

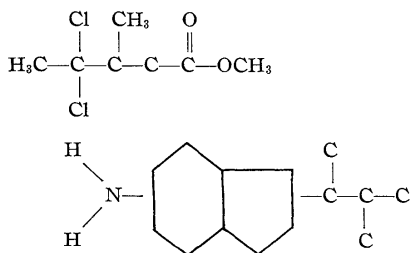
A Topological Application of Computing Machines

ASCHER OPLER

THE use of computing machines in the solution of problems arising in analysis and geometry is undergoing active exploration at the present time. These machines have been used for logical operations and have proved of considerable value. This report will describe the application of digital computers to an elementary problem in combinatorial topology.

The treatment given here will probably be of little interest to those interested in theory, but will prove to have considerable practical application. It developed as a by-product of an ostensibly unrelated research into the cataloguing of organic chemicals.

Organic chemicals, such as vitamins, insecticides, synthetic hormones, etc., are actually aggregates of complex molecules. To the organic chemist, they are more simply represented as single structures composed of atoms and bonds. The synthesis, the properties and the reactions of these compounds may be understood largely in terms of these structural diagrams. Thus, when the organic chemist wants a catalogue of thousands of chemicals, he visualizes collections of structures like:



There are over half a million of these structures known, and the list is growing at a rapid and ever increasing rate. Numerous questions arise in the daily work of chemists that make constant recourse to this catalogue a necessity. For instance: Has a certain compound ever been made before? What compounds have certain substructures in common? The members of a group have certain properties in common; what in the structures might be responsible?

Attempts have been made to devise

mechanical aids to search these catalogues but have relied, in whole or in part, on chemical characteristics. Working with Dr. T. R. Norton, an organic chemist, the writer has attempted to solve the cataloguing problem by reducing it to one of topology alone and then solving it in this form.

This has now been tested and proven feasible using digital computing machines. The method of transforming the chemical structures to topological networks will not be described here; only the solution of the corresponding topological problem will be covered.

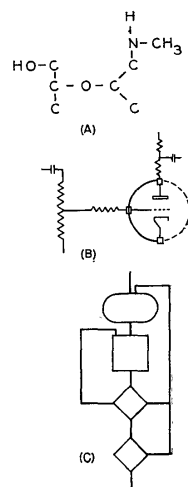
The problem treated here concerns networks composed of members of various types containing a number of positions of potential attachment, some or all of which are used in the assembly of the network. The problem is to represent the topology of the network with a digital code and to search among collections of such codes to locate the networks which have given topological features. As a means of visualization, consider structures made from an Erector set constructed of flexible beams.

Practical uses, aside from chemical cataloguing, should be numerous. As an example close to the computing-machine field, the flow diagrams used to characterize computer programs fall into the category described. Computer charts may be reduced to codes and stored for later recovery. In stored form, they may be searched for common features and help the development of automatic coding and compiling systems. For those employed in electronic design of computers, the circuit diagrams also fall into this category, and the amusing prospect of computers storing their own circuit diagrams may be considered. With the storage of numerous electronic layouts, the search for common features should assist in the mechanization of production through more judicious selection of common subcircuits for automatic production of packaged plug-in assemblies. A third use in the computer field arises in the representation of connection patterns for analogue computers of the electronic differential analyzer type.

In general, the practical use of this method is indicated where large files of

networks with elements used repeatedly in various arrangements are encountered. The extension to switching circuitry and communication networks is called for. Still another use might be in traffic engineering, where flow of vehicles and pedestrians can be catalogued and studied. In Fig. 1, a collection of familiar diagrams that are reducible to topological networks is shown.

Fig. 1 Typical arrangements which may be represented topologically



Elements of the System

MEMBERS

These comprise a finite set of specified types of objects representable as linear figures. For purposes of this representation, many branched and rounded figures may be converted to linear form. The choice of types to form the set of members will depend on the nature of the networks and the relative frequency of occurrence of objects of different types. In some cases, members will be only the most primitive elements; in others, simple combinations occurring more frequently may serve.

POSITIONS

Every member must have a finite number of points of potential attachment to other members. These need not be equally spaced, but should be consecutively numbered from end to end. Numbering may be arbitrary or based on some generally accepted system.

NODES

The conjunction of single positions in each of two different members is termed a node.

SUBSTRUCTURES

An ensemble of two or more connected

ASCHER OPLER is with the Western Division of The Dow Chemical Company, Pittsburg, Calif.

representing any member-node combination. If these limits are 32, 32, and 1,024 respectively, a 35 binary digit number may be used.

In using these representations with conventional fixed-size word computers, the following has been found to be a useful order. Each structure makes up a block of computer words. Within each block, every member-node designation occupies one word. Each word then is divided into six parts with an arbitrary assignment of groups of digits to represent the required designations. In addition to these words, a first word identifies the block and gives the total number of words in the block. In our work, a second word is used as an internal redundancy check based on chemical principles. Thus, a typical structure appears as a block of $n + 2$ words where n is the number of members.

Details of Trials with Digital Computers

Earliest computer trials were made using the Datatron (Electrodata Corporation, Pasadena, Calif.). Here the word form was somewhat simplified and allowed 10-digit decimal words to describe the structures. These tests revealed the shortcomings of the original program and indicated that magnetic drum computers would be rather slow for the application, since nearly 3,500 logical commands were required to examine each 8-member structure studied. On the basis of these tests, the program was revised and improved, and a binary fast-access machine decided upon for further work.

The 701 Electronic Data Processing Machine of the New York International Business Machines Corporation (IBM) Data Processing Center was used for a group of more successful trials of the method. It may be of some interest to

note that the programming, coding, debugging, and running of the program were all done through correspondence and wire communication. The results and methods were adjudged highly satisfactory.

The structures were all of the open type and were represented by blocks of ten full (35 - bit + sign) words, since they all had eight members. Nearly 200 of these blocks occupied the major part of the cathode-ray tube memory of the machine. In the testing

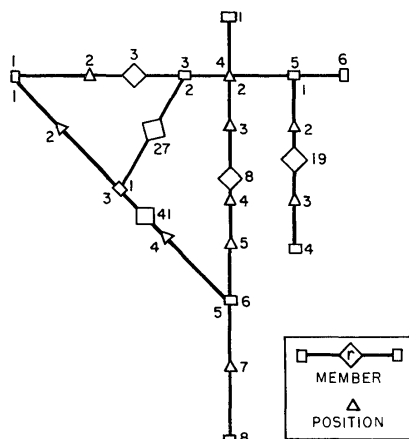


Fig. 3. Example of representation of a closed structure

$$\begin{aligned}
 & /m_1^3/ \\
 & /(\mathbf{p}_{1,4}-\mathbf{n}-\mathbf{p}_{2,2}) m_2^8/ \\
 & /(\mathbf{p}_{1,5}-\mathbf{n}-\mathbf{p}_{3,1}) m_3^{19}/ \\
 & /(\mathbf{p}_{1,1}-\mathbf{n}-\mathbf{p}_{4,1}) m_4^{41}(\mathbf{p}_{4,5}-\mathbf{n}-\mathbf{p}_{2,6})/ \\
 & /(\mathbf{p}_{4,3}-\mathbf{n}-\mathbf{p}_{5,1}) m_5^{27}(\mathbf{p}_{5,2}-\mathbf{n}-\mathbf{p}_{1,3})/
 \end{aligned}$$

stage, each structure was contained on a single binary card along with the necessary input, redundancy and locating words. (Later work will enter words stored in large "super blocks" on magnetic tape.) After the structures were loaded, the program described below was fed to the machine which was now in a

ready status for searching for subnetworks. Each search was completely described in a single "criterion card" which was fed to the machine. Complete examination of nearly 200 blocks required 1 to 3 seconds before the printer listed the structure identities that met the criterion. Since neither the program nor the structure representations were destroyed, after this examination, the 701 was ready for more searches.

The program used in searching was designed to operate at the highest speed possible. Therefore, it called for the block-by-block examination of the structures. As soon as the first nonmatch occurred, the block was instantly rejected and the next block brought into operating position. Since there were a number of match tests, this shortened the time considerably. If a block remained in place and survived all match tests, the corresponding identity word was transferred to a block reserved for printing identities of successful structures.

Because of the latitude in the sequence of member selection, all matches had to be tried in all possible positions before nonmatch could be reported. Furthermore, since there could be considerable multiplicity, each criterion had to be tested against all nodes of a given type. Thus, if we require a member of type 13 to connect to a member 17 and there are two 13's and four 17's, we must make eight tests before we can definitely report no 13-to-17 attachment. The 701 program allowed for numerous types of multiplicity and for substructures of considerable complexity.

At the present time, work is starting to reprogram the task for a 704 computer, operating on approximately 15,000 structures stored on tapes. This program will allow for even greater complexity than the 701 program and will also handle closed structures.