Reprinted from FORMAL REPRESENTATION of HUMAN JUDGMENT, Benjamin Kleinmuntz (ed.), 1968. Copyright © 1968 by John Wiley & Sons, Inc.

CHAPTER 7

Joshua Lederberg
and Edward A. Feigenbaum
Stanford University

MECHANIZATION OF INDUCTIVE INFERENCE IN ORGANIC CHEMISTRY*

INTRODUCTION

A paradigm of the scientific method is the successive alternation of reference to hypothesis and datum. We start with a hypothesis, which moves us to select some aspect of the real world for empirical enquiry. Sensory impressions are translated by convention into data. More refined hypotheses are then *induced* by a poorly understood process which contains at least two elements: (1) the data somehow suggest a hypothesis, and (2) deductive algorithms are applied to the hypothesis to make logically necessary predictions; these are then matched with the data in a search for contradictions. A hypothesis is regarded as inductively proven (i.e., we have achieved a scientific discovery) when its predictions are satisfied, and when we have the illusion of inductive exhaustion that no other hypothesis will lead to equally concordant predictions. It is rare for inductive exhaustion to be rigorously justified. Usually, the process is reiterated many times: each refinement of hypothesis suggests the examination of new data; each new datum leads to a discrimination among existing hypotheses, or suggests another refinement.

Our main contribution is, perhaps, the suggestion that organic chemistry is an apt field for the mechanization of the process of scientific induction. It can be circumscribed so that the first studies are simplified without undue loss of

**The research reported here was supported in part by the Advanced Research Projects Agency of the Office of the Secretary of Defense (SD-183), and in part by the National Aeronautics and Space Administration (NsG 81-60).*

utility or generality. Real data of any desired level of complexity can be adduced. Few natural sciences are so rich in inductive analysis from information that can be presented in a simple, uniform format. By contrast, genetics or embryology are sciences that might invoke models of most of external reality. Above all, the hypotheses of organic chemistry can be abstractly represented, that is, as structure diagrams. These lead, in turn, to an algebra for inductive exhaustion rarely available in any other scientific field at the present time.

As will be seen, our program (named DENDRAL) is firmly rooted in this algebra which can generate an exhaustive and irredundant list of hypotheses from initial contextual data (1,2,3,4). The problem of induction is then reduced to efficient selection from this prospective list. This is only feasible if the experimental data are recurrently consulted to guide the hypothesis-generator. Unproductive branches of the generation tree are anticipated and avoided as soon as possible; conversely, the data are used for heuristic reordering of the priorities with which the hypotheses are brought up for examination. The program thus simulates a systematic idealization more than it does the haphazard evocation of new concepts in human intelligence.

HEURISTIC DENDRAL: A SUMMARY FROM THE STANDPOINT OF ARTIFICIAL INTELLIGENCE RESEARCH

The intent of this section is to present in a succinct and compact fashion information relevant to a general understanding of what our program does and how it does it.

Motivation and Task Environment

We have been interested in exploring processes of empirical inquiry, particularly discovery processes involved in searching a hypothesis space for hypotheses meaningful and relevant to the explanations of real-world data. Some practical considerations concerning the automation of routine scientific endeavor in particular environments, using heuristic programming techniques, also supplied some of the motivation. We were interested also in exploring man-machine interaction in the context of scientific problem solving, not only as an

augmentation to human problem solving processes ("smart scratch paper") but also as a means for "educating" a suitably receptive program, by making it easy for a human skilled in the task area to impart to the program his heuristic search rules and other information relevant to good performance in the task.

The task area chosen was the analysis of mass spectra of organic molecules. The hypotheses relevant to explaining mass spectral data in organic chemical analysis are essentially graphs--molecular graphs consisting of atoms and bonds, such as are seen in textbooks on organic chemistry.

The main tasks presented to the program at present are:

1. Given a chemical (compositional) formula, output a list of the chemically most plausible isomers (structural variants) of the composition, ordered from most plausible through least plausible if that is requested.

2. Given a mass spectrum and a composition, output a list of the most plausible isomers of the composition in the light of the spectral data given. Restated, generate a hypothesis or list of hypotheses to best explain some given spectral data.

Processes, Algorithmic and Heuristic

The program that solves these problems is called Heuristic Dendral. It is a LISP program of some 30-40 thousand words, developed on the SDC Q-32 time-sharing system and presently operating on the PDP-6 at the Stanford Artificial Intelligence Project. Though the program consists of many functions, the most important activities can be summarized as follows:

1. At the most basic level, there is an algorithm, called the Dendral Algorithm, rarely exercised without constraints. Given a chemical composition, it will generate all of the topologically possible noncyclical connected graphs that can be made from the atoms of the composition, given the valences of these atoms. Associated with the Dendral Algorithm is a notation for these graph structures, called Dendral notation. Canonical forms of the graph structures in Dendral notation exist and are used. The Dendral Algorithm is a systematic

and exhaustive "topologist" and knows nothing about chemistry, except the valences of atoms. But, using a chess analogy, it is the "legal move generator," the ultimate guarantor of the completeness of the hypothesis space.

2. Heuristic processes control and limit the generation process (i.e., prune the implicit generation tree). Taken together, these heuristics constitute the program's "chemical model." This model includes: a list of denied embedded subgraphs, the existence of any one of which in a strucutre rules out that structure as a plausible hypothesis; a list of well-known, stable, and generally highly significant radicals which are treated in an aggregate fashion as "superatoms," or essentially higher level concepts; an evaluation function not dependent on spectral data that evalutes the potential fruitfulness of attempting to generate structures from a collection of as yet unassigned atoms (i.e., evaluates the worth of pursuing a particular subproblem); a data matching process that we sometimes call "the zero-order theory of the mass spectrometer" that makes decisions about the relevance of the subproblems based on the actual mass numbers present in the given spectrum; a "rote memory," called the Dictionary, which is the memory of previously solved subproblems, corresponding to the theorem memory in theorem-proving programs; and a few other heuristic processes of lesser importance.

3. Learning processes in Heuristic Dendral are relatively simple, and a high order of learning by the program itself remains more of a goal than an accomplished fact. The main "internal" learning process is the Dictionary building activity. Learning on the subproblem evaluation function a la Samuel's Checker Program is possible, but not implemented. "Extrinsic" learning, in the sense of a human expert communicating to the program the elements of the chemical model, has been extensively and successfully used. Perhaps this is high level programming, but in the same sense that pedagogy in general is high level programming activity.

ORGANIC CHEMISTRY THE PROBLEM-CONTEXT OF DENDRAL

The fundamental problem of organic chemistry is the topological structure of a molecule. This was first brought into

focus by the Swedish chemist, Jons Jakob Berzelius (1779-1848) when he established the occurence of chemical isomers. These are different organic molecules having the same chemical composition or ensemble of atoms; hence they have different structures (i.e., connectivities of the atoms with respect to atom-to-atom bonds). For one of the simplest examples, take C_2H_6O, which has the two isomers, dimethyl ether and ethanol (Fig. 7.1). To determine that the composition of a compound once obtained as a pure sample, say C_2H_6O, is essentially a mechanical process of quantitative analysis. To assign it to one of the possible isomers is a much more demanding intellectual exercise.

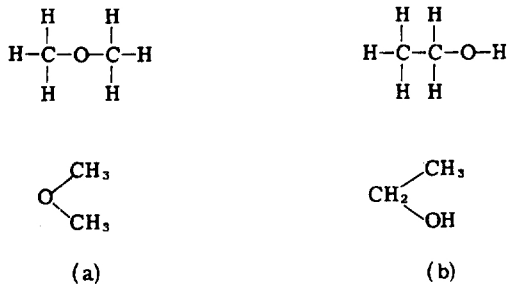


Fig. 7.1 Two isomers of C_2H_6O: (a) Dimethyl ether (O..CH3 CH3), (b) Ethanol or ethyl alcohol (CH2..CH3 OH).

Each of these may also be represented by isomorphic graphs, for example, for ethanol: (CH3.CH2.OH) or (OH.CH2.CH3) or (CH2..OH CH3). The previous notation is in canonical DENDRAL form, being initialized by the center of the graph, followed by a dot for each radical, and then a list of radicals in order of an algorithmically defined value. For internal representation or more compact coding, the H's can be dropped, leaving us with O..CC and C..CO respectively. Where symmetries prevail, we can go one step further, using the '/' as a ditto mark, as in (O./C) for (O..CC). This economy is, of course, trivial here, but not so in more complex formulas.

At this level of analysis, structure means connectivity, not geometry. In fact, with the help of X-ray diffraction analysis, a great deal can be learned about the actual disposition in space of the atoms in a molecule in the crystalline state. However, the molecules, especially in the liquid or gaseous states, may be undergoing a variety of dynamic transitions—linear, rotational, and rocking modes about every chemical bond. Chemical geometry is beyond the scope of the present

discussion, but what we know of it could be superimposed upon the topological frameowrk developed below.

The preceding paragraph can be summarized: a chemical structure is represented by an undirected graph whose nodes are atoms, whose edges are chemical bonds. While the analogy was recognized 100 years ago, this outlook has still to penetrate the teaching of organic chemistry.

In practical problem solving the chemist uses every possible datum. For example, smell can help him decide between dimethyl ether and ethanol, if he did not already recognize that the ether would be much more volatile than its isomeric alcohol. He also has a repertoire of reagents that can help to detect various fragments (called radicals) in the molecule (e.g., -OH). More recently a specialized instrument, the mass spectrometer, has been developed which facilitates a unified systematic attack on structural problems. Briefly, a molecule is bombarded by an electron beam which sputters off an electron, leaving a positively charged molecule-ion. A fraction of these fragment, giving radical ions of various sizes corresponding to different modes of cleavage, often complicated by further rearrangements and reactions of fragments. Finally, the ensemble of molecule- and radical-ions is resolved by careful acceleration through electrostatic and magnetic fields.

The utility of the mass spectrometer and some examples of the logical inference employed in exploiting it are reviewed by McLafferty (1966).

The mass spectrum is a paired list of mass numbers and their relative intensities. Mass spectrometers of very high resolution have been built, capable of distinguishing between radicals of different composition but the same integer atomic weight. For example, the radical -NH, $M = 15.0110$ can be distinguished from the radical $-CH_3$, $M = 15.0215$. This capability is especially useful for determining the formula of the intact molecule. Unless we specify otherwise, however, we have in mind the more ordinary low resolution mass spectrometer which lumps together species having the same integral mass. However, more precise data are readily accommodated and avidly used by the program logic.

The stated goal of our program is then an inductive solution of the mass spectrum. That is, a molecular formula and its mass spectrum are given as data. We must induce the

structure (hypothesis) that best satisfies the data. Our basic approach to this has been first to furnish the computer with a language in which chemical structure hypotheses can be expressed, then to interrogate chemists and their literature for the rules and techniques they have used in problem solving and attempt to translate these into computer algorithms. In the course of searching for these heuristics, we have in fact discovered a number of algorithms which are much more systematic than the approaches commonly used by chemists in this field.

ISOMERS

Underlying the solution of virtually every problem and subproblem in structural organic chemistry is the potential exhaustion of the list of possible isomers of a given molecule or radical. It is remarkable that while hundreds of thousands of students of elementary organic chemistry are challenged in this way every year, no algorithm for generating and verifying complete lists of isomers has hitherto been presented. Each student is left to work out his own intuitive approach to this problem, which may account for the bafflement with which very many students approach the subject upon their first exposure to it.

The core of DENDRAL is a notation for chemical structures and an algorithm capable of producing all distinct isomers and casting each of them into a canonical representation. This will be outlined in more detail further on.

The lowest level of DENDRAL might be called the topologist. This machine considers only the valence rules and elementary graph theory in constructing lists of isomers. It uses two elementary concepts, one, the center of a graph as a point of departure, and two, a recursive procedure for evaluating a radical as a way of specifying the canonical representation of a given molecule. After the center of the map is fixed, being either a bond or an atom of known valence, the radicals pendant on the center must be listed in nondecreasing value. The apical node of each radical is then regarded as a new center and the process continues recursively.

The same approach can be used to make a generator from DENDRAL. From the formula or composition list, a bond or a

given species of atom is first taken as the central feature and the remaining atoms partitioned in appropriate ways, and these partitions assigned tentatively to the pendant radicals. For each radical then successive allocations are made for the apical node and then partitions are allocated to the pendant subradicals, and so forth.

TABLE 7.1 Canons of Dendral Order* (Hierarchy of Vector Valuation in Decreasing Order of Significance)

The DENDRAL-VALUE of a Radical Consists of the Vector:

COUNT

- Rings by number of rings†
- Other atoms (except H)

COMPOSITION of radical

- Rings† by valuation of ring
  - Composition, Vertex Group, Path List, Vertex List, Substituent Locations
- Other atoms by atomic number (S, P, O, N, C)

UNSATURATIONS (afferent link included; ring paths excluded)

APICAL NODE

- Ring Value†
- Degree: number of efferent radicals
- Composition: e.g. (S, P, O, N, C)
- Afferent link: (:, :, .)

APPENDANT RADICALS if any

- (nested vectors in canonical order); nil if current apex is terminal
- Enantiomerism around apex (DL, D, L, unspecified) if applicable

*From J. Lederberg (1964), DENDRAL-64, NASA CR-57029, Star No. N65-13158
†Rings are not discussed in this paper.
Each line above is a separate cell or subcell of the vector.

Table 7.1 summarizes the order of allocation for evaluating a radical, and for generating the next structure, a procedure used recursively in the DENDRAL program in LISP. At this time, the operational program is confined to acyclic structures. However, the specifications have been detailed for a complete system, including ring structures of arbitrary complexity as well as consideration of optical isomerism (1,2). Table 7.2 lists the computation of all the isomers generated by the topologist for the formula $C_3H_7NO_2$, one of whose isomers

is the common amino acid, alanine. This exercise is already at the very margin of human capability, barring the possible rediscovery of this algorithm. In practice no intelligent human has the patience to attempt to generate such a list by the intuitive process. The chemist will often then demand redundant information at this point in order to narrow the range of possibilities he is obliged to consider before he will make the effort to produce an exhaustive list.

The topologist knows only the valence rules as quasi-empirical data, that is, that four bonds must issue from each carbon atom, three from any nitrogen, two from any oxygen, and but one from hydrogen. With this very limited quota of chemical insight, the topologist produces many structures that would be regarded as absurdities by the experienced chemist, for example the radical (·O·NH·OH) in no. 4 of Table 7.2. The next stage in the development of DENDRAL is then to impart a certain amount of additional chemical information taken from the real world. In doing this a definite context is implied, even if this is not immediately overt. There are probably many realms of organic chemistry, for example, at ultra low temperatures, of which we have only limited experience. The implicit context we have in fact adopted is that of the natural product, that is to say, molecular species that might be reasonably stable at ambient temperatures, and therefore stand some chance of persisting or being isolated from natural sources. However, this rule has been applied rather cautiously and the lists that will be adduced for further illustration still contain a number of items which would be regarded as quite dubious by this criterion.

The program is quite amenable to adjustment to any given set of facts. Indeed, a certain stage in the program can be switched on to interrogate the chemist to help to find the context in which various rules will be applied or not. At this stage chemical insight is given most explicitly by providing a list of forbidden substructures. Whenever these substructures are encountered during the building of a potential molecule, the generator is adjusted to ignore that entire branch of synthetic possibilities. In order to effectuate this use of a "BADLIST," a graph matching algorithm has been incorporated into the DENDRAL program. At best, however, graph matching is an expensive proposition and it soon became necessary to seek ways of economizing on redundant computation. The last

TABLE 7.2 The Isomers of Alanine, C3H7NO2, without Chemical Common Sense*

```
>(ISOMERS *ALANINE)

>BADLIST
 NIL

>GOODLIST
 NIL

>SPECTRUM
 NIL

>DICTLIST
 NIL

>(ILLEGAL ATTACHMENTS)
 (NIL NIL NIL)

C3H7NO2
MOLECULES
((U . 1.) (C . 3.) (N . 1.) (O . 2.))
 1.     .  C3H7    O.N = O,
 2.     .  CH..CH3 CH3    O.N = O,
 3.     .  CH2.CH = CH2    NH.O.OH,
 4.     .  CH2.CH = CH2    O.NH.OH,
 5.     .  CH2.CH = CH2    O.O.NH2,
 6.     .  CH2.CH = CH2    N..OH OH,
 7.     .  CH = CH.CH3    NH.O.OH,
 8.     .  CH = CH.CH3    O.NH.OH,
 9.     .  CH = CH.CH3    O.O.NH2,
10.     .  CH = CH.CH3    N..OH OH,
11.     .  C. = CH3 CH2    NH.O.OH,
12.     .  C. = CH3 CH2    O.NH.OH,
13.     .  C. = CH3 CH2    O.O.NH2,
14.     .  C. = CH3 CH2    N..OH OH,
15.     .  CH2.CH2.NH2    O.CH = O,
16.     .  CH2.CH2.NH2    *COOH,
17.     .  CH2.NH.CH3    O.CH = O,
18.     .  CH2.NH.CH3    *COOH,
19.     .  NH.C2H5    O.CH = O,
20.     .  NH.C2H5    *COOH,
21.     .  CH..CH3 NH2    O.CH = O,
22.     .  CH..CH3 NH2    *COOH,
23.     .  N..CH3 CH3    O.CH = O,
24.     .  N..CH3 CH3    *COOH,
25.     .  CH2.CH = NH    CH2.O.OH,
26.     .  CH2.CH = NH    O.CH2.OH,
27.     .  CH2.CH = NH    O.O.CH3,
28.     .  CH2.CH = NH    CH..OH OH,
29.     .  CH = CH.NH2    CH2.O.OH,
30.     .  CH = CH.NH2    O.CH2.OH,
31.     .  CH = CH.NH2    O.O.CH3,
```

TABLE 7.2 (Continued)

```
32.    .  CH = CH.NH2    CH..OH OH,
33.    .  CH2.N = CH2    CH2.O.OH,
34.    .  CH2.N = CH2    O.CH2.OH,
35.    .  CH2.N = CH2    O.O.CH3,
36.    .  CH2.N = CH2    CH..OH OH,
37.    .  CH = N.CH3    CH2.O.OH,
38.    .  CH = N.CH3    O.CH2.OH,
39.    .  CH = N.CH3    O.O.CH3,
40.    .  CH = N.CH3    CH..OH OH,
41.    .  NH.CH = CH2    CH2.O.OH,
42.    .  NH.CH = CH2    O.CH2.OH,
43.    .  NH.CH = CH2    O.O.CH3,
44.    .  NH.CH = CH2    CH..OH OH,
45.    .  N = CH.CH3    CH2.O.OH,
46.    .  N = CH.CH3    O.CH2.OH,
47.    .  N = CH.CH3    O.O.CH3,
48.    .  N = CH.CH3    CH..OH OH,
49.    .  C. = CH3 NH    CH2.O.OH,
50.    .  C. = CH3 NH    O.CH2.OH,
51.    .  C. = CH3 NH    O.O.CH3,
52.    .  C. = CH3 NH    CH..OH OH,
53.    .  C = .CH2 NH2    CH2.O.OH,
54.    .  C = .CH2 NH2    O.CH2.OH,
55.    .  C = .CH2 NH2    O.O.CH3,
56.    .  C = .CH2 NH2    CH..OH OH,
57.    .  CH2.CH2.OH    CH2.N = O,
58.    .  CH2.CH2.OH    CH = N.OH,
59.    .  CH2.CH2.OH    NH.CH = O,
60.    .  CH2.CH2.OH    N = CH.OH,
61.    .  CH2.CH2.OH    O.CH = NH,
62.    .  CH2.CH2.OH    O.N = CH2,
63.    .  CH2.CH2.OH    *CONH2,
64.    .  CH2.CH2.OH    C = .NH OH,
65.    .  CH2.O.CH3    CH2.N = O,
66.    .  CH2.O.CH3    CH = N.OH,
67.    .  CH2.O.CH3    NH.CH = O,
68.    .  CH2.O.CH3    N = CH.OH,
69.    .  CH2.O.CH3    O.CH = NH,
70.    .  CH2.O.CH3    O.N = CH2,
71.    .  CH2.O.CH3    *CONH3,
72.    .  CH2.O.CH3    C = .NH OH,
73.    .  O.C2H5    CH2.N = O,
74.    .  O.C2H5    CH = N.OH,
75.    .  O.C2H5    NH.CH = O,
76.    .  O.C2H5    N = CH.OH,
77.    .  O.C2H5    O.CH = NH,
78.    .  O.C2H5    O.N = CH2,
79.    .  O.C2H5    *CONH2,
80.    .  O.C2H5    C = .NH OH,
81.    .  CH..CH3 OH    CH2.N = O,
82.    .  CH..CH3 OH    CH = N.OH,
83.    .  CH..CH3 OH    NH.CH = O,
84.    .  CH..CH3 OH    N = CH.OH,
```

TABLE 7.2 (Continued)

```
 85.    .  CH..CH3 OH    O.CH = NH,
 86.    .  CH..CH3 OH    O.N = CH2,
 87.    .  CH..CH3 OH    *CONH2,
 88.    .  CH..CH3 OH    C = .NH OH,
 89.    .  CH2.CH = O    CH2.NH.OH,
 90.    .  CH2.CH = O    CH2.O.NH2,
 91.    .  CH2.CH = O    NH.CH2.OH,
 92.    .  CH2.CH = O    NH.O.CH3,
 93.    .  CH2.CH = O    O.CH2.NH2,
 94.    .  CH2.CH = O    O.NH.CH3,
 95.    .  CH2.CH = O    CH..NH2 OH,
 96.    .  CH2.CH = O    N..CH3 OH,
 97.    .  CH = CH.OH    CH2.NH.OH,
 98.    .  CH = CH.OH    CH2.O.NH2,
 99.    .  CH = CH.OH    NH.CH2.OH,
100.    .  CH = CH.OH    NH.O.CH3,
101.    .  CH = CH.OH    O.CH2.NH2,
102.    .  CH = CH.OH    O.NH.CH3,
103.    .  CH = CH.OH    CH..NH2 OH,
104.    .  CH = CH.OH    N..CH3 OH,
105.    .  O.CH = CH2    CH2.NH.OH,
106.    .  O.CH = CH2    CH2.O.NH2,
107.    .  O.CH = CH2    NH.CH2.OH,
108.    .  O.CH = CH2    NH.O.CH3,
109.    .  O.CH = CH2    O.CH2.NH2,
110.    .  O.CH = CH2    O.NH.CH3,
111.    .  O.CH = CH2    CH..NH2 OH,
112.    .  O.CH = CH2    N..CH3 OH,
113.    .  C. = CH3 O    CH2.NH.OH,
114.    .  C. = CH3 O    CH2.O.NH2,
115.    .  C. = CH3 O    NH.CH2.OH,
116.    .  C. = CH3 O    NH.O.CH3,
117.    .  C. = CH3 O    O.CH2.NH2,
118.    .  C. = CH3 O    O.NH.CH3,
119.    .  C. = CH3 O    CH..NH2 OH,
120.    .  C. = CH3 O    N..CH3 OH,
121.    .  C = .CH2 OH    CH2.NH.OH,
122.    .  C = .CH2 OH    CH2.O.NH2,
123.    .  C = .CH2 OH    NH.CH2.OH,
124.    .  C = .CH2 OH    NH.O.CH3,
125.    .  C = .CH2 OH    O.CH2.NH2,
126.    .  C = .CH2 OH    O.NH.CH3,
127.    .  C = .CH2 OH    CH..NH2 OH,
128.    .  C = .CH2 OH    N..CH3 OH,
129.    =  CH.C2H5    N.O.OH,
130.    =  C..CH3 CH3    N.O.OH,
131.    =  CH.CH2.NH2    CH.O.OH,
132.    =  CH.CH2.NH2    C..OH OH,
133.    =  CH.NH.CH3    CH.O.OH,
134.    =  CH.NH.CH3    C..OH OH,
135.    =  N.C2H5    CH.O.OH,
136.    =  N.C2H5    C..OH OH,
137.    =  C..CH3 NH2    CH.O.OH,
```

TABLE 7.2 (Continued)

```
138.    =  C..CH3 NH2    C..OH OH,
139.    =  CH.CH2.OH     CH.NH.OH,
140.    =  CH.CH2.OH     CH.O.NH2,
141.    =  CH.CH2.OH     N.CH2.OH,
142.    =  CH.CH2.OH     N.O.CH3,
143.    =  CH.CH2.OH     C..NH2 OH,
144.    =  CH.O.CH3      CH.NH.OH,
145.    =  CH.O.CH3      CH.O.NH2,
146.    =  CH.O.CH3      N.CH2.OH,
147.    =  CH.O.CH3      N.O.CH3,
148.    =  CH.O.CH3      C..NH2 OH,
149.    =  C..CH3 OH     CH.NH.OH,
150.    =  C..CH3 OH     CH.O.NH2,
151.    =  C..CH3 OH     N.CH2.OH,
152.    =  C..CH3 OH     N.O.CH3,
153.    =  C..CH3 OH     C..NH2 OH,
154.    CH...  CH3    CH = NH    O.OH,
155.    C. = .  CH3   CH.NH2     O.OH,
156.    CH...  CH3    N = CH2    O.OH,
157.    C. = .  CH3   N.CH3      O.OH,
158.    CH...  CH3    CH2.OH     N = O,
159.    C.. =  CH3    CH2.OH     N.OH,
160.    CH...  CH3    O.CH3      N = O,
161.    C.. =  CH3    O.CH3      N.OH,
162.    CH...  CH3    CH = O     NH.OH,
163.    CH...  CH3    CH = O     O.NH2,
164.    C. = .  CH3   CH.OH      NH.OH,
165.    C. = .  CH3   CH.OH      O.NH2,
166.    C = ..  CH2   CH2.NH2    O.OH,
167.    C = ..  CH2   NH.CH3     O.OH,
168.    C = ..  CH2   CH2.OH     NH.OH,
169.    C = ..  CH2   CH2.OH     O.NH2,
170.    C = ..  CH2   O.CH3      NH.OH,
171.    C = ..  CH2   O.CH3      O.NH2,
172.    CH...  NH2    CH = CH2   O.OH,
173.    C. = .  NH2   CH.CH3     O.OH,
174.    CH...  NH2    CH2.OH     CH = O,
175.    C.. =  NH2    CH2.OH     CH.OH,
176.    CH...  NH2    O.CH3      CH = O,
177.    C.. =  NH2    O.CH3      CH.OH,
178.    C = ..  NH    C2H5       O.OH,
179.    C = ..  NH    CH2.OH     CH2.OH,
180.    C = ..  NH    CH2.OH     O.CH3,
181.    C = ..  NH    O.CH3      O.CH3,
182.    CH...  OH     C2H5       N = O,
183.    C.. =  OH     C2H5       N.OH,
184.    CH...  OH     CH = CH2   NH.OH,
185.    CH...  OH     CH = CH2   O.NH2,
186.    C. = .  OH    CH.CH3     NH.OH,
187.    C. = .  OH    CH.CH3     O.NH2,
188.    CH...  OH     CH2.NH2    CH = O,
189.    C.. =  OH     CH2.NH2    CH.OH,
190.    CH...  OH     NH.CH3     CH = O,
```

TABLE 7.2 (Continued)

191.	C.. =	OH	NH.CH3	CH.OH,
192.	CH...	OH	CH = NH	CH2.OH,
193.	CH...	OH	CH = NH	O.CH3,
194.	C. = .	OH	CH.NH2	CH2.OH,
195.	C. = .	OH	CH.NH2	O.CH3,
196.	CH...	OH	N = CH2	CH2.OH,
197.	CH...	OH	N = CH2	O.CH3,
198.	C. = .	OH	N.CH3	CH2.OH,
199.	C. = .	OH	N.CH3	O.CH3,
200.	C = ..	O	C2H5	NH.OH,
201.	C = ..	O	C2H5	O.NH2,
202.	C = ..	O	CH2.NH2	CH2.OH,
203.	C = ..	O	CH2.NH2	O.CH3,
204.	C = ..	O	NH.CH3	CH2.OH,
205.	C = ..	O	NH.CH3	O.CH3,
206.	N...	CH3	CH = CH2	O.OH,
207.	N...	CH3	CH2.OH	CH = O,
208.	N...	CH3	O.CH3	CH = O,
209.	N...	OH	C2H5	CH = O,
210.	N...	OH	CH = CH2	CH2.OH,
211.	N...	OH	CH = CH2	O.CH3,
212.	C....	CH3	CH3	OH N = O,
213.	C....	CH3	NH2	OH CH = O,
214.	C....	CH3	OH	OH CH = NH,
215.	C....	CH3	OH	OH N = CH2,
216.	C....	NH2	OH	OH CH = CH2.

*This is a complete list of the topological possibilities. The restraints of BADLIST and of a filtered DICTIONARY have been relaxed. Compare with Table 7.4: the additional structures here are chemically implausible for the standard context of the intended use of DENDRAL. For example, no structures are empirically known which contain the radical (.O.NH.OH).

In these and following tables, the text is all computer output except lines prefixed with >, which are input from the teletype.

important feature merely exploits an idiosyncrasy of the DENDRAL program that makes it easy to detect linear sequences of nodes that might be on a list of illegal attachments, for example, -N-N-N or -O-O.

Of far greater generality is the use of a dictionary of solved subproblems. As soon as the program has gone a short way towards a solution of any practical problem, DENDRAL would find itself constantly redoing the same subproblems over and over again as it rebuilds radicals on one side of the

TABLE 7.3 Calling the Function (PRINDICTER) Results in a Dump of the Dictionary in Its Current State*

```
(PRINDICTER)
U00C01N01001    1.    .CH2.NH.OH
  2.      .CH2.O.NH2
  3.      .NH.O.CH3
  4.      .O.NH.CH3
  5.      .N..CH3 OH

U01C02001    1.    .CH2.CH = O
  2.      = CH.CH2.OH
  3.      = CH.O.CH3
  4.      .O.CH = CH2
  5.      .C. = CH3 O
  6.      = C..CH3 OH

U01C01N01001    1.    .CH = N.OH
  2.      = CH.NH.OH
  3.      = CH.O.NH2
  4.      .NH.CH = O
  5.      = N.O.CH3
  6.      .O.CH = NH
  7.      .O.N = CH2
  8.      .*CONH2

U00C02001    1.    .CH2.CH2.OH
  2.      .CH2.O.CH3
  3.      .O.C2H5
  4.      .CH..CH3 OH

U00C01002
(NO STRUCTURES)

U00C01001    1.    .CH2.OH
  2..     .O.CH3

U01C02N01    1.    .CH2.CH = NH
  2.      .CH2.N = CH2
  3.      .CH = N.CH3
  4.      = CH.CH2.NH2
  5.      = CH.NH.CH3
  6.      .N = CH.CH3
  7.      = N.C2H5
  8.      .C. = CH3 NH
  9.      = C..CH3 NH2

U01C01N01    1.    .CH = NH
  2.      = CH.NH2
  3.      .N = CH2
  4.      = N.CH3

U01C01002    1.    .O.CH = O
  2.    .*COOH
```

TABLE 7.3 (Continued)

```
U01C01001    1.    .CH = O
  2.      = CH.OH

U00C02N01    1.    .CH2.CH2.NH2
  2.      .CH2.NH.CH3
  3.      .NH.C2H5
  4.      .CH..CH3 NH2
  5.      .N..CH3 CH3

U00C01N01    1.    .CH2.NH2
  2.    .NH.CH3

U00N01002
(NO STRUCTURES)

U00N01001    1.    .NH.OH
  2.      .O.NH2

U00N01    1.    .NH2

U01C03    1.    .CH2.CH = CH2
  2.      .CH = CH.CH3
  3.      =CH.C2H5
  4.      .C. = CH3 CH
  5.      = C..CH3 CH3

U01C02    1.    .CH = CH2
  2.      = CH.CH3

U01C01    1.    =CH2

U01N01002    1.    .O.N = O

U00002
(NO STRUCTURES)

U01N01001    1.    .N = O
  2.    = N.OH

U01N01    1.    = NH

U00001    1.    .OH

U01002
(NO STRUCTURES)

U01001    1.    = O

U00C03    1.    .C3H7
  2.    .CH..CH3 CH3

U00C02    1.    .C2H5
```

TABLE 7.3 (Continued)

U00C01 1. .CH3

DONE

*This example shows the dictionary that was built for Table 7.4, and contains the radicals needed to generate the molecules isomeric to C3H7NO2. The headings encode the compositions in the form UaaCbb NccOdd where C, N, O have their usual connotation of atoms, and U stands for "unsaturations." This is calculated as double-bond-equivalents, or the number of pairs of H by which the composition falls short of a saturated, that is, double-bond-free molecule.

molecules after reconstructing the other side. In order to avoid the waste involved in this redundancy, the program automatically generates a list of compositions which is consulted whenever a new radical is to be generated. If the composition of the new radical appears in the dictionary, the dictionary contents are simply copied out. If not, the problem is solved and a new dictionary item is entered for further use later. Insofar as the dictionary has already been filtered with respect to BADLIST, a great deal of effort can be saved, and in fact the program would not be practical for molecules of even moderate complexity were it not for this feature. As an example, the dictionary that has been generated in the solution of the alanine problem is given in Table 7.3, and the filtered list of isomers is Table 7.4. It is also feasible and desirable to give chemical insight into the program by overt manipulation of the dictionary. That is to say, when a given context calls for it, the radicals corresponding to a given composition can be entered directly, usually with the aim of excluding certain idiosyncratic items. This must be done with great care, since the list of larger radicals that may be generated later relies upon the dictionary already established for smaller radicals.

A serious problem encountered in practice is managing the trade-off between the growth of the dictionary and the corresponding loss of scratch space for the LISP program to

TABLE 7.4 The Isomers of Alanine, C3H7NO2, Restrained by Common Sense*

```
>(ISOMERS *ALANINE)

>BADLIST
 ((C (1. (N O)) (1. (N O))) ((N O) (1. C (1. (N O))))
 (C (3. C (1. (N O) (1. H)))) (C (3. C) (1. (N O) (1. H)))
 ((N O) (1. H) (1. C (3. C))) (C (2. C (1. (N O) (1. H))))
 (C (2. C) (1. (N O) (1. H))) ((N O) (1. H) (1. C (2. C)))
 (N (2. C (1. O (1. H)))) (C (2. N) (1. O (1. H))) (O (1. H)
 (1. C (2. N))) (O (1. O)) (O (1. N (1. O))) (N (1. O)
 (1. O)) (C (1. H) (1. N (2. O))) (N (1. C (1. H)) (2. O))
 (*CO* (1. O (1. H))) ((N O) (1. C (1. O (1. H)) (2. O))))

> GOODLIST
  NIL

> SPECTRUM
  NIL

>DICTLIST
  NIL

>(ILLEGAL ATTACHMENTS)
 (NIL ((N N N)) ((O) (*CH2OH*)))

 C3H7NO2
 MOLECULES
 ((U . 1.) (C . 3.) (N . 1.) (O . 2.))
  1.    .  C3H7    O.N = O,
  2.    .  CH..CH3 CH3     O.N = O,
  3.    .  CH2.CH2.NH2     O.CH =O,
  4.    .  CH2.CH2.NH2     *COOH,
  5.    .  CH2.NH.CH3      *COOH,
  6.    .  NH.C2H5     O.CH = O,
  7.    .  CH..CH3 NH2     *COOH,
  8.    .  N..CH3 CH3      O.CH = O,
  9.    .  CH2.CH2.OH      CH = N.OH,
 10.    .  CH2.CH2.OH      NH.CH = O,
 11.    .  CH2.CH2.OH      O.CH = NH,
 12.    .  CH2.CH2.OH      O.N = CH2,
 13.    .  CH2.CH2.OH      *CONH2,
 14.    .  CH2.O.CH3     CH = N.OH,
 15.    .  CH2.O.CH3     *CONH2,
 16.    .  O.C2H5     CH = N.OH,
 17.    .  O.C2H5     NH.CH = O,
 18.    .  CH..CH3 OH      CH = N.OH,
 19.    .  CH..CH3 OH      *CONH2,
 20.    .  CH2.CH = O    CH2.NH.OH,
 21.    .  CH2.CH = O    CH2.O.NH2,
 22.    .  CH2.CH = O    NH.O.CH3,
 23.    .  CH2.CH = O    O.NH.CH3,
 24.    .  CH2.CH = O    N..CH3 OH,
 25.    .  C. = CH3 O    CH2.NH.OH,
```

TABLE 7.4 (Continued)

26.	.	C. = CH3	O	CH2.O.NH2,	
27.	.	C. = CH3	O	NH.O.CH3,	
28.	.	C. = CH3	O	O.NH.CH3,	
29.	.	C. = CH3	O	N..CH3 OH,	
30.	.	CH.CH2.OH		CH.O.NH2,	
31.	=	CH.CH2.OH		N.O.CH3,	
32.	=	CH.O.CH3		CH.O.NH2,	
33.	=	CH.O.CH3		N.O.CH3,	
34.	C.. =	CH3	CH2.OH	N.OH,	
35.	C.. =	CH3	O.CH3	N.OH,	
36.	CH...	CH3	CH = O	NH.OH,	
37.	CH...	CH3	CH = O	O.NH2,	
38.	C = ..	CH2	CH2.OH	O.NH2,	
39.	CH...	NH2	CH2.OH	CH = O,	
40.	C = ..	NH	CH2.OH	CH2.OH,	
41.	C = ..	NH	CH2.OH	O.CH3,	
42.	CH...	OH	CH2.NH2	CH = O,	
43.	CH...	OH	CH = NH	CH2.OH,	
44.	C = ..	O	C2H5	NH.OH,	
45.	C = ..	O	C2H5	O.NH2,	
46.	C = ..	O	CH2.NH2	CH2.OH,	
47.	C = ..	O	CH2.NH2	O.CH3,	
48.	C = ..	O	NH.CH3	CH2.OH,	
49.	N...	CH3	O.CH3	CH = O,	
50.	N...	OH	C2H5	CH = O,	

*This restraint is implemented by systematic graph-matching against a BADLIST which contains the worst monstrosities of fragments, as indicated in the dialogue that precedes the output table.

maneuver in. If left unchecked the dictionary building can easily reach the point of exhausting available computing room and paralyzing the program. A heuristic management of the dictionary would be a close analog to the human solution to this problem and is being studied at the present time. For example, very large dictionaries could be stored on external memories, and only those segments kept in core that are needed for the current operations of the program.

These facilities have been built into the DENDRAL generator program in such a way as to leave it in a state of high efficiency. Thus the filters are not applied at the end after the production of a larger redundant list, they are applied at the earliest possible stage in the tree building program. When $C_3H_7NO_3$ is examined by this filtered DENDRAL generator the results of Table 7.4 are obtained. Each of these is a moderately

plausible chemical isomer. No. 7 is the actual structure of alanine. The order of output is the canonical DENDRAL sequence.

It may be of some interest that three of the structures in Table 7.4 have apparently not yet been reported in the chemical literature, although they would appear to be reasonable candidates for synthesis by a chemistry graduate student. With even slightly more complex molecules, one should expect to find that only a small minority of the potential structural species are in fact already known to chemical science. Without an algorithmic generator, however, it has not hitherto been possible to make any realistic estimates of the extent of empirical coverage of the theoretical expectations.

It should be perfectly obvious that again with a small increase in complexity the number of possible isomers will grow very quickly and one may have to rely upon a heuristic rather than an exhaustive approach to the generation of hypotheses apt to a given set of data. In particular it might be desirable to use some *a priori* notions of plausibility in the generator and then to seek ways of adjusting the program so that the parameters for plausibility sequences were already sensitive to qualities in the data themselves. One approach to this uses GOODLIST, an ordered list of preferred substructures. That is to say, we would assign the highest plausibility and therefore priority for deductive corroboration of those molecules which contain items in GOODLIST. In order to accomplish this each GOODLIST item is regarded as a "super atom" of appropriate valence, and the corresponding subset of atoms from the compositional formula is allocated to the super atom. Thus the very common radical -COOH, the carboxyl radical, is a very common ensemble of a double bond, a carbon atom, and two oxygen atoms, (·C·:OH O). Insofar as the molecular formula permits, various numbers of these sets of atoms are assigned to carboxyl groups, and the construct -COOH is then regarded as if it were a univalent superatom.

Certain housekeeping details must be looked after to be sure of avoiding redundant representations and to reconvert the constructions to canonical form. They will, however, no longer be in canonical sequence, but rather have some implicit order of plausibility in the sequence with which they are put out. When alanine is subjected to such a procedure, the ordering of Table 7.5 is obtained. It will be noted that alanine is a very early entry in this table.

TABLE 7.5 The Isomers of Alanine, as in Table 7.4, but Resequenced by the Application of GOODLIST*

```
>(SETQ GOODLIST SAVEGOODLIST)
 ((*COOH* (1. C (1. O) (2. O)) 100 . 0) (*CO* (2. C (2. O))
 100. 0.) (*CHNH2* (2. C (1. N)) 100. 0.) (*CH2OH*
 (1. C (1. O)) 100. 0.) (*NOH* (2. N (1. O)) 100. 0.)
 (*CHNH* (1. C (2. N)) 100. 0.) (*NCH2* (1. N (2 . C))
 100. 0.))
>(ISOMERS *ALANINE)
 MOLECULES
 ((U . 0.) (C . 1.) (*COOH* . 1.) (*CHNH2* . 1.))
  1.     .  CH2.CH2.NH2    *COOH,
  2.     .  CH..CH3 NH2    *COOH,

 MOLECULES
 ((U . 0.) (C . 2.) (N . 1.) (*COOH* . 1.))
  1.     .  CH2.NH.CH3     *COOH,

 MOLECULES
 ((U . 0.) (*CO* . 1.) (*CHNH2* . 1.) (*CH2OH* . 1.))
  1.    C = ..  O    CH2.NH2    CH2.OH,
  2.    CH...  NH2    CH2.OH    CH = O,

 MOLECULES
 ((U . 0.) (C . 1.) (O . 1.) (*CO* . 1.) (*CHNH2* . 1.))
  1.    C = ..  O    CH2.NH2    O.CH3,
  2.    .  CH2.CH2.NH2    O.CH = O,
  3.    CH...  OH    CH2.NH2    CH = O,

 MOLECULES
 ((U . 0.) (C . 1.) (N . 1.) (*CO* . 1.) (*CH2OH* . 1.))
  1.    C = ..  O    NH.CH3    CH2.OH,
  2.    .  CH2.CH2.OH    NH.CH = O,
  3.    .  CH2.CH2.OH    *CONH2,

 MOLECULES
 ((U . 0.) (C . 2.) (*CO* . 1.) (*NOH* . 1.))
  1.    C = ..  O    C2H5    NH.OH,
  2.    N...  OH    C2H    CH = O,
  3.    .  CH2.CH = O    CH2.NH.OH,
  4.    .  CH2.CH = O    N..CH3 OH,
  5.    .  C. = CH3 O    CH2.NH.OH,
  6.    .  C. = CH3 O    N..CH3 OH,
  7.    CH...  CH3    CH = O    NH.OH,

 MOLECULES
 ((U . 0.) (C . 2.) (N . 1.) (O . 1.) (*CO* , 1.))
  1.    .  CH2.O.CH3    *CONH2,
  2.    .  CH2.CH = O    CH2.O.NH2,
  3.    .  C. = CH3 O    CH2.O.NH2,
  4.    .  NH.C2H5    O.CH = O,
  5.    .  CH2.CH = O    NH.O.CH3,
  6.    .  C. = CH3 O    NH.O.CH3,
  7.    .  O.C2H5    NH.CH = O,
```

TABLE 7.5 (Continued)

```
 8.      .  CH2.CH = O    O.NH.CH3,
 9.      .  C. = CH3 O    O.NH.CH3,
10.      C = ..  O    C2H5    O.NH2,
11.      .  CH..CH3 OH    *CONH2,
12.      CH...  CH3    CH = O    O.NH2,
13.      .  N..CH3 CH3    O.CH = O,
14.      N...  CH3 O.CH3    CH = O,
MOLECULES
((U . 1.) (*CHNH2* . 1.) (CH2OH* . 2.))
*NO ALLOWABLE STRUCTURES

MOLECULES
((U . 1.) (C . 1.) (O . 1.) (*CHNH2* . 1.) (*CH2OH* . 1.))

MOLECULES
((U . 1.) (C . 2.) (O . 2.) (*CHNH2* . 1.))
*NO ALLOWABLE STRUCTURES

MOLECULES
((U . 0.) (*CH2OH* . 2.) (*CHNH* . 1.))
*NO ALLOWABLE STRUCTURES

MOLECULES
((U . 0.) (*CH2OH* . 2.) (*NCH2* . 1.))
*NO ALLOWABLE STRUCTURES

MOLECULES
((U . 1.) (C . 1.) (N . 1.) (*CH2OH* . 2.))
 1.      C =..  NH    CH2.OH    CH2.OH,

MOLECULES
((U . 1.) (C . 2.) (*CH2OH* . 1.) (*NOH* . 1.))
 1.      .  CH2.CH2.OH    CH = N.OH,
 2.      C.. =  CH3    CH2.OH    N.OH,

MOLECULES
((U . 0.) (C . 1.) (O . 1.) (*CH2OH* . 1.) (*CHNH* . 1.))
 1.      .  CH2.CH2.OH    O.CH = NH,
 2.      CH...  OH    CH = NH    CH2.OH,

MOLECULES
((U . 0.) (C . 1.) (O . 1.) (*CH2OH* . 1.) (*NCH2* . 1.))
 1.      .  CH2.CH2.OH    O.N = CH2,

MOLECULES
((U . 1.) (C . 2.) (N . 1.) (O . 1.) (*CH2OH* . 1.))
 1.      =  CH.CH2.OH    CH.O.NH2,
 2.      =  CH.CH2.OH    N.O.CH3,
 3.      C = ..  CH2    CH2.OH    O.NH2,
 4.      C = ..  NH    CH2.OH    O.CH3,

MOLECULES
((U . 1.) (C . 3.) (O . 1.) (*NOH* . 1.))
```

TABLE 7.5 (Continued)

```
 1.     .  CH2.O.CH3    CH = N.OH,
 2.     .  O.C2H5     CH = N.OH,
 3.     .  CH..CH3 OH     CH = N.OH,
 4.    C.. =  CH3     O.CH3     N.OH,

MOLECULES
((U . 0.) (C . 2.) (*CHNH* . 1.))
*NO ALLOWABLE STRUCTURES

MOLECULES
((U . 0.) (C . 2.) (O . 2.) (*NCH2* . 1.))
*NO ALLOWABLE STRUCTURES

MOLECULES
((U . 1.) (C . 3.) (N . 1.) (O . 2.))
 1.     .  C3H7    O.N = O,
 2.     .  CH..CH3 CH3     O.N = O,
 3.     =  CH.O.CH3     CH.O.NH2,
 4.     =  CH.O.CH3     N.O.CH3,
```

*Substructures defined in GOODLIST are prevented from reappearing except under the corresponding superatom. Thus the final block of four molecules is the group containing none of the defined superatoms: *COOH*, *CO*, *CHNH2*, *CH2OH*, *NOH*, *CHNH*, *NCH2*. In many applications, the count of a given superatom will be set to zero for a particular context, or conversely, to non-zero. For example, the superatom *NCH2* is quite likely to be suppressed if the chemist knows that formaldehyde was not used in the synthesis of the molecule being analysed.

These computations are brought to the surface here only in order to reveal the heuristic revision of priorities that is available to DENDRAL. In actual problem solving, many of these hypotheses would be rejected long before a trial molecule was completed.

REFERENCE TO DATA

With these facilities we are now ready to attempt to apply DENDRAL to explicit data. The actual processes in the mass spectrometer are too complicated to be dealt with head-on in the first instance. We therefore deal with various models of

the behavior of the mass spectrometer, the theories of mass spectrometry. To exercise the simpler logical elements of heuristic DENDRAL, we begin with a zero order theory, one which postulates that the mass spectrum is obtained by assigning a uniform intensity to each fragment that can be secured by breaking just one bond in the molecule. We neglect the splitting of bonds affecting only a hydrogen atom. To test the program we do not at first use a real spectrum, but rather the spectrum predicted by this idealized theory for some given isomer.

As before, the predicter is deeply embedded within the DENDRAL generator, so that the structure building tree is truncated at the earliest point that a violation of the theory by the data set is encountered. This leads to a very efficient set of trials, not of completed, but of tentative and partial structures when the program is given a molecular composition and a hypothetical zero-order spectrum. The essence of the program is to generate all of the partitions at a given level, and then to scan these for compatibility with the mass list of the fragments. There are also some pertinent *a priori* considerations about the partitioning of molecular compositions, and this has been used to reorder the primary partitions in the most plausible sequence. We manage the sequence with which hypotheses are tested but still retain the exhaustive and irredundant character of the generator. Owing to imperfect memory and nonstandard formats, human judgment rarely succeeds so well at this.

Each of the plausibility operations plainly should and can be related to a statement of context. For example, in setting up the GOODLIST, the chemist will be interrogated about the likelihood of certain radicals, and cues for this can also be obtained directly from the mass spectrum. For example, the program is aware that mass number 45 is almost pathognomic for the radical -COOH. Hence, this superatom will be set to zero in the absence of a signal at that mass. Conversely, in a high-resolution analysis, the occurrence of mass number 44.998 would justify fixing -COOH as nonzero.

PERFORMANCE

The description, so far, characterizes an operational

program. Its main features can be routinely demonstrated without special preparation by remote teletypewriter interactions with the PDP-6 computer at Stanford University. DENDRAL has been tested in a number of ways in an attempt to evaluate its performance as a working tool. It will, of course, vastly outdo the human chemist in such contrived but potentially useful exercises as making an exhaustive and irredundant list of isomers of a given formula (Table 7.4 shows this for $C_3H_7NO_2$). In many cases, particularly when an adequate dictionary has been previously built and no further entries are being made, the computer will output its solutions at teletype speed. The program is also slightly faster than the human operator at subgraph-matching, that is, searching a series of molecular structures for the presence of any member of a given list of forbidden embedded subgraphs. It will outdo the human by approximately 100 : 1, or perhaps better if accuracy is given due weight in converting structural representations into canonical form and testing for isomorphism.

A few real spectra have been input, with surprisingly crisp results in view of the known imperfections of the zero order theory of mass spectrometry.

Thus heuristic DENDRAL was run with data on *threonine* obtained with a Bendix time-of-flight instrument (Fig. 7.2). The program returned two solutions, *threonine*, the correct structure, and one other (Fig. 7.3). The second isomer has not, to our knowledge, been analyzed by mass spectrometry. However, its spectrum can be predicted to resemble that of *threonine* very closely in its qualitative features.

When Dendral was challenged with $C_4H_9NO_3$ under the conditions of Table 7.4 it returned 238 "plausible isomers," of which only these two satify the data according to the program's model of the theory of mass spectrometry. The inclusion of the data shortens the computation time from about 30 minutes to about 3 minutes.

It is not easy to test the exhaustiveness of the DENDRAL generator without extensive files of known structures. However, it is possible to write recursive combinatorial expressions to count the expected numbers of isomeric alkane molecules (C_nH_{2n+2}) and alkyl radicals ($-C_nH_{2n+1}$) as shown in Table 7.6. These numbers have been verified by DENDRAL for radicals through C_9H_{19} and for molecules through $C_{12}H_{26}$, after which the LISP program structure becomes too unwieldy to

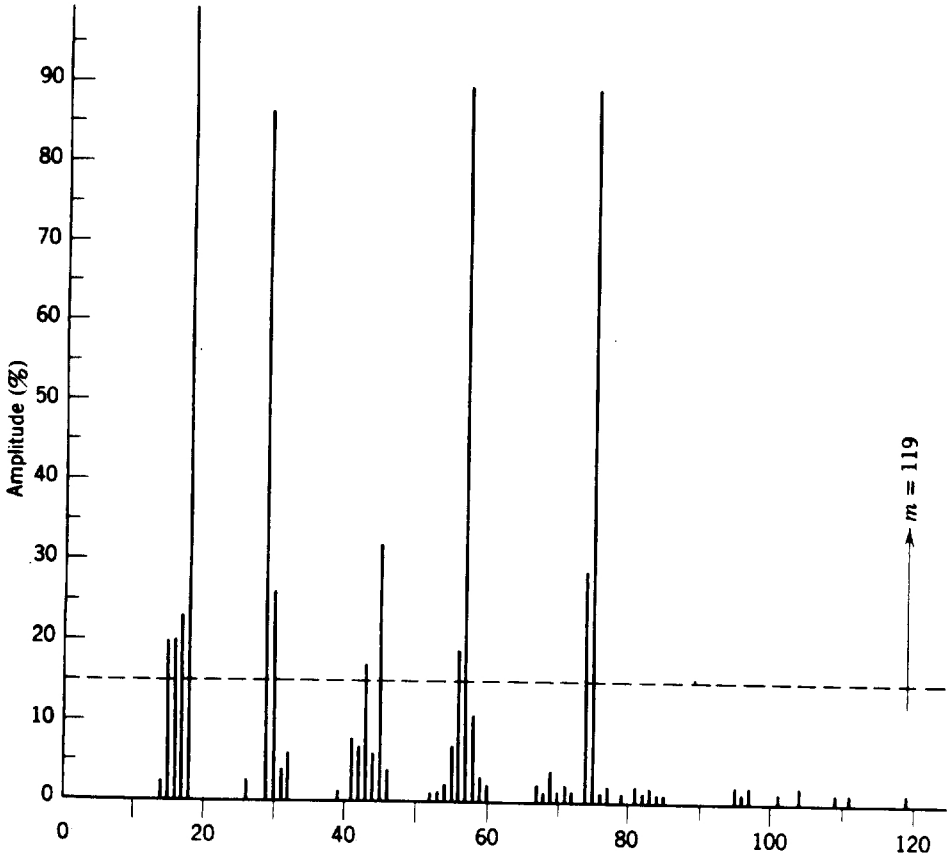
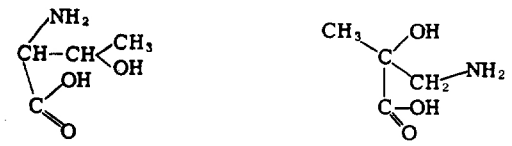


Fig. 7.2 The mass spectrum of threonine (CH...NH2 CH..CH3 OH C. =OH O) presented as a bar diagram from Martin (1965). This yielded a list of mass numbers: (15 16 17 18 29 30 43 45 56 57 74 75 119). This list was input to DENDRAL, which responded to (ISOMERS C4H9N03) with two solutions: 1. CH...NH2 CH..CH3 OH C. =OH O threonine
2. C....CH3 OH CH2.NH2 C. =OH O See Fig. 5.3.



(a) (b)

Fig. 7.3 (a) Threonine, (CH...NH2 CH..CH3 OH C. =OH O), and (b) 2-Methyl, 2-hydroxy, 3-aminopropionic acid, (C....CH3 OH CH2.NH2 C. =OH O).

TABLE 7.6 Counting the Isomeric Alkanes and Alkyl Radicals: C_nH_{2n+2} and C_nH_{2n+1}*

C	Alkane	Alkyl	C	Alkane	Alkyl
1	1	1	11	159	1238
2	1	1	12	355	3057
3	1	2	13	802	7639
4	2	4	14	1858	19241
5	3	8	15	4347	48865
6	5	17	16	10359	124906
7	9	39	17	24894	321198
8	18	89	18	60523	832019
9	35	211	19	148284	2156010
10	75	507	20	366319	5622109

*The figures were generated by a computer program following the algorithm of Henze and Blair (1931). Cayley's historic algorithm is incorrect, but is still quoted by a recent monograph on applications of graph theory (1965).

continue in core memory. Since there are no chemical prohibitions, the list (Table 7.7) of 75 isomeric decanes may illustrate the systematic combinatorial aspects of DENDRAL more vividly to a human reader than the preceding outputs do. Isomers are, of course, vastly more numerous for compositions containing some N and O atoms.

Facilities have been provided in the past, but are not available on our present computer system owing to hardware limitations, for providing two-dimensional graphic displays of structural maps as translations of DENDRAL notations. These programs also enabled man-computer interactions where the chemist could manipulate chemical structures to a substantial degree.

Where DENDRAL begins to be shaky is, as usual, when confronted with subtle changes of context which the user may often find difficult to communicate precisely to the program, even when he can do this readily to his fellow scientists. As far as possible we seek to get out of this difficulty by building interrogation subroutines into the program so that the chemist can provide data rather than obliging him to write new program text in the LISP language. Present efforts are concentrated on elaborating the theory of mass spectrometry as represented in

TABLE 7.7 The 75 Isomers of Decane, $C_{10}H_{22}$

```
(ISOMERS C10H22)

MOLECULES
((U . 0.) (C . 10.))
 1.     .  CH2.CH2.C3H7    CH2.CH2.C3H7,
 2.     .  CH2.CH2.C3H7    CH2.CH2.CH..CH3 CH3,
 3.     .  CH2.CH2.C3H7    CH2.CH..CH3 C2H5,
 4.     .  CH2.CH2.C3H7    CH2.C...CH3 CH3 CH3,
 5.     .  CH2.CH2.C3H7    CH..CH3 C3H7,
 6.     .  CH2.CH2.C3H7    CH..CH3 CH..CH3 CH3,
 7.     .  CH2.CH2.C3H7    CH..C2H5 C2H5,
 8.     .  CH2.CH2.C3H7    C...CH3 CH3 C2H5,
 9.     .  CH2.CH2.CH..CH3 CH3    CH2.CH2.CH..CH3 CH3,
10.     .  CH2.CH2.CH..CH3 CH3    CH2.CH..CH3 C2H5,
11.     .  CH2.CH2.CH..CH3 CH3    CH2.C...CH3 CH3 CH3,
12.     .  CH2.CH2.CH..CH3 CH3    CH..CH3 C3H7,
13.     .  CH2.CH2.CH..CH3 CH3    CH..CH3 CH..CH3 CH3,
14.     .  CH2.CH2.CH..CH3 CH3    CH..C2H5 C2H5,
15.     .  CH2.CH2.CH..CH3 CH3    C...CH3 CH3 C2H5,
16.     .  CH2.CH..CH3 C2H5    CH2.CH..CH3 C2H5,
17.     .  CH2.CH..CH3 C2H5    CH2.C...CH3 CH3 CH3,
18.     .  CH2.CH..CH3 C2H5    CH..CH3 C3H7,
19.     .  CH2.CH..CH3 C2H5    CH..CH3 CH..CH3 CH3,
20.     .  CH2.CH..CH3 C2H5    CH..C2H5 C2H5,
21.     .  CH2.CH..CH3 C2H5    C...CH3 CH3 C2H5,
22.     .  CH2.C...CH3 CH3 CH3    CH2.C...CH3 CH3 CH3,
23.     .  CH2.C...CH3 CH3 CH3    CH..CH3 C3H7,
24.     .  CH2.C...CH3 CH3 CH3    CH..CH3 CH..CH3 CH3,
25.     .  CH2.C...CH3 CH3 CH3    CH..C2H5 C2H5,
26.     .  CH2.C...CH3 CH3 CH3    C...CH3 CH3 C2H5,
27.     .  CH..CH3 C3H7    CH..CH3 C3H7,
28.     .  CH..CH3 C3H7    CH..CH3 CH..CH3 CH3,
29.     .  CH..CH3 C3H7    CH..C2H5 C2H5,
30.     .  CH..CH3 C3H7    C...CH3 CH3 C2H5,
31.     .  CH..CH3 CH..CH3 CH3    CH..CH3 CH..CH3 CH3,
32.     .  CH..CH3 CH..CH3 CH3    CH..C2H5 C2H5,
33.     .  CH..CH3 CH..CH3 CH3    C...CH3 CH3 C2H5,
34.     .  CH..C2H5 C2H5    CH..C2H5 C2H5,
35.     .  CH..C2H5 C2H5    C...CH3 CH3 C2H5,
36.     .  C...CH3 CH3 C2H5    C...CH3 CH3 C2H5,
37.    CH...  CH3    CH2.C3H7    CH2.C3H7,
38.    CH...  CH3    CH2.C3H7    CH2.CH..CH3 CH3,
39.    CH...  CH3    CH2.C3H7    CH..CH3 C2H5,
40.    CH...  CH3    CH2.C3H7    C...CH3 CH3 CH3,
41.    CH...  CH3    CH2.CH..CH3 CH3    CH2.CH..CH3 CH3,
42.    CH...  CH3    CH2.CH..CH3 CH3    CH..CH3 C2H5,
43.    CH...  CH3    CH2.CH..CH3 CH3    C...CH3 CH3 CH3,
44.    CH...  CH3    CH..CH3 C2H5    CH..CH3 C2H5,
45.    CH...  CH3    CH..CH3 C2H5    C...CH3 CH3 CH3,
46.    CH...  CH3    C...CH3 CH3 CH3    C...CH3 CH3 CH3,
47.    CH...  C2H5    C3H7    CH2.C3H7,
48.    CH...  C2H5    C3H7    CH2.CH..CH3 CH3,
49.    CH...  C2H5    C3H7    CH..CH3 C2H5,
```

TABLE 7.7 (Continued)

50.	CH...	C2H5	C3H7	C...CH3 CH3 CH3,
51.	CH...	C2H5	CH..CH3 CH3	CH2.C3H7,
52.	CH...	C2H5	CH..CH3 CH3	CH2.CH..CH3 CH3,
53.	CH...	C2H5	CH..CH3 CH3	CH..CH3 C2H5,
54.	CH...	C2H5	CH..CH3 CH3	C...CH3 CH3 CH3,
55.	CH...	C3H7	C3H7	C3H7,
56.	CH...	C3H7	C3H7	CH..CH3 CH3,
57.	CH...	C3H7	CH..CH3 CH3	CH..CH3 CH3,
58.	CH...	CH..CH3 CH3	CH..CH3 CH3	CH..CH3 CH3,
59.	C....	CH3	CH3	C3H7 CH2.C3H7,
60.	C....	CH3	CH3	C3H7 CH2.CH..CH3 CH3,
61.	C....	CH3	CH3	C3H7 CH..CH3 C2H5,
62.	C....	CH3	CH3	C3H7 C...CH3 CH3 CH3,
63.	C....	CH3	CH3	CH..CH3 CH3 CH2.C3H7,
64.	C....	CH3	CH3	CH..CH3 CH3 CH2.CH..CH3 CH3,
65.	C....	CH3	CH3	CH..CH3 CH3 CH..CH3 C2H5,
66.	C....	CH3	CH3	CH..CH3 CH3 C...CH3 CH3 CH3,
67.	C....	CH3	C2H5	C2H5 CH2.C3H7,
68.	C....	CH3	C2H5	C2H5 CH2.CH..CH3 CH3,
69.	C....	CH3	C2H5	C2H5 CH..CH3 C2H5,
70.	C....	CH3	C2H5	C2H5 C...CH3 CH3 CH3,
71.	C....	CH3	C2H5	C3H7 C3H7,
72.	C....	CH3	C2H5	C3H7 CH..CH3 CH3,
73.	C....	CH3	C2H5	CH..CH3 CH3 CH..CH3 CH3,
74.	C....	C2H5	C2H5	C2H5 C3H7,
75.	C....	C2H5	C2H5	C2H5 CH..CH3 CH3,

the predictor subprogram. This is giving very promising results, the chief limitations being (1) the precise definition of the rules actually used by the chemist and operant in nature, and (2) the translation of these conceptual algorithms into viable program. These two issues are, however, not as independent as might be imagined. It is the clumsiness of the program writing and debugging that impedes rapid testing of the correctness with which a rule has been formulated. In our experience each half hour of conference has generated approximately a man-month of programming effort. It is obvious that despite the simplicity of the DENDRAL notation for chemical structures, we still have a long way to go in the development of a language for the simple expression of other conceptual constructs of organic chemistry, particularly context definitions and reaction mechanisms.

Insofar as programs are also graphs and an effective subroutine may be regarded as a hypothesis that matches its intended functions, the latter being both logically deducible and

operationally testable by running the subroutine, program writing may be regarded as an inductive process roughly analogous to the induction of structural formulas as solutions to sets of chemical data. We believe it may be necessary to produce a solution to this meta language puzzle before the implementation of human ideas in computer subroutines can proceed efficiently enough for the rapid and effective transfer of human insights into machine judgment. Nevertheless, by the rather laborious process that we have outlined, DENDRAL has proceeded to that stage of sophistication where it is at least no longer an occasion of embarrassment to demonstrate it to our scientific colleagues and friends who have no interest whatsoever in computers per se.

The deferral of cyclic structures will weaken the casuistic impact of the program upon chemists. However, the acyclic molecules give sufficient play for analyzing the inductive process. Furthermore, it may be advantageous to leave a blemish that diminishes the latent threat of artificial intelligence to human aspirations. However, a complete notation and specifications for cyclic DENDRAL have been documented (Lederberg, 1965) and this is being programmed now in response to the utilitarian demands of chemist friends.

BUILDING DENDRAL

DENDRAL was developed in the LISP 1.5 and 1.6 dialects. The original package was composed by Mr. William White working from the specifications summarized in Table 7.1, and a version of DENDRAL which almost worked was generated on the IBM 7090 with the help of a time-shared editing system run on the PDP-1. When the LISP system on System Development Corporation's Q-32 became available to us, we pursued a vigorous programming effort by remote teletype communication from Stanford to Santa Monica. This proved to be a very powerful and remarkably reliable system, and Mr. White and Mrs. Georgia Sutherland perfected the program (Sutherland, 1967) on that computer with a total effort of about one year.

In retrospect it is quite obvious that the program simply could never have been written and debugged without the help of the rapid interaction provided by the time-sharing system. We stress *never* advisedly, in the light of our own experience

with the human frustrations involved in the typical turnaround times for error detection and error correction under the operating system for the IBM 7090. In November 1966 we moved our operations to LISP 1.5 on the PDP-6 computer installed for the Artificial Intelligence Project at Stanford. Despite the avowed close compatibility of the LISP systems, approximately 3 man-months of effort were required to transfer the program from one dialect to the other.

PATTERN RECOGNITION

As the candidate structures become more and more complex we have to abandon the idea of exhaustive enumeration of possible structures. Instead, the data are scrutinized for cues that offer any preference for certain kinds of structures as starting points. As we keep examining the problem we do find more and more ways in which such cues can be exploited. For example, an elementary pattern analysis of the period with which mass numbers are represented, for example, for gaps in the sequence of mass numbers with significant intensity around a period of about 14 mass units (CH_2), can give significant hints about the existence of a number of branch points within the molecule. If these can be limited, the extent of the necessary tree building can be drastically curtailed from first principles. Likewise, an examination of mass numbers approximating half the total molecular weight can lead to some trial hypotheses about the major partition of the molecule, which again can truncate the development. We do not, however, yet have a program sophisticated enough to make a profound reexamination of its own strategy at any level more complicated than the resetting of numerical parameters, a limitation closely related to the meta language challenge mentioned above. In sum, we find that the development of this program has not encountered very much that is fundamentally new in principle: problem solving in this field has much the same flavor as the solutions already adduced for chess, checkers, theorem proving, etc. One possible advantage of pursuing investigations in artificial intelligence and heuristic programming within this framework is that the practical utility of what has already been produced should suffice to engage the attention of a considerable number of human chemists

working on practical problems in a fashion that lends itself to machine observation and emulation of their techniques.

PROGRAMMING AS INDUCTION

The game of writing programs becomes more and more an experimental science as the complexity of the programs increases. At the limit, the programmer has the insecure hope that his text will (1) run and (2) accomplish the intended goals, that is, his program is a hypothesis that needs deductive elaboration to verify it. This suggests that program writing ought to be mechanized by a process analogous to the induction of chemical hypotheses by DENDRAL and starting with mechanized observations of human techniques of problem-solving.

The pervasive role of analogy in human judgment suggests that much would be gained in artificial intelligence if a large compatible tool kit of successful programs were available both to the human and the mechanized programmer. Unfortunately, artificial intelligence researchers go to such excesses in their originality and improvisation of idiosyncratic dialects, that there is no easy way in which past successes of unpredictable relevance can be immediately tried out for a new problem. Experimental science, on the other hand, is replete with important advances that resulted from the provocative availability of a new technique waiting on the shelf to find a use. Indeed, mass spectrometry itself has exactly that history.