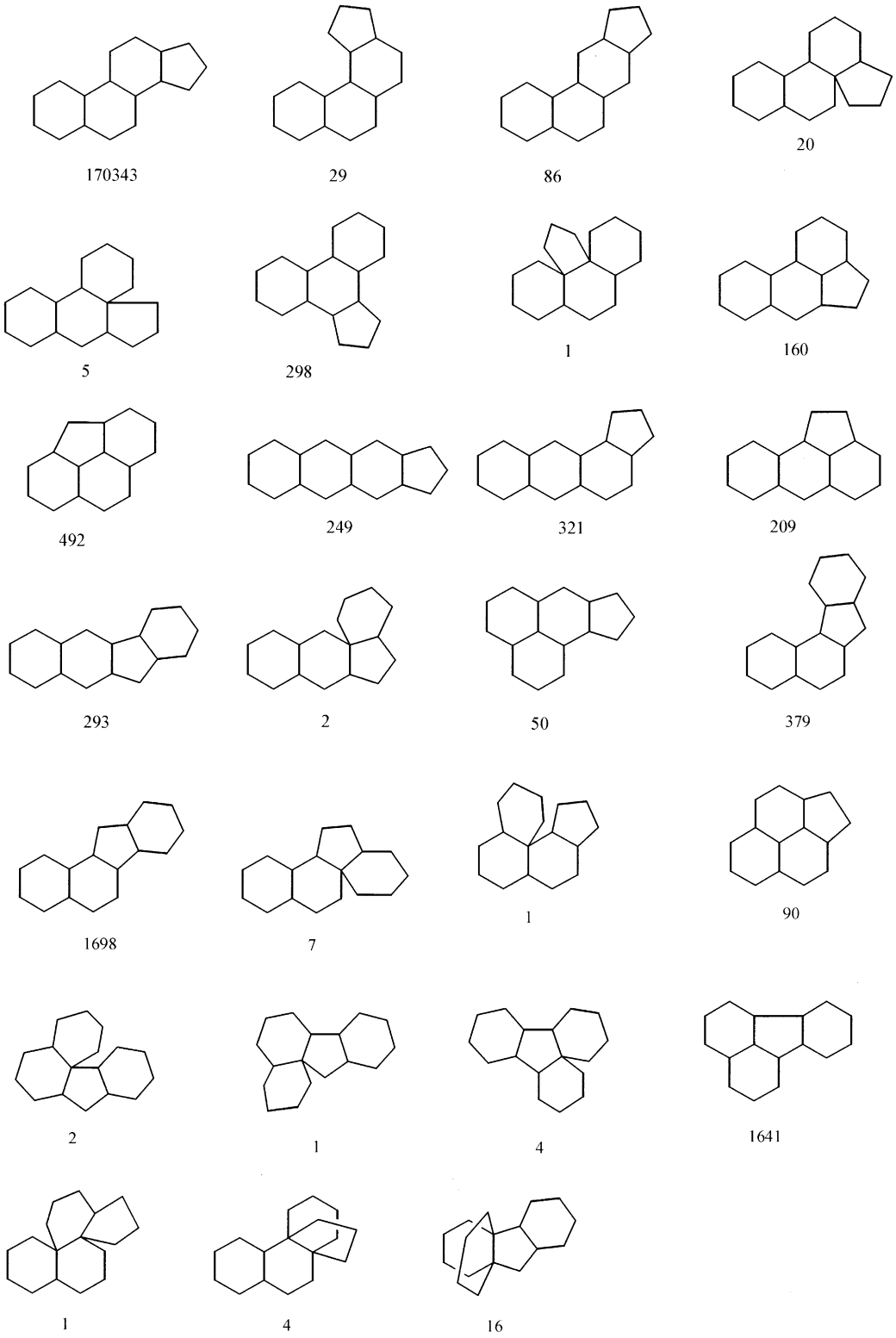


Figure 1. Fused structures found in the Beilstein database for the 6, 6, 6, 5 combination. The number of hits is displayed below each structure.

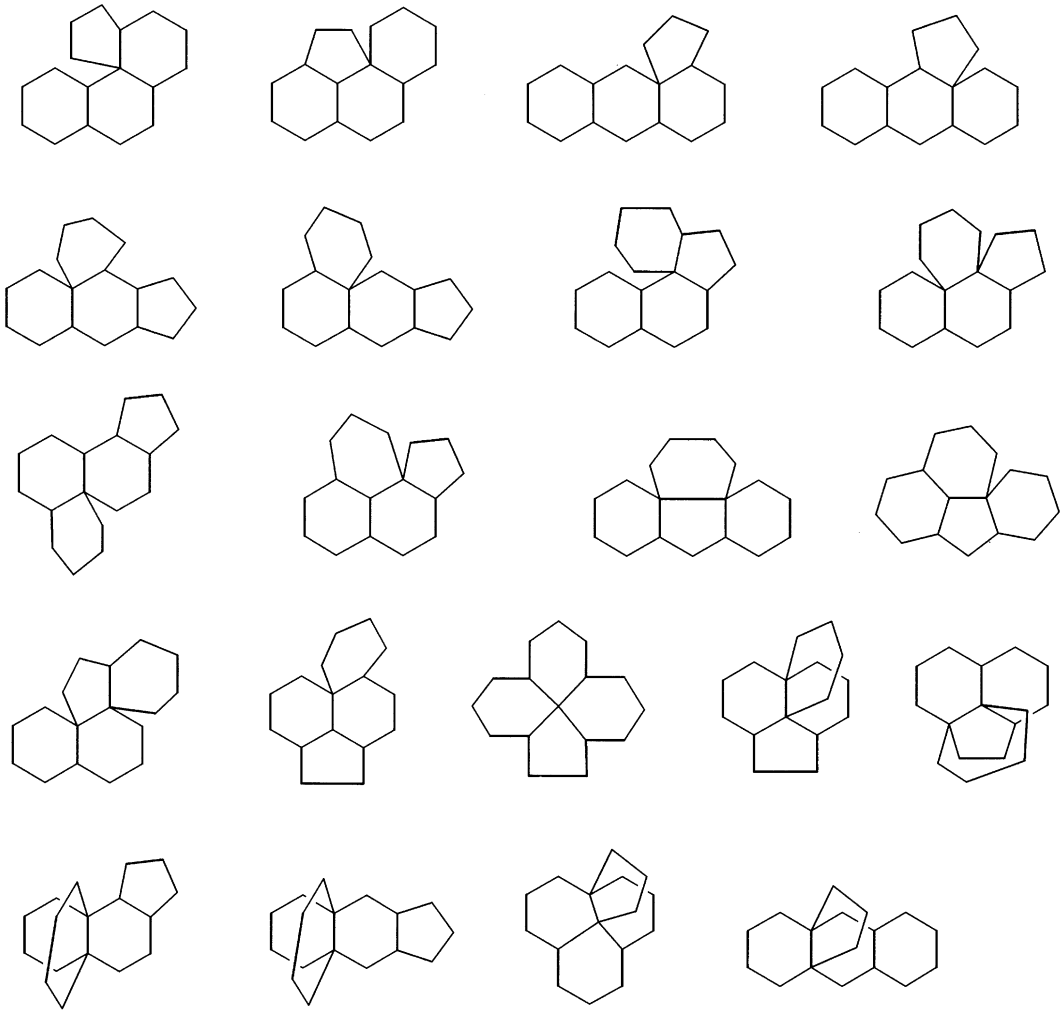


Figure 2. New fused structures generated for the 6, 6, 6, 5 combination.

ring is added on a pair of atoms, GASP uses the equivalency of atoms to discard equivalent pairs: this control reduces the number of generated structures.

Nevertheless identical structures can be generated by different combinations. For example combination (4,5,6) and (4,6,5) generate the same structure (Scheme 4). It is thus, again necessary to verify for each structure if it is actually a new one, by comparing their canonical names.

When all structures have been generated GASP calculates the relationships between the different rings present in each structure, i.e. the number of spiro, fused, bridged. This can be done only at the end of the program: structure **3** is spiro, but the spiro relationship disappears in structure **4** (Scheme 4), in which there are two fused rings. Some problems arise when searching these relationships. They are briefly described below:

When GASP counts the common atoms between the ring-4 and ring-6 in **4**, it finds one atom. To verify if it is a true spiro atom, it deletes the spiro atom and the bonds starting from it, obtaining structure **5** (Scheme 5).

If the final connectivity table is composed of one structure this one is not counted as spiro. If there are two structures such as in **6**, it is a true spiro (Scheme 6).

Some cases are not obvious such as in **7** and necessitate an accurate analysis (Scheme 7).

The first analysis finds the following relationships:

Ring #1 and ring #2: bridged.
Ring #1 and ring #3: fused.
Ring #2 and ring #3: fused.

If the first relationship is true, the two others are not, since ring #1 and ring #2 generate a fourth ring which is bridged with ring #3 (**7b**). For such structures GASP finds two bridges, and no fused rings.

These relationships are stored with the structures and they can be used for the visualisation of the results: the user can see all the structures generated, but an option allows him to see the structures with a given number of fused, spiro or bridges, these numbers can be any, $<$, $>$ or $=$ to a given value. For example he can see the structures where all the rings are fused and having neither spiro atoms nor brigdes.

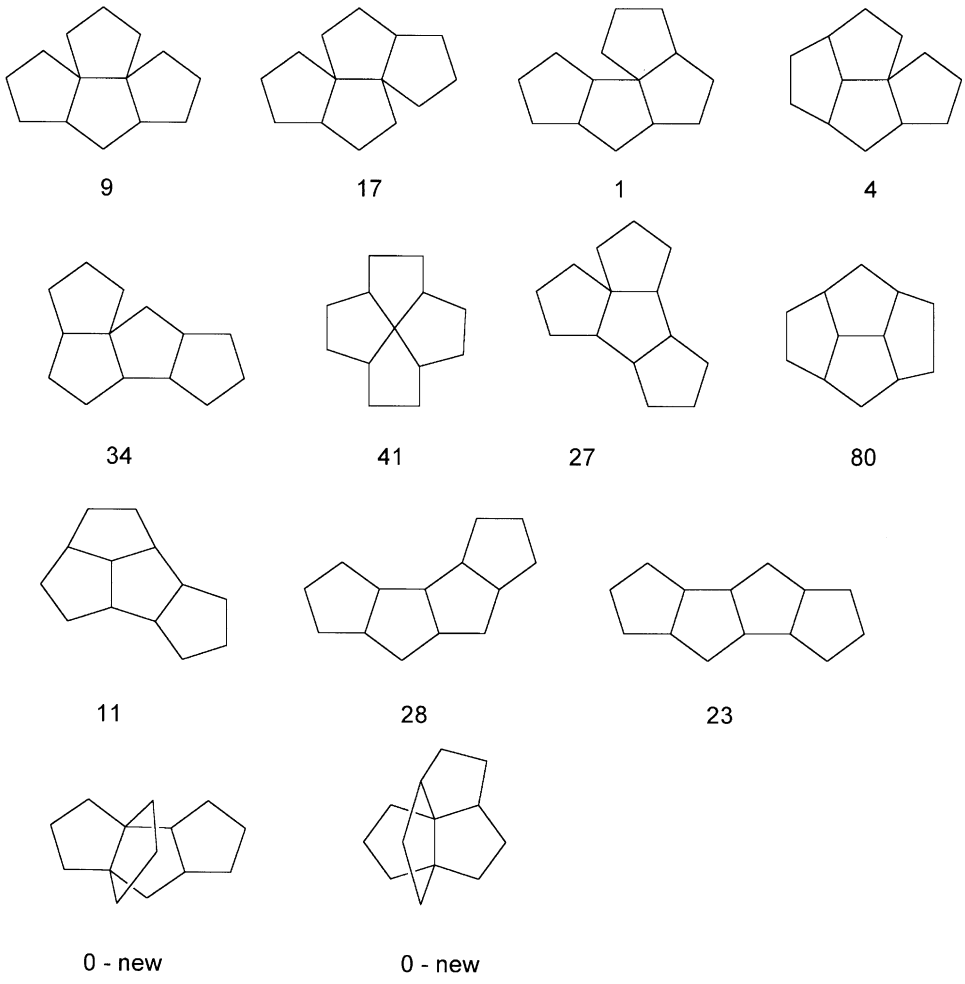


Figure 3. Fused four five membered rings and the number of hits in the Beilstein database.

GASP is written in VisualBasic. It has been tested on a 300 MHz compatible IBM/PC. The time for generating structures with three rings is less than one minute. For four rings the time can reach half an hour. It is due to the number of comparisons to perform the elimination of identical structures and access to the disk. This approach could certainly be improved, but our aim was not to have a very fast but a simple program able to verify our basic idea and to rapidly provide the chemist with a complete answer to the question 'Generate all the structures possessing a given number of rings whose size is determined' in view to suggest him new structures. The flowchart of Scheme 8 summarises the main steps of the program.

3. Results

GASP has been tested on many combinations. We give here the results obtained for the generation of some structures with three and four rings. The first case is composed of four

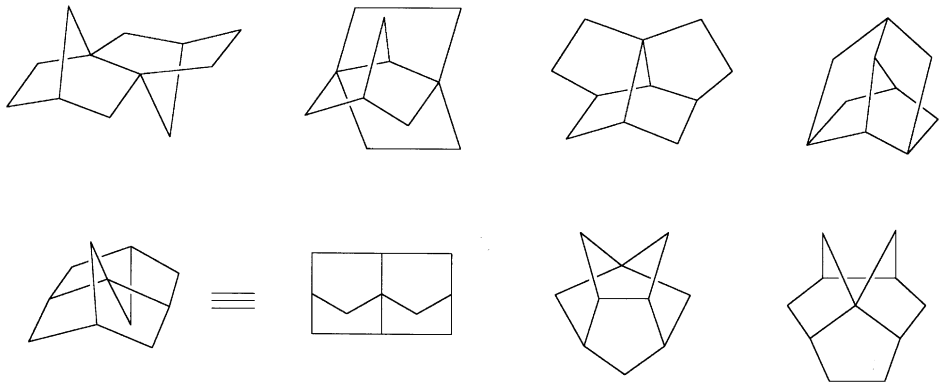


Figure 4. Complex structures with four five membered rings.

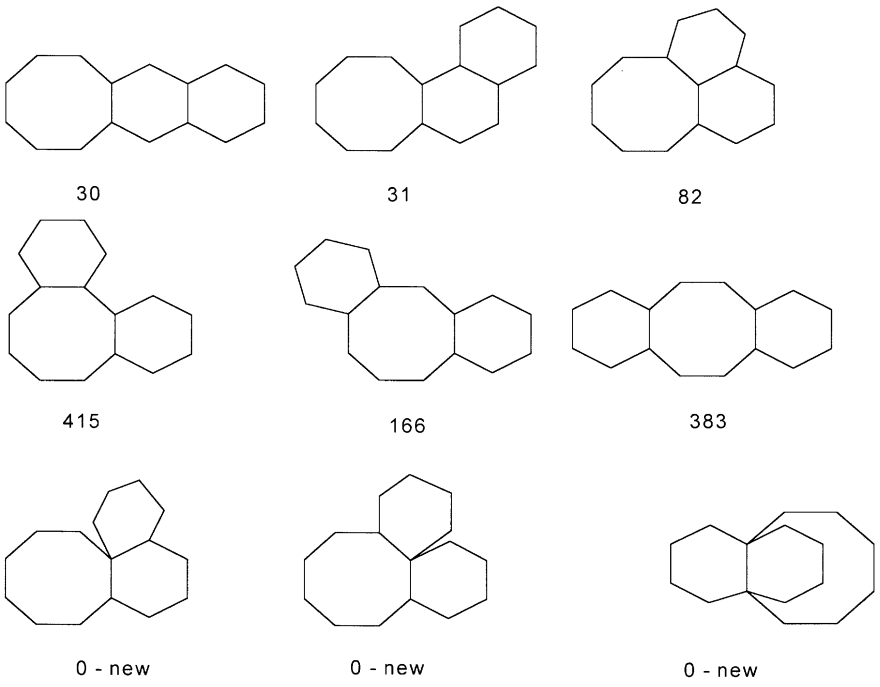


Figure 5. Fused rings of size 6, 6 and 8.

rings, three of size 6 and one of size 5. It corresponds to the steroid skeleton since it is a structure of particular interest. The second case corresponds to a tetraquinane structure with four rings of size 5 which can be found in the crinipellin skeleton and the last case is composed of three rings of size 6, 6 and 8 which can be encountered in the taxol skeleton (Scheme 9).

For the 'steroid' case GASP generated 988 structures in 35 min. Since the most frequent rings in organic chemistry are fused ones, we present the results given by the option which allows to display the desired structures. GASP found 48 structures with fused rings, they are displayed in Figs. 1 and 2. We interrogated the Beilstein database[9] to know if these skeletons have already been synthesised. In the query the bond order was set to 'any' allowing one to search not only skeletons but also structures having double bonds, and all atoms sites were set to free, i.e. they can

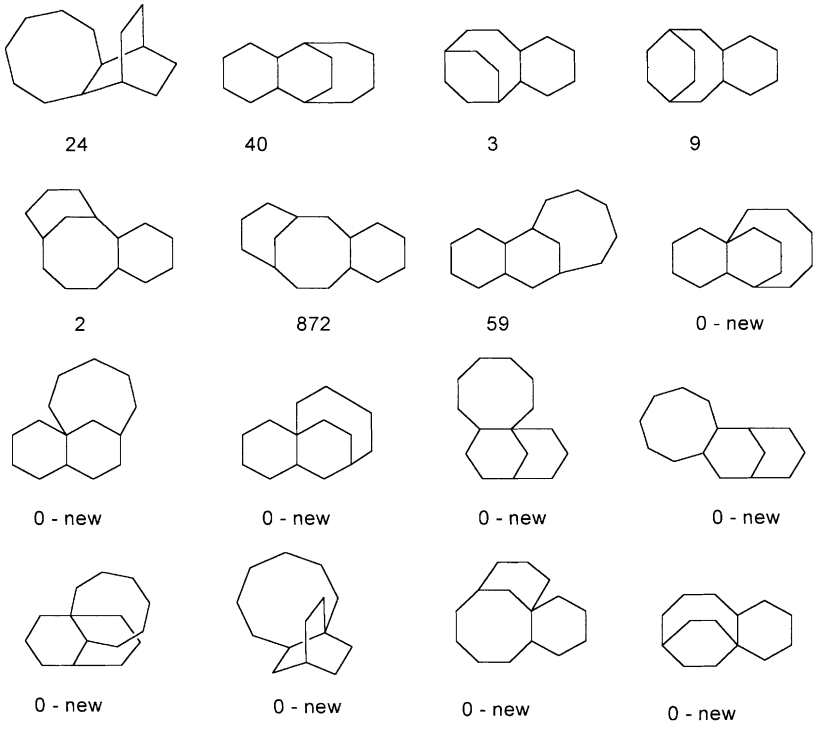


Figure 6. Bridged rings of size 6, 6, 8.

have any substituents. The number of hits found by this search for known structures is displayed in Fig. 1. It is interesting to see that several new structures have been generated (Fig. 2).

For the tetraquinane case GASP generated 161 structures in less than one minute. Among these 161 structures, only 13 skeletons are composed of fused rings. These results are displayed in Fig. 3 with the number of hits found in the Beilstein data base. This case presents less possibilities, nevertheless two structures seem new. We show in Fig. 4 some new complex structures generated in this case.

In the last case (three rings of size 6, 6, 8) 78 structures were generated in less than one minute. We show in Fig. 5 the nine skeletons in which the rings are fused and in Fig. 6 the skeletons with one bridge as in the case of taxol. Several new structures were generated.

These results are encouraging and could give new directions of research for the synthetic chemist.

4. Conclusion

We developed an original program to generate new structures from a simple algorithm which generates structures with a given number of rings of given sizes, for example, the user can ask the computer to generate all structures with 3 rings the size of which are 5, 6 and 8. In the case of the steroid like skeleton (4 rings of size 6,6,6,5) it generates 988 structures. Among them there are 48 structures in which all four rings are fused, and among these 48 skeletons, 21 are new, which is interesting in view of detecting new lead structures. In the infinite world of polycyclic structures, GASP can help the chemist in the recognition of *terra incognita*.

References

1. (a) Dolle, R. E. *J. Combi. Chem.* **2000**, *2*, 383–433. (b) *Applications of Combinatorial Chemistry*. Siegel, M. G., Kaldor, S. W., Eds., *Tetrahedron* **1999**, *55*, 11609–11710. (c) Bunin, B. A. *The Combinatorial Index*; Academic: New York, 1998. (d) Warr, W. A. Combinatorial Chemistry. In *The Encyclopedia of Computational Chemistry*; Schleyer, P. v. R., Allinger, N. L., Clark, T., Gasteiger, J., Kollman, P. A., Schaefer III, H. F., Schreiner, P. R., Eds.; Wiley: Chichester, 1992; pp 407–417.
2. (a) Afferent (http://www.afferent.com). (b) Synopsis (http://www.synopsys.co.uk). (c) CHIC (CHImie Combinatoire). Barone, R.; Chanon, M. Unpublished results. (d) Gugish, R.; Kerber, A.; Laue, R.; Meringer, M.; Weidinger, J. *MATCH* **2000**, *41*, 189–203.
3. (a) Xu, J.; Stevenson, J. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 1177–1187 and references therein. (b) Xue, L.; Godden, J. W.; Bajorath, J. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 1227–1234.
4. (a) Murray, C. W.; Clark, D. E.; Byrne, D. G. *J. Comput.-Aided Mol. Des.* **1995**, *9*, 381–395. (b) Gillett, V. J.; Newell, W.; Matta, P.; Myatt, G.; Sike, S.; Zsoldos, Z.; Johnson, P. A. *J. Chem. Inf. Comput. Sci.* **1994**, *34*, 207–217. (c) Clark, D. E.; Firth, M. A.; Murray *J. Chem. Inf. Comput. Sci.* **1996**, *36*, 137–145. (d) Nilakantan, R.; Bauman, N.; Venkataraghavan, R. *J. Chem. Inf. Comput. Sci.* **1991**, *31*, 527–532. (e) Devillers, J. *J. Chem. Inf. Comput. Sci.* **1996**, *36*, 1061–1066. (f) Chau, P. L.; Dean, P. M. *J. Comput.-Aided Mol. Des.* **1992**, *6*, 385–396. (g) Van Drie, J. H.; Weininger, D.; Martin, Y. C. *J. Comput.-Aided Mol. Des.* **1989**, *3*, 225–251. (h) Moon, J. B.; Howe, W. J. *Proteins: Struct., Funct., Genet.* **1991**, *11*, 314–328. (i) Johnson, P. A.; Green, S. M. de Novo Design Systems. In *The Encyclopedia of Computational Chemistry*; Schleyer, P. v. R., Allinger, N. L., Clark, T., Gasteiger, J., Kollman, P. A., Schaefer III, H. F., Schreiner, P. R., Eds.; Wiley: Chichester, 1992; pp 650–657. (j) Funatsu, K. Structure Generators. In *The Encyclopedia of Computational Chemistry*; Schleyer, P. v. R., Allinger, N. L., Clark, T., Gasteiger, J., Kollman, P. A., Schaefer III, H. F., Schreiner, P. R., Eds.; Wiley: Chichester, 1992; pp 2811–2818.
5. (a) Le Bret, C. *MATCH* **2000**, *41*, 79–97 and references therein. (b) Barone, R.; Barberis, F.; Chanon, M. *MATCH* **1995**, *32*, 19–25 and references therein. (c) Hendrickson, J. B.; Parks, C. A. *J. Chem. Inf. Comput. Sci.* **1991**, *31*, 101–107. (d) Parks, C. A.; Hendrickson, J. B. *J. Chem. Inf. Comput. Sci.* **1991**, *31*, 334–339. (e) Wieland, T.; Kerber, A.; Laue, R. *J. Chem. Inf. Comput. Sci* **1996**, *36*, 413–419. (f) Ivanciuc, O.; Balaban, A. T. Graph Theory in Chemistry. In *The Encyclopedia of Computational Chemistry*; Schleyer, P. v. R., Allinger, N. L., Clark, T., Gasteiger, J., Kollman, P. A., Schaefer III, H. F., Schreiner, P. R., Eds.; Wiley: Chichester, 1992; pp 1169–1190.
6. (a) Bauer, J.; Herges, R.; Fontain, E.; Ugi, I. *Chimia* **1985**, *39*, 43–58. (b) Herges, R. *Angew. Chem., Int. Ed. Engl* **1994**, *33*, 255–276. (c) Hendrickson, J. B. *Recl. Trav. Chim. Pays-Bas* **1992**, *111*, 323–334. (d) Höllering, R.; Gasteiger, J.; Steinhauer, L.; Schulz, K. P.; Herwig, A. *J. Chem. Inf. Comput. Sci.* **2000**, *40*, 482–494.
7. Brinkman, G.; Dobrynin, A. A.; Krause, A. *MATCH* **2000**, *41*, 137–144.
8. Moreau, G. *Nouv. J. Chim.* **1980**, *4*, 17–22.
9. (a) http://www.beilstein.com. (b) *The Beilstein Online Database. Implementation, Content and Retrieval*; Heller, S. R., Ed.; ACS Symposium Series 436, Am. Chem. Soc.: Washington, D.C., 1990.