12

On Generality and Problem Solving: A Case Study Using the DENDRAL Program

E. A. Feigenbaum, B. G. Buchanan and J. Lederberg
Computer Science Department
Stanford University

In discussing the capability of a problem-solving system, one should distinguish between generality and expertness. Generality is being questioned when we ask: how broad a universe of problems is the problem solver prepared to work on? Expertness is being questioned when we ask: how good are the answers and were they arrived at with reasonable cost? Generality has great utility in some ways, but is not often associated with superior performance. The experts usually are specialists.

In analytical chemistry, there is a domain of inductive inference problems involving the determination of molecular structure by analysis of certain physical spectra of the molecule. We have written a problem-solving program (Heuristic DENDRAL) that is prepared to attempt to solve any problem in this very large domain. By now, it has solved hundreds of structure-determination problems in many different chemical families. For some families of molecules, it is an expert, even when compared with the best human performance. For the other families, i.e., most of chemistry, it performs as a novice, or worse.

This paper will use the design of Heuristic DENDRAL and its performance on many different problems it has solved as raw material for a discussion of the following topics:

1. the design for generality;
2. the performance problems attendant upon too much generality;
3. the coupling of expertise to the general problem-solving processes;
4. the symbiotic relationship between generality and expertness, and the implications of this symbiosis for the study and design of problem-solving systems.

We conclude the paper with a view of the design for a general problem-solver that is a variant of the 'big switch' theory of generality.

Previous papers have given a detailed exposition of the workings of the Heuristic DENDRAL program (Buchanan *et al.* 1969) and a discussion of some general issues of representation and theory formation suggested by the DENDRAL work (Buchanan *et al.* 1970). It is fair to ask for an integrated presentation of the results of this application of heuristic programming to an important chemical inference problem. Several papers presenting these results to chemists have appeared or are in press (Lederberg *et al.* 1969, Duffield *et al.* 1969, Schroll *et al.* 1969, Buchs *et al.* 1970), but no summary of these results is available in the artificial intelligence literature.

Yet the attention given to the program as an application of artificial intelligence research has tended to obscure the more general concerns of the project investigators. These are:

1. To study and construct detailed information processing models of processes of scientific inference. By scientific inference we mean the inferential process by which a model is constructed to explain a given set of empirical data.
2. To study experimentally the 'operating characteristics' and the effectiveness of different designs (strategies) for the deployment of task-specific knowledge in a scientific area.
3. To develop a method for eliciting from an expert the heuristics of scientific judgement and choice that he is using in the performance of a complex inference task.
4. To solve real problems in an area of significance to modern science, and to do so with a level of performance high enough to have a noticeable impact upon that area of science.
5. To discover the heuristics which lie behind efficient selection. As we conclude later, the significant problem may not be so much 'tuning' a specialist with a new set of heuristics as learning how to acquire these heuristics.

THE TASK ENVIRONMENT

For the sake of completeness and review, we include here a brief description of the scientific problem that was chosen as the task environment in which to pursue the project's goals (publications listed in the references will give the interested reader the complete story). The problem given to the program is the usual problem of the analytical chemist: to determine the molecular structure of an unknown compound. While the chemist may use many analytical techniques, the program uses only two of the most important tools to collect data about the unknown sample. The primary source of empirical data is a mass spectrometer, an instrument that fragments molecules of a chemical sample (using an electron beam) and records the results. A mass spectrum, the output of the mass spectrometer, is a two-dimensional record of the abundance of various fragments plotted as a function of their molecular weights. A secondary source of data is a nuclear magnetic resonance (NMR) spectrometer, which uses variations in magnetic field strengths to provide information about certain specific kinds of structure internal to a molecule.

(In addition, there is no difficulty in utilizing a third source of data, the infrared (IR) spectrometer, as soon as it becomes sufficiently important to do so.)

The problem solver is given the mass spectrum, the NMR spectrum if it is available, and the elementary formula if it is available (number of atoms of each element). For the classes of molecules reported in this paper, the program need not be given the formula but can infer it directly from the spectrum by a heuristic procedure.

The output of the problem solver is a graph, that is, a topological model, of the molecular structure of the unknown compound. Or, if more than one graph is a plausible explanation of the given data, the output is a list of the plausible molecular graphs, rank ordered, with their relative plausibility scores.

The determination of molecular structure by these electronic instrumental techniques is seen by physical chemists to be a significant advance over older chemical methods, and is enticing because of the speed and economy of the analysis and the generality of the approach. However, the almost bewildering variety of fragmentations and reactions that can be induced by the high energy of the electron beam in a mass spectrometer are far from being completely understood, so that the science of mass spectrum analysis, though no longer an infant, has still not reached its maturity.

GENERALITY VERSUS SPEED AND ECONOMY

> 'A view of existing problem solving programs would suggest, as common sense would also, that there is a kind of "law of nature" operating that relates problem solving generality (breadth of applicability) inversely to power (solution successes, efficiency, etc.) and power directly to specificity (task-specific information).' (Feigenbaum 1968)
>
> 'Evidently there is an inverse relationship between the generality of a method and its power. Each added condition in the problem statement is one more item that can be exploited in finding the solution, hence in increasing the power.' (Newell 1969)

One does not need a view of generality in problem-solving systems of the scope of GPS (Ernst and Newell 1969) to appreciate the importance of this tradeoff between generality (breadth of applicability) and effectiveness in solving a given problem (particularly speed and cost). The story of the DENDRAL program's success as an application is in part a story of this tradeoff, which the remainder of this paper will sketch. We approach this discussion of generality of problem-solving systems with some caution since the history of the search for generality in problem solvers (primarily the GPS effort) will tend to color the discussion no matter what we say or do not say about it.

Structure determination by mass spectral analysis is a technique pursued by its scientific practitioners because of its generality: its broad applicability

to all types of molecules. The designer of a problem-solving system to interface with this empirical data is inclined, at least initially, to try to match the generality of the physical process with generality of the reasoning process. Yet he soon finds, paradoxically, that he cannot afford this match, that he must retreat and rework his analysis into more specialized forms if he is to be able to use his problem solver on real problems.

The Heuristic DENDRAL program has solved hundreds of structural inference problems, most recently of structures in the family of organic amines, for which the analysis is reasonably complex. The difference in running speed between solving these problems by the most general methods known to the program and solving them by its heuristic methods specialized for this type of problem is estimated to be as large as a factor of thirty thousand!

The world known to the DENDRAL program is the world of organic chemical structures. For the purposes of this paper DENDRAL's world will be taken to be the world of non-ringed (acyclic) organic molecules, although not all parts of the program are so constrained. [As of July 1970, the Structure Generator could delineate all acyclic isomers and all mono-cyclic (single-ringed) isomers of a given chemical formula, the Predictor could predict mass spectra for acyclic molecules (and manipulate the internal structure of any cyclic molecules), and the Planner could infer structural information from the spectral data of any saturated acyclic monofunctional molecule.]

In the discussion to follow generality will mean breadth of applicability within the confines of the DENDRAL world. Some procedures apply to all possible structures in this world, and they will be considered the most general. If there were a procedure that applied only to a single molecule, that procedure would be the least general. Thus, generality is to be taken to mean relative generality in the DENDRAL world.

THE GENERAL PROBLEM-SOLVERS OF THE DENDRAL WORLD

In another place (Lederberg *et al.* 1970), we have summarized our overall design philosophy as follows:

'Some of the essential features of the DENDRAL program include:

1. Conceptualizing organic chemistry in terms of topological graph theory, that is, a general theory of ways of combining atoms.
2. Embodying this approach in an exhaustive hypothesis generator. This is a program which is capable, in principle, of "imagining" every conceivable molecular structure.
3. Organizing the generator so that it avoids duplication and irrelevancy, and moves from structure to structure in an orderly and predictable way. The key concept is that induction becomes a process of efficient selection from the domain of all possible structures. Heuristic search and evaluation are used to implement this efficient selection.'

This is a design philosophy which is clearly aimed at the most general kind of problem-solving capability within the DENDRAL world, that is, any mass spectrum and associated chemical formula within the DENDRAL world can be treated.

From another point of view, the DENDRAL program can be seen to be implemented within a generate-and-test paradigm, to use Newell's terminology (Newell 1969). The 'generate' part is the Structure Generator program and the 'test' part is the Predictor program. Hypothesis generation and hypothesis validation are equally appropriate labels for these two stages of the problem solving.

The Structure Generator incorporates:

1. an algorithm that allows it to proceed systematically from one possible candidate to the next, that is, a legal move generator that defines the space;
2. general criteria for instability of organic molecules that allow it to avoid working on chemically irrelevant structures;
3. procedures for treating subgraphs as if they were atoms, allowing particularly important combinations of atoms to be treated as a unit in the combinatorial work of the generator. Because of the structure of molecular graphs, this task environment lends itself to partial solutions using the techniques described below.

The Structure Generator program knows nothing of the theory of mass spectrometry. Given a chemical formula, it will generate all the isomers (structural variants) that are chemically plausible *a priori*. These are the candidates that are input to the 'test' part of the generate-and-test procedure.

The Structure Generator, even when used alone, has performed valuable

Table 1. Numbers of possible non-cyclic molecular structures of selected formulas. These numbers define the size of the search space for problems involving molecules of a given chemical formula. The size of the space increases dramatically with both the number of carbon atoms and the number of other types of atoms in the formula. This table is abstracted from Lederberg *et al.* (1969).

	Number of carbon atoms						
Chemical Formula	4	5	6	7	8	9	10
$C_nH_{(2n+2)}$	2	3	5	9	18	35	75
$C_nH_{(2n+2)}O$	7	14	32	72	171	405	989
$C_nH_{(2n+3)}N$	8	17	39	89	211	507	1238
$C_nH_{(2n+3)}NO$	50	137	365	995	2727	—	—

service for chemists by exhibiting the sizes and structures of the analytical chemist's problem spaces. The number of chemically-possible structural models, as shown in table 1, is an important boundary on a chemist's problem hitherto known only for a few classes of problems (*see* Lederberg *et al.* 1969).

The Predictor program is the 'expert' on the general theory of mass spectrometry. It answers this question for the system: Though the candidate may be chemically plausible on *a priori* grounds, is it a good candidate to explain the given mass spectrum? In other words, does its predicted spectrum fit the data?

Table 2. Amino acid results

Name of 'unknown'	Chemical formula	Size of problem space[1]	Number of plausible structures[2]	Number of structures generated[3]	Rank order of correct candidate[4]
Glycine	$C_2H_5NO_2$	38	12	8	1st, 7 excluded
Alanine	$C_3H_7NO_2$	216	50	3	1st
Serine	$C_3H_7NO_3$	324	40	10	1st, 9 excluded
Threonine	$C_4H_9NO_3$	1758	238	2	1st
Leucine	$C_6H_{13}NO_2$	10000 (approx.)	3275	288	Tied for 2nd, 277 excluded

[1] The total size of the problem space is the number of topologically-possible molecular structures generated within valence considerations alone.
[2] The number of plausible structures is the number of molecular structures in the total space which also meet *a priori* conditions of chemical stability. The *a priori* rules have greater effect with increased numbers of non-carbon, non-hydrogen atoms.
[3] The number of structures generated is the number of molecular structures actually generated by the program as candidate explanations of the experimental data. Pruning has been achieved by using the 'zero-order' theory during structure generation.
[4] The rank order of the correct structure is the validation program's assignment of rank to the actual molecular structure used as a test 'unknown'. The number of structures excluded in the validation process is also indicated.

The Predictor incorporates a general theory of the fragmentation and recombination processes that can take place in a mass spectrometer, insofar as these are known to our chemist collaborators. The Predictor program is continually under development as the theory of mass spectrometry develops.

Any chemical structure in the DENDRAL world can be handled by the Predictor. In this sense, the Predictor is as general a problem-solving element as the Structure Generator; in fact, it is the necessary complement.

The Heuristic DENDRAL program contains a great deal more than just

this generate-and-test team, as will be described subsequently. But it is instructive to ask: How powerful are these 'generalists' in solving mass spectral analysis problems?

Table 2 exhibits the results for selected members of the family of amino acids. This family is distinguished from the other families with which we have worked by virtue of containing a relatively large number of heteroatoms (atoms not carbon or hydrogen) relative to the number of carbon atoms. For each entry, we give its common name, its chemical formula, the size of the problem space in terms of the number of topologically-possible isomers, the number of chemically-plausible isomers actually generated by the Structure Generator (using the 'zero-order' theory explained below), and the rank order assigned to the correct candidate (that is, the 'right answer') by the Predictor. It will be seen that the heuristics concerning unstable molecules have a substantial effect for amino acids, i.e., the number of chemically-plausible molecules is much less than the number of topologically-possible candidates. This will not in general be true for molecules with fewer types of atoms, for example, ketones, ethers, and amines, as we shall see later.

PROBLEMS ATTENDANT UPON TOO MUCH GENERALITY

Experiments such as those just summarized pointed up design problems that were consequences of the program's generality. As a result of having to be prepared to handle in a homogeneous and complete manner any formula or any structure presented, the programs are costly in terms of computer running time and use of main memory. With respect to the Predictor, this means that it is feasible to test only a relatively small number of candidate solutions. In the Structure Generator, this means that it is feasible to start with only a small collection of atoms.

The generality of the Structure Generator, which employs only relatively weak *a priori* constraints and no constraints imposed by the data, tends toward producing too many 'plausible' candidates. The generate-and-test procedure breaks down because the generator is too prolific and the test is too expensive.

The solution to this design problem is to strengthen the heuristic controls over the generation of candidate solutions. There are a number of ways available to do this, some of which were tried with success, some with failure. The failures were at least as illuminating as the successes.

The most obvious way will be mentioned first, and then discussed no further in this paper. It is this: review carefully the tricks in the heuristic programmer's toolkit (particularly those that apply to the search of AND–OR problem reduction trees) and do not fail to apply them when they are applicable. The following examples from the Structure Generator illustrate the point:

1. At an OR node (in DENDRAL, the selection of a particular partitioning of the remaining unassigned atoms), try the easiest subproblem first. At an

AND node (in DENDRAL, making radicals from partition elements), try the hardest subproblem first.
2. Limit the number of subproblems considered at an OR node by evaluating the 'quality' of subproblems and discarding those below a threshold value.
3. For difficult problems, allow human intervention in the choice of subproblems (this potentially-powerful heuristic procedure is available in DENDRAL, but has never been used in solving problems).

HEURISTICS RELATED TO PROBLEM DATA: THE EMERGENCE OF 'SPECIALISTS'

By far the most powerful method of gaining effective control over the generator is to force its search to be relevant to specific problem data given as the input (the spectral data). That is, the candidates produced by the generator must be not only chemically plausible *a priori* but also likely solutions to the specific problem at hand.

In DENDRAL, one method for doing this is as follows: whenever a move in the problem space defines a new piece of an emerging structure, validate the move with respect to mass spectral theory by predicting its consequences in terms of expected spectral lines; and prune moves that cannot be so validated. In other words, reduce the search in light of the problem data by applying the theory of mass spectrometry to nodes in the problem space. For example, prune all structures to be built out of a cluster of 2 carbon atoms, 3 hydrogens, and 2 oxygens if there is no corresponding data point (mass = 59). A simple version of this method was used in early versions of the DENDRAL program. The theory of mass spectrometry used was so oversimplified that we called it derisively 'the zero-order theory of mass spectrometry'. Yet it turned out to be a cheap and effective pruning criterion for some problems, namely, the amino acids, for whose fragmentation the zero-order theory was not a bad theory.

The zero-order theory failed, of course, on more complex problems, but a better theory was available, the general theory in the Predictor. A procedure was developed by which the Predictor was called every time there was a need for validation of a partial structure.

When in doubt, consult the 'generalist'! But the design experiment failed, for these reasons:
1. The 'generalist', as we have said, is too expensive even for partial structures; and it was called too frequently.
2. The theory is most powerful in making statements about fragmentation at termini of chemical graphs; but the Structure Generator builds candidate graphs by starting at the center of the graph and building toward the termini. Thus the theory was most powerful precisely when it was having the least heuristic effect! This representational mismatch could have been remedied by considerable reprogramming (although a total correction would have benefited by a complete reconceptualization and reprogramming of the

Structure Generator), but it points up how critical are the problems of representation when one considers using the knowledge held by one process to control another.

There are other heuristic methods available in this concrete, running program, however. These we shall call 'aggregation' and 'planning'. Both have general (and well-recognized) importance quite apart from their power in the DENDRAL application. In DENDRAL, both are employed prior to the search for candidate solutions, and serve to 'preset' the generator to work only on those families of structures that meet certain conditions inferred from the problem data. To be effective, these processes must be cheap, relative to a search unconstrained by their inferences. As we shall see, this is achieved by the use of highly-specialized rules for interpreting the 'meaning' of the problem data (spectral lines). These rules are the formal representation of what the chemist considers to be his good judgement in properly organizing his inference problem.

Aggregation is a self-evident general technique for reducing the number of alternatives produced by any combinatorial generator. Aggregate the combinatorial elements into bigger units and treat these as if they were elements. In DENDRAL, any subgraph can be treated as a 'superatom' with a valence. The internal structure of the superatom is not manipulated by the combinatorial generator.

The most general view of the aggregation heuristic in DENDRAL is this: Use whatever specialized knowledge and processes and whatever auxiliary data are available to infer pieces (partial structures) of the solution. Make these superatoms. For the remaining atoms, uncommitted to superatoms, use the general structure-generating machinery to build the interstitial structures in all the ways allowed by the heuristics defining chemical plausibility.

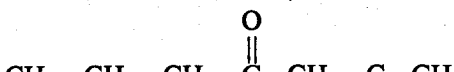
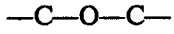
This general approach has been used in many particular ways. For example:

1. The Structure Generator can be supplied with a list of superatoms that are known *a priori* to be highly stable and therefore likely to occur in nature.
2. A nuclear magnetic resonance spectrum, important auxiliary data to a mass spectrum analysis, often provides clear and easily-obtained information about the number of methyl superatoms (CH_3) in the structure. Infra-red and ultra-violet spectra can reveal other kinds of substructure, which can be similarly treated as superatoms.
3. The key subgraphs of a molecule (those containing the heteroatoms) usually leave their particular 'fingerprints' in the lines of the mass spectrum. Complex pattern recognition criteria have been developed by us for identifying these key subgraphs, which are then treated as superatoms. A few of these rules are shown in table 3.
4. Sequence extrapolation and deft numerology have been used to infer some simple structures, such as the longest unbranched chain in the molecule. Once identified, they become superatoms.
5. By direct human intervention, any aggregation – any superatom – can be

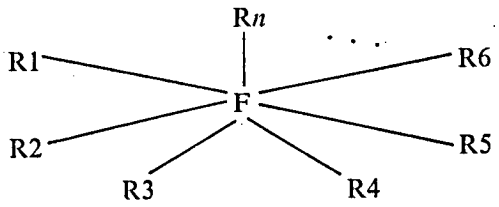
established. This is of great importance when the program is used as an 'assistant' in a very complicated problem. The human chemist often knows in advance basically what kind of structures he is working with, that is, he knows most of the structure *ab initio*. The known piece of structure is input as a superatom; DENDRAL then is of assistance in analyzing the unknown part and connecting all parts to form complete molecules.

Aggregation, as just described, is a part of the more formal, more organized,

Table 3. Heuristics used for identifying three superatoms. *See* Duffield *et al.* (1969), Schroll *et al.* (1969), and Buchs *et al.* (1970a) for fuller discussions of these and other sets of heuristics used in planning.

Superatom Name	Structure	Identifying conditions
Ketone		There are 2 peaks at mass units $x1$ & $x2$ such that (a) $x1+x2=M+28$, (b) $x1-28$ is a high peak, (c) $x2-28$ is a high peak, (d) At least one of $x1$ or $x2$ is high.
N-Propyl Ketone3		1. 71 is a high peak, 2. 43 is a high peak, 3. 86 is a high peak, 4. 58 appears with any intensity.
Ether		1. M-18 is a peak of 0 or 1% intensity, 2. M-17 is a peak of 0 or 1% intensity, 3. There are 2 peaks corresponding to the alpha-cleavage fragments.

more complete heuristic process in DENDRAL that we call planning. [The aggregation heuristics are currently the most important parts of our planning process, but not the only parts. For example, the heuristics which infer the weights of radicals attached to the central subgraph for later use in search control in the generator are not aggregation heuristics. Planning, in our view, can be a much broader process than just aggregation. A plan can contain any information that subsequently will be useful in controlling the search for solutions.] We have organized the planning process around a planning model shown below, where F is the key subgraph of the molecules (that which determines its chemical family), and R1 . . . R*n* are the subgraphs (radicals) that are connected to it. At the planning stage in a particular analysis, more than one F may be possible. The number of radicals attached to the various possible Fs may differ.



A plan given to the Structure Generator by the Planner consists of:

1. one or more Fs, as superatoms;
2. for each F, the 'molecular' weights of the radicals attached to the various valence bonds;
3. other information about aggregation.

The plan delineates the subset of the set of all plausible structures that will be allowed as solution candidates. In effect, it determines that the search for solutions will take place in some particular subtree of the DENDRAL space. How far below the root of the tree (that is, how much of the 'upper levels' need not be searched) is a function of how much aggregation there is in the Fs.

In the early forms of the planning process (previously called a 'preliminary inference' process), the Fs and the pattern recognition rules for identifying Fs were determined in a basically *ad hoc* fashion, by the thorough, careful, but painstaking technique involving chemist, computer, and DENDRAL staff member that has been described as 'Eliciting a Theory from an Expert' (Buchanan *et al.* 1970). In a series of carefully-chosen steps up the ladder of structural and mass spectral complexity, heuristically-powerful sets of Fs and rules for the acyclic monofunctional (that is, one F at a time) chemical families were worked out. The aggregation heuristics previously discussed were employed. The Planner developed into the system's 'specialist' on the meaning of spectral lines – a collection of special facts and special-purpose heuristics organized around particular chemical families.

The use of the Planner as a specialist controlling a general search process is powerful. Results for the analysis of mass spectra of the chemical families of ketones and ethers are illustrative. See tables 4 and 5. The differences between numbers of structures in the columns labeled 'Number of chemically plausible structures' and the columns labeled 'Number of structures generated' exhibit the power of planning in limiting search in these problems.

Table 4. Ketone results.

Name of 'unknown'	Chemical formula	Size of problem space[1]	Number of plausible structures[2]	Number of structures generated[3]	Rank order of correct candidate[4]
2-Butanone	C_4H_8O	11	11	1	1st
3-Pentanone	$C_5H_{10}O$	33	33	1	1st
3-Hexanone	$C_6H_{12}O$	91	91	1	1st
2-Methyl-hexan-3-one	$C_7H_{14}O$	254	254	1	1st
3-Heptanone	$C_7H_{14}O$	254	254	2	Tied for 1st
3-Octanone	$C_8H_{16}O$	698	698	4	1st
4-Octanone	$C_8H_{16}O$	698	698	2	1st, 1 excluded
2,4-Dimethyl-hexan-3-one	$C_8H_{16}O$	698	698	4	Tied for 1st, 1 excluded
6-Methyl-heptan-3-one	$C_8H_{16}O$	698	698	4	1st
3-Nonanone	$C_9H_{18}O$	1936	1936	7	1st
2-Methyl-octan-3-one	$C_9H_{18}O$	1936	1936	4	1st[5]
4-Nonanone	$C_9H_{18}O$	1936	1936	4	1st[5]

[1] The total size of the problem space is the number of topologically-possible molecular structures generated within valence considerations alone.
[2] The number of plausible structures is the number of molecular structures in the total space which also meet *a priori* conditions of chemical stability. The *a priori* rules have no effect with formulas containing a single non-carbon, non-hydrogen atom.
[3] The number of structures generated is the number of molecular structures actually generated by the program as candidate explanations of the experimental data. Pruning has been achieved by using the planning information from the Planning program.
[4] The rank order of the correct structure is the validation program's assignment of rank to the actual molecular structure used as a test 'unknown'. The number of structures excluded in the process is also indicated.
[5] Previous publication showed the correct structure excluded. The general rules of the program have since been modified to improve its performance.

THE PLANNING PROCESS

The primary fact of life for heuristic program designers is that increases in complexity of problems are accompanied by exponential increases in the size of the problem spaces to be searched. Successful heuristic designs cope by increasing the number and/or power of the heuristics to match increases in the size of the space.

The chemical family of amines presents such a challenge for DENDRAL. Amines contain a nitrogen atom as the key heteroatom. Since nitrogen has three valence bonds compared with oxygen's two, amines represent the next logical step up in complexity from the ketones and ethers. For any fixed number of carbon atoms, there are many more amines than either ketones or ethers. That is, there is a marked increase in the size of the spaces to be searched.

Early experiments with amines showed the usual pattern of system breakdown symptomatic of too little heuristic power for the size of the spaces. Since for amines the *a priori* stability heuristics that define chemical plausibility for the generator have little or no heuristic power, all of the heuristic control over the generator must come from the plan. Producing plans simply by extrapolating the techniques used for the ketone and ether families was grossly inadequate.

In such a situation, a sensible design change is to give the Planner the ability to specify more completely the form of acceptable solution candidates. The generator is thereby constrained to search a smaller space. One way to do this is by more aggregation – to cause more pieces or larger pieces of structure to be 'predetermined' by special-purpose inference schemes.

In the DENDRAL development, increased aggregation in the planning stage was designed in as follows:

1. In a systematic way, the size of the Fs was increased to incorporate more carbon and hydrogen atoms. If the set of Fs is to be logically complete within the size bound chosen, then by the ordinary combinatorics, the number of possible Fs from which selections will be made must increase. This complicates the classification decision by which it is inferred that the spectral data indicates a particular F (or set of Fs).

The systematic method used for enumerating the set of Fs for amines was chosen very carefully to mate best with that part of the theory of mass spectrometry that seemed most powerful in aiding the classification decision. The system for constructing the Fs and the mass spectral theory to which it mates (alpha-carbon fragmentation theory) are described in detail elsewhere (Buchs *et al.* 1970) and will not be explicated here.

2. Heuristics for the interpretation of nuclear magnetic resonance spectra were added to the Planner. As previously mentioned, these auxiliary data are useful for inferring the number of CH_3 superatoms in the structure (also how many of these superatoms are linked to a carbon, how many to the heteroatom). A complete interpretation of the NMR spectrum is often

Table 5. Ether and alcohol results.

Alcohol		Number of $C_nH_{(2n+2)}O$ isomers[1,2]	Number of inferred isomers[3] A	B
n-butyl	C_4	7	2	1
Iso-butyl		7	2	1
Sec-butyl		7	3	2
2-methyl-2-butyl	C_5	14	1	1
n-pentyl		14	4	1
3-pentyl		14	1	1
2-methyl-1-butyl		14	4	2
2-pentyl		14	2	1
3-hexyl	C_6	32	2	1
3-methyl-1-pentyl		32	8	4
4-methyl-2-pentyl		32	4	1
n-hexyl		32	8	1
3-heptyl	C_7	72	4	1
2-heptyl		72	8	1
3-ethyl-3-pentyl		72	1	1
2,4-dimethyl-3-pentyl		72	3	1
n-heptyl		72	17	1
3-methyl-1-hexyl		72	17	6
n-octyl	C_8	171	39	1
3-octyl		171	8	1
2,3,4-trimethyl-3-pentyl		171	3	1
n-nonyl	C_9	405	89	1
2-nonyl		405	39	1
n-decyl	C_{10}	989	211	1
6-ethyl-3-octyl		989	39	9
3,7-dimethyl-1-octyl		989	211	41
n-dodecyl	C_{12}	6045	1238	1
2-butyl-1-octyl		6045	1238	25
n-tetradecyl	C_{14}	38322	7639	1
3-tetradecyl		38322	1238	1
n-hexadecyl	C_{16}	151375	48865	1

A=Inferred isomers when only mass spectrometry is used.
B=Inferred isomers when the number of methyl radicals is known from NMR data.
[1] The total size of the problem space is the number of topologically-possible molecular structures generated within valence considerations alone.

Table 5 cont.

Ether		Number of $C_nH_{(2n+2)}O$ isomers[1,2]	Number of inferred isomers[3] A	B
Methyl-n-propyl	C_4	7	2	1
Methyl-*iso*-propyl		7	3	1
Methyl-n-butyl		14	2	1
Methyl-*iso*-butyl		14	2	1
Ethyl-*iso*-propyl		14	1	1
Ethyl-n-butyl	C_6	32	4	1
Ethyl-*iso*-butyl		32	4	2
Ethyl-*sec*-butyl		32	2	2
Ethyl-*tert*-butyl		32	1	1
Di-n-propyl		32	1	1
Di-*iso*-propyl		32	1	1
n-propyl-n-butyl	C_7	72	2	1
Ethyl-n-pentyl		72	4	1
Methyl-n-hexyl		72	8	1
Iso-propyl-*sec*-butyl		72	3	2
Iso-propyl-n-pentyl	C_8	171	4	1
n-propyl-n-pentyl		171	4	1
Di-n-butyl		171	3	1
Iso-butyl-*tert*-butyl		171	2	1
Ethyl-n-heptyl	C_9	405	34	1
n-butyl-n-pentyl		405	8	1
Di-n-pentyl	C_{10}	989	10	1
Di-*iso*-pentyl		989	18	7
Di-n-hexyl	C_{12}	6045	125	2
Di-n-octyl	C_{16}	151375	780	1
Bis-2-ethylhexyl		151375	780	21
Di-n-decyl	C_{20}	11428365	22366	1

[2] The number of plausible structures is the number of molecular structures in the total space which also meet *a priori* conditions of chemical stability. The *a priori* rules have no effect with formulas containing a single non-carbon, non-hydrogen atom.
[3] The number of structures generated is the number of molecular structures actually generated by the program as candidate explanations of the experimental data. Pruning has been achieved by using the planning information from the Planning program.

impossible to make, whether the interpreter is human or DENDRAL, but in any event is not necessary. Whatever partial interpretation can be done unambiguously by the heuristics will be reflected in the plan by corresponding aggregation information.

A new Planner [for historical reasons called 'Inference Maker' in Buchs *et al.* (1970)] implements these ideas. The structure of this program is very simple, but the mass spectrum interpretation heuristics are quite complex. These rules, developed by the DENDRAL group, stand on their own as a contribution to the methodology of mass spectrum analysis. Because of their complexity, however, they are best applied by a computer program, not a human chemist, giving DENDRAL a substantial performance edge over human analysts for the class of problems handled by the rules.

The Planner has the following organization:

1. If an NMR spectrum is given as problem information, infer all that can be inferred about the methyl superatoms. Include this information in the plan. In addition, use it in the test part of step 4 below.
2. Generate a list of the relevant Fs for the chemical family being considered (for example, generate the 31 Fs relevant to amines).
3. Associate with each F a property list which contains a number of criteria of applicability ('diagnostic' criteria) for that F. In large measure these criteria are inferred from mass spectral theory. (We mentioned earlier that the method of structuring the Fs was chosen to make this application of theory easy.)
4. Test each superatom against the given mass spectrum to ascertain whether all of the 'diagnostic' criteria for it are satisfied by the data. If any part of this validation test series fails, discard the F.
5. All Fs not discarded are included in the plan. For each of these, infer the weights of the attached radicals from the spectral data and include these sets of weights in the plan.

Table 6 exhibits the results of using this planning process on a group of amine compounds. There are some noteworthy things about the data in this table, for example:

1. The size of the problem spaces for some of the amines (over 14 million isomers of $C_{20}H_{43}N$!);
2. The impotence of the mass spectrum alone in finding the answer (or a small set of answers). This difficulty is not caused by a lack of expertise in the program. Human experts are in exactly the same situation, or perhaps worse.
3. The extraordinary effect of the NMR data to assist the mass spectrum analysis. Every time a '1' appears in the extreme right column, it indicates that the plan contained so much information about the solution, that the plan in fact uniquely determined the solution! Even in the other cases, the number of isomers in the plan-constrained space is trivially small.

This is remarkable. The Planner, which is the specialist at 'understanding'

the data and inferring conditions on the solution, is so powerful that the need for the general problem-solving processes of the system is obviated. Another way to view this is that all the relevant theoretical knowledge to solve these amine problems has been mapped over from its general form in the Predictor ('first principles') to efficient special forms in the Planner ('cookbook recipes'). The details of how each specialist works have been described elsewhere. In each particular case, new constraints on the problem lead to new heuristics for shortcutting the general combinatorial theory. When the shortcuts can be discovered, a specialist emerges; otherwise, the program relies on its general capabilities.

On the average, the problems of table 6 each took about 0·5 seconds of computer time to solve, whereas the average ketone or ether problem shown in previous tables took a few minutes to solve; and the average amine problem done by the method used for the ketones would take much longer.

PLANNING RULE GENERATOR

At this point, we will review the most important features of the planning process.

Though it houses a few general practitioners performing aggregation, the Planner is primarily a house of specialists. The areas of specialty are chemical families such as ketones, ethers, and amines. One process makes the necessary plan-formulation decisions for all the specialists. The expertness of a specialist is contained in what it knows about its family of specialization, particularly the expected patterns of mass spectral lines for a set of subclasses of the family.

There is, in effect, an N-position switch at the very front end of DENDRAL, which is set when a heuristic procedure or human intervention declares the family of molecules to be considered. [Deciding on an appropriate setting of the switch may involve some 'active' processing, for example, some search. Unless told by human intervention, DENDRAL does not know at the outset what the appropriate specialist is. It discovers this by some trial-and-error search. This involves, first, guessing the correct heteroatom (assuming that the empirical formula is not given). If, as a result of this guess, the specialist that is appropriate cannot validate even one F, a 'backtracking' takes place in which the guess is abandoned, and a new guess as to heteroatom is made.]

Setting the switch calls the appropriate specialist. If there is none, the switch is set to a default position which calls only general practitioners. The specialist knows how to generate the central superatoms relevant to its family and the associated validation criteria for each superatom.

The specialist was given this information by us, the designers. The designers, who know the theory of mass spectrometry, have selected some of this theory – first-order effects – as the basis for a preliminary interpretation of the data. The slice of theory so selected determines what size and structural form the central superatoms must have. The designers then deduce the actual structures of all of the logically-possible central superatoms of that size and form.

Table 6. Amine results.

Amine		Number of $C_nH_{(2n+3)}N$ isomers[1,2]	Number of inferred isomers[3] A	B
n-propyl	C_3	4	1	1
Iso-propyl		4	2	1
n-butyl	C_4	8	2	1
Iso-butyl		8	2	1
Sec-butyl		8	4	2
Tert-butyl		8	3	1
Di-ethyl		8	3	1
N-methyl-n-propyl		8	4	1
Ethyl-n-propyl	C_5	17	5	1
N-methyl-di-ethyl		17	4	1
n-pentyl		17	4	1
Iso-pentyl		17	4	2
2-pentyl		17	2	1
3-pentyl		17	5	1
3-methyl-2-butyl		17	4	1
N-methyl-n-butyl		17	4	1
N-methyl-*sec*-butyl		17	3	1
N-methyl-*iso*-butyl		17	4	1
n-hexyl	C_6	39	8	1
Tri-ethyl		39	2	1
2-hexyl		39	8	1
Di-n-propyl		39	8	1
Di-*iso*-propyl		39	8	1
N-methyl-n-pentyl		39	8	1
N-methyl-*iso*-pentyl		39	8	2
Ethyl-n-butyl		39	6	1
N,N-dimethyl-n-butyl		39	10	1
n-heptyl	C_7	89	17	1
Ethyl-n-pentyl		89	16	1
n-butyl-*iso*-propyl		89	11	1
4-methyl-2-hexyl		89	16	4
N-methyl-di-*iso*-propyl		89	15	3

A=Inferred isomers when only mass spectrometry is used.
B=Inferred isomers when the number of methyl radicals is known from NMR data.
1 The total size of the problem space is the number of topologically-possible molecular structures generated within valence considerations alone.

Table 6 cont.

Amine		Number of $C_nH_{(2n+3)}N$ isomers[1,2]	Number of inferred isomers[3] A	B
n-octyl	C_8	211	39	1
Ethyl-n-hexyl		211	24	1
1-methylheptyl		211	34	1
2-ethylhexyl		211	39	9
1,1-dimethylhexyl		211	32	4
Di-n-butyl		211	24	1
Di-*sec*-butyl		211	33	8
D-*iso*-butyl		211	17	5
Di-ethyl-n-butyl		211	17	3
3-octyl		211	26	2
n-nonyl	C_9	507	89	1
N-methyl-di-n-butyl		507	13	1
Tri-n-propyl		507	2	1
Di-n-pentyl	C_{10}	1238	83	1
Di-*iso*-pentyl		1238	109	16
N,N-dimethyl-2-ethylhexyl		1238	156	9
n-undecyl	C_{11}	3057	507	1
n-dodecyl	C_{12}	7639	1238	1
n-tetradecyl	C_{14}	48865	10115	1
Di-n-heptyl		48865	646	1
N,N-dimethyl-n-dodecyl		48865	4952	1
Tri-n-pentyl	C_{15}	124906	40	1
Bis-2-ethylhexyl	C_{16}	321988	2340	24
N,N-dimethyl-n-tetradecyl		321988	3895	1
Di-ethyl-n-dodecyl		321988	2476	1
n-heptadecyl	C_{17}	830219	124906	1
N-methyl-bis-2-ethylhexyl		830219	2340	24
n-octadecyl	C_{18}	2156010	48865	1
N-methyl-n-octyl-n-nonyl		2156010	15978	1
N,N-dimethyl-n-octadecyl	C_{20}	14715813	1284792	1

[2] The number of plausible structures is the number of molecular structures in the total space which also meet *a priori* conditions of chemical stability. The *a priori* rules have no effect with formulas containing a single non-carbon, non-hydrogen atom.
[3] The number of structures generated is the number of molecular structures actually generated by the program as candidate explanations of the experimental data. Pruning has been achieved by using the planning information from the Planning program.

Table 7. Thioether and thiol results.

Thioether		Number of $C_nH_{(2n+2)}S$ isomers[1,2]	Number of inferred isomers[3] A	B
Methyl-ethyl	C_3	3	1	1
Methyl-n-propyl	C_4	7	1	1
Methyl-*iso*-propyl		7	2	1
Di-ethyl		7	1	1
Methyl-n-butyl	C_5	14	3	1
Methyl-*iso*-butyl		14	5	2
Methyl-*tert*-butyl		14	1	1
Ethyl-*iso*-propyl		14	1	1
Ethyl-n-propyl		14	2	1
Ethyl-n-butyl	C_6	32	3	1
Ethyl-*tert*-butyl		32	1	1
Ethyl-*iso*-butyl		32	3	2
Di-n-propyl		32	2	1
Methyl-n-pentyl		32	10	1
Di-*iso*-propyl		32	1	1
Ethyl-n-pentyl	C_7	72	4	1
n-propyl-n-butyl		72	5	1
Iso-propyl-n-butyl		72	5	2
Iso-propyl-*tert*-butyl		72	1	1
n-propyl-*iso*-butyl		72	3	2
Iso-propyl-*sec*-butyl		72	4	3
n-propyl-n-pentyl	C_8	171	4	1
Ethyl-n-hexyl		171	8	1
Di-n-butyl		171	5	1
Di-*sec*-butyl		171	3	1
Di-*iso*-butyl		171	3	1
Methyl-n-heptyl		171	21	1
Di-n-pentyl	C_{10}	989	12	1
Di-n-hexyl	C_{12}	6045	36	1
Di-n-heptyl	C_{14}	38322	153	1

A=Inferred isomers when only mass spectrometry is used.
B=Inferred isomers when the number of methyl radicals is known from NMR data.

Table 7 cont.

Thiol		Number of $C_nH_{(2n+2)}S$ isomers[1,2]	Number of inferred isomers[3] A	B
n-propyl	C_3	3	2	1
iso-propyl		3	1	1
n-butyl	C_4	7	3	1
iso-butyl		7	3	1
Tert-butyl		7	1	1
2-methyl-2-butyl	C_5	14	1	1
3-methyl-2-butyl		14	2	1
3-methyl-1-butyl		14	6	3
n-pentyl		14	4	1
3-pentyl		14	5	3
2-pentyl		14	6	3
n-hexyl	C_6	32	8	1
2-hexyl		32	12	5
2-methyl-1-pentyl		32	8	4
4-methyl-2-pentyl		32	4	2
3-methyl-3-pentyl		32	1	1
2-methyl-2-hexyl	C_7	72	8	3
n-heptyl		72	17	1
2-ethyl-1-hexyl	C_8	171	39	9
n-octyl		171	39	1
1-nonyl	C_9	405	89	1
n-decyl	C_{10}	989	211	1
n-dodecyl	C_{12}	6045	1238	1

[1] The total size of the problem space is the number of topologically-possible molecular structures generated within valence considerations alone.
[2] The number of plausible structures is the number of molecular structures in the total space which also meet *a priori* conditions of chemical stability. The *a priori* rules have no effect with formulas containing a single non-carbon, non-hydrogen atom.
[3] The number of structures generated is the number of molecular structures actually generated by the program as candidate explanations of the experimental data. Pruning has been achieved by using the planning information from the Planning program.

The designers also deduce from the first-order theory specific values for the validation criteria to be associated with each central superatom. The results of these two deductive steps (superatoms and criteria) taken together constitute a set of planning rules to be used at the time the specific plans are formulated. Thus a set of planning rules makes the Planner a specialist for a chemical family. Once alive and tested, the new specialist is added to the 'big switch'.

It is evident that when the designer has chosen the slice of theory he wishes to use for planning purposes, the remainder of his work, the generation of planning rules, can be, in fact should be, done by program. As the molecular families treated become more complex, necessitating the addition of heuristic power in the planning stage if the generator is to be properly controlled, the planning analysis involves increasingly more theory, which in turn leads to increased difficulty for humans in generating logically-complete and accurate sets of planning rules. In addition, a Planning Rule Generator program can create, automatically, specialists for each of the member-families of the broad class of families to which the theory now applies. This is an automatic mass production process that can replace the tedious and expensive process of eliciting knowledge from an expert that we have used in the past.

A Planning Rule Generator has been written for DENDRAL. It deals with the very general class of saturated (that is, no double bonds or rings), acyclic monofunctional compounds. Plan schemata have been generated by this program for the following families: thiols and thioethers (heteroatom is sulfur); ethers; alcohols; and amines. These planning rules were then used by DENDRAL in solving problems in these areas (that is, the ordinary DENDRAL performance mode). The results are shown in tables 5, 6, and 7. The comments we made earlier concerning table 6 apply also to tables 5 and 7.

The Planning Rule Generator is a complex program, the details of which cannot be described here. Those interested can find a description of the program from a chemical point of view in a recent publication (Buchs *et al.* 1970).

The DENDRAL Planner is a performance process. The Planning Rule Generator is not. It is a higher-level planning process by which it is determined how planning shall be done in particular classes of problems. For us it is the first small step up the ladder of programs for theory manipulation and theory formation 'meta' to the DENDRAL performance program. We view the building of such programs as a promising endeavor. DENDRAL as a performance program is complex enough and rich enough in internal structure and theory to provide many firm foundation points on which to erect a meta-level for the study of theory formation processes.

GENERALITY AND THE DESIGNS FOR PROBLEM-SOLVING SYSTEMS

We shall conclude this paper with a return to the theme with which we began:

generality, expertness, and the design of problem solvers. As a case study, we have traced the evolution of designs for a system that solves difficult scientific inference problems. The forcing function for the evolution of designs was primarily the set of demands placed upon the organization of the DENDRAL program by increasingly more complex and difficult tasks. The design which we now have is 'natural' (that is, shaped by the real world), not 'artificial' or abstract.

Many threads have been woven into our discussion: general processes and representations in DENDRAL; the cost of generality; heuristic power; the specialization of knowledge in the planning process; planning as a method for translating problem data into search constraints and solution conditions; higher-level planning as a method for building specialists from general theory. We now ask whether these threads form a meaningful fabric.

The study of generality in problem solving has been dominated by a point of view that calls for the design of 'universal' methods and 'universal' problem representations. These are the GPS-like and Advice-Taker-like models. This approach to generality has great appeal, but there are difficulties intrinsic to it: the difficulty of translating specific tasks into the general representation; and the tradeoff between generality and power of the methods.

In recognition of these difficulties, a viewpoint at the other extreme has emerged, informally called 'the big switch hypothesis' (Ernst and Newell 1969).

In this view, general problem-solvers are too weak to be used as the basis for building high-performance systems. The behavior of the best general problem-solvers we know, human problem-solvers, is observed to be weak and shallow, except in the areas in which the human problem-solver is a specialist. And it is observed that the transfer of expertness between specialty areas is slight. A chess master is unlikely to be an expert algebraist or an expert mass spectrum analyst, etc. In this view, the expert is the specialist, with a specialist's knowledge of his area and a specialist's methods and heuristics.

The 'big switch hypothesis' holds that generality in problem solving is achieved by arraying specialists at the terminals of a big switch. The big switch is moved from specialist to specialist as the problem solver switches its attention from one problem area to another. [In this paper, we merely state the hypothesis without discussing it. The kinds of problem-solving processes, if any, which are involved in 'setting the switch' (selecting a specialist) is a topic that obviously deserves detailed examination in another paper.]

Our case study of the DENDRAL program suggests a synthesis of these extreme points of view. The features that characterize a general problem-solving process are present. Within the DENDRAL world, the search for solution candidates in the Structure Generator and the validation procedure of the Predictor are 'universal' methods, and the representation employed is 'universal'. The general methods do solve DENDRAL problems, sometimes

well, as with some amino-acid spectra, but they are relatively weak and inefficient.

To increase accuracy and efficiency, specialists emerged, but in a design which called for compatibility and coexistence with the general processes. The existing internal representation was maintained throughout as a 'common language' understood by both generalist and specialist. The specialists did not replace the generalists. They were written to function as planners, providing search constraints and solution conditions. The 'big switch' in DENDRAL is at the front end of the Planner Program. Despite the array of powerful specialists on the switch, perhaps the most important position is the default position – the 'not elsewhere classified' bypass – that calls the general problem-solving processes when the knowledge of a specialist is not available.

The Planning Rule Generator makes the symbiosis of generalist and specialist mutual. The theory of mass spectrometry that is used by the Predictor to validate candidates (or some part of it) is used by the Planning Rule Generator to deduce a new specialist for the 'big switch'.

Herein, we think, lies the germ of another method for problem solvers. A general problem-solving process in part achieves generality because it employs a general theory of the nature and behavior of the objects and operators of its world. This theory can be used in what we might call 'execute mode', as, for example, when DENDRAL's Predictor is validating a candidate solution. But this theory can also be used in what might be called 'compile mode', as, for example, when the Planning Rule Generator is deducing a new specialist.

This idea needs an extended discussion, which we are not prepared to give here. But we shall make two brief observations.

The first observation is that the idea closely parallels the line of argument given by Simon in his book of essays on heuristic programming, *The New Science of Management Decision* (Simon 1960). In discussing human decision-making, particularly in organizations, Simon draws a dichotomy between the routine repetitive decision problems, which he calls 'programmed decisions' and the novel one-shot decision problems, which he calls 'nonprogrammed decisions'. Concerning 'programmed decisions', the organization 'develops specific processes for handling them'. Examples are: habits (an individual's 'compiled subroutines'), standard operating procedures (an organization's 'compiled subroutines'), mathematical models from Operations Research, and EDP procedures. The 'nonprogrammed' decision problems are 'handled by general problem-solving processes'. To a large extent, it is the repetitiveness with which a decision problem presents itself that determines whether it is economic for an organization to invest resources in routinizing and specializing the decision-making process, that is, to 'compile' general processes into special-purpose routines.

The second observation is that the idea may be much more difficult to

implement than it appears at first for the simple reason that the tradeoff between generality and power holds for processes at the meta-level just as it holds for performance level processes. Thus, for example, DENDRAL's Planning Rule Generator is powerful for the supra-family of all saturated, acyclic, monofunctional compounds, but is useless for all other classes of compounds. When we extend DENDRAL's capability to families of cyclic molecules, we may have to write a new Planning Rule Generator. Or is there yet another process lurking at a higher level, a Generator of Planning Rule Generators?

The appropriate place for an attack on the problem of generality may be at the meta-levels of learning, knowledge transformation, and representation, not at the level of performance programs. Perhaps for the designer of intelligent systems what is most significant about human general problem-solving behavior is the ability to learn specialties as needed – to learn expertness in problem areas by learning problem-specific heuristics, by acquiring problem-specific information, and by transforming general knowledge and general processes into specialized forms.

Acknowledgements

This research was supported by the Advanced Research Projects Agency (SD-183). Much of the work reported here was performed by Dr Armand Buchs and Mr Allan Delfino. The assistance of Mrs Georgia Sutherland, Dr Alan Duffield, and Professor Carl Djerassi is also gratefully acknowledged.

REFERENCES

Buchanan, B.G., Sutherland, G.L. & Feigenbaum, E.A. (1969) Heuristic DENDRAL: A program for generating explanatory hypotheses in organic chemistry. *Machine Intelligence 4*, pp. 209–54 (eds Meltzer, B. & Michie, D.). Edinburgh: Edinburgh University Press.

Buchanan, B.G., Sutherland, G.L. & Feigenbaum, E.A. (1970) Rediscovering some problems of artificial intelligence in the context of organic chemistry. *Machine Intelligence 5*, pp. 253–80 (eds Meltzer, B. & Michie, D.). Edinburgh: Edinburgh University Press.

Buchs, A., Delfino, A. B., Duffield, A. M., Djerassi, C., Buchanan, B. G., Feigenbaum, E.A. & Lederberg, J. (1970). Applications of artificial intelligence for chemical inference VI. Approach to a general method of interpreting low resolution mass spectra with a computer. *Helv. Chim. Acta*, in press.

Buchs, A., Duffield, A.M., Schroll, G., Djerassi, C., Delfino, A.B., Buchanan, B.G., Sutherland, G.L., Feigenbaum, E.A. & Lederberg, J. (1970a). Applications of artificial intelligence for chemical inference IV. Saturated amines diagnosed by their low resolution mass spectra and nuclear magnetic resonance spectra. *J. Amer. chem. Soc.*, **92**, in press.

Duffield, A.M., Robertson, A.V., Djerassi, C., Buchanan, B.G., Sutherland, G.L., Feigenbaum, E.A. & Lederberg, J. (1969) Applications of artificial intelligence for chemical inference II. Interpretation of low resolution mass spectra of ketones. *J. Amer. chem. Soc.*, **91**, 11.

Ernst, G.W. & Newell, A. (1969) *GPS: A case study in generality and problem solving*. New York: Academic Press.

Feigenbaum, E.A. (1968) Artificial intelligence: themes in the second decade. Final Supplement to *Proc. IFIP68 International Congress, Edinburgh.*

Lederberg, J., Sutherland, G.L., Buchanan, B.G., Feigenbaum, E.A., Robertson, A.V., Duffield, A.M. & Djerassi, C. (1969) Applications of artificial intelligence for chemical inference I. The number of possible organic compounds: acyclic structures containing C, H, O, and N. *J. Amer. chem. Soc.*, **91**, 11.

Lederberg, J., Sutherland, G.L., Buchanan, B.G. & Feigenbaum, E.A. (1970) A heuristic program for solving a scientific inference problem: summary of motivation and implementation. *Theoretical Approaches to Non-Numerical Problem Solving*, pp. 401–9 (eds Banerji, R. & Mesarovic, M.). New York: Springer-Verlag.

Newell, A. (1969) Heuristic programming: Ill-structured problems. *Progress in Operations Research III* (ed Arnofsky, J.S.). New York: Wiley.

Schroll, G., Duffield, A.M., Djerassi, C., Buchanan, B.G., Sutherland, G.L., Feigenbaum, E.A. & Lederberg, J. (1969) Applications of artificial intelligence for chemical inference III. Aliphatic ethers diagnosed by their low resolution mass spectra and NMR data. *J. Amer. chem. Soc.*, **91**, 26.

Simon, H. (1960) *The new science of management decision.* New York: Harper.