

3.3 Visual Perception Projects

One important aspect of the general vision system is the adaptation of the input mechanisms to the visual environment. Selective attention must be implemented at the level of the vision hardware by choosing accommodative strategies which reflect current perceptual goals. For example, the camera could be sensitized to a specific color characteristic of a desired object (via a color filter). This may yield a gross reduction in the volume of information which must be processed.

The camera parameters currently under computer control are the pan and tilt angles, lens choice (turret) and gray scale range. There are two hard problems in adaptation which arise from the need for a common world model. When the camera is panned, it gets a new view. The images of objects in this new view must be placed in correspondence with the old images of the same objects. An even more difficult problem is to compute accurately the perspective transformation [16] applicable in the new situation. Sobel [20] is developing techniques for these problems, relying heavily on the literature of photogrammetry.

A major area of interest has been the development of edge and line finders. There have been extensive analytical and practical studies of various spatial filtering and edge finding techniques [7, 22]. More recently, we have begun to look at feature verifiers which will use global information and a prediction to help identify a feature.

There are also programs which do fairly well at corner finding, region extraction, etc. These are flexible and might be incorporated into a vision system organized as we have suggested. The real problem is to develop routines for these tasks which are sensitive to possible errors

and ambiguities and know when to ask for help. A related issue is the language for communicating between vision programs at various levels.

Much of the work, to date, on visual perception by machines has centered around plane-bounded man-made objects. Unfortunately, most naturally occurring objects do not have well defined edges. This raises the question of whether many of the presently used edge and object recognition techniques are likely to be useful and if not, what other perception techniques should we be looking at?

Mike Kelly, a Ph.D. student, is approaching this by looking at people in a scene. An overall look at the scene with special purpose people-detection operators determines the areas of interest. A profile recognizer determines some gross parameters about the person. Using automatic pan, tilt, and lens selection we obtain a close-up view of the faces for further analysis.

Traditional edge detectors and region analyzers yield a great deal of spurious information. Therefore, we are using goal directed feature extraction procedures for the determination of location and shape eyes, nose, mouth, hair line etc. These procedures are not critically dependent on edge detection or thresholds.

In addition, we are investigating the use of color, texture, and environmental constraints in visual perception by machine. For any given lighting conditions one or more of these features can be expected to exhibit sharp discontinuities at region boundaries. During the next year we hope to solve the following specific problems concerning color and texture of a scene:

- 1) Using three color filters to obtain digitized images consisting of (toy) houses, cars, trees and rocks.

2) Develop operators which permit picture segmentation using a similarity or homogeneity criterion based on their color parameters at each point.

3) Extension of the operators to use depth, light intensity and texture features.

Two dimensional Fourier transform of the whole scene has been suggested for obtaining texture parameters. Besides being computationally expensive, this cannot conveniently be used in conjunction with color, light intensity, and depth parameters. What is needed is the concept of local texture, i.e., texture parameters at a given point, so that one may decide whether that point belongs to one object or another in the scene.

Local texture at any point can be determined by Fourier transform of a small neighborhood around the point (perhaps convolved with a Bell-shaped or a Cone-shaped weighting function). This would still be computationally expensive. An alternative is to count the number of local maxima and minima (at a certain distance apart) in successive neighborhoods around the point. We hope to develop a local texture determination program and use it with 3-D operators to separate regions and objects.

We are currently completing a first version of our vision scheme which, to facilitate experimentation, allows human intervention at several stages in the scene analysis process. As intermediate stages of analysis are displayed, the user will be able to interrupt and add information to the system. Using this system and some hard thought, we hope to come up with a reasonable first cut at the multi-level vision system. We will then begin the process of refining this system and adding to its basic capabilities.

3.4 Arm Control

Work is also proceeding on improvements to the hand part of the hand-eye system. Along with improved mechanical design, we are generally concerned with the computer control of moving manipulators. This latter is an area presenting several challenging problems, including dynamics of the manipulator, path or space constraints, and the effects of computer control when the sampling time approaches the response time of the controlled element. Pieper approached the manipulator control problem kinematically [11], developing a method for iteratively solving the positioning for manipulators which are unsolvable analytically. In addition, he developed a method by which a manipulator may be moved from one position to another while avoiding certain obstacles. The path chosen does not account for the dynamics of the manipulator and in practice the manipulator is lead slowly along the path by the computer's control program.

Mike Kahn has approached the problem from the viewpoint of dynamics, obtaining a near-minimum time control for moving the manipulator from one position to another. The solution accounts for the dynamic reactions of one joint on another and the gravity loads on the manipulator. Certain linearization assumptions in the solution restrict the applied torque-ratios on the manipulator to given ranges and also limit the validity of the solution to a range of motions in which the inertia changes seen by a given joint are not excessively large. The resulting control is bang-bang and the solution is in the form of switching curves for the control of each joint. The resulting path is a strictly dynamic one, in that it does not account for obstacles which

may exist in the working space.

Several problems remain in the area of computer control of manipulators. 1) For each joint the reversal curve describes the point (dependent on both position and velocity) at which bang-bang deceleration must be applied to the joint so that the arm will arrive at the goal at rest. The problem is to develop a simplified approach to constructing reversal curves for each joint of the manipulator. A possible way of doing this would be to approximate the reversal curves by tangents at the expected switching points. In addition, a simplified method of dealing with large inertial changes seen by the joints would be desirable.

2) To combine the dynamical and kinematical viewpoints is obtaining a control scheme which accounts for both the dynamics of a rapidly moving manipulator and the path constraints imposed by obstacles in the workspace. This problem in multivariable optimal control with state-space constraints also has application in other areas of control theory.

3) An investigation of the influence of sampling rate and computation time on how well a fast manipulator can be controlled. An increase in the sophistication of a computer control scheme could allow an increase in manipulator speed relative to sampling rate.

3.5 Visual Control of a Vehicle

Included in the visual manipulation area is the development of a computer controlled vehicle. We have a battery powered cart with a TV link to the computer and a radio link back. Rod Schmidt, who adapted the hardware has also written a program that, under good seeing conditions, successfully follows a white line that is similar to those found on some roads. At present, programs are being written to analyze road scenes for the boundaries of the road, cars, obstacles such as dogs and holes, and other relevant features. We believe that within five years a computer-controlled vehicle can be brought to a pseudo-practical level; e.g., a computer controlled car could drive with low reliability from our lab to the Bayshore Freeway, paying proper attention to the road, other cars, pedestrians, animals, and traffic signals. A high reliability, reasonable cost system would then still require extensive development.

From a scientific point of view the computer driven vehicle presents several features of interest. First of all, the outputs required are quite simple: turning the wheels, and using the brakes and accelerator. Secondly, on a higher level the reaction to identified objects is simple, they are either to be ignored (shadows), avoided, or waited for. On the other hand, the class of objects that has to be dealt with is very large compared to that for manipulation problems. Many kinds of obstacles might be avoided without every identifying them precisely.

However, the advantages of working on computer controlled vehicles using visual information are not merely scientific. Once the basic problems of locating and identifying the relevant objects in a scene have been solved, technology will provide us with computers that can perform these tasks faster and more reliably than man. This will make possible

a number of applications of both technological and economic importance.

There are, of course, major scientific problems to be solved before applications are feasible. The most apparent of these are the representation problem of scenes where obstacle avoidance is the goal and the consequent scene analysis problem under varying lighting conditions.

We are presently writing programs for locating major features such as the white line or dashed white line and the edge of the road. Our goals for the next year include finishing these programs and also recognizing some class of obstacles (such as cars). The programs will be checked by driving the cart around the building under various conditions.

REFERENCES

1. Earnest, L.D., "On Choosing an Eye for a Computer", Memo AI-51, Computer Science Dept., Stanford University, Stanford, California, 1967.
2. Ernst, H.A., "MH-1 a Computer Operated Mechanical Hand", Doctoral Thesis in Electrical Engineering, M.I.T., Cambridge, Massachusetts, 1961.
3. Feldman, J., and Rovner, P.D., "An Algol-based Associative Language", Memo AI-66, Computer Science Dept., Stanford University, Stanford, California, to appear in Comm. ACM, August 1969.
4. Gibson, J., The Senses Considered as Perceptual Systems, Houghton-Mifflin, Boston, 1966.
5. Guzman, A., "Decomposition of Visual Scene into Three-dimensional Bodies", Proj. FJCC, 1968.
6. Guzman, A., "Some Aspects of Pattern Recognition by Computer", MAC-TR-37, Project MAC, M.I.T., Cambridge, Massachusetts, 1967.
7. Hueckel, M., "Locating Edges in Pictures", forthcoming A.I. Memo, Computer Science Dept., Stanford University, Stanford, California (to appear).
8. Nilsson, N., "The Stanford Research Institute Robot Project", Proc. Joint Int. Conference on Artificial Intelligence, Washington, D.C., 1969.
9. Paul, R., Falk, G., Feldman, J., "The Computer Representation of Simply Described Scenes", Proc. Illinois Graphics Conference, April 1969.
10. Paul, R., "A Representation of a Tower of Plano-convex Objects", Artificial Intelligence Laboratory, Operating Note No. 41, Stanford University, Stanford, California, August 1968.
11. Pieper, D., "The Kinematics of Manipulators under Computer Control" (Ph.D. Dissertation in Mechanical Engineering), Memo AI-73, Computer Science Dept., Stanford University, Stanford, California, 1968.
12. Pingle, K., "Hand-eye Library File", Artificial Intelligence Laboratory Operating Note No. 35, Stanford University, Stanford, California, August 1968.
13. Pingle, K., "A List Processing Language for Picture Processing", Artificial Intelligence Laboratory Operating Note No. 33, Stanford University, Stanford, California.
14. Pingle, K., Singer, J.A., and Wichman, W.M., "Computer Control of a Mechanical Arm Through Visual Input", Proc. IFIP Conference, Edinburgh, 1968.

15. Pohl, I., "Search Processes in Graphs", Stanford University Dissertation, forthcoming.
16. Roberts, L.G., "Machine Perception of Three-Dimensional Solids" in Optical and Electro-Optical Processing of Information, MIT Press, Cambridge, Massachusetts, 1965.
17. Samuel, A., "Studies in Machine Learning Using the Game of Checkers", IBM Journal, November 1967.
18. Shapiro, G., "Advanced Hand-Eye Manipulating", Internal Memo, Stanford Artificial Intelligence Project, Stanford University, Stanford, California.
19. Scheinman, V., "Design of a Computer Controlled Manipulator", Engineering Thesis in Mechanical Engineering, Stanford University, Stanford, California, 1969.
20. Sobel, I., forthcoming Electrical Engineering Thesis on "Visual Accommodation in Machine Perception", Stanford University, Stanford, California.
21. Swinehart, D., "Golgol III Reference Manual", Artificial Intelligence Laboratory Operating Note No. 48, Stanford University, Stanford, California.
22. Tenenbaum, J., "An Integrated Visual Processing System", forthcoming.
23. Wichman, W.H., "Use of Optical Feedback in the Computer Control of an Arm", Engineers Thesis, Stanford University, Stanford, California, August 1967.
24. McCarthy, J., Earnest, L.D., Reddy, D.R., Vicens, P.J., "A Computer with Hands, Eyes, and Ears", Proc. FJCC 1968.

4. Speech Recognition

Our research in speech recognition has been mainly concerned with developing systems for fast and accurate voice input to computers. At present the PDP-10 System can recognize limited sets of words, (the largest vocabulary tested is 560 words) phrases, and a few connected speech utterances made out of these. The time for recognition is around 2 to 15 seconds (depending on the size of the vocabulary) per word with an accuracy of 95 to 98 percent. The system has been used for voice control of a computer-controlled mechanical hand. Some relevant references are given at the end of this section.

Both the goals and methodologies of various ARPA supported speech projects are different and complementary to one another. Our effort differs from BBN's effort in that we are more interested in connected speech than limited vocabularies. BBN's research attempts to extract and use traditional phonological features in recognition. The speech analysis-synthesis project at University of California at Santa Barbara is directed more at vocoder-type applications with no attempt at machine recognition. UCSB's effort attempts to obtain an accurate and compact representation of speech parameters so that the original utterance can be effectively resynthesized from these parameters. Our recognition effort is based on easily extracted acoustic distinctive features (which may or may not be related to phonological distinctive features) and depends strongly on structural relationships of word-level and sentence-level syntax. Our system is designed not to be critically dependent on obtaining accurate parameters since it is usually difficult to obtain high quality speech in a computer environment. Whether we use crude or accurate representation of speech the main problem is the separation

of the characteristics of the intended utterance from that of the speaker and the environment.

Future Plans

There are many outstanding unresolved questions in speech analysis and perception research. During the next few years we expect to concentrate on those problems which, when solved, should significantly enhance our knowledge about speech communication with machines, namely, normalization of speaker differences, languages for man-machine voice communication, and a better understanding of problems involved in carrying on speech conversations with machines. The following paragraphs outline our research plans in these directions.

1. Speaker Normalization. At present, we have to train the system with all the speakers. To recognize the speech of a random speaker, without explicitly training the system with each word or phrase he is likely to utter, requires some form of speaker identification and normalization. Previous attempts at speaker recognition have been disappointing. Our approach will involve having the speaker speak a kernel set of sentences initially and then every time he begins to use the machine he would identify himself by saying "I am John McCarthy" or some such. The system will then modify the recognition algorithm to correspond to his "acoustic signature".

2. Language Design for Man-Machine Voice Communication. A language for speaking to machines should not be unwieldy and ambiguous like English, nor should it be unnatural like a programming language. We believe a Lisp-like quasi-functional notation might be more convenient for some applications, i.e., it is more desirable to say "add Alpha to Beta" rather than "Beta Equals Alpha Plus Beta Semi-colon". The structure

of spoken machine languages will vary from task to task. We hope to have a system in which the user will specify the language as a BNF grammar or some such. The system will then analyze the grammar for possible phonemic, word boundary, and syntactic ambiguities and suggest possible modifications. The system will also provide a skeleton recognizer which the user can use to train the system to recognize his language for one or several speakers.

3. We hope to use Signature-Table learning techniques to achieve more accurate phoneme recognition, (see Section 5.1).

4. Conversational Systems. To be able to carry on speech conversation, a machine must not only have speech recognition and synthesis capabilities but also a good semantic model. At present we have a highly simplified conversational system which can recognize and respond to about 20 phrases using time-domain compressed speech as output, and based on a simple semantic model. The main purpose of this research will be to help us isolate and identify the main bottlenecks in achieving our goal, and to evaluate the performance of various recognition and synthesis models in an exacting task environment.

In the next five to ten years we expect to have a processor of the speed of a PDP-10 servicing 3 to 6 terminals with speech input and output capability. Such a system could be used as:

1. An extra motor process when hands and feet are already in use. (Possible applications: aeronautics, space and interactive graphics.)
2. A fast response, and fast data rate motor process (when a real time control of devices by a computer is performed under human supervision).
3. Natural mode of communication between man and machine. Some possibilities are day-to-day calculations (voice controlled desk calculator), information retrieval and question-answering systems, dictation (of programs,

letters, etc.), medical diagnosis, and psycho-therapy.

If, as it seems likely, a computer 5 to 10 times the speed of the PDP-10 becomes available for about the same cost as a PDP-10 in the next 5 to 10 years, then we believe it will be possible to extend the above system to service 30 to 50 terminals with voice input and output capability. Such a system should then be competitive with conventional input-output equipment.

REFERENCES

1. Reddy, D.R. and Robinson, A.E., "Phoneme to Grapheme Translation of English", IEEE Trans. Audio and Electroacoustics, AU-16, 2, 240-246 (1968).
2. Reddy, D.R., "Digital Speech Compression in Time Domain", Presented at the 76th meeting of Acoustical Society of America, JASA, 44, 1, 391 (1968) (Abstract form only).
3. Reddy, D.R., "On Computer Transcription of Phonemic Symbols", J. Acoust. Soc. Am., 44, 2, 638-639 (1968) (letter).
4. Reddy, D.R., "Consonantal Clusters and Connected Speech Recognition", Proc. of the 6th International Congress on Acoustics, Tokyo (1968).
5. Vicens, P., "Preprocessing for Speech Analysis", Artificial Intelligence Memo AI-71, (October 1968).
6. Reddy, D.R., and Vicens, P., "A Procedure for Segmentation of Connected Speech", J. Audio Engr. Soc., 16, 4, 404-412, (1968).
7. McCarthy, J., Earnest, L.D., Reddy, D.R., and Vicens, P., "A Computer with Hands, Eyes and Ears", Proc. of the FJCC 1968, 329-337, (1968).
8. Reddy, D.R., "On the Use of Environmental, Syntactic and Probabilistic Constraints in Vision and Speech", Memo AI-78, (January 1969).
9. Reddy, D.R., and Neely, R.B., "Contextual Analysis of Phonemes of English", Memo AI-79, (January 1969).
10. Vicens, P., "Aspects of Speech Recognition by Computer", Memo AI-85, (April 1969).

5. Heuristic Search

5.1 Machine Learning

Enough progress has been made in the development of the new signature table procedure as a machine learning technique to justify a serious attempt to apply these methods to problems of practical import. At the same time there is still much to be learned about machine learning in general; enough, in fact, to warrant the continuation and even the expansion of basic work using the more formal context of a game. As has been said many times, the use of a game environment enables one to exclude the many extraneous and complicating details of real life situations and to concentrate one's entire attention on the basic problems under study.

The game of checkers still appears to offer many advantages as a study vehicle and we are accordingly continuing work on this game. The Samuel checker program has been completely rewritten for the PDP-10 and the playing portion is now thought to be reasonably bug free. The program is a good deal more complicated than was the IBM-7094 version and the playing portion quite successfully trades memory space for time so as to more than compensate for the difference in machine speeds. Even without benefit of a satisfactory amount of learning the program plays quite well. The learning aspect of the program is, however, still not entirely satisfactory as we have not yet been able to compensate for the difference in machine speeds in this portion of the program. As the program now stands, many hours of hard-to-come-by machine time with large amounts of core are required to test out each fairly minor modification in learning. Steps are being taken to alleviate this situation both by rewriting the well understood portions of the program to increase speed and by exercising

extreme care in the design of experiments which will yield the greatest amount of information per hour of machine usage. There do appear to be some rather fundamental limitations as to what can be done along these lines and we may always be faced with the need for large amounts of machine time as long as we continue to experiment with different learning techniques.

Work is also continuing on the game of Go which offers another formal system but one in which the look-ahead procedure (so useful in checkers and chess) cannot usually be applied. This work is continuing at a slow pace with but one graduate student involved. The difficulties of the game and lack of knowledgeable players makes it hard to recruit people for this work. Go does, however, present a quite different environment from checkers and chess while still retaining the essential characteristic of formal simplicity and is therefore well worth study.

Speech recognition is engaging our attention as a very fruitful field in which to apply the signature table learning scheme developed by Samuel for checkers. Messrs. Samuel and Reddy are collaborating in this work. The general method of attack is to write a collection of carefully thought out general purpose subroutines which can be directly used in speech recognition but which will hopefully be useful for machine learning applications in quite disjoint fields of endeavor. Speech recognition was chosen largely because of the prior existence of significant work in this field in our laboratory. We now feel that this was a particularly happy choice in view of the fact that the signature table technique seems to be unusually well suited for this specific application.

Our initial efforts are in phoneme identification although we are also beginning to think about ways in which the same techniques can be

brought to bear on the problems of segmentation and word recognition.

Already several quite new ideas have emerged and we are extremely hopeful of success in this work.

5.2 Automatic Deduction

A theorem-proving program based on the Resolution Principle [1] for First Order Logic has been implemented using the PDP-10 LISP system. This program incorporates all of the efficiency strategies for proof search that are the subject of references [2,3,4,5,6], and has been used to compare the relative merits of various combinations of these strategies. In addition, it also has a special replacement rule for equality, [7]. The program is capable of dealing with all of the basic theorems of elementary algebra (Theory of Groups, Rings, Fields, Boolean Algebras and Modular Lattice Theory) and number theory that have been proved by programs discussed in the above references, and also with simple questions that arise in information retrieval, question-answering and automatic program writing [8]. A rudimentary interactive system has been incorporated into the program to allow the user to question and make suggestions to the program in the course of a proof search.

It is proposed to continue research in this area along both theoretical and practical lines towards the goal of developing this program into a practicable basis for automatic mathematics systems (proof checking and proposition testing), program writing, information retrieval, and other projects in Artificial Intelligence. Specific lines of research will include the following: (1) the further development of the interactive feature, (2) the addition of a proof-tree analyzer for extracting solutions to existential statements from proofs of them, (this is a basic part of the program writing and information retrieval applications), (3) the extension of the efficiency strategies mentioned above to a proof system incorporating rules of inference for the basic relations of equality, associativity, commutativity, etc., (4) the use of the algebraic simplification system,

Reduce [9] in some of the phases of the proof search procedure, (5) the incorporation of decision procedures for certain classes of problem, (6) the extension of the basic logical system to a many-sorted calculus, and possibly the introduction of modal logic and w-order logic.

At present time, this is an area of intense endeavor and, using the facilities available here, we propose to test each theoretical possibility on practical problems.

REFERENCES

1. Robinson, J.A., "A Machine-Oriented Logic Based on the Resolution Principle", JACM, Vol. 12, No. 1, pp. 23-41, January 1965.
2. Luckham, D., "Some Tree-Paring Strategies for Theorem-Proving", Machine Intelligence 3, D. Michie (ed), Edinburgh University Press, pp. 95-112, 1968.
3. Luckham, D., "Refinement Theorems in Resolution Theory", Memo AI-81, Computer Science Dept., Stanford University, Stanford, California, March 1969.
4. Wos, L., Carson, D., Robinson, G., "The Unit Preference Strategy in Theorem Proving", AFIPS Conf. Proc. 26, Washington, D.C., Spartan Books, pp. 615-621, 1964.
5. Wos, L., et. al., "Efficiency and Completeness of the Set of Support Strategy in Theorem Proving", JACM, Vol. 12, No. 4, pp. 536-541, October 1965.
6. Wos, L., et. al., "The Concept of Demodulation in Theorem-Proving", JACM, Vol. 14, No. 4, pp. 698-704.
7. Robinson, G., and Wos, L., "Paramodulation and Theorem-Proving in First-Order Theories with Equality", Machine Intelligence IV, D. Michie (ed), (to appear - 1969).
8. Green, C., "Theorem-Proving by Resolution as a Basis for Question-Answering Systems", Machine Intelligence 4, D. Michie (ed), American Elsevier, New York, 1969.
9. Hearn, A.C., "Reduce Users Manual", Memo AI-50, Computer Science Dept., Stanford University, Stanford, California 1968.

6. Models of Cognitive Processes

6.1. The Heuristic Dendral Project

The Heuristic DENDRAL Project explores the processes of problem solving, hypothesis formation, and theory formation in scientific work. The particular task chosen for the exploration initially was the determination of the molecular structure of organic molecules from mass spectral data, but is now being extended to include other spectral data.

In the interpretation of mass spectra of organic molecules, a chemist seeks a hypothesis about molecular structure that will serve to explain given empirical data produced by an instrument called a mass spectrometer. Heuristic DENDRAL is a LISP program that performs in this task environment. The overall structure of the program is a "scientific method" cycle, consisting of phases of preliminary inference making, systematic hypothesis generation, prediction from hypotheses, and test against empirical data with evaluation of goodness-of-fit.

The problem solving behavior of the program is remarkably good for the limited subset of organic molecules with which we have worked, in a task that is not a "toy" problem but a "real life" problem of considerable interest to science. For these molecules, the program's ability is at least equal to, and usually better than, that of professional mass spectrum analysts.

The project personnel are: Professor Edward Feigenbaum of the Computer Science Department and Professor Joshua Lederberg of the Department of Genetics; Dr. Bruce Buchanan, Instructor of Genetics and Computer Science; Georgia Sutherland, Research Associate in Computer Science; and Al Delfino, staff programmer. Affiliated with the project

from the Chemistry Department are Professor Carl Djerassi, Professor of Chemistry, and Dr. Alan Duffield and Dr. Gustav Schroll of his Mass Spectrometry Laboratory. Others affiliated with the project are: Professor Malcolm Bersohn, Chemistry Department, University of Toronto, visiting on sabbatical; Mr. Dennis Brown, Computer Science Department student NSF Fellow, and Mr. Isu Fang, Computer Science Department student AID Fellow.

Eleven scientific reports describing the progress of the project have been published or are in press. The most complete and detailed description of the Heuristic DENDRAL program and its implementation was published in the book Machine Intelligence 4 (7). The most concise description was included as part of Feigenbaum's artificial intelligence survey for the IFIP68 Congress in Edinburgh (8). Results that deal specifically with isomers of organic molecules and with mass spectrometry of simple chemical groups have been submitted to (3), and accepted for (1,2), the Journal of the American Chemical Society. An early discussion of Heuristic DENDRAL from the point of view of the modeling of induction and heuristic rules of judgment in scientific problem solving was published in the book Formal Representations for Human Judgment (9). Heuristic DENDRAL is, of course, a precisely specified and working model of induction and empirical inquiry in a scientific task. As such, it is of interest to Philosophy of Science, which has always had as its main concern an understanding of the nature of scientific inquiry. This interest is addressed in an article on Heuristic DENDRAL to appear in the British Journal for the Philosophy of Science (5). Lederberg's DENDRAL algorithm and his DENDRAL structure

notation form the basis for the heuristic problem solving system, and are described in various publications that address themselves to the topology of organic molecules and the manipulation of these organic graphs by computer (6, 10, 11, A).

Chemists supported by other grants and fellowships will continue to be working with the project in the next period as they did in the last period.

Work to be Undertaken During the Period of the Proposal:

1. Application of A.I. to Chemical Inference

As an application to chemistry, Heuristic DENDRAL is open-ended (at least in relation to mass spectrum analysis and some closely allied problems). Increments of new chemical knowledge, formalized by the man-machine technique we have been using, produce beneficial increments in the problem solving capability of the program. This type of work -- of broadening and deepening the application to chemistry -- will continue, with the assistance of our chemist affiliates. This will include greater concentration on ringed structures; adding many more organic "functional groups" to the analysis; bringing to bear additional chemical data (particularly infrared, ultra-violet, and N.M.R. spectra); improving the theory of molecular fragmentation used in the Predictor; and improving the goodness-of-fit evaluation criteria. We will also attempt to make the program available to the rest of the scientific community in LISP on the IBM 360.

2. The Problem of Representation for Problem Solving Systems

The problem of representation has been much discussed as a key problem of artificial intelligence research (8). We will be studying

this problem in a number of different ways in connection with the various representations of the theory of fragmentation of organic molecules that are currently used by the Heuristic DENDRAL program. We will be attempting to write programs that will automatically alter the representation of chemical knowledge, from forms ill-adapted to a particular type of problem solving in the system (but perhaps more convenient for the input of knowledge by chemists) to forms more efficient in the problem solving. Initial experiments will involve programs that will move knowledge about fragmentation processes from the Predictor's form to the Preliminary Inference Maker's form; and programs that will re-represent the Predictor itself in a more convenient flexible form.

3. The Problem of Knowledge Assimilation

This problem is closely allied to the problem of representation. The Heuristic DENDRAL program has its knowledge (its "model") of mass spectrometric and other chemical processes represented in at least four different forms, each particularly well-suited to the performance of a specific type of inference process. But how can the system check the consistency of these various forms of the same knowledge? When a change is made to one of the representations, involving the input of new chemical knowledge at that point, can it be guaranteed that the proper changes will be made in the other representations to preserve consistency in the system's knowledge of chemistry? We have a way of attacking this problem that involves some of the experience to be gained in studies of the problem of representation discussed above, and we plan to devote considerable effort in this period to working this out.

4. The Problem of Man-Machine Communication for Transfer of Knowledge

In the DENDRAL system, man-machine interaction is used not merely to debut and run the program but also as the primary vehicle for the communication of new knowledge about chemistry to the program. The problem of how to do this effectively is intimately bound up with the problems of representation and knowledge assimilation. Chemists do not necessarily (or even usually) have their knowledge about the chemical world represented in the fashion most convenient for our various programs to use. Thus, automatic re-representation of knowledge from the "ground state" forms used by our chemists to more "finely tuned" forms for use by the program is essential.

The work on knowledge assimilation will allow us to experiment with more intelligent man-machine communication. In addition, to facilitate the transfer of knowledge via the man-machine interaction, we plan to develop a "problem-oriented" language for chemistry, oriented toward the problem of expressing chemical knowledge to the system and translating this knowledge to the appropriate internal representation.

5. The Problem of Efficient Design

How knowledge of the chemical world is employed by DENDRAL at the various points in its "scientific method" cycle effects markedly the efficiency of the problem solving process. Thus, some important questions of systems design are raised, which must be carefully studied and understood -- a task that we propose to do. Since there is no epistemological theory which can serve as a guide, this understanding must be gained experimentally. For example, under what general conditions does it "pay" to apply knowledge in the DENDRAL hypothesis generation stage rather than in the hypothesis validation stage (and

vice-versa)? Can, indeed, any general answers be given to system design questions of this sort?

Summary and Generalization:

A problem solving program has been written which exhibits a high level of performance on a complex scientific inference task. Using this performance system as a springboard, a series of additional program-writing endeavors, addressing themselves to what we believe are key problems in artificial intelligence research, are proposed. These involve the problems of: representation; knowledge assimilation; man-machine communications for transfer of knowledge; and efficient systems design. Though these problems are to be studied concretely in the context of chemical inference, they are clearly of considerable general interest. All of them, for example, are important in the development of computer programs for the intelligent "management" of large data bases. The key questions being asked involve how to organize and "manage" knowledge: how to do this most efficiently to control search, and how to get programs to do this automatically, in an effective manner, with minimal human intervention.