match no. 14 pp. 235-246 1983

COMPUTER-ASSISTED GENERATION OF MOLECULAR STRUCTURES FROM A GROSS FORMULA. I. ACYCLIC SATURATED COMPOUNDS

Ivan P. Bangov

Institute of Organic Chemistry, Bulgarian Academy of Sciences, Sofia 1113, Bulgaria

(Received: May 1983)

ABSTRACT

An algorithm for computer-assisted generation of molecular structures from a gross formula for acyclic saturated molecules is suggested. Its task is to facilitate, to some extent, the solution of the combinatorial problem. Various options of a program based on this approach are discussed.

Computer-assisted structure generation is an important part of any artificial intelligence program for chemical inference. Several approaches to the solution of this problem were developed.[1-4] A common feature of all of them is that they are based on topological description i.e. the atoms of a chemical structure are vertices and the bonds are edges of a chemical graph, but they differ in their practical realization. For our further investigations most attractive appears the algorithm suggested by Munk and coworkers.[1] It consists of combining of numbers representing bonding sites (half-bonds) of a series of structural fragments. The connectivity within a structure is represented by a pair of numbers. Using the chemical terminology the bonding site may be considered as a free valence of the fragment, and the pair of numbers as a chemical bond.

This approach provides a general scheme which allows generating different classes of structures (acyclic,cyclic, saturated,unsaturated). Two severe problems arise, however if the fragments are reduced to single atoms: First, the practical implementation of such an algorithm requires generating all of the N_V! permutations of the N_V valences of the free atoms; restrictions built in the program ensure the generation of chemically reasonable structures. Since the number of permutations sharply increases with the increase of the number of valences, the combinatorial problem becomes a wasteful one. Second, the number of duplicated structures sharply increases along with the increase of the number of permutations, their discrimination being difficult. Thus,for larger molecules the combinatorial problem becomes unmanageable.

In this paper an alternative algorithm is reported, whose main task is to facilitate, to some extent, the solution of the combinatorial problem. This algorithm was developed for acyclic saturated structures, but its generalization to other classes of structures is in progress. It is based on the following strategy:

The starting point of the process of structure generation is the gross formula. We consider the notion of gross formula in a wider sense than the molecular formula, e.g. such a formula can be constituted not only of single atoms, but of groups such as CH_3,OCH_3,OH,Ph etc., if information for their presence is available. This allows some unneeded structures to be a priori eliminated. If, for example, it is known that the only oxygen in the molecular formula C_3H_8O is a hydroxyl group oxygen, we can input it in the form of gross formula C3OH1H7, thus only hydroxy-group containing structures are generated, and all ether structures are a priori eliminated.

Only skeleton atoms (C,O,N,S), univalent heteroatoms as Br,Cl,F,etc. as well as some functional groups (mentioned above), considered in the same way as in Ref. 2 as "superatoms" are involved in the combinatorial problem, i.e. the hydrogen atoms are excluded from the process of permutation generation.

Each structure of a saturated acyclic compound might be considered as a methane substituted by one, two,three or four substituents which are in fact either some of the skeleton atoms, or the heteroatoms and groups mentioned above, by filling the unoccupied valences with hydrogen atoms.Those substituents are substituted then with some of the remaining atoms and groups, which are also substituted etc. till they exhaust the atoms and groups available in the gross formula. Thus, the structure is built as a tree-like graph, the first carbon atom being the root, the other skeleton atoms are the branches and the univalent atoms such as hy - drogen, bromine and chlorine atoms, as well as OH, CH_3, Ph and OCH_3 groups are the leaves of the tree. It should be noted that such a depiction does not provide a direct view on the algorithm itself. In fact the generation of structures is a purely combinatorial problem, but an appropriate representation of the connectivity ensures the generation of the tree described above.

The algorithm is implemented in program STRGEN(STRucture GENerator) written in BASIC for a Hewlett-Packard 9845 B computer. Since the computer's speed is low, only small and medium size molecules can be manipulated. Translation of the program into PL1 for larger and faster computers is in progress. The program is based on the following stepwise procedure (see Fig. 1A and Fig. 1B):

Step 1: The input gross formula is transformed into two string arrays: vector array Subs$ and two row-matrix array Graph$. Subs$ consists of all of the atoms or groups indicated in the gross formula, except the first carbon atom, each of them taken once. The first row of the Graph$ matrix (Graph$(1)) is constructed by the same atoms but taken n-1 times, where n is the atom valence (n=4 for carbon atom, n=3 for nitrogen, n=2 for oxygen etc.).Evidently, univalent heteroatoms such as F, Br, Cl and groups as OH, CH_3, NH_2 etc. will not appear in Graph$(1). The only atom taken n times is the first carbon atom. It should be mentioned here that such an allocation of atoms and groups in the Graph$ and Subs$ arrays is conducive to generation of acyclic saturated structures only. In the further development of the program, now in progress, the index of unsaturation (Eqv) is computed from the gross formula, and according to its value the process of structure generation is directed either to acyclic saturated (Eqv=0) or to one-ring saturated and one-double bond unsaturated (Eqv=1) etc. structures.The filling of the Graph$(1) entries with atoms follows the order they appear in the gross formula. The second row Graph$ (2) is filled with hydrogen atoms (see Fig. 1A and Fig. 1B) in this step.Since the connectivity is represented by matching the two rows, Graph$(1) and Graph$(2), in this step we have the first carbon bonded to four hydrogens, representing the methane molecule, and the other skeleton atoms, n-1 times bonded to hydrogens, being radicals which are the potential substituents of the methane molecule as it was described above.

Step 2: The different transpositions of the substituent atoms are carried out by generating $^{m}P_{N}$ permutations of m elements, selected from N elements without repetition, where N is the number of hydrogen atoms, m of them being substituted, i.e. N is the dimension of Graph$(2) row, and m is the number of the atoms in the Subs$ array. Each permutation number indicates the entry in the row Graph$(2) whose hydrogen atoms will be substituted with

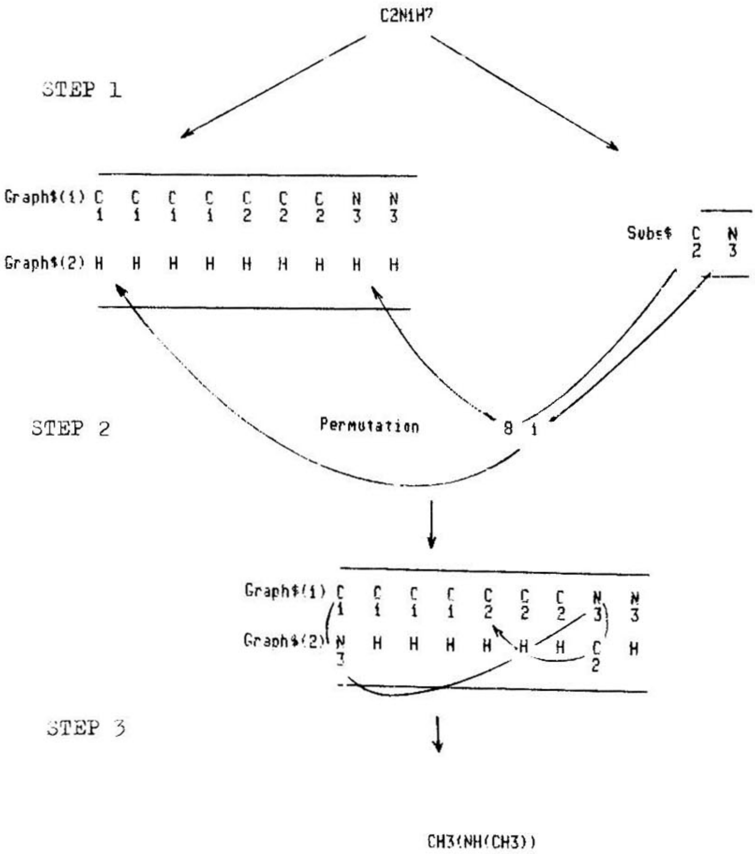

C2N1H7
STEP 1
Graph$(1) C C C C C C C N N
1 1 1 1 2 2 2 3 3
Graph$(2) H H H H H H H H H
Subs$ C N
2 3
STEP 2
Permutation
8 1
Graph$(1) C C C C C C C N N
1 1 1 1 2 2 2 3 3
Graph$(2) N H H H H H H C H
3 2
STEP 3
CH3(NH(CH3))

FIGURE 1A. Structure generation of dimethylamine. The numbers below the atoms indicate the atom numbering The curved arrows in STEP 1 and STEP 2 show the mode of hydrogen atom substitution. The curved arrows in STEP 3 represent the connectivity between the nonhydrogen atoms.

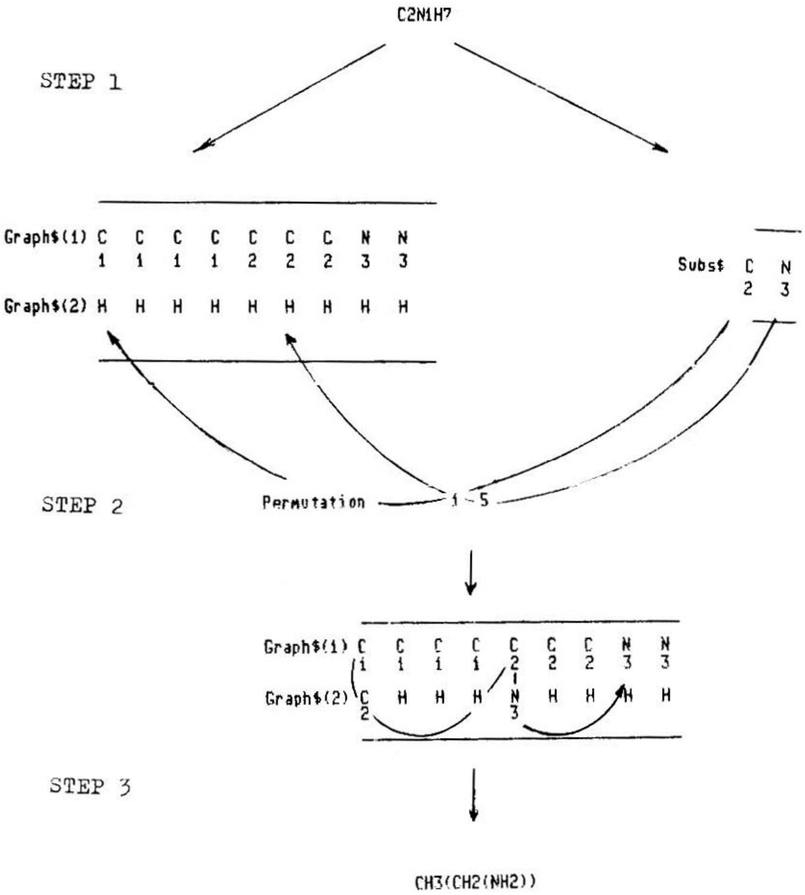

C2N1H7
STEP 1
Graph$(1) C C C C C C C N N
1 1 1 1 2 2 2 3 3
Graph$(2) H H H H H H H H H
Subs$ C N
2 3
STEP 2
Permutation 1 - 5
Graph$(1) C C C C C C C N N
1 1 1 1 2 2 2 3 3
Graph$(2) C H H H N H H H H
2 3
STEP 3
CH3(CH2(NH2))

FIGURE 1B. Structure generation of ethylamine.

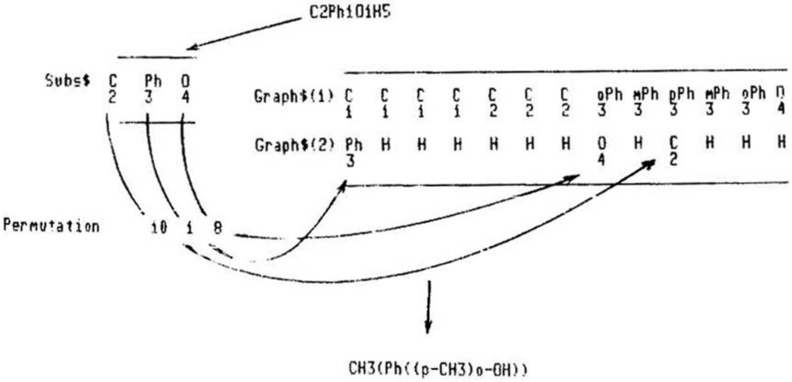
atoms from the Subs$ array, taken in the order they appear in the latter. For example in the case presented in Fig. 1A the permutation (8 1) indicates that atoms C2 and N3 have to substitute hydrogens 8 and 1 in Graph$(2). As it was stated above, the connectivity is formed by juxtaposing the two Graph$ array rows. The solid lines in Fig. 1A show the skeletal branching of the tree-like structure. It is obvious that instead of $N_V!$ permutations (N_V being the number of free valences of the atoms) only ${}^{m}P_{N} = \frac{N!}{(N-m)!}$ permutations are generated in the present approach. In effect, their number is even less, because of the proviso that no more than three hydrogens can be attached to the first atom, otherwise the latter will be cut off from the molecule.Since the ${}^{m}P_{N}$ permutations are generated in lexicographical order, a control is built in the program which ensures its stopping if all of the numbers of the current permutation are greater than four i.e. if there is no more atoms or groups attached to the first carbon. A second, very important control is built ensuring that no atom is bonded to itself, thus some of the permutations are additionally ruled out.

The generation of ${}^{m}P_{N}$ permutations of m elements selected from N elements is realized by generating the ${}^{m}C_{N}$ combinations of N elements taken m at a time in lexicographical order by means of Mifsud's algorithm[5] and for every combination m! permutations of the m digits of the current combination are generated by means of the Shen's algorithm.[6]

Step 3: Every structure whose connectivity is represented by juxtaposing rows Graph$(1) and Graph$(2) is transformed into linear notation form, which in some degree is similar to that reported in Ref. 3. However, in our program the different levels of branching the tree (substitution in the molecular structure) are indicated by enclosing them in round brackets. On the other hand, the brackets represent the structural fragment valences: the left bracket is a free valence, and the right bracket is a saturated valence.

Here, as in Refs. 3,4,8 and 9, the branching obeys a hierarchical order. By contrast with the latter algorithms, in our program the order is determined by the order in which the atoms appear in the gross formula. In the course of analysing the tree-like structure and transforming it into linear notation form, the priority goes to the branching which starts with an atom first appearing in the gross formula. Such a constraint is conducive to elimination of most but not all of the duplicated structures. The current structure is checked whether it coincides with any of the structures previously generated. It is stored in a string array and printed if no structure alike is found. The transformation into linear notation form is carried out by means of "search with backtrack" algorithm.[7] This procedure is the most time-and memory-consuming part of the program.

Although aromatic constitution is not discussed in this article, the phenyl group might be also included among the groups, because its generation in the program is analogous to the other single atoms. It is considered as a "superatom" Ph with valence n=6. This abbriviation is at variance with the usual one which considers Ph=C_6H_5 without any other substituent. In our case Ph=C_6H_5 but some of the hydrogens can be substituted; they are counted, however, in the gross formula. In the same way as for the other atoms, the Graph$ and Subs$ arrays are constructed (they are presented in Fig. 2), Ph group appearing once in Subs$ and five times in the first row of the Graph$ array. However, as one sees from Fig. 2, in contrast with the other atoms, all of the five phenyl valences are naturally labeled. Those are the two <u>ortho</u>, two <u>meta</u>, and one <u>para</u>-positions.



```
Subs$  C  Ph  O
       2  3   4

Graph$(1) C  C  C  C  C  C  C  oPh mPh pPh mPh oPh O
          1  1  1  1  2  2  2  3   3   3   3   3   4
Graph$(2) Ph H  H  H  H  H  H  O   H   C   H   H   H
          3                    4       2
```

FIGURE 2. Structure generation including a substituted benzene group(2,5-dimethylphenol).

There is an additional option for including arbitraryly named structural fragments, considered also as "superatoms". Those fragments are parts of the structure with constitution known to the user. The only additional input information is the free valence of the "superatom". Thus, those fragments are manipulated in the process of the permutation generating in the same way as the other atoms. Since the single atoms forming their inner structure do not participate in the permutation generation, inclusion of such fragments facilitates the combinatorial problem.

References

1. D.B.Nelson,M.E.Munk,K.B.Gash and D.L.Herald,Jr.,J.Org. Chem.,34,3800(1969).
2. R.E.Carhart,D.H.Smith,H.Brown and C.Djerassi, J.Am.Chem. Soc.,97,5755(1975).
3. L.M.Masinter,N.S.Sridharan,J.Liderberg and D.H.Smith, J.Am.Chem.Soc.,96,7714(1974).
4. S.Sasaki,I.Fujiwara and H.Abe, Analyt.Chim.Acta,122,87 (1980).
5. C.J.Mifsud, Algorithm 154 from "Collected algorithms from CACM".
6. M.K.Shen, Algorithm 202 from "Collected algorithms from CACM".
7. E.M.Reingold,J.Nievergelt and N.Deo,"Combinatorial algorithms.Theory and practice","MIR",Moscow 1980 p.124.

8. Y.Kudo and S.I.Sasaki,J.Chem.Inf.Comp.Sci.,16,43(1976).

9. H.L.Morgan, J.Chem.Doc.,5,107(1965); IUPAC "Rules for IUPAC Notation for Organic compounds",Wiley,New York and Longmans,London,1961; E.G.Smith,"The Wiswesser Line-Formula Chemical Notation",McGraw-Hill,New York,1968.