

Proposal to
THE ADVANCED RESEARCH PROJECTS AGENCY

For continuation of
THE HEURISTIC PROGRAMMING PROJECT

ARPA Order No. 2494

February 1977

Computer Science Department
STANFORD UNIVERSITY

Proposal to
THE ADVANCED RESEARCH PROJECTS AGENCY

for continuation of

THE HEURISTIC PROGRAMMING PROJECT

Edward A. Feigenbaum, Professor of Computer Science
Joshua Lederberg, Professor of Genetics
Bruce G. Buchanan, Adjunct Professor of Computer Science

Co-Principal Investigators

Abstract

The research project in Heuristic Programming is proposed for continuation of Contract DAHC-15-73-C-0435 for two years beginning July 1, 1976 at a total cost of \$650,000.00. This project will be administratively distinct within the Computer Science Department.

STANFORD UNIVERSITY HEURISTIC PROGRAMMING PROJECT

Principal Investigators
Prof. Edward A. Feigenbaum
Prof. Joshua Lederberg
Prof. Bruce G. Buchanan

Contract No. DAHC 15-73C-0435
July 1, 1977 to June 30, 1979

1 OBJECTIVES

The objectives of the Heuristic Programming project are to provide conceptual and programming tools for building knowledge-based programs. More specifically the proposed work centers on three major themes:

I-A. Generalization of knowledge-based systems design and implementation techniques.

I-B. Extension of current research toward new or more powerful techniques for knowledge-based systems.

II. Orientation toward other ARPA-IPTO advanced R&D efforts.

2 BACKGROUND AND TECHNICAL NEED

2.1 Symbolic Computation

Computer scientists have long recognized that a computer is a general symbol-manipulating device. Arithmetic constitutes a special case of this capability--the manipulation of those symbols that are numbers. In this proposal we will be discussing non-numeric symbol manipulation by computers. In thinking about non-numeric computation, it is useful to think about:

a. inference methods (as opposed to calculation and algorithms)

b. qualitative "lines of reasoning" (as opposed to quantitative formulations)

c. symbolic facts (not merely numeric parameters and formulas)

d. decision rules of expertise and judgment (as opposed to mathe-matical decision rules)

Symbolic computation, though general and powerful, has hardly begun to be exploited in real applications. The specialty within Computer Science that has studied complex methods of symbolic computation is "Artificial Intelligence Research".

2.2 The Intelligent Agent Viewpoint

Though artificial intelligence research has a number of different goals, one of them can be described as the study, design and implementation of "intelligent agent" computer programs. A succinct version of this view was presented by Feigenbaum in a report to the Director of ARPA in 1973, as follows:

Artificial Intelligence research is that part of Computer Science that is concerned with the symbol-manipulation processes that produce intelligent action. By "intelligent action" is meant an act or decision that is goal-oriented, arrived at by an understandable chain of symbolic analysis and reasoning steps, and is one in which knowledge of the world informs and guides the reasoning. The potential uses of computers by people to accomplish tasks can be "one-dimensionalized" into a spectrum representing the nature of instruction that must be given the computer to do its job. Call it the WHAT-TO-HOW spectrum. At one extreme of the spectrum, the user supplies his intelligence to instruct the machine with precision exactly HOW to do his job, step-by-step. Progress in Computer Science can be seen as steps away from that extreme "HOW" point on the spectrum: the familiar panoply of assembly languages, subroutine libraries, compilers, extensible languages, etc. At the other extreme of the spectrum is the user with his real problem (WHAT he wishes the computer, as his instrument, to do for him). He aspires to communicate WHAT he wants done in a language that is comfortable to him (perhaps English); via communication modes that are convenient for him (including perhaps, speech or pictures); with some generality, some abstractness, perhaps some vagueness, imprecision, even error; without having to lay out in detail all necessary subgoals for adequate performance - with reasonable assurance that he is addressing an intelligent agent that is using knowledge of his world to understand his intent, to fill in his vagueness, to make specific his abstractions, to correct his errors, to discover appropriate subgoals, and ultimately to translate WHAT he really wants done into processing steps that define HOW it shall be done by a real computer. The research activity aimed at creating computer programs that act as "intelligent agents" near the WHAT end of the WHAT-TO-HOW spectrum can be viewed as the long-range goal of AI research.

Thus, the Intelligent Agent is a knowledge-based system--the essential system component is a body of knowledge representing the user's problem domain.

2.3 Knowledge-based Systems Research and Design

Early work in artificial intelligence aimed toward the creation of generalized problem solvers. Work on programs like GPS [by Newell and Simon] and theorem proving [Nilsson71], for instance, was inspired by the apparent generality of human intelligence and motivated by the belief that it might prove possible to develop a single program applicable to all (or most) problems. While this early work demonstrated that there was a large body of useful general purpose techniques (such as problem decomposition into subgoals, and heuristic search in its many forms), these techniques did not by themselves offer sufficient power for expert levels of performance. Recent work has instead focused on the incorporation of large amounts of task specific knowledge in what have been called "knowledge-based" systems. Rather than non-specific problem solving power, knowledge based systems have emphasized high performance based on the accumulation of large amounts of knowledge about a single domain. A second successful focus in work on intelligent systems has been the emphasis on the utility of solving "real world" problems, rather than artificial problems fabricated in simplified domains. This is motivated by the belief that artificial problems may prove in the long run to be more a diversion than a foundation for further work, and by the belief that the field has developed sufficiently to provide techniques that can aid working scientists. While artificial problems may serve to isolate and illustrate selected aspects of a task, solutions developed for those selected aspects often do not generalize well to the complete problem.

There are numerous current examples of successful systems embodying both of these trends, systems which apply task-specific knowledge to real world problems.

Our project is widely regarded as the initiator of this line of endeavor. Over the past 12 years, with support from ARPA, NIH, and NSF, our scientists have studied extensively the problems of knowledge-based systems design and have implemented (or are in the process of implementing) a number of such systems. The accomplishments are summarized in the proposal section on Accomplishments, but a narrative here may prove useful.

DENDRAL: An intelligent assistant to an analytic and structural chemist. It infers the structures of complex organic molecules from structural constraints. These constraints are either supplied interactively by the user from his "private"

knowledge and intuition, or are inferred automatically from instrument data, such as mass spectral data, nuclear magnetic resonance data, etc. For those families of molecules for which the knowledge base has been carefully elaborated, the DENDRAL program performs at levels equalling or exceeding the best human experts. The DENDRAL program now has a significant user community in university laboratories and in industry, and is being used to solve difficult real problems.

Meta-DENDRAL: This program is focused on the problem of elaborating DENDRAL's knowledge base for specific families of compounds. It infers an empirical theory (a body of fragmentation rules) of the mass spectrometry of specific families from recorded mass spectral data. It has not only "rediscovered" rules previously acquired from chemists, but has discovered novel rules for certain families--rules that have recently warranted publication in the chemical literature.

MYCIN: This program is an intelligent assistant to a physician diagnosing infectious diseases. In conjunction with its diagnoses, it recommends therapeutic action. It is capable of explaining its line-of-reasoning in any (and varying) level of detail to the user in English. It can accept new decision rules from the user in English. It keeps an updated model of its own knowledge base, which it uses to critique the introduction of new rules into the system. It is capable of acquiring and using measures of the uncertainty of the knowledge, and produces a "believability" index with each inference, i.e. it is capable of approximate implication. A version called EMYCIN, sans infectious disease knowledge, has been developed to extend the use of the system to other domains. (One of the project scientists, E. Shortliffe, was recently awarded the Grace M. Hopper award of the Association for Computing Machinery for his distinguished work on MYCIN).

HASP: Project scientists working in a classified environment led the development of a signal-understanding program for continuous surveillance of certain objects of military interest. The program ran successfully in a number of highly varied test situations, and is being further developed in a currently-funded ARPA program. The program used a design for incremental hypothesis formation that was a modification of the HEARSAY design for the CMU speech-understanding system. Symbolic knowledge from a number of sources was used to aid the interpretation of the primary signal data. Time-dependent analysis was novel in this system and played an important role.

AM: This remarkable program conjectures "interesting" mathematical concepts. Its knowledge base encompasses the (usually private) knowledge of a mathematician as to what constitutes an "interesting" construct in mathematics. Starting with the simplest set-theory concepts, and hundreds of rules

defining "interestingness" of mathematical concepts, it has conjectured such concepts as addition, multiplication, factorization, primes, unique factorization into primes (the fundamental theorem of arithmetic), and an almost unstudied concept in number theory called "maximally divisible numbers".

MOLGEN: (under development) This program is being designed to be an intelligent assistant to an experimental molecular geneticist in formulating plans for laboratory experiments involving the manipulation of short DNA strands with restriction enzymes. The program is concerned with representing knowledge about planning and with the automatic formulation of plans to the level of detail demanded by the user. The program's knowledge must be represented at various levels--biological, genetic, topological, and chemical--and these levels must be incorporated into the reasoning.

Crystallographic Image Interpretation: (under development) This program is being designed to interpret ambiguous, incomplete three-dimensional image data obtained in x-ray crystallography of protein structures. The image input data is the so-called electron density map and the answer desired is an approximately correct protein molecule (or portion thereof). As with HASP, many sources of symbolic data support the interpretation of the primary signal data. The HASP program organization has been imported into this program as a test of its generality. The interpretation problem is difficult because the best wavelength available (x-rays) is too long to resolve atoms and interatomic separations; hence the need for additional sources of symbolic knowledge, e.g. the amino acid sequence of the protein.

In short, as the capsule sketches above indicate, the main themes of our work involve: the acquisition and maintenance of knowledge bases; the utilization of this knowledge in a variety of ways for data interpretation, problem solving, and planning; and the representation of this knowledge for computer inference. These programs constitute superb "laboratories" in which to study the problems of design of knowledge-based systems. Though built to a large extent with non-ARPA support, they serve well the purposes of the ongoing and proposed ARPA-sponsored research.

2.4 Production-rule-based Systems: Methodology and System Features

We sketch below our approach to issues of knowledge representation, management and use. It consists of (a) a representation of knowledge in terms of production rules and (b) interactive programs for acquiring, using, and explaining the knowledge in the system. An example of a production rule from the MYCIN domain is shown below, in both the internal format

(LISP), and the English translation which is generated from it. applicable approach to knowledge representation, management and use. That foundation consists of (a) a representation of knowledge in terms of production rules and (b) interactive programs for acquiring, using, and explaining the knowledge in the system. An example of a production rule from the domain of infectious disease diagnosis is shown below, in both the internal format (LISP), and the English translation which is generated from it.

```

PREMISE  ($AND (SAME CNTXT GRAM GRAMNEG)
            (SAME CNTXT MORPH ROD)
            (SAME CNTXT AIR ANAEROBC))

ACTION   (CONCLUDE CNTXT IDENT BACTEROIDES TALLY .6)

```

If 1) the gram stain of the organism is gram negative, and
 2) the morphology of the organism is rod, and
 3) the aerobicity of the organism is anaerobic,
 then there is suggestive evidence (.6) that the identity of
 the organism is Bacteroides.

The premise of the rule is a Boolean combination of preconditions, each of which is a predicate testing the value of an associative triple, and is thus value triple.

Such rules contain modular "chunks" of knowledge about the domain, represented in a form that will be comprehensible to someone who is acquainted with the field, but may know nothing about computer programming. Many unique and important capabilities are made possible by encoding knowledge in this form, capabilities reflected in the organization and architecture of the system.

The knowledge base of a system contains the collection of domain specific rules which give the system its competence in a particular domain. The problem data base contains information about a specific problem given to the system.

An interactive "intelligent agent" program uses the collection of rules in the knowledge base to reason about the task at hand. If, for, instance, the MYCIN program is attempting to determine the identity of an organism responsible for a particular infection, it retrieves the entire list of rules which, like the one above, conclude about identity. It then attempts to ascertain whether the conclusion of the first rule is valid, by evaluating in turn each of the clauses of the premise. Thus, for the rule above, the first thing to find out is the gram stain of the current organism. If this information is already

available in the data base, the program retrieves it. If not, determination of gram stain becomes the objective of a new rule, and the program retrieves all rules which conclude about it, and tries to use each of them to obtain the value of gram stain. If, after trying all the relevant rules, the answer still has not been discovered, the program asks the user for the relevant information which will permit it to establish the validity of the premise clause. Thus, the rules unwind to produce a succession of goals (i.e., the system performs an exhaustive depth first search of the and/or goal tree formed by the rules), and it is the attempt to achieve each goal that drives the consultation.

The use of a rule-based representation of knowledge also makes it possible for the system to offer explanations of all intermediate and final conclusions. During the course of interaction with the user, for example, the consultation program requests many pieces of information. If the motivation for requesting any particular piece of information is unclear, the user may temporarily put off answering and instead inquire why such information is needed and how it will be used. The explanation program which allows him to do this can explore past, current, and future (potential) lines of reasoning, at varying levels of detail according to instructions from the user.

A second form of explanation is available after the consultation has ended. For example, again using MYCIN to illustrate, if a user asks "How did you determine the identity of the organism?" the program answers by displaying the rules which were actually used, and explaining, if requested, how each of the premises of the rules was established. This is something which someone familiar with the domain (in this case, a physician) can readily understand without having to know very much about either computers in general or this particular system. Note that it also provides a far more comprehensible and acceptable explanation that would be possible if the methodology employed were to be based statistical approaches to decision making.

The knowledge acquisition program can aid in the task of adding new "chunks" of knowledge to the program's expanding knowledge base. The rule-based representation of knowledge means that the expert himself can specify the new knowledge by expressing it in this format. He can thus help make the program more competent, without having to know anything about computer programming. In addition, since the rules are designed to be independent of one another, and are used by the program as necessary in order to deal with the particular situation under consideration, the addition of a new rule or modification of an existing rule does not require alteration of other items in the knowledge base, as is often necessary with systems using other methodologies.

2.5 Judgmental and Inexact Knowledge

Since we want to deal with real-world domains in which reasoning is often judgmental and inexact, we require some mechanism for being able to say that "A suggests B", or "C and D tend to rule out E." The numbers used to indicate the strength of a rule have been called Certainty Factors (CFs). The methods for combining CFs are embodied in a model of approximate implication. These are derived from and are related to probabilities, but are distinctly different. For the rule given above, the evidence is "suggestive" (.6 out of 1), but not absolutely certain. Evidence confirming an hypothesis is collected separately from that which disconfirms it, and the truth of the hypothesis at any time is the algebraic sum of the current evidence for and against it. This is an important aspect of the truth model, since it makes plausible the simultaneous existence of evidence in favor and against the same hypothesis. We believe this is an important characteristic of any model of inexact reasoning.

2.6 Incomplete knowledge

Another fundamental characteristic of the information in real world problems is its incompleteness. In a military domain for instance, information about an unfolding situation may be incomplete sometimes because it is totally unavailable, or more typically, because it is not possible to collect all necessary data before deciding upon a course of action. Since both of these are common to problems in a wide range domains, a basic requirement of an intelligent system is that it be able to reach a decision based on whatever information is currently available.

Unlike many previous methodologies for decision making (e.g., decision trees), the goal-directed chaining of that we used above as our example rules is quite flexible with respect to the amount of information present, because the reasoning chains are constructed dynamically. Decision constraints (decision "logic") are computed at time of execution. There is thus no need to attempt to create a pre-determined sequence of decisions that takes account of every contingency. Instead the information present at the time of the problem-solving interaction will itself determine the direction in which reasoning ought to proceed.

2.7 Multiple Uses of Knowledge Bases

The accumulation of a large store of task-specific knowledge presents a number of unique opportunities. First, of course, it supports high performance, and this is typically why

it is assembled. But there are numerous other ways in which the same knowledge base can be employed, with little or no extra overhead. One area of investigation is its use as a reference source for computer-assisted education. That is, in much the same way that an expert taught the system about the domain, is it not possible to have the system teach a student? In a similar vein, it might be used as an on-line reference source, one that was easily searched, read, and modified as knowledge in the domain accumulated.

2.8 Summary of Advantages of Production Rule Methodology

2.8.1 Modularity and Ease of Modification by Non-programmer User:

As discussed above, the individual rules, as "units" of knowledge, can be removed or replaced quickly by the user-analyst. This makes for a highly adaptive man-computer system, one which gives the user the (correct) feeling that he is in charge of the nature and course of the analysis method and strategy. Such a system is bound to be more powerful and responsive than a computer-only inference system.

2.8.2 Flexibility with Robustness--Knowledge is separated from software

Most large programs are currently similar to a house of cards in that changes or additions must be made with the utmost care, lest unanticipated remote effects make the entire structure collapse. Generally there are too many unknown, unstable, and critical interdependencies. Using production rules, the effects of changes are propagated naturally and non-destructively.

2.8.3 Highly integrated and dynamic documentation

Documentation is natural, self-evident, and (where necessary) enforced. The rules themselves are represented in a language "natural" to the user-analyst (e.g. English for a physician-user of MYCIN, chemical graphs for a user of DENDRAL). The system can answer the primary explanatory questions concerning use of rules: HOW is the rule being used and WHY was the rule invoked in the line-of-reasoning. Rule forms govern the acquisition of a new rule, and include all information appropriate and useful for documentation (including ancillary information about when, by whom, and why it was added to the knowledge base). In addition, the same kind of "natural" questioning can be used to interrogate the knowledge base. Thus a Manual for the knowledge base is unnecessary.

2.8.4 Rule criticism during Acquisition

Because the acquisition of new rules is controlled by the so-called "rule models" inferred from an examination of the knowledge base, the system can catch many errors of commission and omission made by the user.

2.9 Technical Need

The defense systems of the United States are as much information processing systems as they are systems of weapons technology and use. Military personnel, particularly those in command and other decision-making positions, are routinely inundated with massive amounts of problem data. They use their expertise and good judgment to convert that data into relevant information on the basis of which intelligent choices and plans can be made. There is a pressing and omnipresent need in the services for computer-based intelligent assistance in this vital data-to-information transformation. Our basic research has been aimed at developing the methodology that will facilitate the programming of such agents; and at implementing real systems in physical, chemical, biological, medical, and military problem domains so as to give credibility to the view that such agents are possible.

Traditional mathematical methods are often powerful, but are sometimes abused, and in any event are scarce. These methods are, for most problems of interest, being pressed to their limits in appropriately representing problems of interest. Most of the world's (and the military's) problems are not basically problems of quantitative formulation and analysis, and the attempt to put them in this straightjacket is often frustrating.

Most real-world problems involving decision making and problem solving are symbolic in nature, involving qualitative reasoning, using knowledge that is informal, experiential, judgmental, and often tacit (i.e. private). As we have said, our research is aimed at this class of problems.

"Expert" knowledge-based intelligent-agent programs meet these kinds of needs (among many others):

a. the problem-dependent knowledge base can be made thorough and more complete than that known to any individual user

b. the use of the knowledge in decision-making and problem solving can be made highly systematic, so that relevant knowledge or appropriate solution paths are not inadvertently ignored.

c. the systems can be made available in crisis situations

that are typically characterized by information overload of the human analysts and decision makers.

d. symbolic knowledge can help to structure and guide expensive computational processes, by focusing the processing on appropriate calculations, thereby reducing the computational time and cost by (sometimes) orders of magnitude (this advantage was most obvious in the HASP domain).

Finally, from the point of view of the people doing analysis, decision-making, planning, etc. (i.e. from the human engineering point of view) there are these needs:

a. for systems that are easily expanded and modified, so that the system can change as human understanding of the problem changes

b. for programs that can represent human expertise and inexactness of human judgment in a natural way.

c. for programs that can build and maintain credibility with their users by offering clearly understandable explanations of their lines-of-reasoning, thereby making the users feel comfortable in accepting assistance from the agent.

2.10 Technology Transfer

In view of the importance of the needs described above, our project scientists have been, and are now, engaged in efforts to transfer our techniques and methods to technologists working on defense systems, and to various scientific disciplines.

Our previous proposals have documented our technology efforts of past years. The two year period just ending has been rich with such activities.

a. Feigenbaum and other project scientists have interacted vigorously with RAND Corporation scientists on the transfer of methods and systems developed here to the RITA system development for command-and-control applications.

b. We have had similar high levels of interaction with scientists at SRI on the development of their systems and methods in military applications. Of particular note is our interaction with them on Inference Nets; and the participation of Feigenbaum and Lederberg at a special meeting relating to EW analysis issues.

c. Feigenbaum and Nii have worked with the scientists at Systems Control Inc. on the follow-on effort to HASP (the SIAP program)>

d. Feigenbaum gave an all-day seminar to Senior Engineers at the National Security Agency on Knowledge-based systems, and our project scientists have held meetings with NSA scientists on this subject.

e. Feigenbaum is consulting with System Development Corporation on the writing of a tutorial report on Knowledge-based Systems for a military agency.

f. Feigenbaum has interacted with scientists at NELC on the formulation of a Navy project applying knowledge-based system methods to a C-C Testbed application.

g. Feigenbaum and project scientists have interacted with engineers at General Motors Research on the application of knowledge-based systems to their industrial problems.

h. Considerable interaction with industrial chemists regarding the use of the DENDRAL programs has been underway for some time. One of our scientists will be spending a year bringing up an "export" version of DENDRAL in England under the sponsorship of the Science Research Council and Shell Oil Co. Other industrial users of the programs include Eli Lilly Co., Monsanto Chemical Co., Du Pont, and Kodak.

i. A general and massive effort at technology transfer has been the initiation of the Handbook of Artificial Intelligence, being written by Feigenbaum and his students (see subsequent proposal material for more details).

j. The AGE project described later in the proposal is a major effort to make available significant portions of our technology in the form of software packages to a broad scientific and engineering community.

3 ACCOMPLISHMENTS

The Heuristic Programming Project has become known as one of the most active artificial intelligence research groups in the country because of our long history of accomplishments as well as the intensity of our current work. Summarized below are the most notable accomplishments in the past two year period, followed by a review of our work in the years prior to that.

3.1 1976 to 1977

1) Production-Rules

The production rule methodology was extended and generalized in two important respects. First, the formalism for the premise clauses was extended to allow comparisons among objects as well as tests on individual objects. For example, instead of merely testing to see if the identity of an infecting organism is e.coli, the MYCIN program can now test to see if the current organism has the same identity as any other organism cultured from this patient. Second, explanation capabilities for production rule systems were extended to answer a broader range of questions. For example, questions about why the MYCIN program did NOT consider e.coli to be a likely infecting organism can now be answered. Negative questions are often highly relevant, but are difficult for a program to answer because their answers come from rules that the program did NOT invoke or procedures that did NOT get executed.

2) Planning...particularly planning of experiments

The framework was laid for a rule-based hierarchical planning program to guide scientific experimentation. The knowledge needed to make the planning inferences can come from multiple sources of scientific knowledge. (Coordinate funding to support much of this work was obtained from NSF.) The testbed for these ideas is the MOLGEN experiment planning program in molecular genetics, largely funded by the NSF. The planning problems are general: how to represent planning strategies, how to represent objects at arbitrary levels of detail, how to cope with imprecision in the planning rules.

3) Meta-DENDRAL: new results=new science

Meta-DENDRAL successfully formulated rules of mass spectrometry that were new to the science. These rules, along with a discussion of the methodology, were published in the scientific literature [\[ref XXII\]](#). The program was tested to see if it could rediscover the rules of mass spectrometry for two classes of chemical compounds that were already well understood (amines and estrogenic steroids). Then it was applied to three classes of compounds whose mass spectrometry was not as well known (mono-, di-, and tri-ketoandrostanes). The program produced three sets of rules that explained much of the significant data for these classes. The time for manual rule formation for these data was estimated to be several months. Automatic rule-formation was completed in less than an hour of processing time.

4) Meta-DENDRAL: generalization

Progress was made on generalizing the Meta-DENDRAL program; and rules for a new domain were successfully discovered by the program. A scientific paper on this application was submitted for publication [\[ref Schwenzer\]](#). The new application was:

learning rules for interpreting signals from C13-NMR spectroscopy. The instrument produces data points in a bar graph in response to the resonance of each carbon-13 nucleus in the sample. The rules describe an environment of a C13 atom and predict a resonating frequency range for every atom that matches the description. The Meta-DENDRAL program needed some modification because the rules are predicting ranges of data points, and not precise processes, as for the mass spectrometry version.

5) Meta-DENDRAL: Methodology

A report was written on the AI methodology underlying Meta-DENDRAL [[ref BGB/TM}. The major idea is using knowledge of the domain to guide a learning program. The major difference between Meta-DENDRAL and statistical learning programs is that Meta-DENDRAL uses a strong model of mass spectrometry, including any assumptions the user cares to make about the domain, to guide the formation of explanatory rules.

6) Meta-DENDRAL: negative results

Our attempt to produce rules of various forms from Meta-DENDRAL was premature because the syntactic form of the rules is intimately tied to the rule generator. We are working on the problem of generalizing Meta-DENDRAL and this task is among those that we consider important. Of course, the program can search an immense variety of rules within a specified family. At issue here is the lack of flexibility and generality across family boundaries.

7) HASP: generalization

The control structure and representational scheme of the HASP program was successfully mapped into a new domain (image analysis of x-ray crystallographic data, as described in the background section). A working paper on this was published [[ref Nii)].

8) Knowledge Engineering: techniques for interactive control

Mechanisms were developed for giving users of rule-based interactive programs more control over the running programs. While a knowledge-based system is making its inferences, in between statements that get printed, a user often wonders what is going on and whether the answer from the program will be worth the wait. We introduced into the CONGEN program two helpful interrupt capabilities that can be generalized to other programs. First, the user can interrupt the computation to ask for an estimate of the amount of work remaining to be done. This requires a model of how to extrapolate from current status to

final status. Second, the user can request any, or all, of the intermediate solutions already generated in order to decide whether to continue the work or to introduce new constraints on the problem.

9) Rule-based teaching program: held in abeyance

Before we could produce an "intelligent teacher" that used an existing production rule knowledge for instructional purposes, it was necessary to work on the problem of providing explanations in many different ways (proposed here). Thus the effort expended on this problem only uncovered new research problems for the time.

10) Distributed computing

The first version of the "contract nets" programming organization was developed and a paper about it written. Contract nets will allow various types of AI methods to be programmed for parallel, asynchronous nets of computers.

11) Documentation of AI concepts, techniques, programming methodology: AI Handbook

Organization of the volume was completed, and first drafts of most of the volume were completed. The publication of a technical report was premature, but informal copies of material were sent to ARPA-IPTO.

3.2 1975 to 1976

1) The methodology and a prototype system for encoding rules of strategy for problem solving were developed for production rule systems. We call these "meta-rules" and have reported their use in [[ref Davis1976}. A meta-rule reorders or prunes the list of relevant domain rules to be invoked on the basis of the encoded strategy to pursue the consultation one way rather than another.

2) A program (named Teiresias) was written for interactive acquisition of new rules, using strong syntactic models of the rules to guide the interaction. This work was reported in [[ref Davis1976}. The system has strong expectations of what to expect in rules that make certain conclusions and in rules that refer to various attributes in their premises. Thus it can remind the user to include additional clauses in the rule and it can interpret many ambiguous statements written by the user.

3) A program was written to formulate models of rule-forms in a production rule system's knowledge base. These we call

"rule schemas" and their automatic formulation and use are reported in [[ref Davis1976}. These models are induced over the set of the existing rules in the system at any moment. They are changed each time a new rule is added so that the information is always current.

4) The MYCIN program was "emptied" of its medical knowledge to produce a program that could accept knowledge of non-medical (or other) domains. This version still makes strong assumptions about the organization of objects in the domain and about the nature of rules. If the structure of the new domain is similar to MYCIN's in these respects, however, the program can accept new rules and offer expert advice in this domain. One test domain was the diagnosis and recommendation for fixing problems in an automobile horn.

5) The RULEMOD program for "fine-tuning" Meta-DENDRAL's newly-discovered rules was finished. This program provides a number of important subtasks, including merging similar rules, making rules more specific or more general, and filtering out the weakest rules. RULEMOD checks for counterexamples to rules and uses this information in all of the named tasks. Because of the expense of computing counterexamples to possible rules, this computation is delayed until Meta-DENDRAL has a set of plausible rules, rather than computing counterexamples on each possible rule examined in the search of the rule space.

6) The RULEGEN component of Meta-DENDRAL was demonstrated to work with heuristic search methods used in many other AI programs (cf. Nilsson's book, Problem-solving Methods in Artificial Intelligence, McGraw-Hill). Guidance from a model of mass spectrometry is an important feature of RULEGEN. Also, the program uses problem data for pruning possible rules (and all more specific rules formed from those). The amount of data examined during the search is very large and the space of rules is immense, so the search needs to be rather coarse in order to produce plausible, but not necessarily optimal, rules.

7) A program was written to discover new, interesting conjectures in mathematics. This program, called AM, starts with a set of fundamental propositions of number theory and explores interesting extensions to the set, for example by looking at analogies and by looking for interesting properties that various subsets of the numbers share. This work was reported in [[Lenat}.

8) A rule-based system for integrating multiple sources of knowledge, in the task of interpreting 3-D image data, was designed and partly implemented. (This task was funded in large part by the National Science Foundation.) The data for this prototype system come from the domain of protein crystallography. KSS are being developed and implemented for identifying

individual molecules in the crystal structure, locating heavy atoms and cofactors, locating the backbone of the polymer, identifying side chains, matching the features in the data to the amino acid sequence, etc. The inferred structure is a multi-level hypothesis, conforming to the levels of detail to which model builders refer in applying their skill. A flexible, rule-based control structure allows the system to be exercised in many different modes, e.g. event-driven or goal-driven. Knowledge sources are kept distinct from the program that deploys them, so that the effectiveness of specific KSs on the system's performance can be readily investigated.

3.3 Years prior to 1975

Under ARPA support, and later under NIH support, the HPP developed a high performance, knowledge-based program for the interpretation and explanation of signal data from a mass spectrometer. This is the Heuristic DENDRAL program, which is now widely acclaimed as one of the most powerful and useful of AI programs.

The problem solving paradigm that has emerged from DENDRAL work is the so-called "plan-generate-test" paradigm. It is based on heuristic search of a space of possible hypotheses with planning before generation of hypotheses and testing of each generated candidate.

The generator for DENDRAL is a general-purpose graph generator which produces a list of all possible graphs containing specified numbers of nodes of various types. The most important features of the generator are that the list of graphs is guaranteed to be complete and non-redundant and, equally important, that the list need not be exhaustively generated. The generator can be constrained to produce only graphs that meet specified criteria that are inferred from the initial problem data.

The planning program is domain-specific. It contains a large amount of knowledge of mass spectrometer fragmentation processes, nuclear magnetic resonance phenomena, chemical stability, etc., and can accept additional knowledge (or assumptions) about the problem area from the scientist using the program.

The testing program uses a theory of mass spectrometry to make testable predictions for each candidate hypothesis. Matching the observed data with the prediction, then, allows the program to disconfirm some hypotheses and confirm the rest to varying degrees. The confirmed hypotheses are ranked and the most plausible hypotheses are given to the user as the best explanations of the initial data.

The success of any reasoning program is strongly dependent on the amount of domain-specific knowledge it contains. This is now almost universally accepted within AI, but DENDRAL's success is one of the reasons for its acceptance. We also showed the overall benefits of separating the knowledge of the domain from the computer code that manipulates that knowledge for problem solving.

Because of the difficulty of extracting specific knowledge from experts to put into the program, we began to explore the problems of transferring knowledge into a program efficiently. We have looked at two alternatives to "hand-crafting" each new knowledge base: interactive knowledge transfer programs and automatic theory formation programs. In this enterprise the separation of domain-specific knowledge from the computer programs themselves has been a critical component of our success.

We first developed a representation of knowledge that is natural and easy for experts to think about. The ideas had been discussed by others (e.g., Newell), but we brought them together in a problem-solving system in science. We codify individual facts about the domain as attributes and values associated with objects -- "the X of Y is Z". In addition, we developed an easy way of encoding and manipulating the degree of certainty (called the "certainty factor" or CF) associated with each fact. The rules of inference between facts are then encoded in conditional sentences, or production rules, which allow new conclusions to be made (with some CF) any time the statements mentioned in the premise clauses are true (or "true enough").

For interactive knowledge transfer we developed programs that carry on a dialogue with an expert in order to help him encode his knowledge of the domain. At first these programs were extremely simple, but we have by now given them the ability to understand stylized (not entirely free form) English sentences typed by the expert and have given them some understanding of the content, as well as the form, of the rules.

Automatic knowledge acquisition is a longer-term research problem. One of the stumbling blocks with the interactive knowledge transfer programs is that for some domains there are no experts with enough specific knowledge to make a high performance problem solving program. We were looking for ways to avoid forcing an expert to focus on original data in order to codify the rules explaining those data because that is such a time-consuming process. Therefore we began working on an automatic rule formation program (called Meta-DENDRAL) that examines the original data itself in order to discover the inference rules for that part of the domain.

The problem solving paradigm for Meta-DENDRAL is also the plan-generate-test paradigm used in Heuristic DENDRAL. In this

case the program generates plausible rules within syntactic and semantic constraints and within desired limits of evidential support. The model used to guide the generation of rules is particularly important since the space is enormous. The planning part of the program collects and summarizes the evidential support. The testing part looks for counterexamples to rules and makes modifications to the rules in order to increase their generality and simplicity and to decrease the total number of rules.

Another application of artificial intelligence has been made by the MYCIN group, now called the Knowledge-Based Consultation Systems group to reflect our interest in basic methodological issues. Work on MYCIN began in 1971 with a small group looking specifically at medical decision making. The resulting program, based on many of the ideas developed during work on DENDRAL, was shown to perform at the level of an expert in one area of medicine. The methodology was general, however, and we sought ARPA support in 1975 to extend that methodology beyond the one applications area. This was the start of our work on generalized production systems, along with programs that could explain the system's reasoning and programs to acquire new production rules from experts. Under separate funding the MYCIN program continues to develop into a high performance system that can provide expert advice about problems that are too complex for a non-expert to deal with alone.

Other research groups have studied MYCIN for their own applications, including ARPA-sponsored groups at SRI and RAND. The conceptual simplicity of the production rule interpreter in MYCIN makes it relatively easy to map the program into new domains, with their own sets of domain-specific rules. MYCIN's rule base contains about 400 rules at present and so we are looking closely at ways to keep the amount of processing from growing with the number of rules.

4 PROPOSED WORK

4.1 Prologue

The Heuristic Programming Project's contract support from ARPA is part of a mosaic of funding that supports basic research in the design and engineering of knowledge-based systems. The work is usually set in the context of domains of knowledge in science or medicine. The ARPA contract supports this work in the direction of research goals of interest to ARPA's IPTO in the area of knowledge-based systems. Grant support from the National

Science Foundation and the National Institutes of Health provides domain expertise as well as some of the basic computer science support. because the task domains and the basic research goals of the ARPA-sponsored research mesh so well with NIH's goals for the SUMEX-AIM computer resource, pdp-10 time and file space is provided by SUMEX-AIM to the ARPA project. The synergism among these various projects and interests has been very great, making this project one of the most effective--and indeed one of the most cost-effective--in the history of artificial intelligence research.

4.2 Major Themes

The statement of tasks that follows is woven of three major themes:

I-A. Generalization of knowledge-based systems design and implementation techniques. The systems we have built are sufficient in number and variety to support an attempt to generalize our techniques so that they will be readily accessible to a broad community of system builders, not merely the small Stanford group. We believe that we are at a significant point of cumulation in this science, and that the product of the scientific generalization should be programs embodying the generalized methods serving future knowledge-based systems design and implementation efforts.

I-B. Extension of current research toward new or more powerful techniques for knowledge-based systems. The line of research we have been pursuing has been very fruitful. Most of the important scientific thrusts are currently active but not yet fully exploited. There are substantial scientific and practical benefits to pushing ahead to a fuller exploitation during the coming period. Much of the work on knowledge acquisition, representation of production rules, integration of multiple sources of knowledge, etc., is to be viewed in this light.

II. Orientation toward other ARPA-IPTO advanced R&D efforts. We are orienting a portion of our total effort to a rule-based approach to some of the problems in the interpretation of image data; and to a study of the program organizations and signal-interpretation methods that might be used in distributed sensor net applications.

4.3 Tasks

The following list summarizes the research tasks to be performed during the two year period. It is broken into two major parts. The first set of tasks are frontier problems of

advanced symbolic computing. The second set includes research problems that relate directly to other ARPA-IPTO programs.

I. Expert Knowledge Based Systems and Heuristic Rule Technique Development

Proposed is a continuation of our basic research on knowledge-based systems design and techniques for representing "expert" symbolic knowledge in the form of heuristic rules. The research is composed of five major parts (A-E) discussed below.

A. Development of program-packages embodying generalized techniques for work on knowledge-based systems ; preparation of a compendium of these techniques and associated methods and programs in artificial intelligence research.

1. AGE package ("Attempt to GEneralize"). Isolate inference, control and representation techniques from previous knowledge-based programs; reprogram them for domain independence; write a rule-based interface that will help a user understand what the package offers and how to use the modules; and make the package available to other ARPA contractors, service labs doing knowledge-based systems development, and the general scientific community.

Detailed Discussion: The goal of this new effort is to construct a computer program to facilitate the building of knowledge-based systems. The design and implementation of the program will be based primarily on the experience gained in building knowledge-based systems at the Heuristic Programming Project in the last decade. The programs that have been built are: DENDRAL, meta-DENDRAL, MYCIN, AM, HASP, and MOLGEN (currently under development). Initially, The AGE program will embody methods used in our programs. However, the long-range objective is to integrate methods and techniques developed at other A.I. laboratories. The final product is to be a collection of useful "building-block" subprograms, combined with a knowledge-based front-end that will assist a user in constructing knowledge-based programs. It is hoped that AGE can speed up this process and facilitate transfer of the technology by: (1) packaging common AI software tools so that they do not need to be reprogrammed for every problem; and (2) helping people who are not knowledge-engineering specialists to write knowledge-based programs.

Two Specific Research Activities of the AGE Effort are:

1. The isolation of techniques used in knowledge-based systems.

It has always been difficult to determine if a particular problem-solving method used in a knowledge-based program is

"special" to a particular domain or whether it generalizes easily to other domains. In the currently existing knowledge-based programs the domain-specific knowledge and the manipulation of such knowledge using AI techniques are often so closely coupled that it is difficult to make use of the programs for other domains. We need to isolate the AI techniques that are general to determine precisely the conditions for their use.

2. Guiding users in the application of these techniques.

Once the various techniques are isolated and programmed for use, an "intelligent front end" is needed to guide users in their application. Initially, we assume that the user understands AI techniques and knows what he wants to do, but that he does not understand how to use the AGE program to accomplish his task. The program at this stage of the development will need to have the basic tools coupled with a package to guide the user in applying these tools. A longer-range interest involves helping the user determine what techniques are applicable to his task. That is, we assume that the user does not understand the necessary techniques of writing knowledge-based programs. Some questions to be posed are: What are the criteria for determining if a particular application is suited to a particular problem-solving framework? How do you decide the best way to represent knowledge for a given problem? There are some smaller, but by no means trivial, questions which also need answering. Is there a "best way" to write production rules which would apply to many task domains? Is there a data representation that would cover many tasks? What is the best way to handle differences in the ability of the users of the AGE program?

Research Plan: The AGE program will be developed along two separate fronts, both of which are divided into incremental development stages. The first of these fronts is the development of the ability to help build many different types of knowledge-based programs (the "generality" front). The second front is the development of "intelligence" in the interaction between the user and the AGE program; i.e. moving from dialogues on "how to use the tools in AGE" to "what tools to use" (the "how-to-what" dialogue front). The current plan for the development of the AGE program, with approximate milestone dates, are listed below.

a. Generality: The development of a program package that will enable the user to build HASP-like knowledge-based programs. The applicable tasks are characterized by the need for: the integration of multiple sources of knowledge, multi-level representation of solution hypothesis, opportunistic problem-solving methods, and explanation capability of the reasoning steps. HASP-like paradigm has been used to solve problems of interpreting large amounts of digitized physical signals, but can also be extended to problems of processing large amounts of symbolic data. Dialogue: The development of dialogue to show the

user how to utilize the packaged components in AGE to build HASP-like programs. The interactive capability will be limited to: specifying how to build multi-level hypothesis structure; how to write production rules to represent domain knowledge; and how to use various techniques available for opportunistic hypothesis formation. (12/77)

b. Generality: Supplement the ability to build HASP-like programs with a capability to build MYCIN-like goal oriented programs. Dialogue: Same level of dialogue capability with additional ability to discuss how to chain rules and how to specify the necessary parameters for the context tree. (9/78) c. Generality: Same level as for 9/78, i.e. ability to build HASP-like, MYCIN-like or combination of HASP- and MYCIN-like knowledge-based programs. Dialogue: Begin to extract from the user some key characteristics of the task, and using that information begin to suggest appropriate knowledge representation and problem-solving techniques for the user's task. This interactive capability will be limited to the generality level at this point in the AGE development. (12/78) d. Test phase: the usefulness of the AGE system by developing an application program in some task domain. (a) An application program will be chosen from among ongoing program development efforts within our own project or within the ARPA contractor community. An application will be chosen whose primary task is that of interpreting large amounts of symbolic data or described signal data. Among the possibly appropriate tasks would be one of rule-based image analysis (see proposal section IIa.). (The search for an application will be concurrent with the development of the dialogue capabilities of 3., 12/78). (b) Collect specific knowledge needed for the application program and begin to develop the program using the AGE system (6/79).

2. PLAN package. Oriented toward the representation of plans-of-action and toward an expert's knowledge of the best problem solving strategies to employ in his domain. Will do inference on components of planning and strategy rules so that new plans and strategies can be constructed readily from previous ones. The representation will allow the manipulation of various "levels of detail" of plans and strategies. The package will be made available as previously mentioned in connection with AGE.

Detailed Discussion:

Before starting a technical presentation of the ideas for the Plan Package, it is worth highlighting some of the issues which motivate its development.

a. How can a variety of types of domain actions be accommodated in a knowledge base?

b. How can a variety of types of strategy and control knowledge be incorporated in a knowledge base?

c. How can a variety of types of problem solving states be expressed and manipulated by the system?

d. How should plans be represented?

e. How can the problem statements for a variety of types of problems be acquired?

f. How does the expression and representation of problem solving states relate to the expression of the domain and strategy knowledge?

The Plan Package consists of two major entities -- the Planning Network and the Strategy Package. The Planning Network is a set of software which manages the representation of the plans created during the problem solving process. When a problem is acquired from a user, it is represented as an initial planning network. Problem solving takes place as the active strategy rules manipulate the planning network to create solutions. The Strategy Package itself is discussed in the next section.

Since the planning state knowledge is important for the expression of strategy in the Plan Package, it is worthwhile exploring briefly the nature of this knowledge. It is useful to consider the planning network as being composed of three parallel planes -- the solution plane, the planning plane, and the focus plane. These planes contain (1) the solution steps (domain rule applications) and world states, (2) the planning and design steps and (3) the focus of attention knowledge respectively. All three planes of the network are built dynamically during the problem solving process. Different types of nodes in the network correspond to the different components of the problem solving process.

A number of issues have been raised about the management of strategy knowledge.

a. How should strategies be expressed?

b. How can strategy information be assimilated so that the system will use it appropriately when designing or explaining solutions?

c. How can a knowledge based system assist a domain expert in structuring and expressing his ideas about strategy?

Means-ends analysis is one of the simplest ideas in the current stock of methods for problem solving. As such, it should exist as a standard strategy in strategy package of artificial intelligence techniques to be used as needed. The current state

of artificial intelligence, where a researcher must re-code Means-ends analysis any time he wishes to use it is akin to a carpenter forging a new hammer for each job.

One approach for making an instance of Means-ends analysis available as a tool would be to provide a packaged program which accepts arguments for the various components of Means-ends analysis (eg. a difference table, difference function, etc.). The alternative being proposed here is a system which uses schemata to drive the strategy acquisition process and which can guide a user through the details. The goal is to create a supportive environment for the painless testing of fairly high level strategies. Such a system should be able to draw on its knowledge base to provide assistance in casting a problem into a Means-ends framework.

In summary, other systems have stumbled over the expression of more complex forms of domain and strategy rules and have been limited to solving a single kind of problem. We propose extending this work by developing what we have termed the Plan Package. The Plan Package consists of two major components - a schema-based representation for the problem-solving states termed the Planning Network and a schema-based representation for domain rules and strategies termed the Strategy Package. The Planning Network will provide a representation for a variety of types of problem solving so that the problem solving system will be able to solve more than one type of problem. The Strategy Package will provide a set of standard artificial intelligence strategies in the form of schemata, which may be instantiated into strategy rules when they are supplied with the particulars of domain knowledge. These schemata will facilitate the acquisition of tailored strategies by guiding a user a step at a time through the particulars of the acquisition process.

Research Plan:

The Plan Package will be developed and tested in the domain of molecular genetics as part of the MOLGEN project. It will be further developed and extended to other domains as a test for generality as part of the AGE project.

| Time | Milestone |
|-------------|--|
| 3mo MORSE | -Molgen Object Rule & Strategy Editor operational objects. |
| 6mo MORSE | operational for rules. System capable of simulating several kinds of laboratory experiments. Geneticists begin using system to build Knowledge Base. |
| 9mo MORSE | operational for some strategies. Capable of planning limited classes of experiments. Primitive performance monitoring operational. Begin CS experiments. |
| 12mo MOLGEN | starts having impact by trying to design real experiments. Penny starts getting involved to use parts of system for experiments in AGE project. |

3. AIHANDBOOK. Documentation of methods, techniques, programs, languages, etc., with tutorial overviews, whose goal is the transfer of knowledge-based systems and other A.I. technology to the broader engineering, scientific, and military community.

Detailed Discussion:

A significant "technology transfer" gap exists in the dissemination of information concerning the concepts, methods, techniques, and programs of artificial intelligence research. The quality of the science is widely recognized (e.g. four of the twelve Turing Award winners of the Association for Computing Machinery are researchers in artificial intelligence). But the use of the science in engineering applications is very small. And the amount of university teaching being done to train new scientists and engineers in the use of the work is also small. Consequently, we were motivated to begin a major work of documentation to fill the gap. The emerging pair of volumes is called the Handbook of Artificial Intelligence. It is being prepared by project scientists and Stanford Computer Science Students. It is a project of low cost and potentially high impact.

The Handbook will consist of approximately two hundred articles on all aspects of artificial intelligence research and development. Each article will be approximately three pages in length; will include brief discussions of the concept, method, technique, program, etc., and a small set of primary source references for further study. In addition, the Handbook will contain numerous "overview" articles summarizing main lines of activity. The entire work will be tied together by an index constructed as a relational network (a "semantic net") so that a user can be led to an article related to his interest even though he did not know that he needed that article.

The outline for the AI Handbook has been put together with the advice and consultation of many AI scientists from laboratories in this country and abroad.

Research Plan:

The Handbook will be produced first in a preliminary edition of two volumes, so that comments, criticism, and correction can be obtained from a wide audience. The preliminary edition of the first volume will be available in Fall, 1977 (so we intend). The preliminary edition of the second volume will be available by Summer, 1978. All the editorial changes necessary, and the entire Handbook preparation for publication will be done during the 1978-79 academic year. More than 150 "first drafts" of articles are already written, and have been made available to ARPA-IPTO for its internal uses.

B. Production rule representation of knowledge.

1. Particular focus will be placed on the representation of expert problem solving strategies. (i.e., rule-directed behavior at the level of program control.)

Detailed Discussion:

The strategies for problem solving are as important for high performance as the domain specific rules themselves. Some of the strategies that experts use can be expressed in meta-rules, which guide the application and determine the relevance of domain rules. We have already developed the meta-rule formalism for representing knowledge about the reordering and pruning of domain rules. We are proposing to extend that to additional kinds of strategy knowledge. For example, one strategy for dealing with a complex situation is to view it initially as if it were a simple one, find the specific complicating factors, and deal with those as special cases. Here is "default information" about the uncomplicated situation that allows us to focus on the important special case rules. We propose to use knowledge about commonly expected associations (not expressed in meta-rules but in general "models" of the domain) to guide the application of rules. This work will be carried out in the context of the MYCIN and MOLGEN programs.

Research Plan:

a. Work with MYCIN's domain experts to formulate at least three models of associated knowledge and expected ("default") values for relevant parameters.

b. Develop a formalism for encoding this kind of knowledge (6/78).

c. Isolate the critical times in MYCIN's performance, specifically in the control structure, when strong expectations from models will be most beneficial for guiding the consultation. Modify the control structure to refer to the stored information in models.

d. Test the cost effectiveness of using models of typical situations in the domain in the context of the three (or more) sample in-depth models (6/79).

2. Incorporate time-dependent relations into the production rule formalism and control structure, i.e., use time as an integral part of the analysis, where appropriate. Correspondingly, in the dynamic model of the situation (current best hypothesis) built up by a reasoning program, keep a "time line" of relations between events. HASP was the ground-breaking program for this work, and the new work will be oriented toward signal data interpretation and also toward MYCIN.

Detailed Discussion:

One area of improvement for production systems is to uniformly handle rules which interpret a sequence of events over time, as opposed to making conclusions based on a single "snapshot" of the decision making situation. This area of interest will allow the expert to capture the more dynamic situations in his problem domain. The main extension to knowledge-based systems is to allow the expert to specify trends to be searched for in data which has occurred up to the present time, to provide a prediction of future trends, and to specify specific milestones which are anticipated. As the processing continues forward in time, the program can compare these anticipated events against the actual data to determine if the previous decision making process was effective. The representation of trends can also be used to "work backwards" to estimate what certain data values might have been at a previous time, but were not then reported.

The major problems include picking both the most effective representation for this type of information and the methods for recognizing and describing trends, and in incorporating the uncertainties in the data or method of data collection into the prediction process. Much of the richness of time oriented analysis is based on mathematical models (e.g. differential equations, simulations, and statistics), which must be integrated with the heuristic information used by the expert to make decisions. The problems of predicting past or future events based on data not known with certainty, or where the current situation must be used to imply how much reliability can be placed on a prediction using the available data must be handled. In a consultation system, it is important to know which type of facts are expected to change, in order to inquire of the user only the minimum facts needed to update the current model of the problem situation.

One area of improvement for production systems is to uniformly handle rules which interpret a sequence of events over time, as opposed to making conclusions based on a single "snapshot" of the decision making situation. This area of interest will allow the expert to capture the more dynamic situations in his problem domain. The main extension to knowledge-based systems is to allow the expert to specify trends to be searched for in data which has occurred up to the present time, to provide a prediction of future trends, and to specify specific milestones which are anticipated. As the processing continues forward in time, the program can compare these anticipated events against the actual data to determine if the previous decision making process was effective. The representation of trends can also be used to "work backwards" to estimate what certain data values might have been at a previous time, but were not then reported.

The major problems include picking both the most effective representation for this type of information and the methods for recognizing and describing trends, and in incorporating the uncertainties in the data or method of data collection into the prediction process. Much of the richness of time oriented analysis is based on mathematical models (e.g., differential equations, simulations, and statistics), which must be integrated with the heuristic information used by the expert to make decisions. The problems of predicting past or future events based on data not known with certainty, or where the current situation must be used to imply how much reliability can be placed on a prediction using the available data must be handled. In a consultation system, it is important to know which type of facts are expected to change, in order to inquire of the user only the minimum facts needed to update the current model of the problem situation.

Research Plan:

Design internal representation (3 months)

Integrate mathematical models (4 months)

Forward and backward prediction under heuristic control (5 months)

Handle uncertainty in the prediction process (6 months)

3. Extend the capabilities of the production rule formalism to include general knowledge of the task domain not easily represented in production rules, such as mathematical relations. Extend the control structure to use this general knowledge in order to direct the reasoning and focus the system's attention on the most relevant subset of rules. Toward similar ends, we will develop further our previous work on the use of meta-rules as guidance for the use of domain rules in a program.

Detailed Discussion:

One very powerful form of reasoning used by a human consultant when he is presented with a new situation is that of classifying and comparing the new situation with "typical" situations he has encountered previously. For example, an auto mechanic who is asked to fix a car which won't start, immediately begins to check for those things which are typically malfunctioning in a "car won't start" situation. That is, he will check the starter, the battery, the supply of gas, etc., and not bother to check the oil level, because he knows that the oil level is not typically at fault in a "car won't start" situation. Sometimes the consultant is unable to classify the new situation as one he has encountered previously, and he must rely upon his basic knowledge of the task domain in order to complete the

consultation. When he is able to classify the new situation as being similar to a "typical" situation, he has, in effect, a model of what to expect in the current consultation. He uses this model of a similar consultation to guide him in performing the current consultation. The model also gives him expectations about what will happen in this new situation because it tells him what typically happens in this situation. And because he can compare his findings with what is "typically" found, he has a way to check the accuracy of the information he obtains in the new situation. We propose to add such a set of models to a rule-based consultation system in order to control the invocation of the rules and to give the system knowledge about what to expect in typical situations. These models will also give us the ability to take appropriate action in response to inconsistent information. That is, given knowledge about what constitutes a "typical" case, atypical findings can be isolated and checked for their validity.

Research Plan:

- a. Plan design of models and model-based system (9/77)
 - b. Add models to data base and implement model-directed consultation system (3/78)
 - c. Provide consistency checking of information obtained during the consultation
 - d. Provide explanation and interactive knowledge-acquisition facilities for the model-based system (12/78)
4. Develop a formalism for measuring the power and performance of a program's knowledge base. Create methods for assessing strengths and weaknesses in the knowledge base and for recommending modifications.

Detailed Discussion:

The objective of this research is to develop principles and methods for measuring the performance of certain kinds of AI real world problem solving systems, and to investigate how to use such measurements to improve system performance. This work is motivated by the belief that for an AI system to solve real world problems effectively, it is just as important for it to know how its knowledge is used as it is to know what knowledge it has in the first place. The work is also motivated by the conviction that performance knowledge is essential for conveying a program's scope and limitations to a user, who can then use the program more intelligently. Furthermore, it is hoped that the principles developed here will facilitate the creation of a general laboratory for testing AI methods, whose use will increase our understanding of the field.

This investigation will primarily be concerned with how well the knowledge base of an AI problem solving system is organized and used. The emphasis will be on such factors as the frequency of accessing and grouping information about domain objects; the cost of evaluating and invoking rules characterizing the domain; and the relative merits of strategies for selecting knowledge sources. There are of course other important determinants of system performance, such as the selection of the best internal representation for a data structure, or the selection of the most efficient method of searching a data file. However, the main concern here is to assist expert users in organizing domain knowledge for effective problem solving in conjunction with the AI system.

We will seek to develop interactive methods for a user or designer of a knowledge based system to express system performance criteria; to describe which parts of the system should be monitored, in what detail, and what special events to look for; and to enter heuristics to help the system decide what changes to recommend on the basis of its observations. The system should provide a description of distribution of effort and an estimate of the costs and benefits of further observation of parts of the system.

Research Plan:

a. develop layered information gathering tools - tools which offer a choice of more or less detail for more or less cost - and establish how to use them effectively.

b. establish ways to characterize which parts of a system to observe and what events to detect.

c. develop a means for stating performance criteria (6/78).

d. provide examples of using a description of behavior and a statement of performance criteria to recommend knowledge base changes.

e. investigate ways of estimating the utility of monitoring the system (6/79).

C. Program designs for expert problem solving, with particular focus on planning methods

1. Develop programs and methods for hierarchical planning in task domains in which knowledge is uncertain. Though neither hierarchical planning methods nor methods of inexact inference are entirely new, their confluence is now an important issue.

Detailed Discussion:

Significant progress has been made in recent years on methods for both hierarchical planning, producing successively more detailed plans until all factors have been considered, and inexact inference, reasoning with rules whose conclusions can only be "trusted" with a measured degree of certainty. Combining these two powerful problem-solving techniques is now an important research issue for complex, real-world domains. Hierarchical planning in these domains is necessary because of the combinatorial explosion that results when all details are considered at all stages of plan formation. Yet, the rules of these domains make use of uncertain knowledge to make uncertain conclusions, conclusions which may change as further details become available at more refined stages of planning.

The major research question is how to properly modify the basic structure of hierarchical planning to handle the reality of inexact domain rules. Certainly Sacerdoti's idea of postponing decisions involving the ordering of steps in a plan until such a decision is necessary seems a reasonable beginning. The system must be able to handle cases where plans fail at low levels of refinement not only because detailed preconditions of rules cannot be satisfied, but also because the very actions of the rules themselves change when all details are considered. This occurs because the abstractions of rules which are valid at the highest levels of hierarchical planning cannot be expected to provide more than a reasonable guess about the likely action of a rule for which a detailed simulation may be necessary. In simple domains the virtues of hierarchical planning were seen by merely considering successively more rule preconditions at each level of planning, the action of the rule stayed constant. The problem for domains with inexact knowledge is not merely one of considering at each level of planning only those details which are necessary, it is one of considering those details which are reasonable, a task which itself may be rule based.

Research Plan:

a. A goal of the first six months should be to categorize the variety of problems which do result when planning in complex domains where knowledge is tentative and uncertain.

b. Develop rule-based systems which provide heuristic control at the problem points in hierarchical planning--expand the notion of domain dependent critics as introduced by Sacerdoti. This process should take between 1 and 1 1/2 years.

c. Test these systems in a variety of problem domains. This will involve methods for expert entry and explanation of the rules described in (b). The process of testing and expanding will start after 1 year and continue through the second year (6/79).

D. Heuristic Rule Acquisition, by automatic rule formation methods and semi-automatic model-directed rule extraction

1. Extend the interactive production rule acquisition system to handle acquisition of other types of information about the task domain, i.e., domain-specific models other than rule-based models. Use the domain-specific models to check for inconsistencies in new rules, i.e., exploit the inevitable redundancy to enforce logical consistency among rules.

Detailed Discussion:

The information about "typical" situations, discussed in section B above, will be very helpful for reasoning and problem solving. It is necessary to be able to acquire this information from experts first, however, before it can be used. This will require giving the production rule system the same acquisition and explanation capabilities for these "models" as it has for its rules.

The current knowledge-acquisition facility of the consultation system will be extended so that these models can be acquired through interaction with the user. The explanation capabilities of the system will also be extended and improved so that "higher-level" explanations of system performance will be provided. For example, we could explain which hypothesis, i.e., which model, is being tried during a given consultation, and we could say why it is being tried.

Research Plan:

a. Define a vocabulary to be used in the acquisition of models.

b. Implement the basic Knowledge-Acquisition program based on this vocabulary.

c. Provide good human engineering features for Knowledge-Acquisition. For example, simple spelling errors should be tolerated and careful prompts should be used when communicating with the user.

d. Extend the explanation facility so that "higher-level", contextual explanations can be handled.

2. Develop programs for acquiring one particular type of non-rule-based auxiliary model of a domain: an expert's taxonomy of the objects in his domain. This taxonomy is a necessary precondition for applying rules that mention the objects.

Detailed Discussion:

In many domains a strict hierarchical relationship exists among an important subset of the objects in that domain. Experts in these domains often form that subset of objects into a taxonomy with important consequences for rules affecting or using those objects. The major idea is that objects can be understood and accessed through this hierarchical relationship; for example, if no rules can be found which specifically mention object A, then perhaps relevant rules can be found which do mention its parent in the taxonomy, object B. Also, a taxonomical classification makes acquisition and explanation affecting objects simpler and more efficient; misunderstood objects can be clarified by showing their relationship to other, better understood ones.

An automated system for extracting from an expert his taxonomy of a complex domain should provide interactive facilities to allow the expert to make completely clear the relationships between objects in the taxonomy. It is important for a rule-based system using the taxonomy to understand if these relationships are based on some underlying theory of the domain, or if they simply reflect a particular expert's view of the pathways through which objects should be accessed. The system should allow diverse types of - relationships within the taxonomy and should make expansion and modification a relatively simple process for even a large taxonomy.

Research Plan:

a. Initially develop an interactive framework for describing and modifying an expert's taxonomy of objects in his domain. This should be accomplished within six months.

b. Combine this taxonomy acquisition system into a variety of knowledge-based systems to test its flexibility and utility. This will be an ongoing process over a several year period.

c. Develop methods for describing the domain theory that underlies the taxonomic classification along with means for testing the supplied taxonomy to check if its organization really does represent that theory. This should be accomplished in some generality within two years.

3. Develop a general representation for checking the form of new knowledge entered about objects and relations in the domain. These so-called "schema" can also guide the acquisition of knowledge.

Detailed Discussion:

One of the most successful systems to address the knowledge acquisition issue is the MYCIN/TEIRESIAS system developed at Stanford. MYCIN/TEIRESIAS developed techniques for the

acquisition of domain knowledge in the form of rules and objects. MYCIN/TEIRESIAS has approximately four hundred production ("If ... then ...") rules in its knowledge base. Problem solving context and expectation models are used to guide the process of acquiring these rules from a user. MYCIN/TEIRESIAS also has several hundred objects in its vocabulary (eg. gramstain, nutrient, and culture). To guide the acquisition of these objects which have varied structure, a schema, that is a description of structure, is kept for each type of object. When a new instance of an object is acquired, MYCIN/TEIRESIAS decomposes the acquisition process into a number of small steps corresponding to the components of the object. A schema for schemata (the "Schema-schema") gives a description of schemata in the knowledge base. This makes it possible to acquire a new kind of object by first acquiring a new schema for it using the schema-schema, and then acquiring an instance using the new schema.

Several extensions to the ideas in MYCIN/TEIRESIAS are needed to build a powerful problem solving system. The part of the medical diagnostic task represented in this system is described in terms of rules of a very limited format. Some common kinds of operations and strategies are awkward to express within this constrained format. For example, iteration can be expressed only by introducing some extra dummy variables. Following the analogy of using schemata to describe different types of objects, it is proposed that rule schemata be developed to describe different types of rules. (MYCIN/TEIRESIAS may be viewed as having only one type of rule which has an "if" component and a "then" component with associated experts for guidance in filling the two components. This "schema" for rules in MYCIN/TEIRESIAS is built into the system.) We propose to develop a variety of schemata to cover a broad range of domain operations and strategies. Just as an object can be decomposed into its component objects, a rule could be decomposed into its component subactions. In the proposed system, a user could select from a menu of standard artificial intelligence strategies and the system would guide the acquisition of the strategy rule using appropriate schemata from the knowledge base.

In summary, the acquisition process for a strategy rule is broken down into a number of small and manageable steps. The schema used to guide the acquisition process inherits many of its specifications for creating the rule from its ancestors in the schemata hierarchy. It is suggested that this process can be used to help prevent required entries from being forgotten when a new rule is acquired. Much of the structure of a rule can be filled in automatically - for example, the iterative loop in the Means-ends analysis example. Tests on the sets of acceptable values for the components of instances can be built into the schemata as a further check on the correctness of what a user enters. The goal of this process of assisted acquisition is to

make the acquisition of domain specific strategy rules as painless and bug-free as possible.

Research Plan:

a. Develop the vocabulary of schemas that extends the MYCIN/TEIRESIAS capabilities to iterative statements.

b. Design a hierarchy of schemas and programs for "inheriting" or transferring knowledge from one level of the hierarchy to others.

c. Design the "mean" of AI strategies and instantiate all schemas below at least one of them.

4. Additional experiments will test the relative power of rule formation under different learning conditions.

Detailed Discussion:

Given a body of data from which rules are to be formed, together with a basic approach to rule induction, there remains a range of ways in which the data may be utilized, which differ in the degree of parallelism involved in the examination of instances. At one extreme are methods in which rules are formed and refined in a sequence of steps, each step involving the examination of one new instance. At the other extreme are methods which involve a single-pass rule formation process, using all available data. There are, of course, many intermediate possibilities. We propose to investigate, within the Meta-DENDRAL framework, whether some of these methods are optimal in the sense of yielding rules of comparatively high quality with the expenditure of comparatively little computing effort. It is hoped that the investigation will lead us to some general insights concerning the optimal utilization of data in automatic rule formation.

Research Plan:

a. Develop and implement one or more procedures for updating an evolving set of rules on the basis of newly examined data. These procedures will make use of existing capabilities of the RULEGEN and RULEMOD programs, and will make possible the implementation of a variety of schemes for data utilization, as described above.

b. Select and implement a representative subset of the class of data utilization schemes indicated above, and test their performance in the application area of mass spectrometry.

c. Describe in a technical report these experiments, their results, and the lessons learned.

E. Explanation by program of its line-of-reasoning

Current capabilities are exemplified by the interactive explanation capability (in English) of the MYCIN system, and non-interactive capability in HASP.

1. Continue program development to extend explanation capabilities to complex procedures in addition to production rules. Also extend facility to allow explanation programs to be provided with additional knowledge about the purposes of requests, thereby allowing them to produce the "most relevant" explanations of lines of reasoning.

Detailed Discussion:

Our previous work demonstrated the explanation capability of rule-based expert systems [Comp. Biomed. Res. ppr, and AJCL microfiche]. We propose to extend the advantages of an explanation system to programs whose knowledge is procedurally embedded. The key concepts we will explore are: 1) an "event history" which records decisions made by the program in the form of traces, 2) an "event structure" which abstractly represents the steps and strategies of a procedure, and 3) an interactive question-answerer which can selectively retrieve traces from the event history, guided by the organization of the event structure. For the sake of illustration we will use an example from antibiotic therapy selection in the MYCIN program. One relevant question at the end of a consultation with MYCIN is "Why did you recommend streptomycin for the streptococcus infection?". In order to answer this question the program needs to keep a record of the major conceptual reasoning steps taken by the various procedures in the therapy selection algorithm at the end of a MYCIN consultation. It must also be able to structure these "events" so that only the relevant steps will be retrieved in answer to the user's question.

Research Plan:

a. Develop a model for types of problem-solving methods which might be explained by this scheme. Is there a broad class of programs to which it might be applicable? (8/77)

b. Develop a general schema for the event structure which can be used to abstractly represent the class of programs determined in 1. (10/77)

2. Extend the explanation capabilities of production systems to allow the explanation of strategy decisions embodied in meta-rules.

Detailed Discussion:

Strategy decisions may sometimes be more important than decisions about more specific steps in the reasoning. We propose to extend the capabilities of the MYCIN explanation system to phrase answers to questions in terms of the general strategies as well as in terms of individual rules.

For example, the answer to the question "Why did you ask me about X?" may sometimes refer to the strategy decision in which X was relevant, e.g., "Because one of the strategies for finding out about Q is only relevant when you know X." That is, the answer may be given in terms of the strategy rules as well as the individual domain rules. Again, the question "Why didn't you use rule R?" may refer to the general strategy under which the program was operating at the time -- "Because my strategy rule S said that R would not be useful in this context." Finally, general questions about the domain, and not about the specific consultation, may refer to strategies. E.g., "In general how do you conclude the identity of infecting organisms?" This may be answered by mentioning the individual domain rules, as MYCIN does now, or by mentioning some of the general strategies for reasoning about this parameter.

Research Plan:

- a. Extend the history list mechanism of the MYCIN program to accept information about the strategies that MYCIN is carrying out during a consultation.
- b. Upgrade the question-answering capabilities of the MYCIN program to answer general questions about the program's reasoning in terms of strategy rules. An important problem to be solved is recognizing the need to frame an answer in terms of strategies rather than individual rules.

II. Research in Support of Other IPTO R&D Programs

A. Rule-based methods in the interpretation of image data

Many of the rule-based methods previously developed by our project, and to be studied and developed in this period, can be applied to assist the interpretation of image data, in at least the following ways:

... modularity/flexibility. Qualitative rules of interpretation can be changed easily when existing rules are deemed inadequate by the image analysis expert.

... acquisition. New rules can be inserted into the program under guidance of a rule acquisition system (similar to MYCIN's) to extend performance to new or changed circumstances.

... explanation. The line-of-reasoning being used by the program to interpret the image can be exhibited for the human analyst to understand and evaluate.

This important application of our work is not now being made, and we propose to orient a portion of our effort in this direction. A.I. researchers in IPTO's Image Understanding program are doing significant work primarily in "front-end" image processing and procedure-based interpretation; but not in rule-based methods such as exist in MYCIN, DENDRAL, and HASP.

Because the start-up costs in money and scientific energy to understand the real problems in image analysis and build base-level software are so great, we propose to develop our prototype system in a domain in which we have considerable expertise, excellent data, extensive software, and (last but not least) some other financial support from NSF. This domain is protein crystallography, involving the interpretation of three-dimensional images of the electron density of a protein molecule in a crystal.

The rule-based system to be developed in this prototype will be engineered to be as portable as possible to facilitate its application to the photo-interpretation prototypes being built by the Image Understanding projects, e.g., those at the Stanford AI Laboratory, SRI, and Carnegie Mellon University.

This proposed activity has additional plausibility because of the following technical circumstances:

... the HASP rule-based program structure developed by some of our staff is applicable to this problem.

... as part of our current activity, to test the generality of that HASP program structure, we have already mapped the basic ideas in the HASP program onto the crystallographic data interpretation problem. Hence we have a running start on the problem.

Detailed Discussion:

For the reasons discussed above, we propose to develop rule-based methods for interpreting 3-D images, within the context of protein crystallographic data. Our ultimate goal is to transfer the rule-based methodology to other IPTO R&D programs in image interpretation. In this respect, preliminary discussions have been held with Prof. Raj Reddy of CMU. Prof. Reddy is interested in our work and eager to collaborate. The technology

transfer is expected to be bi-directional in this effort: the Image Understanding Projects stand to benefit by employing a rule-based structure in their programs, and we expect to benefit from their expertise in front-end processing of the primary image data.

The interpretation of an electron density image, derived from the reduction of X-ray crystallographic data, is a necessary step in understanding the 3-D structure of proteins and other macromolecules. When crystallographers use the term "electron density map" they usually have in mind some pictorial representation of the electron density defined over a certain region of 3-space (usually some fraction of the unit cell of the crystal). By carefully studying the map the experienced interpreter can find features which allow him to infer important features of the image: approximate atomic locations, molecular boundaries, groups of atoms, the backbone of the polymer, etc. An "interpretation" is a model of the molecular structure which conforms to the electron density map and is also consistent with his knowledge of protein chemistry, stereochemical constraints and other available chemical and physical data (e.g., the amino acid sequence).

Our goal is to build an intelligent assistant to the image analyst, i.e., a computational system that can generate and verify its own structural hypotheses, and explain the steps taken in the interpretive process. This capability implies (1) a representation of the 3-D image data suitable to machine interpretation, (2) a substantial knowledge base appropriate to the task domain, (3) a wide assortment of model building algorithms and heuristics, in order to achieve acceptable performance, and (4) a flexible, rule-based system for incorporating both the knowledge sources and the control strategy. An elaboration of these points has been recently documented (Engelmore and Nii, HPP-77-2). The fourth point, concerning knowledge representation and utilization, is of primary concern here, because of its applicability in other image understanding contexts.

The expert analysts who interpret these images move continually across a large field of basic facts, special features of the data and implications of the partial model already built, looking for any and all opportunities to add another piece to their structure. There are several desiderata to working in this "opportunistic" mode of hypothesis formation: (1) the inference making rules and the strategies for their deployment should be separated from one another, (2) the rules should be separated from the mechanics of the program in which they are embedded, and (3) the representation of the hypothesis space should be compatible with the various kinds of hypothesis generating rules available. (The hypothesis structure represents an a priori established plan for problem solving.) The modularity of such a

system allows users to add or change rules for manipulating the data base, as well as to investigate different solution strategies, without having to make major modifications to the system. These issues, which have general applicability beyond the specific task domain of protein crystallography, are discussed further in Appendix A.

The formal and informal procedures which comprise our knowledge sources are expressed as rules, as in the DENDRAL, HASP and MYCIN systems. These rules are collected into sets of rules, each set being appropriate to use on a particular class of events. The events generally reflect the level on which the inference is being made, which in turn reflects the level of the detail of the model. The correspondence between event classes and rule sets is established by another set of rules, the event rules. The event rules thus form a second layer of rules which direct the system's choice of knowledge sources for a given event, reflecting the system's knowledge of what it knows. (A similar set of rules, the job rules, perform the same role when the system operates in goal-driven mode.) Maintaining the rule-based structure affords a flexibility in choosing different combinations of knowledge sources to work together, without having to make any changes in the knowledge sources themselves. Thus, yet a higher level knowledge source, the strategy rules, can manipulate the events in order to choose the appropriate combination of KSs suited to a particular stage or state in the solution hypothesis. Examples of strategy rules are the merging of two or more events to produce a new event that can lead to further progress, or shifting from event-driven to goal-driven mode.

We thus have a completely rule-based control structure, employing three distinct levels of rules (or knowledge): the specialist, commonly called the knowledge sources, the event processing rules (or job processing rules), representing knowledge about the capabilities of the specialist, and the strategy rules which know when to use all available knowledge to solve the problem. Although this pyramidal structure of rules and meta-rules could continue indefinitely, the flexibility of knowledge deployment offered by our three-tiered system would appear to be sufficient for this problem solving system. Similar ideas in a simpler context have been explored by Davis (HPP-76-7) for the MYCIN system.

Research Plan:

We propose the following tasks and dates of completion:

1. Continue the implementation of the rule-based image interpretation system in the crystallographic context. Sub-tasks include:

a. Implementation of additional knowledge sources for matching templates of predicted sub-structures with primary and/or processed image data. (3/78)

b. Partitioning of the KSs into functionally related sets of rules. (3/78)

c. Development and implementation of new strategy rules for directing the utilization of the KSs. (6/78)

2. Implement an explanation subsystem, using the history list feature presently incorporated, which interfaces conveniently with the user. (6/78)

3. Develop contacts with at least one IPTO Image Understanding group, in order to understand their particular problems and associated technical issues. (9/77)

4. Select one of the Image Understanding groups for collaboration, and begin working with it, in order to orient our prototype development towards its particular needs. (6/78)

5. Begin the transfer of ideas and software from the prototype system to the collaborating Image Understanding project. (6/79)

B. Distributed Sensor/Computer Networks

We propose a two-year effort of relatively small size to study issues in signal interpretation, artificial intelligence methods, and computer organization raised by the IPTO initiative in Distributed Sensor/Computer Networks.

The general problem to be studied is an inference problem of interpreting and integrating the sensory information being received by an array of sensors/computers geographically distributed, for which there is no designated central processing agent. Programs resident at the local sensor/ computer would compute a "local" understanding, communicating it to neighbors as appropriate and calling upon neighboring computational assistance as needed. The local sensor/computer stations may be integrating more than one type of instrumental data.

Our experience with this type of problem arises partly from our work on the geographically distributed, multi-sensor (but central processing) HASP task; and partly from our planning for multi-instrument data interpretation in analytic chemistry (arising from the work on DENDRAL, and currently under funding consideration by NIH).

The problem described above is inextricably tied up with two other problems, only one of which is being studied elsewhere

in the program. The first is one of conceiving of how the methods of artificial intelligence research that are currently well understood can be mapped into a highly-parallel asynchronous computing environment. Attempts will be made to study this problem for (at least) heuristic search methods, rule-based inference methods and graph matching. The second is one of conceiving of a flexible enough architecture of small computers and communications to support the requirements of the AI methods, especially in their application to the interpretation of geographically distributed signals. We have begun initial and relatively abstract studies of both of these problems.

We define "distributed processing" to mean processing that involves one or both of the following attributes: physical decomposition of the processor into relatively independent processor nodes, and functional decomposition of the top level task into relatively independent subtasks. There are several applications, particularly distributed sensor networks and distributed command and control systems, in which the confluence of distributed processing and AI techniques appears attractive. We propose to study and develop techniques for symbolic inference in a distributed processing environment; and to discover the constraints (if any) placed on a distributed processor architecture by these applications. These techniques are based on an emerging formalism we have been developing, which we call the CONTRACT NET. In a contract net, individual tasks are dealt with as contracts that exist between pairs of processor nodes that take on the roles of contract manager and contractor respectively. In any such system with distributed executive control, there must be a method for choosing the small number of tasks that can be processed at a given time instant from the large number generally available. To this end, we wish to explore the concept of a "priority description" which is intended to offer a more detailed specification of task characteristics and importance than the usual integer ratings.

Our approach to the architectural problem is toward a much more loose coupling between the computers of the net (i.e., much more local processing, much less communication) than is implicit in the design of the c.m star machine architecture being studied at Carnegie-Mellon. A possible and important by-product of this study could be a computational route to extreme speed-ups in symbolic computation arising from the collective processing activity of large numbers of inexpensive microcomputers. The slow speed of symbolic computation, in contrast to numeric calculation, has been a major hurdle in the development of AI science and AI applications.

Joining us in these studies have been a member of the staff of the Canadian Defense Research Establishment Atlantic (a Stanford student) and staff member of the Norwegian Defense Research Establishment (a Stanford post-doctoral Fellow). We

have had some indications of possible industry support for the research on distributed architectures. We propose three tasks:

1. Interact with the other investigators in the IPTO Distributed Sensor Network project to understand and help to further refine the scientific and practical problems therein; and produce a report (or series of reports) on the methods available from artificial intelligence research that might be applicable to the development of a total system. Our methodology will be computer simulation (not hardware building). The sense data interpretation problem that will give specificity to our study will be an image data interpretation problem, probably the same image data interpretation problem discussed in (a.) above (the crystallographic image data interpretation problem) for reasons of economy of effort and funds.

2. Prepare a report on the realization of at least two of the well-understood AI methods in a distributed computing environment. Again, the methodology will be computer simulation but the tasks chosen for study will range more broadly than just signal data interpretation.

3. Prepare report(s) on studies of various highly parallel asynchronous architectures suited to symbolic computation. The methodology will involve both abstract analytic studies and computer simulation. We do not at this time intend to construct any network until we understand the issues from a theoretical point of view.

Research Plan:

1. Complete design of CONTRACT NET formalism for expressing control of problem solving in a distributed network, and incorporate the completed design into a simulation program.

2. Report on the variations in control that can be achieved through the use of priority descriptions.

3. Expand the simulator to allow exploration of the suitability of different processor node interconnection alternatives for a distributed processor (6/78).

4. Test the CONTRACT NET formalism on a variety of simple, yet representative problems, using the simulator, and report results.

5. Report on the problems raised by the requirement for global information in a distributed network in which there is limited memory at each processor node (6/79).

Appendix A. Extended Discussion of Proposed Rule-based System for Image Understanding

In this appendix we amplify some issues on knowledge representation and utilization in the context of the proposed 3-D image understanding research.

Representation of Knowledge in the Image Interpretation System

The problem of representing a large and diverse body of knowledge, in a form which will allow it to be used cooperatively and efficiently in the search for plausible hypotheses, is of central concern to this research. The system currently under development draws upon many concepts which have emerged in the design of other large knowledge-based systems, e.g., the use of production rules (as in DENDRAL, HASP, etc.) and a multi-level space of hypotheses (as in Hearsay-II).

Knowledge consists of facts, algorithms and heuristics (rules of good guessing). Facts required for protein structure inference are general physical, chemical, stereochemical and crystallographic constraints. Typical factual knowledge stored in the system includes molecular structure and chemical properties of the twenty amino acid building blocks. These facts are encoded as tables or in property lists attached to specific structural entities.

Algorithms and heuristics comprise the formal and informal knowledge which generate and/or verify hypothesis elements. We have been guided by two general principles in the representation of the knowledge sources:

- 1) decompose identifiable areas of knowledge into elementary units, each of which increments the hypothesis when specified preconditions are met.
- 2) represent the elementary units as situation-action rules.

To illustrate, consider the relatively simple example of heavy atom location. This subproblem is decomposed into two independent parts: 1) inferring the presence of heavy atoms and 2) determining their spatial locations. These two independent parts are represented as two separate KSs, invoked under different conditions. A rule contained in the first KS is:

IF the composition list contains a cofactor of type heme,
THEN:

- 1) create a superatom node of type heme in the model plane,
- 2) create an atom node of type iron in the model plane,
- 3) create membership links between the iron and the heme,
- 4) put "cofactor_posited" on the event-list,
- 5) put "heavyatom_posited" on the event-list.

Note that several actions may be performed for a given situation. An action may itself be a situation-action rule, and may be iterative. Also note that at least one of the actions of each rule is to place a token on an event-list. The event-list is used by the interpreter, discussed in the next section, to determine what to do next, i.e., which set of knowledge sources will be invoked after the current event has been processed.

Control Structure for the Image Interpretation System

There are several choices of control structure faced by the designer of a knowledge-based system. Basically the choices are among points on a spectrum, at the extremes of which are goal-driven and event-driven systems. In a goal-driven system (of which MYCIN is a well-known example (Shortliffe, HPP-74-2)) the rule interpreter selects a rule which concludes with the goal being sought. In our system, we might imagine having such a goal rule as follows:

```
IF
  1) the topological description is complete, and
  2) the coordinates of all atoms in the structure are assigned,
    and
  3) the structure satisfies stereochemical constraints, and
  4) the structure is consistent with the electron density map,
    and
  5) the structure is consistent with auxiliary chemical data,
THEN:
  signify that a model has been completed.
```

The interpreter would then attempt to verify each of the premises in the goal rule. To do that, other rules would be selected whose conclusions (the right-hand sides) verified the premises under consideration and the interpreter would attempt to verify the premises of these rules, and so on, working through the list of rules in this recursive fashion. The program's focus of attention is determined by the current rule whose premises are being evaluated. Many levels of recursion may occur before a rule is reached which is relevant to the current state of the system. A goal-driven monitor is attractive, in that it pursues a logical chain of reasoning, in which the purpose of each move is clearly revealed by the tree of subgoals.

A vital requirement for the success of a goal-driven control structure is the relative independence of the subgoals. Although some aspects of the protein image interpretation problem can be solved in this fashion, the independence requirement is not generally satisfied. For example, the location of each atom in the molecule must obey stereochemical constraints with respect to neighboring atoms, so that one cannot simply determine each atom's position independently.

An alternate way to focus attention is to employ an event-driven control structure. In this scheme the current state of the hypothesis space determines what to do next. The monitor continually refers to a list of current events - the event-list mentioned in the rules discussed above - which is used to trigger those knowledge sources most likely to make further headway. As a knowledge source makes a change in the current hypothesis, it also places a symbol on the event-list to signify the type of change made. Thus as events are drawn from the event-list for processing, new events are added, so that under normal conditions the monitor always has a means for choosing its next move. An explanation capability is provided by keeping a list of events processed, their predecessor and successor events, and the KSS invoked.

The system we are currently developing operates in both goal-driven and event-driven modes (with an emphasis on the latter), and is closely related to the HASP system design. The normal iterative cycle of problem solving uses the event-list to trigger knowledge sources, which create or change hypothesis elements and place new events on the event-list. Thus the system's behavior is "opportunistic" in that it is guided primarily by what was most recently discovered, rather than by a necessity to satisfy subgoals. The choice of an event-driven control structure as the primary mode of operation is based partly on subgoal interdependence, partly on efficiency in selecting appropriate knowledge sources and partly on conformity with the structure modeling process normally employed by protein crystallographers.

BIBLIOGRAPHY

Heuristic Programming Project Memos Since 1964

HPP-64-1

J. Lederberg,

"DENDRAL-64-A System for Computer Construction, Enumeration and Notation of Organic Molecules as Three Structures and Cyclic Graphs", (technical reports to NASA, also available from the author and summarized in (12)).

(1a) Part I. Notational algorithm for tree structures (1964) CR.57029

(1b) Part II. Topology of cyclic graphs (1965) CR.68898

(1c) Part III. Complete chemical graphs; embedding rings in trees (1969)

HPP-64-2

J. Lederberg,

"Computation of Molecular Formulas for Mass Spectrometry", Holden-Day, Inc. (1964).

HPP-65-1

J. Lederberg,

"Topological Mapping of Organic Molecules", Proc. Nat. Acad. Sci., 53:1, January 1965, pp. 134-139.

HPP-65-2

J. Lederberg,

"Systematics of Organic Molecules, Graph Topology and Hamilton Circuits. A General outline of the DENDRAL system." NASA CR-48899 (1965)

HPP-65-3

AIM-30, AD785056

Edward A. Feigenbaum, Richard W. Watson,
"An Initial Problem Statement for a Machine Induction Research Project", (April 1965).

HPP-66-1

AIM-38, AD785066

Donald A. Waterman,

"A Filter for a Machine Induction System",
(January 1966).

HPP-66-2

AIM-46, CS-50, PB176761

Staffan Persson,

"Some Sequence Extrapolating Programs: a Study
of Representation and Modeling in Inquiring
Systems",

Ph.D. Thesis in Computer Science, (U.C. Berkely September 1966).

HPP-66-3

AIM-47

Bruce Buchanan,

"Logics of Scientific Discovery",

Ph.D. Thesis in Philosophy

(Michigan State University December 1966).

HPP-67-1

AIM-49

Georgia Sutherland,

"DENDRAL -- a Computer Program for Generating and Filtering Chemical
Structures", (February 1967).

HPP-67-2

J. Lederberg,

"Hamilton Circuits of Convex Trivalent Polyhedra (up to 18 vertices),
Am. Math. Monthly, (May 1967).

HPP-67-3

AIM-54

Joshua Lederberg, Edward A. Feigenbaum,

"Mechanization of Inductive Inference in Organic Chemistry",

(August 1967). Also in Formal Representations For Human Judgment,
B. Klainmuntz, Editor, Wiley, 1968.

HPP-68-1

J. Lederberg,

"Online Computation of Molecular Formulas from Mass Number".

NASA CR-94977 (1968)

HPP-68-2

AIM-62

Bruce G. Buchanan, Georgia Sutherland, E. A. Feigenbaum

"Heuristic DENDRAL: a Program for Generating Explanatory
Hypotheses in Organic Chemistry", (July 1968).

Also in Machine Intelligence-4, Edinburgh Univ., Press, 209-254, 1969.

HPP-68-3
AIM-67 AD680487
Edward A. Feigenbaum,
"Artificial Intelligence: Themes in the Second Decade",
Proceedings of IFIP International Congress, Edinburgh, (August 1968).

HPP-68-4
AIM-74, CS-118, AD681027
Donald Waterman,
"Machine Learning of Heuristics",
Ph.D. Thesis in Computer Science, (Stanford University December 1968).

HPP-68-5
E. A. Feigenbaum and B. G. Buchanan, "Heuristic DENDRAL: A
Program for Generating Explanatory Hypotheses in Organic Chemistry,"
in Proceedings, Hawaii International Conference on System Sciences,
B. K. Kinariwala and F. F. Kuo (eds.), University of Hawaii Press, 1968.

HPP-69-1
AIM-80, AD685612
Georgia Sutherland,
"Heuristic DENDRAL: a Family of LISP Programs", (March 1969).

HPP-69-2
AIM-99
Bruce G. Buchanan, G.L. Sutherland, E.A. Feigenbaum,
"Toward an Understanding of Information Processes of
Scientific Inference in the Context of Organic Chemistry",
(September 1969). Also in Machine Intelligence-5, Edinburgh Univ., Press,
1970.

HPP-69-3
AIM-102
Donald A. Waterman,
"Generalization Learning for Automating the Learning
of Heuristics", J. Artificial Intelligence Vol. L, No. 1/2
(July 1969).

HPP-69-4
AIM-104
Joshua Lederberg, Georgia Sutherland, Bruce G. Buchanan,
Edward A. Feigenbaum
"A Heuristic Program for Solving A Scientific Inference Problem:
Summary of Motivation and Implementation", (November 1969).
Proceedings of IV Systems Symposium, Case Western Reserve Univ.,
Springer-Verlag, 1970.

HPP-69-5
J. Lederberg,
"Topology of Molecules", in the Mathematical Sciences -
A Collection of Essays, (ed.) Committee on Support of
Research in the Mathematical Sciences (COSRIMS), National
Academy of Sciences - National Research Council, M.I.T.
Press, (1969).

HPP-69-6

J. Lederberg, G.L. Sutherland, B.G. Buchanan, E.A. Feigenbaum, A.V. Robertson, A.M. Duffield, and C. Djerassi, "Applications of Artificial Intelligence for Chemical Inference I. The Number of Possible Organic Compounds: Acyclic Structures Containing C, H, O and N". Journal of the American Chemical Society, 91 (1969).

HPP-69-7

A.M. Duffield, A.V. Robertson, C. Djerassi, B.G. Buchanan, G.L. Sutherland, E.A. Feigenbaum, and J. Lederberg, "Application of Artificial Intelligence for Chemical Inference II. Interpretation of Low Resolution Mass Spectra of Ketones". Journal of the American Chemical Society, 91:11 (May 21, 1969).

HPP-69-8

C.W. Churchman and B.G. Buchanan, "On Design of Inductive Systems: Some Philosophical Problems". British Journal for the Philosophy of Science, 20 (1969).

HPP-69-9

G. Schroll, A.M. Duffield, C. Djerassi, B.G. Buchanan, G.L. Sutherland, E.A. Feigenbaum, and J. Lederberg, "Application of Artificial Intelligence for Chemical Inference III. Aliphatic Ethers Diagnosed by Their Low Resolution Mass Spectra and NMR Data". Journal of the American Chemical Society, 91. (1969).

HPP-70-1

A. Buchs, A.B. Delfino, C. Djerassi, A.M. Duffield, B.G. Buchanan, E.A. Feigenbaum, J. Lederberg and G. Schroll, "Application of Artificial Intelligence for Chemical Inference. IV. Saturated Amines Diagnosed by Their Low Resolution Mass Spectra and Nuclear Magnetic Resonance Spectra", Journal of the American Chemical Society, 92, (1970).

HPP-70-2

AIM-123

Bruce G. Buchanan, Thomas E. Headrick, "Some Speculation About Artificial Intelligence and Legal Reasoning", (May 1970). Stanford Law Review Fall 1970.

HPP-70-3

Y.M. Sheikh, A. Buchs, A.B. Delfino, G. Schroll, A.M. Duffield, C. Djerassi, B.G. Buchanan, G.L. Sutherland, E.A. Feigenbaum and J. Lederberg, "Applications of Artificial Intelligence for Chemical Inference V. An Approach to the Computer Generation of Cyclic Structures. Differentiation Between All The Possible Isomeric Ketones of Composition C₆H₁₀O", Organic Mass Spectrometry, 4, 493, (1970).

HPP-70-4

A. Buchs, A.B. Delfino, A.M. Duffield, C. Djerassi, B.G. Buchanan, E.A. Feigenbaum and J. Lederberg, "Applications of Artificial Intelligence for Chemical Inference VI. Approach to a General Method of Interpreting Low Resolution Mass Spectra with a Computer", *Helvetica Chimica Acta*, 53, 1394 (1970).

HPP-70-5

AIM-131, CS-176, AD715128

Edward A. Feigenbaum, Bruce G. Buchanan, Joshua Lederberg, "On Generality and Problem Solving: a Case Study Using the DENDRAL Program", (August 1970). Also in *Machine Intelligence-6*, Edinburgh Univ. Press, 1971.

HPP-71-1

A. Buchs, A.B. Delfino, C. Djerassi, A.M. Duffield, B.G. Buchanan, E.A. Feigenbaum, J. Lederberg, G. Schroll, and G.L. Sutherland, "Applications of Artificial Intelligence for Chemical Inference VII. The Application of Artificial Intelligence in the Interpretation of Low-Resolution Mass Spectra", *Advances in Mass Spectrometry*, 5, 314 (1971).

HPP-71-2

AIM-141, CS-205, AD730506

Bruce G. Buchanan, Joshua Lederberg, "The Heuristic DENDRAL Program for Explaining Data", In proceedings of the IFIP Congress 71, Ljubljana, Yugoslavia (February 1971).

HPP-71-3

AIM-147, CS-216, AD732457

Robert E. Kling, "Reasoning by Analogy with Applications to Heuristic Problem Solving: a Case Study", Thesis: Ph.D. in Computer Science (August 1971).

HPP-71-4

AIM-141

B.G. Buchanan, E.A. Feigenbaum, and J. Lederberg, "A Heuristic Programming Study of Theory Formation in Science". In proceedings of the Second International Joint Conference on Artificial Intelligence, Imperial College, London (September 1971).

HPP-71-5

B.G. Buchanan, A.M. Duffield, A.V. Robertson, "An Application of Artificial Intelligence to the Interpretation of Mass Spectra", *Mass Spectrometry Techniques and Appliances*, G.W.A. Milne, Ed., John Wiley & Sons, Inc., p. 121, (1971).

HPP-72-1

D.H. Smith, B.G. Buchanan, R.S. Engelmores, A.M. Duffield, A. Yeo, E.A. Feigenbaum, J. Lederberg, and C. Djerassi,
"Applications of Artificial Intelligence for Chemical Inference. VIII. An Approach to the Computer Interpretation of the High Resolution Mass Spectra of Complex Molecules. Structure Elucidation of Estrogenic Steroids", Journal of the American Chemical Society, 94, 5962 (1972).

HPP-72-2

B.G. Buchanan, E.A. Feigenbaum, and N.S. Sridharan,
"Heuristic Theory Formation: Data Interpretation and Rule Formation". In Machine Intelligence 7, Edinburgh University Press (1972).

HPP-72-3

AIM-181, CS-325

Bruce G. Buchanan,

"Review of Hubert Dreyfus' 'What Computers Can't Do': a Critique of Artificial Reason", (December 1972).

Computing Reviews (January, 1973).

Also in Computing Reviews, January 1973.

HPP-72-4

J. Lederberg,

"Rapid Calculation of Molecular Formulas from Mass Values". Journal of Chemical Education, 49, 613, (1972).

HPP-72-5

Joshua Lederberg,

"Use of a Computer to Identify Unknown Compounds:

The Automation of Scientific Inference",

Biochemical Applications of Mass Spectrometry, (1972).

HPP-72-6

STAN-CS-318

H. Brown, L. Masinter, and L. Hjelmeland,

"Constructive Graph Labeling Using Double Cosets". Discrete Mathematics, 7, 1 (1974).

HPP-73-1

D.H. Smith, B.G. Buchanan, R.S. Engelmores, H. Aldercreutz and C. Djerassi,

"Applications of Artificial Intelligence for Chemical Inference IX. "Analysis of Mixtures Without Prior Separation as Illustrated for Estrogens". Journal of the American Chemical Society, 95, 6078 (1973).

HPP-73-2

D.H. Smith, B.G. Buchanan, W.C. White, E.A. Feigenbaum, C. Djerassi and J. Lederberg,

"Applications of Artificial Intelligence for Chemical Inference X. "INTSUM. A Data Interpretation Program as Applied to the Collected Mass Spectra of Estrogenic Steroids". Tetrahedron, 29, 3117, (1973).

HPP-73-3

AIM-205, CS-370, AD764288

N.S. Sridharan, et.al,

"A Heuristic Program to Discover Syntheses for Complex Organic Molecules", (June 1973).

HPP-73-4

STAN-CS-73-381

N.S. Sridharan,

"Computer Generation of Vertex Graphs", (July 1973).

HPP-73-5

R. Carhart and C. Djerassi,

"Applications of Artificial Intelligence for Chemical Inference XI: The Analysis of C13 NMR Data for Structure Elucidation of Acyclic Amines", Journal of the American Chemical Society (Perkin II), 1753, (1973).

HPP-73-6

AIM-215, CS-387, AD769380

B.G. Buchanan and N.S. Sridharan,

"Analysis of Behavior of Chemical Molecules:

Rule Formation on Non-Homogeneous Classes of Objects".

In proceedings of the Third International Joint

Conference on Artificial Intelligence, (August 1973).

Also in Machine Intelligence-7, Edinburgh Univ. Press, 1974.

HPP-73-7

D. Michie and B.G. Buchanan,

"Current Status of the Heuristic DENDRAL Program for Applying Artificial Intelligence to the Interpretation of Mass Spectra", in "Computers for Spectroscopy",

[Also: University of Edinburgh, School of Artificial Intelligence, Experimental Programming Report No. 32 (1973).]

HPP-73-8

AIM-217, CS-391, AD770610

N.S. Sridharan

"Search Strategies for the Task of Organic Chemical Synthesis", (August 1973).

HPP-73-9

E.H. Shortliffe, S.G. Axline, B.G. Buchanan, T.C. Merigan,

S.N. Cohen,

"An Artificial Intelligence Program to Advise Physicians Regarding Antimicrobial Therapy", Computers and Biomedical Research, 6:544-560 (1973).

HPP-73-10

AIM-216, CS-389, AD71299

Larry Masinter, N.S. Sridharan, J. Lederberg, D.H. Smith
"Applications of Artificial Intelligence for Chemical Inference
XII: Exhaustive Generation of Cyclic and Acyclic Isomers",
(September 1973). Also in J. Am. Chem. Soc. Vol. 96, No. 25,
December 11, pp. 7702-7714, (1974).

HPP-74-1

E.H. Shortliffe, S. G. Axline, B.G. Buchanan, S.N. Cohen,
"Design Considerations for a Program to Provide Consultations
in Clinical Therapeutics, Proceedings of San Diego Biomedical
Symposium 1974 (February 6-8 1974).

HPP-74-2

AIM-251, CS-465, ADA001373

Edward H. Shortliffe,

"MYCIN: A Rule-Based Computer Program for Advising Physicians
Regarding Antimicrobial Therapy Selection", Thesis: Ph.D. in
Medical Information Sciences,
Thesis: Ph.D. in Medical Information Sciences, (October 1974).

HPP-74-3

B.G. Buchanan,

"Scientific Theory Formation by Computer". In Proceedings of NATO
Advanced Study Institute on Computer Oriented Learning Processes,
Bonas, France, (1974).

HPP-74-4

E.A. Feigenbaum,

"Computer Applications: Introductory Remarks", in "Proceedings of
Federation of American Societies for Experimental Biology", 33, 2331
(1974).

HPP-74-5

L.M. Masinter, N.S. Sridharan, R.E. Carhart and D.H. Smith,
"Applications of Artificial Intelligence for Chemical Inference XIII.
Labeling Of Objects Having Symmetry^{1,2}", Journal of the American
Chemical Society, 96, 7714, (1974)/

HPP-74-6

D. Michie and B. G. Buchanan, "The Scientist's Apprentice."
In Computers for Spectroscopy (ed. R.A.G. Carrington)
London: Adam Hilger, 1974.

HPP-75-1

E.H. Shortliffe, B.G. Buchanan,

"A Model of Inexact Reasoning in Medicine", Mathematical
Biosciences, 23:351-379, (1975).

HPP-75-2

E.H. Shortliffe, R. Davis, S.G. Axline, B.G. Buchanan, C.C. Green,
S.N. Cohen,

"Computer-Based Consultations in Clinical Therapeutics;
Explanation and Rule Acquisition Capabilities of the MYCIN System",
Computers and Biomedical Research, 8:303-320, (August 1975).

HPP-75-3

E.H. Shortliffe, F.S. Rhame, S.G. Axline, S.N. Cohen, B.G. Buchanan, R. Davis, A.C. Scott, R. Chavez-Pardo, and W.J. Van Melle,
"MYCIN: A Computer Program Providing Antimicrobial Therapy Recommendations".
Clinical Medicine, (August 1975).

HPP-75-4

E.H. Shortliffe,
"Judgmental Knowledge as a Basis for Computer-Assisted Clinical
Decision Making, Proceedings of the 1975 International Conference on
Cybernetics and Society, (September 1975).

HPP-75-5

E.H. Shortliffe, S. Axline, B.G. Buchanan, R. Davis, S. Cohen,
"A Computer-Based Approach to the Promotion of Rational Clinical Use of
Antimicrobials, International Symposium on Clinical Pharmacy and Clinical
Pharmacology, (September 1975), Boston, Mass. Also in
Clinical Pharmacy and Clinical Pharmacology, William A. Gouveia,
Gianni Tognoni, Eppo van der Kleijn, Editors, Elsevier/North-Holland
Publishing Company, pp. 259-274, (1976).

HPP-75-6

AIM-266, CS-517, ADA019641
Randall Davis, Bruce Buchanan, Edward Shortliffe,
"Production Rules as a Representation of a Knowledge-Based Consultation
Program", (October 1975).

HPP-75-7

AIM-271, CS-524,, ADA019702/OWC
Randall Davis, Jonathan King,
"An Overview of Production Systems", Machine Representations of
Knowledge", Dordrecht, D. Reidel Publ. Co., 1976. Proceedings of
NATO ASI Conference, (October 1975).

HPP-75-8

R.G. Dromey, B.G. Buchanan, J. Lederberg and C. Djerassi,
"Applications of Artificial Intelligence for Chemical Inference. XIV.
"A General Method for Predicting Molecular Ions in Mass Spectra", Journal
of Organic Chemistry, 40, 770 (1975).

HPP-75-9

D.H. Smith,
"Applications of Artificial Intelligence for Chemical Inference. XV.
Constructive Graph Labelling Applied to Chemical Problems.
Chlorinated Hydrocarbons". Analytical Chemistry, 47, 1176 (1975).

HPP-75-10

R.E. Carhart, D.H. Smith, H. Brown and N.S. Sridharan,
"Applications of Artificial Intelligence for Chemical Inference XVI.
Computer Generation of Vertex Graphs and Ring Systems", Journal of
Chemical Information and Computer Science, 15, 124 (1975).

HPP-75-11

R.E. Carhart, D.H. Smith, H. Brown and C. Djerassi,
"Applications of Artificial Intelligence for Chemical Inference XVII.
An Approach to Computer-Assisted Elucidation of Molecular Structure",
Journal of the American Chemical Society, 97, 5755 (1975).

HPP-75-12

B.G. Buchanan,
"Applications of Artificial Intelligence to Scientific Reasoning", In
Proceedings of Second USA-Japan Computer Conference, American Federation
of Information Processing Societies Press, (August 1975).

HPP-75-13

R.E. Carhart, S.M. Johnson, D.H. Smith, B.G. Buchanan, R.G. Dromey,
J. Lederberg,
"Networking and a Collaborative Research Community: A Case Study Using
the DENDRAL Program", in "Computer Networking and Chemistry", P. Lykos,
Ed., American Chemistry Society, Washington, D.C., (1975).

HPP-75-14

D.H. Smith,
"The Scope of Structural Isomerism", Artificial Intelligence for Chemical
Inference. XVIII. Journal of Chemical Information and Computer Sciences,
15, 203 (1975).

HPP-75-15

E.H. Shortliffe, R. Davis,
"Some Considerations for the Implementation of Knowledge-Based Expert
Systems", SIGART Newsletter, 55:9-12, (December 1975).

HPP-76-1

D.H. Smith, J.P. Konopelski and C. Djerassi,
"Applications of Artificial Intelligence for Chemical Inference. XIX.
Computer Generation of Ion Structures", Organic Mass Spectrometry, 11,
86, (1976).

HPP-76-2

Raymond E. Carhart and Dennis H. Smith,
"Applications of Artificial Intelligence for Chemical Inference XX.
Intelligent Use of Constraints in Computer-Assisted Structure Elucidation".
(1976).

HPP-76-3

C.J. Cheer, D.H. Smith, and C. Djerassi,
"Applications of Artificial Intelligence for Chemical Inference XXI.
Chemical Studies of Marine Interbrates - XVII. The Computer-Assisted
Identification of [+]-Palostrol in the Marine Organism Cespitularia
sp., aff. subviridis". Tetrahedron. Vol. 32, pp. 1807 to 1810, Pergamon
Press, (1976).

HPP-76-4

B.G. Buchanan, D.H. Smith, W.C. White, R.J. Gritter, E.A. Feigenbaum, J. Lederberg, and Carl Djerassi,
"Application of Artificial Intelligence for Chemical Inference XXII. Automatic Rule Formation in Mass Spectrometry by Means of the Meta-DENDRAL Program", Journal of the American Chemical Society, 98, 6168 (1976).

HPP-76-5

T.H. Varkony, R.E. Carhart and D.H. Smith,
"Applications of Artificial Intelligence for Chemical Inference XXIII. Computer-Assisted Structure Elucidation. Modelling Chemical Reaction Sequences Used in Molecular Structure Problems^{1,2}". (1976).

HPP-76-6

D.H. Smith and R.E. Carhart
"Applications of Artificial Intelligence for Chemical Inference XXIV. Structural Isomerism of Mono and Sesquiterpenoid Skeletons 1,2-", Tetrahedron, Vol. 32, pp. 2513 to 2519, Pergamon Press (May 1976).

HPP-76-7

AIM-283, CS-552
Randall Davis,
"Applications of Meta Level Knowledge to the Construction, Maintenance and Use of Large Knowledge Bases",
Thesis: Ph.D. in Computer Science, (July 1976).

HPP-76-8

AIM-286, CS-570
Douglas Lenat,
"AM: An Artificial Intelligence Approach to Discovery in Mathematics as Heuristic Search",
Thesis: Ph.D. in Computer Science, (July 1976).

HPP-76-9

AIM-291, CS-577
Bruce G. Buchanan, Joshua Lederberg, John McCarthy,
"Three Reviews of J. Weizenbaum's Computer Power and Human Reason",
(November 1976).

HPP-76-10

Bruce G. Buchanan and Dennis Smith,
"Computer Assisted Chemical Reasoning", in Proceedings of the III International Conference on Computers in Chemical Research, Education and Technology", Pleum Publishing, (1976).

HPP-77-1

STAN-CS-77-593

A. C. Scott, W. Clancey, R. Davis, E. H. Shortliffe,
"Explanation Capabilities of Knowledge-Based Production Systems",
American Journal of Computational Linguistics, Microfiche 62,
Knowledge-Based Consultation Systems, (1977).

HPP-77-2

STAN-CS-77-589

Robert S. Engelmores and H. Penny Nii
"A Knowledge-Based System for the Interpretation of Protein X-Ray
Crystallographic Data", (January 1977).

HPP-77-3

B.G. Buchanan, R. Davis, V. Yu and S. Cohen,
"Rule Based Medical Decision Making by Computer",
(Submitted to MEDINFO.77, 1977).

HPP-77-4

T.M. Mitchell and G.M. Schwenzer,
"Applications of Artificial Intelligence for Chemical Inference.
XXV. A Computer Program For Automated Empirical 13C NMR Rule
Formation", (Submitted to JACS, January 1977).

HPP-77-5

STAN-CS-77-596

Mark Stefik and Nancy Martin,
"A Review of Knowledge-Based Systems as a Basis for a Genetics Experiment
Designing System", (February 1977).

HPP-77-6

STAN-CS-77-597

Bruce G. Buchanan and Tom Mitchell.
"Model-Directed Learning of Production Rules", Submitted to the Proceedings
for the Workshop on Pattern-Directed Inference Systems in Hawaii, (February, 1977).

HPP-77-7

H. Penny Nii and Edward A. Feigenbaum,
"Rule-based Understanding of Signals", to be presented at Workshop on
Pattern-directed Inference Systems, (May 1977).

Hpp-77-8

R. Davis,

"Knowledge Acquisition on Rule-based Systems: Knowledge about
Representations as a Basis for System Construction and Maintenance",
submitted to Pattern-Directed Inference Conference, (May 1977).

HPP-77-9

R. Davis,

"interactive Transfer of Expertise I: Acquisition of New Inference
Rules", submitted to Fifth IJCAI, (August 1977).

HPP-77-10

R. Davis,

"A Decision Support System for Medical Diagnosis and Therapy Selection in Data Base", SIGBDP Newsletter, 8, pp. 58-72, (Winter 1977).

HPP-77-11

Dennis H. Smith and Raymond E. Carhart,

"Structure Elucidation Based on Computer Analysis of High and Low Resolution Mass Spectral Data". Proceedings of the Symposium on Chemical Applications of High Performance Spectrometry, University of Nebraska, Lincoln, (1976 in press).

Budget
 Heuristic Programming Project
 Proposal for Renewal of Contract DAHC 15-73-C-0435
 July 1, 1977 - June 30, 1979

| | 7-1-77/6-30-78 | 7-1-78/6-30-79 | Total Budget 7-1-77/6-30-79 |
|---|----------------|----------------|--------------------------------|
| <u>Salaries</u> | | | |
| <u>Faculty</u> | | | |
| Feigenbaum, Edward A. Professor 33%, Summer Quarter, 1977 17%, Academic Year, 1977-78 100%, Summer Quarter, 1978 18%, Academic Year, 1978-79 | 8,089 | 15,829 | 23,918 |
| Buchanan, Bruce G. Adjunct Professor 50%, Calendar Year | 15,750 | 16,695 | 32,445 |
| Michie, Donald Visiting Professor in Artificial Intelligence 38%, One Quarter | - | 3,750 | 3,750 |
| <u>Research Staff</u> | | | |
| Brown, Dennis Research Associate 100%, Summer Quarter | 4,667 | 4,947 | 9,614 |
| Creary, Lewis G. Research Associate 100%, Summer Quarter | 4,375 | - | 4,375 |
| Davis, Randall (or replacement) Research Associate 54%, Calendar Year | 10,000 | - | 10,000 |
| Engelmore, Robert Research Associate 50%, Calendar Year | 15,120 | 16,027 | 31,147 |
| Research Associate (to be named) 100%, Calendar Year | - | 19,000 | 19,000 |
| Nii, H. Penny Scientific Programmer 100%, Calendar Year | 24,538 | 26,010 | 50,548 |
| White, William C. Programmer 50%, Calendar Year | 9,896 | 10,490 | 20,386 |
| <u>Five Student Research Assistants</u> (to be named) 50%, Academic Year; 100% Summer Quarter | 34,965 | 37,065 | 72,030 |
| <u>Secretarial and Clerical Assistance</u> 100%, Calendar Year | 9,600 | 10,368 | 19,968 |
| Subtotal, Salaries | 137,000 | 160,181 | 297,181 |

7-1-77/6-30-78 7-1-78/6-30-79 7-1-77/6-30-79

Staff Benefits

19.0% 9-1-78/8-31-77
 20.0% 9-1-77/8-31-78
 20.8% 9-1-78/8-31-79

27,126 33,109 60,235

Travel

10 West Coast trips
 6 East Coast trips
 1 Foreign trip

5,000 5,000 10,000

Expendable Supplies and Materials

Telephone, postage, office
 supplies, consulting, etc.

7,000 7,100 14,100

Publications and Report Costs

3,000 3,300 6,300

Equipment Rental and Maintenance

6,000 6,500 12,500

Equipment Purchase

Terminal Equipment

7,500 10,000 17,500

Subtotal Direct Costs

192,626 225,190 417,816

Indirect Costs, 58% (excluding equipment)

107,374 124,810 232,184

TOTAL BUDGET

300,000 350,000 650,000