MIT/LCS/TM-140

ARTIFICIAL INTELLIGENCE AND CLINICAL PROBLEM SOLVING

Peter Szolovits

September 1979

Artificial Intelligence and Clinical Problem Solving

Peter Szolovits

July 1979

This investigation was supported (in part) by National Institutes of Health Grant No. 1 P41 RR 01096-02 from the Division of Research Resources and by the National Library of Medicine Grant No. 1 P01 LM 03374-01.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LABORATORY FOR COMPUTER SCIENCE

CAMBRIDGE MASSACHUSETTS 02139

Abstract

An ambitious, but intriguing, possibility for radically increasing the availability and adequacy of health care, while containing its cost, is to use the computer as a consultant to augment and extend the skills of all health care providers. We propose to pursue a program of fundamental research in representation of knowledge, decision-making, problem-solving, program explanation and clinical cognition, to understand how to construct computer programs that, as an integral part of the health care system, can amplify the knowledge and reasoning powers of medical decision makers. We plan to apply the techniques so developed to problems in the diagnosis and therapy of acid/base and electrolyte disturbances, diagnosis of birth defects using an existing data-base of diseases and associated manifestations, the development of multi-modal cancer therapy protocols, and the application of the methods of decision analysis to produce general tools for physicians to use in analysing difficult clinical cases.

Key Words: knowledge based systems, artificial intelligence, computer diagnosis, clinical decision making, problem solving

Preface

This Memo is (in large part) a reproduction of a proposal submitted by the Clinical Decision Making Group of M.I.T.'s Laboratory for Computer Science to the National Library of Medicine (NLM). Our proposal was in response to the announcement by the NLM of a new program of support for fundamental research "in the area of knowledge representation, linguistic theory, decision-making, problem-solving, data manipulation, and information engineering" and its applications in medicine. The work proposed here has been approved by the NLM with the intention that it will be funded for five years (1979-1984).[1] Therefore, it seems appropriate for us to make available to our colleagues and students our current assessment of the state of the art in these domains and the most interesting problems which promise to provide the basis for fruitful research over the next half decade.

The writing of this proposal was itself a major collaborative task, bringing together investigators from M.I.T., Tufts University-New England Medical Center Hospital, Boston University-University Hospital, and the Baylor College of Medicine. Portions of the text and many of the ideas were contributed by the following people:

Peter Szolovits, Ph.D., Assistant Professor of Computer Science and Engineering, Director, Clinical Decision Making Group, M.I.T.

Marylou Buyse, M.D., Assistant Professor of Medicine, Assistant Pediatrician; Tufts University-New England Medical Center Hospital

Kenneth W. Church, Research Assistant, Electrical Engineering and Computer Science, M.I.T.

Byron Davies, Research Associate, Hematology Service, Department of Medicine, New England Medical Center Hospital

Randall Davis, Ph.D., Assistant Professor of Electrical Engineering and Computer Science, M.I.T.

Jane Desforges, M.D., Professor of Medicine, Senior Physician and Hematologist, Tufts University-New England Medical Center Hospital

Robert Friedman, M.D., Assistant Professor of Medicine, Chief, Medical Information Systems Unit, University Hospital, Boston University

---

1. Part of the research of our Group which is not described in this document and computational support for all of the research proposed here will continue to be funded by the Division of Research Resources, NIH, under grant number RR-01096.

Harold A. Goldberger, M.D., Research Assistant, Electrical Engineering and Computer Science, M.I.T.

G. Anthony Gorry, Ph.D., Professor of Health Management, Director, Program for Health Management Research, Baylor College of Medicine

Jerome P. Kassirer, M.D., Associate Chairman, Department of Medicine, Associate Physician-in-chief, Tufts University-New England Medical Center Hospital

William J. Long, Ph.D., Division of Sponsored Research Staff, Electrical Engineering and Computer Science, M.I.T.

William A. Martin, Ph.D., Associate Professor of Electrical Engineering and Computer Science, and Associate Professor of Management, M.I.T.

Ramesh S. Patil, Research Assistant, Electrical Engineering and Computer Science, M.I.T.

Stephen G. Pauker, M.D., Associate Professor of Medicine, Assistant Physician, Tufts University-New England Medical Center Hospital

William B. Schwartz, M.D., Vannevar Bush University Professor of Medicine, Tufts University School of Medicine

Howard B. Sherman, Research Assistant, Electrical Engineering and Computer Science, M.I.T.

Brian C. Smith, Research Assistant, Electrical Engineering and Computer Science, M.I.T.

William R. Swartout, Research Assistant, Electrical Engineering and Computer Science, M.I.T.

The problems outlined in this proposal and the ones we encounter throughout the coming years of continued research should provide fertile ground for interesting research, both by us and our colleagues at other institutions.

CONTENTS

1. Introduction

As Schwartz has noted [III] "Many discussions during the past decade have considered the use of computers as an adjunct to medicine. Few, however, have fully explored the possibility that the computer as an intellectual tool can reshape the present system of health care, fundamentally alter the role of the physician, and profoundly change the nature of medical manpower recruitment and medical education--in short, the possibility that the health-care system by the year 2000 will be basically different from what it is today.

"Much has, of course, already been said about the role of the computer in improving the efficiency of the health-care system. These now familiar projections envision the computer performing a wide variety of functions such as the scheduling of hospital admissions, the keeping of medical records, and the operation of laboratory and pharmacy. Such developments in the area of 'housekeeping' activities offer considerable hope for the improvement of both hospital and outpatient operations but do not come to grips with the more fundamental problems of the health-care system--the increasing shortage of physician manpower and the geographic maldistribution resulting from the reluctance of today's doctor to practice in rural or depressed urban communities. Even less do they give hope of dealing with the difficult challenge of maintaining a high level of physician competence in the face of a continued expansion of medical knowledge that tends to widen progressively the gap between what a doctor should know and what he can retain and utilize. The computer thus remains (in the light of conventional projections) as an adjunct to the present system, serving a palliative function but not really solving the major problems inherent in that system. There is, in fact, little reason to believe that any of the current proposals for solving these problems, technologic or other, will do more than mitigate their severity".

One radical and intriguing possibility for improving the efficiency and effectiveness of the health care system is to use the computer as an intellectual or deductive instrument--a consultant that is built into the very structure of the health care system and augments the abilities of physicians and paramedical personnel. Clearly, however, considerable intellectual and technological resources must be marshalled and a long term research commitment must be made if this possibility is to be realized.

1.1 Objectives

In our view, the primary scientific and technical insights needed to realize the above goal must come from the collaboration of medical professionals and computer scientists. It seems highly unlikely that researchers in artificial intelligence (AI) will be able to duplicate the capabilities of the human intellect in the near future. This being so, there will be no direct replacement of medical personnel by computers. Rather, it is required to understand more deeply and in detail the information processing which must take place in order to perform specific clinical diagnosis and management functions. Only medical experts have the knowledge required to perform this analysis.

While this essential knowledge rests in the medical community there is, within the computer science community, a growing body of techniques and software tools which promise to provide a revolutionary approach to the analysis and augmentation of expert problem solving. Our objective is to pursue collaborative research, bringing together medical experts and computer scientists to develop a formal understanding of the physicians' art in terms of computational models

which may be applied in building the intellectual instruments mentioned above.

Although the physicians with whom we collaborate are working in specific areas such as cardiology, acid/base disorders, or cancer, the techniques we are supplying and developing are not restricted to any particular area of medicine. We might classify them as follows.

1.1.1 Representation of Knowledge

A key problem is to represent medical knowledge in a computer. In looking at the weaknesses of programs produced to date it becomes clear that a much higher level of performance can be reached merely by the addition of a large number of specific techniques, normally quite simple ones, which apply only in certain situations. One must represent these techniques and the problem situations under which they apply in the computer. To do this requires the definition of a large number of terms. Unless this is done in an orderly fashion, the system soon becomes hopelessly complex. Fortunately the complexity can be reduced by addressing a series of generic issues.

For example, if we are to capture clinical expertise in a machine, we must equip the machine with an understanding of time and events which take place in time. Thus the machine needs a minimal ability to place events and intervals on some form of time-line, and to make appropriate deductions about this arrangement. But much more is required. For example, we must develop ways to capture the concept of episodes. The machine needs to understand such fragments as "the gradual onset of the disease" and "an abrupt cessation of symptoms."

In addition to time, notions of causality and co-occurrence are required.

It is easy to cite examples of diagnosis which rely on a knowledge of anatomy and physiology. For example, a series of symptoms are correlated with successive infringement on regulatory mechanisms by a tumor. This knowledge must be represented in the computer.

To avoid an explosion of facts one must be able to capture the relationship between, for example, edema, pedal edema, peri-orbital edema, symmetric peri-orbital edema, and the edema of a specific patient. Systems which simply deal with vectors of diseases and symptoms cannot do this. Progress in research on English language processing by computer now suggests certain linguistically motivated forms of representation which must be explored for their adequacy in describing the innate interrelationships among facts.

1.1.2 Computational Models of Reasoning

The knowledge represented in a computer must express not only the relevant facts of medicine and an adequate description of the patient under consideration but also the diagnostic hypotheses and therapeutic plans to be contemplated and reasoning techniques which may extend and modify them. We believe that a diagnostic program must include an overall coherent hypothesis which represents not only the suspected disorders but also the uncertainties about them, their causal and temporal interconnections, and the manner in which they explain or fail to account for known findings. In this view, reasoning is the sequential construction, alteration and refinement of the coherent hypothesis structure. A similar model is indicated for reasoning about therapeutic planning and evaluation.

Directly related to the design of computational models of reasoning is the investigation of human cognition. An important element of our research has been the study of protocols for clues to the mechanisms of expert clinical behavior. Many of our past programs have been built on the principles elicited by these studies, and we consider this an effective research methodology for both the elucidation of teachable models of human cognition and the development of computer programs. Of special interest in this domain is the investigation of the numerical weighting of alternative hypotheses.

I.1.3 Inquiry and Explanation

Related to the representation of knowledge and models of reasoning is the development of mechanisms which allow a user to employ a natural and direct mode of interaction with a program and allow a program the ability to explain its behavior in terms which are readily understandable to a clinician. This capability must be developed to meet several needs:

a) the need for users to understand the basis for a program's advice, particularly when the clinical problem is a serious one or when its advice runs counter to the user's conception.

b) the need for clinicians working on programs to have access to facts and procedures used by the program in arriving at a particular conclusion.

c) the need for a program to relate its facts and strategies to the underlying medical reality.

d) the need for students to interrogate the program to learn about its strategies.

I.1.4 Decision Analysis

When a decision problem is ultimately narrowed down to a choice among a small number of well-understood alternatives, the techniques of decision analysis are most appropriate for solving or exploring the problem. These techniques have proven quite useful in tackling a number of medical problems and we want to continue to supply and develop this expertise as we feel it will be important in any comprehensive solutions to the administrative aspects of problem solving. The most difficult aspect of using decision analysis, both for a complex medical program and for a physician, is the selection of the appropriate formulation of the problem, the collection or estimation of relevant probabilities and utilities, and the interpretation of the results of analysis in terms of sensitivities and uncertainty.

Several of the clinicians in our research group have been active in applying decision analytic techniques to actual patient care and in training other physicians in their use. From this experience, we have identified useful techniques to employ in the formulation of new analyses. We intend to develop a new program to aid physicians in the implementation and exploration of new decision analytic formulations of medical problems.

1.1.5 Model-Based Decision-Making

There are a number of important areas of clinical medicine in which a formal (generally mathematical) model is available upon which certain diagnostic or management decisions could conceivably be based. In many of these cases, however, the model in question is of little clinical use. Although the model often surpasses the ability of even the best physician to deal with certain aspects of the problem or with classic cases, it cannot cope with a variety of patient-specific factors which should be factored into the decisions, or certain emergency conditions which should cause a reordering of the priorities in the model. In addition, physicians usually interact with a patient over an extended time and modify their plans and expectations based on previous clinical experience. In general, physicians understand how to alter and refine their approach to a problem in the light of such factors, but computer programs unfortunately remain very rigid in this regard.

To achieve more flexible models, we need new ways to combine medical common sense with mathematical models. The models themselves must be represented in such a way as to allow this common sense to be applied. Hence it must be clear to some supervisory program what the basis for a particular model is, and how changes or assumptions about the patient affect this basis, and hence the model.

Our group has been developing expertise in the above areas for a period of years.

1.2 Background

Several years ago a medical decision making effort was started at the MIT Sloan School of Management. The nucleus of the group consisted of Drs. Jerome P. Kassirer and William B. Schwartz of Tufts-New England Medical Center Hospital (NEMCH) and Professor G. Anthony Gorry of MIT. Drs. Schwartz and Kassirer had been working on the problem of encoding the protocols of experts in computer programs, and had developed a program for acid-base problems [111]. Gorry had developed a program which used statistical decision theory to solve diagnostic problems [41]. Because of the common interest in automating processes for clinical decision-making, the three joined forces.

The initial efforts of the group were directed along the lines suggested by the decision theory program. The work was considerably deepened and expanded during the two years following the initial formation of the group. A series of papers describing the work were published [112, 42]. These two papers consider in detail the application of decision analysis to clinical decision making, both insofar as the automation of the process is concerned, and with respect to use of this formalism by clinicians.

Dr. Stephen G. Pauker joined the group in 1971, bringing to it a rare combination of expertise both in medicine and in computer science.

During the latter stages of their work on decision analysis, the group began to see certain difficulties in using decision analysis as the sole basis for a system to deal with real problems of crisis medicine. After further definition of these difficulties, they were given a research grant from HISMA under which they explored these problems. From this exploration emerged a need for a close cooperation with skilled computer scientists.

In order to promote a closer union between computer scientists and medical personnel in the group, a week long conference on the problems of clinical decision making and the relevance of computer science to these problems was held. Attending were MIT Profs. Minsky, Fredkin, Moses, Martin, and Sussman (all in Computer Science), and the members of the group already mentioned.

Subsequent to the meeting Prof. Gorry joined the Laboratory for Computer Science and began collaborating with Prof. William A. Martin in a graduate course, Knowledge Based Application Systems. A number of graduate students in computer science were brought into the group and Profs. Gerald J. Sussman, Edward Fredkin, and Marvin Minsky took a role in supervising these students. That year, 1974, a three year grant under Section 769A P.L. 92157 was obtained with Prof. Gorry as principal investigator. These funds, which were eventually extended for a fourth year, expired August 31, 1978.

During the next several years the network of collaborators has further expanded. In discussions with Dr. Schwartz, Dr. Jane Desforges at NEMCH became interested in the application of decision analysis to diagnostic planning for patients with Hodgkin's Disease. Collaborating with members of our group a team at NEMCH published two papers [108, 109] and received a three year grant from the National Cancer Institute starting July 1976. After several more publications [104, 105] and the successful demonstration of their programs, this group is now seeking continued support from NCI as well as continuing their collaboration with us. We recruited Prof. Barnett of the MIT Sloan School to help them with decision analysis, and we also supply students, computer time, and computer expertise.

A student, Mr. Howard Silverman, working with Dr. Pauker and Prof. Gorry, developed a digitalis therapy advisor [43]. Then, in the summer of 1975, Prof. Gorry went to Baylor Medical School to head the Program for Health Management. Since the digitalis program performed quite well in tests at NEMCH, Prof. Gorry obtained funding from NHBLI to run a more exhaustive retrospective test of the program at Baylor. A first round of such tests has been completed and the analysis of the resulting data is now underway.

Prof. Szolovits joined the MIT faculty in 1974 in order to work in the area of knowledge based application systems. He began to teach the knowledge Based Applications course, decided to focus his interest on medical applications, and when Prof. Gorry left he assumed the major technical responsibility for the work at MIT

Because much of the interest in the digitalis program has been on its acceptability to the potential user community and its use as an educational tool, Mr. William Swartout, a student of Profs. Martin and Szolovits, implemented a new version of the program in the OWL language which allowed his development of an explanation capability, on which research continues.

Dr. William Long joined the group as a research staff member to conduct the MIT end of the digitalis program's testing (all the cases were collected and abstracted in Houston but actually run at MIT) and to improve the program's modeling techniques and models based on our experience with it.

Through his association with Prof. Gorry, Prof. Martin became convinced that the medical area was ideally suited for application of his work in knowledge representation and natural language dialogue. Although that work was and is adequately supported by the Advanced

Research Projects Agency (ARPA), that agency will not sponsor any work in medical applications and so on the departure of Prof. Gorry, Prof. Martin assumed the role of Principal Investigator of the Section 769A grant to make possible applications in that area.

Concurrent with the final year of the Section 769A funding, we were awarded a three-year grant from the Division of Research Resources, to run from Sept. 1, 1977 to Aug. 31, 1980. On Sept. 1, 1978, Prof. Szolovits took over as Principal Investigator of the Resource, and Prof. Martin has turned full time to his work on knowledge representation and natural language. Although Prof. Martin is no longer officially part of the project, he and Prof. Szolovits are still collaborators and his OWL language is being used by us.

Mr. Ramesh Patil has joined the group to work on issues of knowledge representation, especially as applied to acid/base and electrolyte problems. Mr. Brian Smith and Dr. Harold Goldberger, a psychiatrist who has entered MIT as a graduate student in computer science have also worked on the development of artificial intelligence techniques for medical reasoning and representation [117].

Prof. Randy Davis joined the MIT faculty in the Fall of 1978 after working on the MYCIN project at Stanford University, and has begun to work on problem solving strategies as applied to medical reasoning tasks. Prof. Robert H. Friedman, a physician and Chief of Medical Information Systems in the Department of Medicine at Boston University-University Hospital has begun to work with us on the application of our modeling techniques to the design of cancer care treatment protocols. Dr. Friedman is Principal Investigator of the Cancer Care Data Management System project at Boston University, which is supported by NCI.

1.2.1 Scientific and Technical Environment

The MIT Laboratory for Computer Science (LCS) and the MIT Artificial Intelligence Laboratory have a combined annual budget of approximately 6.3 million dollars and employ roughly 300 technical people. Founded in 1963 as Project MAC (for Multiple Access Computer and Machine Aided Cognition), LCS developed the Compatible Time-Sharing System (CTSS), one of the first time-shared systems in the world, and Multics--an improved time-sharing system that introduced several new concepts. These two major developments stimulated research activities in the application of on-line computing to such diverse disciplines as engineering, architecture, mathematics, biology, medicine, library science, and management. Since that time, the Laboratory's objectives expanded, leading to a broad front of research activities in many areas of computer science, including: computer and operating systems, data bases, real-time computation, programming languages, methodology, and technology, theory of computation, semantics, symbolic mathematical manipulation, and knowledge-based systems. The AI Laboratory is the largest such laboratory in the world and has a distinguished record as the source of ideas in time sharing, list processing (LISP), graphics, computer image processing, and other software technologies supportive of AI research. In 1967 Profs. Martin and Moses completed doctorates in the AI group and joined the Laboratory for Computer Science to build the MACSYMA system for algebraic manipulation. MACSYMA and DENDRAL are often cited as the two most successful systems incorporating the knowledge of experts and having their roots in AI research. MACSYMA is currently in wide use over the ARPA network and is supported by subscriptions from a consortium of users. Since 1972 Prof. Martin has been working to develop means of producing similar systems in other areas of expertise. Under ARPA support Prof. Martin has developed, with a team of about half a dozen

researchers, a first version of OWL [46, 66, 123], a system for computing with knowledge representations based on English. Continued ARPA support of the OWL effort is anticipated with the goal of producing a system which interacts with users in typed English and assists them in maintaining a description of their files and data bases. As mentioned above, ARPA will not support the application of this technology in the medical area. Nevertheless, this area is highly attractive to both students and staff for humanitarian as well as technical reasons. Profs. Minsky, Fredkin, Martin and Sussman have supervised the work of students in this area in the past and we expect them to do so in the future, although we do not plan to charge their time to the project.

A number of the research activities of MIT colleagues within the two Laboratories indirectly support the research proposed here and help to create a stimulating intellectual environment. Prof. Martin heads the Knowledge Based Systems Group at LCS, in which supportive work in natural language processing and knowledge representation is being carried out. Under Prof. Sussman, the AI lab has a number of investigators working on the creation of expert reasoning programs in electronics and programming.

The members of our collaborative group have also played a significant role in the developing national AI in Medicine community. We have been participants in the yearly AIM Workshops held since 1975 at Rutgers University, and we are planning to hold this year's workshop at MIT. We stay in close touch with others in our discipline, and several of us serve on the oversight or executive committees of other AIM research groups (the Rutgers resource, and Stanford's SUMEX-AIM).

1.2.2 Rationale

We believe that expert medical advising programs, for most medical areas, will operate most successfully by incorporating the clinical knowledge, intuitions, and techniques which the best expert doctors employ in their practices rather than by implementing newly invented medical techniques. Our efforts concentrate, therefore, on eliciting the heuristic principles on which current medical experts, our collaborators, operate; we then try to solve the technical computer science issues which let us implement those principles in our programs. A newly hypothesized component of the cognitive theory often poses an implementation challenge which can lead to novel computer science developments. The working program in its behavior on a series of cases often points out an inadequacy of or fallacy in the proposed clinical principle and challenges us to a new round of discussion, introspection and protocol analysis to create an improved theory. It is the detailed comprehensibility of a program based on these clinical principles coupled with the incremental changability and immediate behavioral feedback of the program's computer implementation which drive this research and development cycle.

Even in the application of normative formal decision-making techniques such as decision analysis, our experience suggests that overcoming the problems of making such tools truly available and acceptable to the medical practitioner forces us to broaden our view beyond the strict confines of the formal technique to the "softer" problems of its applicability and environment. We have argued elsewhere that although such formal techniques may be of great value, they need to be applied within a context determined according to a human cognitive theory.

Our particular research efforts vary over a considerable range along this dimension. In our core research and the acid/base and electrolyte project, we are specifically focusing on building formal models of human-style reasoning; in our behavioral analysis studies we seek to discover aspects of that model from observing human problem solvers. In our other subprojects, we accept a much stronger mathematical (statistical or functional) model as representing the underlying "reality" of the application domain. There, we study AI techniques as the setters of context for applying the more formal methods.

1.2.3 Organization of The Project: Core Research and Subprojects

The objectives described above identify a number of intellectual problems which form the core of our research activity. They represent fundamental research plans "in the area of knowledge representation, linguistic theory, decision-making, problem-solving, data manipulation, and information engineering," as specified in the National Library's program announcement.

We have found that, in addition to a concerted program of core research on underlying theoretical issues which impact on all our objectives, it is most effective to organize our investigations around specific subprojects which use the knowledge of a particular medical domain. This is true even when the principal thrust of some subproject may indeed turn out to generate techniques which are universally applicable. The following are our reasons for this viewpoint:

1) The study of major research problems such as knowledge representation and reasoning is virtually impossible in the abstract. One must choose some set of knowledge to be represented, some problem domain to reason about. In the history of AI research, in fact, this same observation has led researchers to define micro-worlds to investigate: moving blocks on a table, understanding children's stories, and playing chess, among others. Although we believe that the medical domain offers problems significantly different from the traditional AI research micro-worlds, we also believe that the techniques developed for one medical domain are likely to be applicable to others. Thus, the choice of a specific medical area serves to help focus our attentions--if done properly, it does not restrict the generality of our results.

2) The choice of "real world" problems as the context for investigating fundamental questions imposes a stricter methodologic discipline than when the application domain is created by the researcher. Even if only unconsciously, the made-up application can be subtly tailored so that even partially adequate solutions may appear complete if their flaws can be defined away. With the proper choice of an actual application area, we gain an independent set of criteria for evaluating the success of the research. For example, in our work on explanation we have already identified specific needs of relating the program's actions to the user's understanding of the application domain which were ignored by previous AI researchers because the "blocks world" is sufficiently simple that everyone understands its fundamental character without need for explanation.

3) Much of our insight must come from practicing doctors because only they know how doctors represent and reason about medical problems. Because our collaborators are experts in various fields of medicine, it is sensible to choose application domains in which they are especially knowledgable. Further, an additional result of our studies is often an illumination of the structure of a medical problem area. Our collaborators and the

medical community consider this an important benefit, especially to improve the manner in which medical students and interns and residents are trained. Every result of our past research efforts in history taking, digitalis administration, and decision analysis has been appropriated by our colleagues to help structure their teaching.

For these reasons, both our proposal and our planned work is organized in terms of application-oriented subprojects. Nevertheless, the fundamental research questions we are addressing are those listed as our objectives.

1.3 Specific Aims

As part of the core research activity of the project, we will pursue the following topics:

1) Computer generated explanation of a program's behavior and its justification in terms of the medical models of the domain. This work will use the Digitalis Therapy Advisor as the target program to be explained.

2) The development of general anatomical and physiological modeling techniques to represent the normal and abnormal functioning of the human organism. These models are to be useful both as a basis common sense explanations and reasoning.

3) The selection and co-ordination of alternate AI problem-solving methods depending on the specifics of the problem and its partial solution. This work is to generalize the problem-solving methods of current AIM programs.

Each of these directly supports the objectives described in this proposal.

In addition to the project core, we will have six subprojects:

1) Acid/Base and Electrolyte Project (ABEL): an expert acid/base and electrolyte diagnosis and therapy advisor,

2) Clinical Cognition Project: the investigation of expert doctors' clinical cognition,

3) Decision Analytic Clinical Consultation Tools Project: the development of AI-based tools for formulating decision analyses,

4) Birth Defects Diagnosis Project: the development of a program to diagnose birth defects using a large database of existing information, and

5) Multi-Modal Cancer Care Therapy Project: a study of multi-modality therapy design in cancer treatment,

The next few paragraphs describe how each of the project's objectives is taken up in the various subprojects.

Knowledge representation is the key to all of our efforts, and therefore plays a role in each subproject. The representation of temporal and causal relations and of varying levels of detail in description are of greatest importance in the ABEL subproject; indeed, that application area was selected largely to address these issues. The investigation of a linguistically-based representation system is most significant in our core project on explanation, although it also arises in ABEL.

Our interest in computational models of reasoning is strong in ABEL, where we address problems of deep reasoning in a very complex problem domain, in the Birth Defects area, where we deal with shallow reasoning over a large knowledge base, and in the Clinical Cognition subproject, where we study how expert doctors reason.

Work on inquiry and explanation is specifically identified as part of the core research activity. In fact, the domain to which we apply this work is the digitalis therapy advisor, which is not listed as a subproject here because its development is adequately supported by other grants. We believe the need for explanation from programs to be so important that all our programs must eventually have such a capacity. For now, however, we concentrate on just one of these to develop the needed capabilities.

The formulation and use of decision analysis will be pursued principally in the Decision Analysis Tools subproject, though some of the issues of how to integrate such numerical decision making techniques with symbolic rules will arise in ABEL and with clinical models in the Cancer Therapy subprojects, as well as the explanation part of the core. The development of anatomical and physiological based models is a separate component of the core.

The detailed methods of procedure of each of the subprojects and of the core will be taken up in the subsequent major sections of the proposal.

1.4 Administrative Structure

1.4.1 Project Administration

The MIT Laboratory for Computer Science is an interdepartmental laboratory drawing the bulk of its staff from the Department of Electrical Engineering and Computer Science. The laboratory head, Prof. M. Dertouzos reports to the MIT Provost for technical matters and to the MIT Office of Special Programs for financial and legal matters. The Principal Investigator, Prof. Peter Szolovits is a faculty member in the Department of Electrical Engineering and Computer Science working in the Laboratory for Computer Science. The LCS staff provides Prof. Szolovits with administrative and logistics support. The staff responsible for the operation of the PDP-10 computers report to Profs. Martin, Moses, and Winston jointly.

Prof. Szolovits, as Director, will be responsible for the overall activities of the project. Each identified subproject will have a designated Principal Investigator, who will report to Prof. Szolovits. All administrative interaction with the funding agency will be handled through the project as a whole, as part of the Director's responsibility.

1.4.2 Collaborative Arrangements

Because the grant is to be administered through MIT and several of the investigators to be supported are at other institutions, suitable administrative arrangements will be created for transferring appropriate funds to be handled by each investigator's own institution. Thus far in other projects, we have handled this via a subcontract mechanism, which has worked well.

Each section of the proposal which describes a subproject ends with an assurance by the Principal Investigator of that subproject that he or she will take responsibility for that subproject.

1.4.3 Facilities

Computing facilities are to be provided under the Research Resource grant (NIH Number 1 P41 RR 01096) on the MIT-ML PDP-10 computer, which is used for our medical research and ARPA sponsored work on knowledge based systems and automatic programming.

This computer is under the direction of Prof. Martin and has been running smoothly for many years with a small staff of excellent systems people shared with other LCS and AI lab PDP-10's. The machine is on the ARPA network and has a number of dial-up lines which have been in service for some time supporting MACSYMA. MACSYMA has now been moved to its own PDP-10, leaving behind this well exercised system.

1.4.4 Human Subjects

During the first year of the proposed project, no aspect of the research we are proposing will involve experiments on human subjects, under the appropriate federal guidelines. Depending on the rate of progress of each of the subprojects, some of them may involve the need for approval of human subjects experimental plans in later years of the project. MIT has an appropriate organization to conduct such reviews, and we intend to seek their advice and approval when the issue becomes current. To date, we have been reluctant to seek anticipatory human subjects approval because (a) the need does not yet exist and (b) we do not wish to engage in speculative proceedings before real plans are formulated.

1.5 Significance

The significance of our proposed work centers on its potential contribution to the theory of how to represent and manipulate medical information in the computer and its importance to the creation of tools which improve the health care system.

1.5.1 Technical and Educational Significance

With the more widespread use of computers in therapeutic and experimental settings, we are on the brink of a new set of opportunities to apply systematic aids to human decision making in medicine. To achieve this potential, however, we must lay the groundwork of technology which will allow the computer itself to deal with the enormous quantity and complexity of data that will be available, since otherwise the new administrative burden on the potential computer user might outweigh his expected benefit. Therefore, programs will need to include realistic and explicit models of the data they work with. Their reasoning steps must also be explicit, not only in terms of

computational steps for instructing the machine but in terms that are meaningful to the user, that is, in the terminology of the application.

Two principal classes of results are expected from our work: contributions to the basic scientific questions of artificial intelligence in medicine, and prototypical programs for aiding their users in medical decision making. In this proposal, we outline a significant set of research objectives and plans which will lead to the development of artificial intelligence techniques to serve as the underlying basis for the potential programs mentioned above. The successful completion of our plans will yield new theories of knowledge representation, reasoning, explanation, and modeling. It will also give us insights into how these theories can in fact be applied to prototypical medical application areas. Further, we will generate a number of working, testable programs which aid (at least experimentally) a decision maker facing a difficult medical problem.

An additional impact will be on the curriculum at Tufts University School of Medicine and on the teaching activities on the wards of New England Medical Center Hospital, where most of our medical collaborators practice and teach. An important component of our effort is concerned with the development of cognitive theories which can explain clinical problem solving and which in turn can be implemented as computer programs which simulate expert performance. We believe, however, that our work on clinical cognition also has the potential for contributing importantly to new educational programs directed toward improving the quality of physician's own problem solving activities. The current emphasis of the educational process is almost entirely on acquisition of knowledge, not on an understanding of the process by which physicians solve problems and acquire "clinical judgment". Such matters of process have been viewed as too subtle and complex to be either understood or taught but we feel that the work we propose (along with our previous studies) will provide the basis of the teaching of better problem-solving techniques. We have been introducing these ideas and insights into the curriculum at Tufts University School of Medicine and also as a part of day-to-day bedside teaching at the New England Medical Center Hospital. We hope that our efforts, if successful, will stimulate other medical schools to introduce training in problem solving into their curricula as well, using techniques and materials which we plan to develop as we implement our own program.

Our new collaboration with physicians at Boston University-University Hospital also opens the possibility of applying AIM techniques in a clinically-used data management system, thereby advancing the state of the art of that system as well as providing an excellent testbed for the AIM methods in a realistic environment.

1.5.2 Impact on Current Health Care Problems

Two of the major issues facing the health care system today are problems of quality of care and of the maldistribution of health services. Each of these problems can potentially be favorably affected in an important fashion by the work described in this application.

1.5.2.2 Quality of Care

The ever-expanding body of knowledge in medicine has created a situation in which it is increasingly difficult, in fact nigh impossible, for even the most conscientious physician to maintain a high degree of competence across the range of medical problems he is likely to encounter. The resulting adverse effect on quality of care has been substantial. It seems clear, therefore, that any

effort which could make expert consulting skills readily available to the practicing physician should have a considerable impact on the health care system and will help ameliorate a problem that is of increasing concern to society.

It is our belief that the work proposed here gives promise of contributing to the development of computer programs which can provide expert consulting capability and which can thereby upgrade physician performance. Many years of work will undoubtedly be necessary before a major effect on the health care system can be anticipated. We believe, however, that the kind of fundamental studies which we have proposed then will make a significant contribution towards the eventual development of an effective, large-scale consulting system.

1.5.2.4 Distribution of Health Services

A second major difficulty facing the country is the maldistribution of physicians. Physicians simply do not wish to live and practice in rural communities and the depressed inner city, with the result that the provision of health services in such areas is grossly inadequate. Over the long run it seems likely that adequate services in such areas can probably be provided only by the extensive use of allied health personnel (nurse practitioners, etc.). This strategy threatens, however, to create two standards of health care, a socially unacceptable arrangement. We believe that the research activities we propose to undertake and the kinds of consultation programs, which they should lead to, can help significantly with the problem making it possible for allied health personnel to deliver higher quality care than would otherwise be possible. The immediate availability of expert advice through the computer (including advice on when to refer a patient to a physician or hospital) should markedly upgrade the performance of non-physicians and thus greatly extend the usefulness and acceptability of such personnel as health care providers.

2. Core Research

Personnel:

Peter Szolovits, Ph.D., Principal Investigator
Randall Davis, Ph.D., Investigator
Brian Smith, Research Assistant
William Swartout, Research Assistant
Kenneth Church, Research Assistant

2.1 Overview

Medical education and training place a great emphasis on experience as the principal means whereby the student develops the "feel", the insight, and the grasp of typical clinical situations which he needs to become a capable doctor. Such "experience" consists not simply of practical exposure to clinical situations, but more specifically of exposure to the expertise of a competent physician at work in a complete medical setting. The expert physician is an expert because of his greater problem solving skills and because he has handled a hundred or a thousand times more cases in his specialty than the typical general practitioner. His refers to a greater stock of prototype cases, and he also has a better ability to handle cases which are divergent from them. His investigative skills more refined; his judgment more certain.

It is this expertise which the medical experience is able to convey to the student. Yet we do not understand just how experience accumulates to give these advantages; we do not know how to teach better, to transmit more effectively this clinical expertise. Nor do we know how, as a physician gains knowledge, how he is able to translate this into increased skill and increased subtlety. We have no definitive theory of how either knowledge or experience can together be distilled into skillful medical reasoning.

The attempt to formalize in a computer program our best theory of how the expert physician performs an important part of his task focuses our attention on just these issues: What is expert medical knowledge? How much knowledge is required? How should it be organized and how should it be applied? The answers to such questions will come only from the careful study of a real problem domain, and the success of such studies will be determined in large part by the boundedness of the problem domain under consideration. We believe that medicine, with its highly developed taxonomy, its codified knowledge-base, the generally repetitive nature of the problem-solving encounters, and the existence of acknowledged experts, constitutes a promising set of problem domains because of its well-bounded character. We therefore feel that, building on current and projected technology, acceptable progress can be made toward the development of decision-making programs that can deal competently with complex clinical problems.

The previous section of the proposal has already introduced the scope and rationale for our core research. In this section, we take up the three identified components of that research. Each contributes in its own way to the whole research program, taking a different "cut" across what would otherwise be an unmanageably large task. The first project is a proposal to formulate a set of conceptual building-blocks or elements out of which computational theories of knowledge

representation and use may be built; the second and third consider specific applications of such knowledge: the explanation and justification of medical decisions, and the use of knowledge to select specific problem solving strategies.

2.2 The Representation of Knowledge

2.2.1 Objective

Investigations into the computational representation of knowledge form one of the central issues in current artificial intelligence research. It is undeniably true that any powerful computational model of medical reasoning will consist of, among other components, a representation of a considerable amount of medical knowledge. Consider a few of the most elementary facts about the heart: it is a pump (and may therefore have to do with blood pressure); it is an organ in the thoracic cavity; it gives off both audible and electrical signals which can be monitored in order to determine its general functioning. These facts are just the smallest fragments of the very large body of knowledge on which any practicing physician constantly relies.

Part of the enterprise of formulating computational models of medical expertise therefore means understanding how such knowledge can be usefully embedded within a computational formalism. This task divides naturally into two highly interdependent, but nonetheless distinguishable, parts. The first of these components is of a more theoretical and abstract nature: it focuses on such questions as just what it means for knowledge to be integrated into a computer program, how active procedures can make use of a wide-ranging body of stored knowledge, how those active procedures (such as search routines or differential diagnosis strategies) may themselves be represented in such a way that they are transparently accessible to other parts of the program (such as explanation facilities and updating routines), and so on. Studies of such issues will make reference to previous work in representing medical knowledge, research in knowledge representation in AI in general, and to much of the work of such related disciplines as computer science, philosophy, psychology, and education.

The second ingredient in the overall enterprise to develop powerful knowledge representation theories is to make use of representation systems to represent and use specific kinds of medical knowledge for specific purposes, such as to aid in a computational model's ability to explain its reasoning, or to assist in a programs selecting and updating a wide range of problem solving strategies. As mentioned above, both of these somewhat more practically focused projects are also part of our core research, and are described in the sections immediately following this current one: the important point to be established here is the interdependence between the two. It is only within a theory of what knowledge representation is, and with reference to a conceptually coherent formal knowledge representation system, that we will ultimately be able to express and use our theories of how medical knowledge participates in intelligent medical reasoning. On the other hand, it is only with constant attention to the potential uses to which it will be put, that a powerful knowledge representation system can be designed.

The interplay between these two styles of investigation can be brought out by considering the following analogy: A knowledge representation project might be understood as providing the architecture for a "knowing", "reasoning", computational machine. Of course we are not building physical computers -- the point is simply that in an abstract sense the conceptual frameworks which we develop can be likened to abstract, conceptual, computational mechanisms. In terms of such a

metaphor, it is not within the scope of a knowledge representation project to say just what the programs are that should be written for the machine -- to do so would be to answer all of the unsolved questions in artificial intelligence research. But it is a view that we take -- a view which, at this general level, we share with other members of the artificial intelligence community -- that progress can, and indeed is, being made towards the formulation of just such a "machine architecture". As well as endorsing this goal, however, we also believe that we have an approach to such a design which will be successful in substantially new ways. In particular, our design is based on the notion of constructing a computational language in which the denotational import -- the *meaning* or *reference* -- of each and every symbol is made explicit within the formal system, in a manner in which it can be used by the processes which manipulate these symbols.

Our goal, then, is to provide the architecture for an integrated computational system both for the representation of general and medical knowledge, and simultaneously for the representation of reasoning procedures. The ideas on which it will be based will be made clearer by considering the history of knowledge representation research, and by articulating our approach in more detail.

2.2.1.2 The History of Knowledge Representation Research

In the long search for a convenient formalism within which to express and manipulate knowledge, two fundamentally distinct traditions can be identified. On the one hand, there are projects whose work with formal systems concentrates on the *manipulation* of formal symbols: we will say that these researchers are working within a *procedural tradition*. This tradition would include much of the work in computer science, such as the development of computer languages such as LISP [67, 73], SMALLTALK [37], SCHEME [118], PLANNER [49], CONNIVER [120], ACTORS [50], production systems [21], etc. Common to all of these approaches is a focus on *how it is that formal symbols can conveniently be manipulated*. What is ignored in such efforts (or is at least given much less prominence) is this issue of what those formal symbols mean. Thus one might have a carefully articulated search strategy which follows out a depth-first search of an AND-OR tree, but just what the AND's and OR's connect is not specified in the language. LISP, for example, neither restricts nor suggests what the S-expressions out of which one constructs data-structures might mean. This of course gives such languages extreme flexibility, but it does not help a potential user know what to encode in his symbol structures.

The second tradition we would call the *declarative tradition*. The effort here is on the *meaning* of symbols -- just what it is that the formal objects one constructs are meant to convey, mean, or signify. It is within such a tradition that attention has been focused on the role and structure of general concepts and relationships. Consider, for example, the many relationships between the following three objects: a specific case of nephritis, the general concept "nephritis", and whatever stereotypical scenarios that a doctor might have in his head of this disease. All of these would typically be explored and formalized by researchers within the declarative tradition. The variables provided in a declarative system might expressly be distinguished into those representing individual objects, those representing concepts, those representing relationships, etc.

The emphasis of a declarative approach, in contrast to the procedural one we identified above, is on *what it is that formal symbols mean*. Members of such a tradition would include both mathematical and philosophical logics, and the modern artificial intelligence "representation languages", such as KRL [5], FRL [39], OWL [46, 123], semantic nets [47], KLONE [9], etc. What is *not* provided in such systems are concepts in terms of which to talk about the reasoning processes

which manipulate the symbols.

2.2.1.4 The Development of an Integrated Representational Calculus

It is our belief that the approaches taken by these two traditions need not be kept so separate. In fact, we believe that it will only be by carefully integrating both kinds of concern that significant progress will be made. One of the members of our group, Mr. Brian Smith, comes to this work with past experience with research in both of these two traditions,[1] and is currently in the process of designing a new *representational calculus* as a demonstration of the power of integrating these two approaches. The calculus is founded on the notion of formalizing the denotational import of computational symbols, and is being developed to honor the following set of regulatory mandates:

a. A representational formalism must encode knowledge in such a way that the appropriate facts can be accessed quickly in appropriate situations. Care must therefore be taken to establish explicit structural relationship between the representations of different facts, which may be different from the encoding of the logical connections between them. In addition -- for reasons of generality, flexibility, and economy -- only a small subset of all possible facts can be stored: formalisms must be provided to encode general principles, and routines built to access these appropriately and to deduce their obvious consequences. Furthermore, a knowledge representation system should provide guidance in determining the most useful subset of all potential facts to represent.

b. All important aspects of the structure of a formal representation should be represented explicitly within the declarative formal language, rather than implicitly within programs or procedural code. This requirement (which is not met by such diverse languages as predicate logic, KRL, and semantic nets) is essential in order to ensure a host of desirable characteristics: that the resulting systems be self-descriptive, modular, explainable, self-analytic, modifiable, etc.

c. To the extent, in spite of the previous point, that programs and code constitute part of a representation system, those programs and code should be expressed in the same declarative descriptive language as the representational data structures. In this way descriptions of the active inferential processes of the resulting systems will be as readily accessible as facts in the main knowledge data base.

d. A formal representational system must be grounded on a denotational theory of semantics, so that the meaning and content of each constituent formal symbol can be justified and explained. An empirical claim being explored in our work is that some kind of denotational meaning is always attributed to a formal symbol by a programmer, in order for him or her to make sense out of the resulting system. If this is true (and we believe that it is), then unless the system is able to help the programmer articulate what

---

1. With Carl Hewitt and others in the design of the ACTOR model of computation and the PLASMA language embodying it [51], and with Terry Winograd and Daniel Bobrow in the design of the KRL representation language [6].

meaning is being attributed, and also help express this meaning explicitly within the formalism, then much of the content and meaning of the program will remain inaccessible to explanatory routines. Put another way, one could say that unless the representation system helps the programmer articulate the meaning of symbols, the programmer's understanding of the program will remain sufficiently vague and informal as to be of no theoretical or intellectual benefit to the research as a whole.

e. A representational calculus should provide for arbitrary meta-description and meta-representation (symbols standing for other symbols) within the same declarative language, in a manner integrated in such a way that whether a formal symbol is used directly or is described at the meta-layer is invisible to other components of the system. Such meta-description is essential in order to represent the explicit use of viewpoints, knowledge-specific advice, hypotheses, the decentralized interpretation of particular formal symbols, etc. For example, both the explanatory expertise explored in the next part of this section, and the problem solving strategies explored in the last part, are forms of meta-description being used for particular purposes. In addition, the ability to distinguish kinds of medical knowledge, such as physiological vs. epidemiological vs. anatomical, is crucial in the management of as large a body of facts as that represented by the medical domain. Without an ability to encode knowledge about how other knowledge is to be organized, used, and understood, only confusion could result from an expansion to a real domain such as even approximately complete medical knowledge about a given disease or organ system.

f. A formal system, while defining certain distinctions and relationships as primitive, must allow the user to define and use other new ones of his own invention. These new relationships should be have to bear the same ontological weight as those which are provided primitively in the language itself. For example, there are fundamental relationships between generic disease prototypes (such as the stereotypical pneumonia that one sees time and time again), individual cases of diseases (such as the pneumonia that is plaguing you right now), and the abstract individual concept of the disease (such as the pneumonia that was identified 2000 years ago by the Greeks). As researchers understand and formalize such relationships, they should be able to construct structures to represent them which have the same status as the primitive ones encoding the relationship between a generic concept and a particular instance. A consequence -- perhaps an inevitable consequence -- of work in representation is that one must bring to it a particular set of ontological, metaphysical, and epistemological biases. A representation system designer should attempt to avoid constraining the system so much that it becomes difficult for other people to use.

g. It is often powerful to describe a single individual in terms of more than one conceptual abstraction. For example, we mentioned above that you might want to describe the heart as "a pump" and also as "an organ of the thoracic cavity". Providing for this kind of *multiple description* allows great power and flexibility, although it introduces complexities in keeping track of inheritance paths. For example, if you want to know what tumors could be pressing on the heart, you would have the choice of looking into the thoracic cavity, or investigating the general properties of pumps. We, as people, know that the latter is absurd, but it is not always easy to control a program to do the right thing. A representation system should help in the identification of the right set of

concepts, in terms of which the *control of reasoning strategies* can be specified.

h. Some properties of general classes are not held by all known instances of those classes. Birds, for example, can in general fly, although penguins are birds which cannot. Representations of many concepts need to include *default* information, that is taken to be true of any instance unless there is specific reason to believe otherwise.

i. Different individuals sometimes share a common sub-part. The shoulder, for example, may be considered to be both part of the arm, and also part of the torso. A door may be part of a room, and also part of the hallway. If the larger objects are separately represented (the room and the hallway, for example), the representation system must be able to know that one and the same door is shared between them. In general, a representation system should provide flexible and sensible notions of individuals and identity, and allow for the shared mentioning of different individuals by different components of the system.

j. It is commonly agreed that it is possible to construct computer programs which, while they perform as claimed on a specific set of examples, nonetheless convey little or no deep theoretical understanding of any substantial subject matter. A representational calculus should fill a clear and well-understood role in the interaction between theories, models, interpretations, in two specific ways: not only should the calculus itself perform its task in a clean and economic way, but just what the task is of the representation system needs to be made clear and precise. It is only when both of these requirements are met that a system can assist the development of theoretically clean and rigorous systems.

k. A representational calculus should evidence the same computational tractability and organizational perspicuity as languages which have derived from the lambda-calculus (such as LISP, SCHEME, etc.), but at the same time bring with it a theory of denotational semantics of the kind that has been developed for the predicate calculus.

2.2.2 Specific Aims and Methods

The list of requirements given above sets some substantial goals for a representational calculus -- goals which would probably be unrealistic if our work in representation were not on a substantial footing already. In fact, however, Mr. Smith's work has already led him to the basic outline of such a representational calculus, and the specific goals we are setting forth are to extend and coalesce these ideas into a workable functioning system.

His design (known informally as "GOBLETS") is based on six primitive notions, which in turn can be broken down into three kinds of object and three kinds of relationship. The three objects are 1) that of an *individual* (such as Jimmy Carter or my left arm), 2) that of a *general structured concept* (such as the concept of a "heart" or of a "chronic disease"), and 3) that of a *set* or *collection* (such as the set of diseased glomeruli in my kidney, or the set of all people in the U.S., under the age of 25, who have had rheumatic fever). The three relations are 1) that of "*falling under*" or being *predicated by* a concept (Carter, for example, falls under the concept "President"; acute glomerular nephritis falls under the concept "disease"), 2) that of a symbol being *coreferential* with another (meaning that they are refer to the same thing -- the description "the current president of the U.S." and the name "Jimmy Carter" are two symbols which are coreferential), and 3) that of

one symbol *referring to* another symbol (the description "a fact whose certainty is only 25%" might, for example, refer to the assertion "Mary has arthritis").

From these six notions it seems that a complete declarative calculus can be built. There are several points of interest about this system: while the constituent entities are not dissimilar to those of standard logic, it nonetheless meets all of the requirements for "defaults", "taxonomic hierarchies", "procedural representation", "structured conceptual objects", that have been raised in traditional attacks on predicate logic as a representation language. In addition, it appears that the further development of the declarative aspects of the calculus will contribute to our understanding of such diverse issues as the function of the symbol "LAMBDA" in LISP and similar languages, some fundamental asymmetries between the concepts "AND" and "OR", etc.

One area within the declarative scope of the language which will receive our immediate attention is the representation of plurality -- sets, sequences, collections, etc. This is an area which has traditionally been particularly weak in its treatment in the declarative tradition, although within such exemplars of the procedural tradition as MACLISP [73] and INTERLISP [127] very powerful iterative procedures have been developed for reasoning with plural structures. A project we will engage in will be to take what has been learned in this procedural approach and fold it back into a more declarative scheme, while at the same time retaining its power as a (declarative) specification of useful procedures for reasoning with plural symbols.

Another focus of the research we are proposing here will be to discern just what is required to extend the calculus as it stands to a complete procedural, as well as declarative, formalism. As mentioned above, we will require that procedural specifications be written in the same declarative language as is all other knowledge. Nonetheless, in order for anything to happen, there must be some active primitives in the system out of which control structures can be built. A generalization of the notion of evaluation used in most applicative languages has already been roughly sketched, and will be investigated in depth. In addition, however, we anticipate requiring a primitive "DO" activity which, given a description of a (very very simple) activity, will carry it out. That we will separate the notion of "evaluation" from a notion of "doing" is part of the leverage we are bringing by integrating the two disparate traditions.

The GOBLET calculus, even as it stands, is presently under consideration for use by research groups on both East and West coasts; it has also been used in courses both at MIT and at Stanford. A certain amount of success has been demonstrated in its dealing with a set of complex issues regarding the declarative representation of general world knowledge. The urgent part of its development, which we are here proposing to carry out, is to carry the ideas through the procedural specification so that it can meet the goal of being a complete procedural calculus, as well as a declarative representation language.

2.2.3 Significance

The development of a useful representational calculus will be of substantial benefit not only to our own work in developing models of medical reasoning, but also to artificial intelligence research in general. In both cases, the benefits are several. First, and most importantly, a better understanding of the ingredients in terms of which knowledge is processed will prove invaluable in the clear and coherent articulation of the kinds of knowledge that play a part in clinical decision making. Secondly, a powerful theory of knowledge representation will allow the implementation of

a powerful knowledge representation system, which would enable us to build, test, and understand our computational models more easily.

Much of current linguistics focuses on the syntax of language -- on the structural components out of which communicative messages and expressions are built. What it is that is said is not so relevant to this kind of inquiry (although of course the two subjects are by no means completely independent). Research in knowledge representation can analogously be considered as research into the *syntax of thought* -- as an investigation into the structural components out of which rational thought processes are constructed. In the following sections we turn to a consideration of the sorts some kinds of "specific thoughts" that play a role in medical reasoning. The goal of the knowledge representation research is to formulate the correct ingredients for such higher level thought processes.

2.3 Techniques for Explanation and Justification

Recently, creators of expert consulting systems have recognized, like human consultants, that a crucial part of the consultation process is explaining the models and processes being employed in terms that the user can understand. There are several factors that make explanation particularly desirable within the context of medical decision making. For the foreseeable future, with very few exceptions, expert systems will be consulted by medical personnel rather than by patients, and to varying degrees, the physician and the program will share in the decision-making process. If the program is unable to explain the bases for its conclusions, to make clear the way in which the facts presented bear upon its recommendations, and the reasons for its rejection of other approaches to the management of the patient, a shared problem-solving process cannot be truly realized. In addition, a program that explains its reasoning may provide a valuable educational function.

Recognizing the value of explanation, several researchers have developed explanation facilities for their systems. These facilities have fallen into two general classes: those that produce explanations by examining the code of the expert system and traces of its execution [20, 121, 137] and those that produce explanations by assembling canned bits of text supplied by the programmer [4]. Unfortunately, these simple techniques fail in a number of important ways. This section of the proposal will discuss some of these failings and outline possible solutions.

Usually, when explanations are to be produced by examining the code, the expert system is programmed in a high-level language, and procedures and variables are given English-like names which are thought to be understandable by the system's users. To explain how a particular procedure works, the system merely reads back the code to the user, performing relatively simple transformations on the code to turn it into readable English. Usually, the system also keeps a record of the execution of the program so that it can answer questions about its actions for a particular case. For example, in the Digitalis Advisor's explanation facility [121] the interpreter kept an execution trace at the level of the implementation language. Thus, if a procedure computed a dose change for a patient, the fact that this procedure ran, what procedure called it, and the result of its execution were all recorded. To state how the dose change had been determined for a particular patient, the system examined the relevant high-level trace left by the system, converted it to English and displayed it to the user (see figure below). The other systems mentioned above are similar in flavor, with some key differences resulting from differences in the high level language employed by the system. For example, MYCIN [20] uses rules which are not given English-like

names, but the variables used are named in a similar close-to-English fashion.

Fig. 1. An Explanation of the Event of Checking for Sensitivity Due to Thyroid-Function

User: How did you check sensitivity due to thyroid-function?

The system responds:
I CHECKED SENSITIVITY DUE TO THYROID-FUNCTION BY EXECUTING THE FOLLOWING STEPS:

1. I ASKED THE USER THE STATUS OF MYXEDEMA. THE USER RESPONDED THAT THE STATUS OF MYXEDEMA WAS PRESENT.

2. SINCE THE STATUS OF MYXEDEMA WAS PRESENT I DID THE FOLLOWING:

2.1 I ADDED MYXEDEMA TO THE PRESENT AND CORRECTABLE CONDITIONS. THE PRESENT AND CORRECTABLE CONDITIONS THEN BECAME MYXEDEMA AND HYPOKALEMIA.

2.2 I REMOVED MYXEDEMA FROM THE DEGRADABLE CONDITIONS. THE DEGRADABLE CONDITIONS THEN BECAME HYPOXEMIA, CARDIOMYOPATHIES-MI, AND POTENTIAL POTASSIUM LOSS DUE TO DIURETICS.

2.3 I SET THE FACTOR OF REDUCTION DUE TO MYXEDEMA TO 0.67. THE FACTOR OF REDUCTION DUE TO MYXEDEMA WAS PREVIOUSLY UNDETERMINED.

2.4 I ADDED MYXEDEMA TO THE REASONS OF REDUCTION. THE REASONS OF REDUCTION THEN BECAME MYXEDEMA AND HYPOKALEMIA.

The quality of the explanations produced in this manner depends to a great degree on how the expert system code is written, since the explanation routines usually perform only simple transformations on that code. In particular, the basic structure of the program is not altered significantly, and the variable names in the explanation are basically the same as those in the program. If the explanations are to be understandable, the expert system must be written so that its structure is easily understood by anyone familiar with its domain of expertise, and the variable and procedure names used in the program must represent concepts which are meaningful to the end user.

This method of producing explanations has some advantages. It is relatively simple; if the right way of structuring the problem can be found, it does not impose too great a burden on the programmer; since the explanations reflect the code directly, consistency between explanation and code is assured, and hence the user may check the correctness of the code. Despite its advantages, this technique has a number of serious failings. It may be difficult or impossible to produce a program structure the user is familiar with. This may be due to structural differences resulting from the sometimes peculiar requirements of computer hardware, or more fundamentally, it may be because the power of an expert often *depends* on his ability to re-structure the problem in a new way that the user is unfamiliar with. This technique can only produce *one* explanation of a particular piece of code, yet a piece of code will often be used for more than one purpose. A multi-purpose piece of code should be explained in multiple ways. Additionally, a simple

restatement of the execution of a piece of code may not provide the clearest explanation of that code. Finally, compared with the amount of knowledge actually contained in a program, the amount of knowledge required to produce or completely explain that program is enormous. In his thesis, Long [63] shows that to write even a simple program for handling savings accounts a large number of models are needed to represent the domain, argument passing and control, data, I/O, and the target language. However, most of the information contained in these models never appears in the final program and cannot be accessed by the user. It is impossible to explain the underlying basis of the system without these additional models.

Another popular technique produces explanations by displaying canned phrases to the user. This method uses pre-formed packets of English text which are assembled and output by the computer. These packages might consist of entire paragraphs (as in some current versions of the acid-base-electrolyte program of Bleich *et al* [4]) or might consist of sentences and phrases assembled about the particular case. The physician may either watch passively, as this output is created, or may ask for further detail with certain key-word responses, for example "WHY". Since the explanations and code of such a system are not coupled, the programmer must remember to change the canned text every time he alters the program. If he forgets, the explanations will no longer correspond to the program's actual behavior. Furthermore, it is necessary to enter an answer for every question the user can ask. Clearly, as the system becomes large, this can become an onerous if not impossible burden.

2.3.1 Specific Aims for Explanation

We propose to construct a system capable of explaining the reasoning of an expert consulting program to a user unfamiliar with programming methodologies. This system will extend the capabilities of existing systems in a number of important ways. Multiple models of domain knowledge and computer programming techniques will be interwoven and structured into a hierarchy allowing the proposed system to give explanations at multiple levels of abstraction, and from multiple points of view. The system will be able to justify the ultimate implementation of the consulting program by examining refinement and debugging structures. If a user does not understand an explanation, the system will often be able to offer alternate explanations. The system will also be able to expand or summarize previous explanations in response to user requests.

We intend to use the Digitalis Advisor as our target expert consulting program and will implement an explanation system for it first. However, we expect that the explanatory techniques we develop for the Digitalis Advisor will be sufficiently general that it will be relatively easy to extend them to other similar consulting programs.

2.3.2 Methods of Procedure

This section outlines some of the structures and techniques we intend to use to give us the sort of explanatory capabilities outlined above.

2.3.2.2 Models and Modeling Techniques

A key element of this research will be the development of multiple structures for explanation. One group of structures will be a set of models containing knowledge about the domain. For example, these models would include a model detailing knowledge about the

physiological aspects of digitalis administration and a model of how expert cardiologists give digitalis. These models correspond to the domain knowledge a programmer would require in creating a Digitalis Advisor. The causal nets developed for the CASNET system [132] may provide an appropriate representation for this knowledge.

Another structure will be a typology of computer modeling techniques. When a programmer writes a program to perform a task in some domain, he employs various techniques to recast the task into a form that will be suitable for computers. That is, we can think of a program as a model of the performance of the original task in its domain. As is often true of models in general, the program will usually not be an exact representation of the task performance, but will employ certain modeling techniques or transformations which expedite the process of programming. The use of these techniques will often necessitate making assumptions and approximations so that a particular piece of the program may not correspond exactly to anything in the performance of the task by humans, and vice versa. For example, in the Digitalis Advisor, a weighted-sum scheme is used (along with other techniques) to determine whether or not the patient is getting better. We use this technique because we do not know exactly how doctors make that decision, yet using it involves making some assumptions, and the results that it gives will only be an approximation to the way a doctor makes that decision (although it may be a good approximation). A key step in justifying a program is to show that these transformations are reasonable, that is, that their use does not invalidate the program as a model of the task performance.

In examining the Digitalis Advisor, we have found that there are a small number of techniques (or models) employed over and over again. It seems that whenever a particular model is employed, it raises similar sorts of issues in terms of approximations and assumptions, and, therefore, in terms of justification. Furthermore, it seems that these techniques are not limited to the Digitalis Advisor, but are widely used. We propose to construct a typology of these "tricks of the trade" and the assumptions they raise which should prove very useful in providing explanations.

The third structure will be a refinement structure tracing the development of the expert program from general abstract plans to specific implementation details. This structure will be used in giving explanations at different levels of detail and for grouping explanations by topic. As we envision it, this structure will build on work by Sacerdoti [106]. He developed a hierarchy of plans for disassembling and reassembling an air compressor. The highest level plans provided a good overview, but they also contained some over-simplifications that could cause trouble. Lower level plans contained more detail and corrected the over-simplifications. Since users can only absorb a certain amount of new information at a time, it is often best to start giving explanations at a high level so that the user has an overview of the situation and is not overwhelmed by all the details of the system. Later, the details can be gradually filled in, and any over-simplifications introduced by the overview can be corrected.

2.3.2.4 Explanation Paths

The explanation facility developed for the Digitalis Advisor [121] often seemed to leave the user "hanging". That is, the explanations seemed to be incomplete because they were given in terms that the user was unfamiliar with, and they did not ultimately relate back to terms he was acquainted with. For example, if a physician asked how the measure of serum calcium was used, the explanation might say that the level of serum calcium was used to set the factor of reduction due to hypercalcemia to 0.75. Unfortunately, the doctor doesn't know what a "factor of reduction" is,

or how it relates to other outputs of the program with which he is acquainted, such as the dosage of digitalis. Some other explanation systems (notably MYCIN) share this problem of giving explanations in terms of rules which are artifacts of the implementation, but not necessarily useful to physicians.

Once we have a set of models, we may be able to deal with this problem by defining a set of *explanation paths* through the models which outline acceptable explanations. For example, if we require that an explanation begin and end with terms that the user is familiar with, then the system will not leave the user hanging in unfamiliar territory. Thus, in the example with serum calcium, a better explanation might be: "Since the level of serum calcium was above 10, the dose was reduced by 25%." In this explanation, the factor of reduction is not even mentioned. Since it is an internal variable, only its final effect on the dose is stated. Under some circumstances, it might be worthwhile to mention the variable if, for example, it was frequently used in the program or if an analogy with some previous explanation could be pointed out.

2.3.2.6 User Response and Strategies for Explanation

In the Digitalis Advisor, there was no way for the user to respond to an explanation he did not understand, and thereby obtain a clarifying explanation. While he could ask for a general explanation, and could then request more detailed explanations of parts, if he didn't understand the top level explanation he was out of luck. There was no way to indicate a lack of understanding to the system, and no way for the system to do anything about one. In addition to providing for general user response, an explanation system should have several strategies for formulating explanations, so that if a user doesn't understand an explanation as stated one way, the system can give him one from another viewpoint.

2.3.2.8 User Modeling

A user model is a representation of the system's knowledge of the user. It may include information about his preferences, what he knows, what he has been told by the system, and so forth. A number of systems [10, 12, 14, 15, 38, 135] have proposed or used user models. Most of these models have concentrated on representing what the user knows, and, to some degree, the user's preferences. This information is used to guide later tutorial efforts. We intend to use information in the user model in two ways. First, it will guide the production of the explanations, by noting user preferences (what he is thought to know and who he is, such as a doctor or programmer). Second, the system will use information in the model when trying to debug an explanation that the user doesn't understand.

2.3.3 Significance

The intent of this research is to find ways of representing programs and the knowledge that goes into creating those programs so that they can be explained and justified. The goal is *not* simply to write a digitalis therapy advisor that explains itself; instead, the research will show in general how the domain knowledge underlying a program may be represented and related to the program so that it is possible to explain and justify the program to a user familiar with the domain but unfamiliar with computers. This important capability is something which previous explanation systems have been unable to do. The key contribution of our particular research will be to show how multiple hierarchical models can be used to accomplish these goals.

2.4 Problem Solving Strategies

2.4.1 Objective

In attempting to arrive at a diagnosis or select an appropriate course of treatment, a physician may (though he is quite unaware of it) employ a wide range of approaches to solving the problem. In forming a diagnosis, for instance, he may first listen to the patient's comments and complaints, quietly drawing conclusions about the patient's condition and forming a number of hypotheses about possible underlying causes. He may then interrupt at some point (perhaps well before the patient has finished describing the problem), in order to ask a number of specific questions concerning something the patient has said, because those comments brought to mind one or more issues with significant diagnostic implications. If, when the patient has finished his description, the physician has not yet formed a clear hypothesis about the diagnosis, he may ask a number of general questions (e.g., "Have you noticed anything else?") intended to elicit any further useful information the patient may have. He may, alternatively, do a routine "review of systems", methodically asking a number of general questions about the various anatomical systems. Finally (although we have certainly not mentioned all the possibilities here), he may have developed a small number of specific diagnostic hypotheses and may proceed to ask a few carefully chosen questions whose answers will help to distinguish between the possible diagnoses.

As we view the problem, a central issue here is the techniques employed to guide the collection and use of information. For instance, When does the physician decide to interrupt, and how does he decide what question to ask? When is a review of systems necessary and when can it be omitted? When is it appropriate to begin asking questions intended to distinguish between diagnoses?

In more general terms, the issue is one of problem solving strategies: how do we know what techniques to use in solving problems? Since no approach has been found to be universally effective, for any range of problems we require a range of techniques. Even within a single problem, the diagnostician (problem solver) equipped with a wider range of techniques is often more effective.

The long-term objective here is to develop a better understanding of the variety of problem solving strategies that exist and the circumstances under which each variety is appropriate. This will be explored in the context of a program which would, in time, have the ability to adaptively change its strategy in response to new developments as work on a problem progresses.

2.4.1.2 Nature of the problem

With few exceptions, previous work on programs to do medical decision making has centered around the use of a single "hardwired" approach to the problem. [By "hardwired" we mean that the particular approach chosen (some of which are described in the next section) was built into the structure of the system, so that any desired change or variation in that approach would require fundamental changes to the basic system structure.] Systems like MYCIN and CASNET (described below), for example, are each designed around one particular problem-solving technique. While the technique chosen is often both reasonably powerful at problem solving, and convenient in terms of supporting other useful capabilities (e.g., explanation or forming a prognosis), there are fundamental limitations in building a system around a single approach.

There is, first, the issue of both power and breadth of application. Since, as noted, no single problem solving paradigm has proved to be universally applicable, nor even nearly so, no system with only a single approach is likely to be capable of handling a sufficiently wide variety of problems to be an effective clinical system. In addition, since clinical expertise appears to depend to some extent on the use of a range of different problem solving techniques on even a single problem, a program with only one technique may be insufficiently powerful.

A second issue involves acceptance of a system by the medical community. A system may be unacceptable because it functions in a fashion sufficiently unfamiliar to the user that he is puzzled by its behavior. Given the view of clinical problem solving outlined above, which emphasized the range of different techniques a clinician commonly uses, it is unlikely that any system designed around a single approach will produce behavior that appears familiar to the physician. A system may also be unacceptable because its single approach precludes certain capabilities deemed desirable. As we discuss in more detail below, a system which does solely "goal-directed" reasoning (i.e., the program asks questions and the user can only respond to the questions asked) is not capable of accepting information volunteered by the user, and seems overly restrictive in its mode of interaction. Similarly, a program which did solely "data-directed" reasoning (drawing inferences from volunteered information) might never reach a firm conclusion unless exactly the right information were volunteered.

2.4.1.4 Previous work

Relevant previous work includes two recent efforts at automating medical decision making (Mycin and Internist), and work on encoding and using strategies (Teiresias).

The Mycin system is an example of a program that makes very effective use of purely goal-directed reasoning. Its operation can be viewed as backward chaining of simple condition-action inference rules: The rules are invoked in a simple backward-chaining fashion that produces an exhaustive depth-first search of an and/or goal tree. Assume that the program is attempting to determine what bacteria might be infecting the patient. It retrieves (the pre-computed list of) all rules which make a conclusion about that topic (i.e., they mention bacteria identities in their action), and invokes each one in turn, evaluating each premise to see if the conditions specified have been met. This may result in the attempt to seek additional information (to confirm or deny a condition in a rule premise), at which point the process recurs, once again collecting all relevant rules about the new (sub)topic, and attempting to invoke each rule. When the program finds it has no way to deduce information about a given topic, it asks the user.

Information about a specific patient can thus be entered only in response to a question by the system, and can only deal with that question. Because the system does strictly goal-directed inference, there is no way for the user to volunteer information. This can be frustrating to a clinician seeking consultative advice who knows a fair amount about a particular case and is prepared to supply some particularly revealing information.

It has also been pointed out that the system's performance at times appears "unnatural" to the physician. This is often the result of the system's "single-minded" approach to problems, in which it methodically follows down a line of reasoning no matter how strongly the information it receives may be suggesting a different line of reasoning.

A number of important concerns motivated the use of this technique, and it does offer some interesting capabilities. For instance, in addition to supporting an interesting degree of problem-solving power in its domain, the backward-chaining technique makes it relatively simple for the program to supply explanations for its results and behavior. The use of simple inference rules also demonstrated a significant advance over earlier, mathematically based techniques (like Bayes' Rule), in terms of providing performance that was more comprehensible to the user. Despite these benefits, the restriction to a single problem-solving strategy appears to present a number of fundamental shortcomings.

Work on the Internist system has made some useful steps in overcoming the problems of using a single strategy. This system, intended to function as a consultant in internal medicine, is capable of accepting from the user a range of information describing symptoms the clinician has observed in the patient, results of tests, etc. As this information is supplied, the system forms hypotheses about the likely underlying cause(s). When all of the "volunteered" information has been entered, the system then begins asking questions, in order to gather additional information and arrive at its best diagnosis. The questions it chooses to ask are determined by one of several "modes" in which it can function. When there are just two competing hypotheses, for instance, the system attempts to find questions whose answer would tend to confirm one of the possibilities and disconfirm the other. If there are numerous hypotheses, the system asks questions whose answers would tend to eliminate one or more of the hypotheses, so that it can then concentrate on those which are more likely to be correct.

While we believe this to be a step in the right direction, our work (described below) is intended to approach the problem at a more fundamental level. (We will not, of course, be attempting to deal with a domain nearly so large as internal medicine, and will be focusing more narrowly on the control issue itself.) We believe, for instance, that the decision about which questioning mode to choose should depend on other factors in addition to the number of currently viable hypotheses.

The proposed research is in part an outgrowth of our previous work on the Teiresias system [20]. That system explored the application of meta-level knowledge to several topics, including knowledge acquisition (i.e., making it possible for the physician to "teach" the program) and the encoding and use of strategies. The work on strategies was embodied in the notion of "meta-rules", which were employed to guide the use of the rest of the rules in the system. A meta-rule might, for example, indicate that "in cases of pelvic abscess, rules suggesting the presence of one class of bacteria (enterobacteriaceae) should be used before those suggesting the presence of another class (gram-positive rods)" [because the first class is much more likely to be the cause of that particular kind of infection].

While this initial work was conceived and implemented only within the context of goal-directed invocation, it soon led to a number of useful observations and provided the foundation for a much more generalized view of the issues. The most fundamental idea was the observation that one body of knowledge (the meta-rules) could be used to direct and control the use of another body of knowledge (the ordinary, or "object-level" rules). While the initial implementation was quite simple, the basic idea is reasonably powerful and can be implemented in more powerful ways. This work also led to the notion of "content-directed" invocation and retrieval, and a more general view of what it means to retrieve and invoke the knowledge in a program [21].

This in turn led to a very preliminary design of a system which would attack the problem of knowledge retrieval and use at a fairly basic level, by having a body of (meta-level) knowledge about that very problem. That is, it would contain the expertise required for deciding how and when to apply its collection of object-level knowledge, and by applying this repeatedly over the course of problem solution, could vary its problem-solving strategy. It is the elaboration and implementation of this design that is the main concern of this part of the proposed work.

2.4.2 Specific Aim and Method of Procedure

The proposed work will proceed in several steps. First, we are formulating a taxonomy of problem-solving strategies, starting with heuristic search techniques. We have chosen this as a starting place first, because there is a range of different techniques, each reasonably well understood, and second, because there is a sufficient number of them that the appropriate organization for them may be easier to discover.

While starting with this restricted set of strategies will help make the problem tractable, it does not overly restrict the effort, for two reasons. First, a large number of common problem-solving strategies can be viewed as one or another variety of heuristic search (e.g., data-directed reasoning can be seen as "working forward", goal-directed as "working backward", etc.). Second, while this set of approaches will form the initial collection, we plan eventually to include a number of other kinds of strategies, since this initial set does not appear to limit the eventual capability of the system.

The strategies we plan to include will come from at least two sources. First, textbooks (e.g., [139]) and research papers describe a number of different techniques, primarily those developed in attacking traditional AI problems. Second, we believe that interaction with the clinicians on the project will also help to uncover a number of useful problem-solving approaches, since some previous work (e.g., [71]) has already demonstrated the rich collection of strategies used by physicians. While these latter appear to be domain-specific (i.e., limited to medicine), we believe they are in fact simply medical instantiations of much more general ideas. The notion of "review of systems", for example, is easily generalized to other domains, and in fact to any exhaustive, methodical framework for selecting questions about a topic.

The taxonomy derived from this work will form the foundation for the system, providing an organization for the "space" of possible search techniques. Preliminary work on this has suggested its plausibility, as we have developed a prototype taxonomy based on six "dimensions" of search characteristics, that provides a description of some 300 varieties of search.

The next step will be to formalize the body of knowledge used for selecting the appropriate search technique. When, for example, is it appropriate to used goal-directed reasoning rather than data-directed reasoning? When is a simple, methodical review (like a review of systems) appropriate? The answers to these questions will form a collection of "strategy-selection" expertise that will be used to guide the program's performance.

This body of knowledge will then be combined with the taxonomy of strategies and the resulting performance evaluated and used as the basis for further development of both the taxonomy and the body of knowledge. The primary piece of performance we are working toward is that noted in the introduction: the program should display the ability to adaptively change its

strategy in response to new developments as work on a problem progresses.

Since the body of knowledge is going to be collected from a range of different sources, we do not expect that the system will simulate any single individual's performance. Rather we will judge its behavior by asking whether it is both effective (does it solve the problem quickly and convincingly?) and plausible (does it behave in ways that clinicians find sensible and acceptable?) The answers to these questions will help guide our continued efforts to develop the system.

2.4.3 Significance of the proposed work

This work will be relevant to fundamental issues in both medicine and AI. Its significance to medicine derives from the issues noted above, of the drawbacks in any medical decision making system that is designed around a single approach to problem solving. We believe that a successful effort on this work will provide concepts that lead to medical consultation systems that are both more powerful than previous efforts and yet more acceptable to clinicians.

The significance to AI work arises from our concern with the fundamental issues of knowledge representation and control. In a sense, this work represents an interesting application of AI concepts to the field of AI itself, with expected benefits similar to those obtained in other domains. Previous applications of AI to medicine, chemistry [11], and mathematics [65] have demonstrated an interesting side effect: the attempt to formalize, represent, and apply knowledge about the application domain often led to a better understanding of that domain itself. It would not be unreasonable, then, to expect that this effort might have a similar effect, and lead to a better understanding of fundamental issues of problem solving strategies.

3. Acid/Base and Electrolyte Program

Personnel:

William B. Schwartz, M.D., Principal Investigator
Peter Szolovits, Ph.D., Investigator
Ramesh S. Patil, Research Assistant

3.1 Background

In recent years there has been a growing interest in the use of the computer as a means of providing expert advice on the diagnosis and management of acid-base and electrolyte disorders [4, 13, 36, 99]. Perhaps the most widely used and tested program for the management of electrolyte and acid-base disorders is that developed by Bleich [4]. Electrolyte and acid-base disorders occur frequently in hospitalized patients suffering from a variety of major illnesses which are in many instances life-threatening and usually curable if treated correctly. Thus a program which can deal with the considerable complexity of such disorders has promise of contributing importantly to patient care. Bleich's program, which is now widely available through a commercial time-sharing system is implemented using a *flow chart* technique and provides both a profile of differential diagnosis and also therapeutic recommendations. The successes of the flow chart approach in dealing with acid-base problems adds credibility to the view that the computer will have a growing and ever more important role, to play as a consultant on difficult clinical problems.

From careful examination of the nature of flow charts, and of the performance of programs such as those dealing with electrolyte and acid-base disorders, it is apparent, however, that there are many problems yet to be solved if we are to create a consulting capability which will closely mimic the performance of an actual consultant. In particular, the flow chart strategy

i. cannot deal with a chief complaint and cannot take a present illness in order to obtain a context for its analysis,

ii. must deal separately with each disorder and cannot integrate all abnormalities into a single coherent picture when there are multiple abnormalities and multiple etiologies.

iii. cannot treat the differential diagnosis in a probabilistic fashion.

iv. cannot evaluate the effects of therapy and respond to the evolving clinical situation with appropriate therapeutic recommendations.

v. cannot explain the exact lines of reasoning that it followed in arriving at its conclusions but can only provide standard text-book like material to explain the underlying activities of the program.

We would emphasize that all these problems are largely inherent to the flow-chart approach and do not represent failures on the part of the programmer. We propose here a series of strategies for breaking these constraints with the intent of creating a program for electrolyte and

acid-base disorders which will be able to carry out each of the functions just described in a fashion analogous to the physician/consultant. After describing our research proposal we will return to the flow-chart approach in order to document, with actual examples, the difficulties and limitations which we hope to resolve.

At the outset, we would emphasize that our concern is not with acid-base and electrolyte disorders per se. Our basic purpose is to use the domain of electrolyte disorders in order to determine whether it is now possible to develop an "artificial intelligence" program which is capable of dealing with all aspects of diagnosis and management at a level comparable to that of an expert.

3.2 Rationale

After gaining a great deal of experience with programs which take the present illness [85, 87, 122], perform sequential Bayesian diagnosis [41, 42, 112], and recommend therapy with a common and dangerous drug [43, 116, 121], we have identified a new medical area in which to focus many of our fundamental research interests: the diagnosis and therapy of acid/base and electrolyte disturbances. We have chosen this area for concentrated attention for the following reasons:

We wanted to explore more deeply the questions of diagnostic and therapeutic reasoning. Therefore, we needed a field in which detailed and sophisticated models of disease and treatment were known, but not so well understood that a complete model existed. In the acid/base and electrolyte field, detailed biochemical models of acid/base equilibrium and buffering, and qualitative models of renal and pulmonary interaction with homeostasis offer a rich background of knowledge for our research. At the same time, the field is neither so well understood that its problems are trivial, nor can any good computational model easily account for all difficulties.

A program in the same medical area but relying on an older computer methodology has existed for a number of years [4] and can serve as both a realistic guide for minimum performance requirements for our eventual program and a point of comparison for investigating the differences of style dictated by our A/I approach. Dr. Bleich has made his program available to us in the past, both to try on cases we considered pivotal for our own work and to consider its breadth of coverage. We emphasize that our work is not an attempt to outperform that program by building another one. We are studying the thinking style and content of expert medicine, and are using the acid/base and electrolyte domain for our study.

Dr. Schwartz is a noted expert in the field of acid/base and electrolyte disorders. Thus, we are able to draw on excellent models of how to approach problems in that domain from one of its leading practitioners. Further, if an effective program should ultimately result from our efforts, it would be based on a level of medical expertise not usually available to the general practitioner.

3.3 Overview

The objective of a program in the diagnostic and treatment domain is "the proper management of the patient." That proper management consists of collecting the relevant information about the patient, identifying the disease process(es) responsible for the patient's illness,

and prescribing a proper course of actions to correct the patient's condition. One of the complexities of the task is that its subcomponents do not have well defined boundaries. This is because the patient may be presented to a clinician at different stages of a diseases evolution and treatment. During the course of management new information about the past history may become necessary as the diagnostic hypotheses evolve. The current diagnosis may also depend on information that is presently unavailable (e.g., serum-electrolytes that have been drawn but not reported by the laboratory). Moreover, the disease itself may evolve through time providing additional clues to its identity. Further, at times the response to a certain treatment itself may be the best clue to the diagnosis. Therefore, the clinician must choose the next course of action from a large range of alternatives, which can be broadly classified as gathering more information (much of which may turn out to be irrelevant in the current clinical evaluation), ordering further tests (involving possibly expensive time delays and/or clinical costs), waiting for further development, prescribing therapy or some combination of the above. Thus at every stage of expert consultation the program must be able to choose between the alternative actions with the objective of maximizing the utility to the patient. This can be achieved in a computer program only by developing a system capable of diagnosis, therapy and making decisions between various alternatives available to a physician during patient care. With this in view we propose to design the Acid/Base and Electrolyte Consultant system which will address each of the above mentioned aspects of diagnosis and management. Further, in keeping with our objective, we have tried to separate and modularize different components of a physician's knowledge and expertise so as to be able to evaluate our understanding about each and their interactions. This should also allow us to further experiment with, redesign, and implement any component of the system without having to reimplement the entire program again.

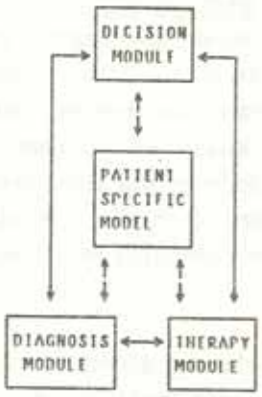


Fig. 3.1. A Schematic for the Overall ABEL System

The Acid/Base and Electrolyte Consultant system will therefore consist of four major components: i) the Patient Specific Model, ii) the Global Decision Making component, iii) the Diagnostic component, and iv) the Therapy component. The patient specific model describes the physician's understanding of the state of the patient at any point during the diagnosis and management. This patient specific model is used as a central data structure with which other components of the system may reason. The global decision making component is the top level program which has the responsibility of calling the other programs with specific tasks. In general the global decision program will call the diagnostic program with a task such as: take the initial history, elaborate some specific diagnosis, etc. The diagnosis component then performs the specified diagnostic task and reports the results to the main program. It also modifies the patient specific

model to reflect the revised state of the patient. Similarly if the global decision making program calls the therapy selection program, it attempts to formulate a set of alternate therapies for the patient along with a check list of items that must be tested before any specific therapy can be recommended. It also identifies information that will help discriminate between alternate therapy recommendations. This information then is reformulated and sent to the diagnostic program as the next problem to be solved. This approach of separating diagnostic, decision-making and therapeutic activities allows us to make explicit the decision making that goes on in a physician's reasoning: e.g., is further diagnosis necessary, what treatment should be selected, should we wait before prescribing further treatment, can we choose some therapeutic action that would also provide diagnostic information thus making further diagnosis at this point unnecessary, etc.

Although the therapeutic capability of the program should ultimately be the most important from a clinical viewpoint, it is the development of the patient specific model and the diagnosis subprogram which first claims our attention. Because diagnosis must encompass at least past evidence of therapy, most of the conceptual problems of therapeutic evaluation occur in a diagnosis program even when decoupled from its eventual therapeutic partner. Therefore, we have concentrated first on understanding the patient description and diagnosis components of the above system outline. Our work on therapy is only sketchily drawn and the problems of exactly how to interface diagnosis and therapy are understood only in a general form.

3.4 Diagnosis

In this section we are dealing with diagnosis, i.e., the process of actively seeking information and identifying disease process(es) causing the patient's illness. In other words ascertaining *what the facts are* and *what the facts mean*. The process of active information gathering is dependent on our understanding and analysis of the available facts. From protocol analysis, researchers [28, 55, 71, 103] have observed that doctors generally use a set of hypotheses (tentative diagnoses) to organize and search for new information efficiently. Therefore, it seems reasonable to organize the program around a problem solving paradigm using hypothesis generation mechanisms. The use of hypotheses will allow the program to organize sets of findings specific to a patient in a small number of chunks. It will also provide the program with a mechanism (commonly shared with the clinicians) to communicate its "thought process" to the clinician.

From our experience with the existing diagnostic systems and a careful evaluation of the Present Illness Program [85, 103, 122], we are convinced that a relatively simple representation of what a physician knows about the patient (the physicians understanding about the patient's illness and the current state of the hypotheses about a patient's condition) will not be able to provide the program with the desired level of expertise. The patient description must unify all the known facts about the patient, their suspected interrelationships, the hypotheses and how the hypotheses account for various known and hypothesized findings. As the physician's knowledge is expressed at various levels of detail, from deep physiological and causal knowledge to global knowledge about syndromic and phenomenological correlations, we need a representation scheme that can unify these descriptions. We also need mechanisms for moving from one level of description to another. We have developed a patient specific (description) model and associated knowledge representation scheme which attempts to construct a hierarchic description with the above objectives in mind. This is the central patient specific data structure of the system.

Focusing our attention on the interaction between the hypothesis generation and the question selection process, let us assume for the time being that we have the capability of generating a small number of hypotheses given a set of findings. Now let us consider a program which alternates between asking a question and generating from scratch a new set of hypotheses after each new fact is entered. The regeneration of a diagnosis with every new fact is inherently inefficient, it requires a large amount of information processing (much of it repetitive) and it lacks the continuity of thought process so essential to human problem solving. This problem can be avoided by using a hypothesize and reformulate paradigm where a set of initial hypotheses is generated from the chief complaint and initial patient information; these hypotheses are then reformulated and extended to incorporate new findings acquired during the active information gathering process.

The task of diagnosis ideally involves confirming the appropriate hypothesis (or hypotheses) and eliminating other competing hypotheses. Note that multiple hypothesized diagnoses arise due to uncertainty and lack of knowledge about the patient. Further, the uncertainty is reflected in the patient description by states with a low belief-value (score) and the lack of knowledge is reflected by multiple competing interpretations. Now the task of the diagnostic problem solver can be reformulated as that of identifying and eliminating the uncertainty and lack of information in the patient description. We propose to do this in the following manner. All the states (in the patient description) with low belief factors and multiple competing interpretations will be collected, forming a problem-set. This problem set will be used to formulate the top level diagnostic goal. A plan for problem solving will be generated by decomposing this goal successively into subgoals in the context of the patient description. Each goal in the plan will be associated with the parts of the patient description relevant to it and the prior expectations about the outcome of the problem solving effort. The association between the goal structure and the patient description will allow us to separate patient specific information relevant to the immediate diagnostic problem from information not directly relevant. The expectations associated with the goal structure will provide us with the context in which to evaluate the incoming information for discrepancies.

The decomposition of the diagnostic goal into subgoals will be done using various diagnostic strategies based on the protocol analyses by Miller [71], Kassirer [55] and Elstein [28]. The strategies are Confirm, Rule-Out, Differentiate, Group-and-Differentiate, Refine and Explore (the first three strategies have also been used in Internist, but their application is quite different). The choice of an appropriate strategy for a given situation will be based on the number of hypotheses in the primary-goal and their relative belief values.

3.4.1 Representation of Medical Knowledge

The knowledge used by a diagnostic program can be divided into two classes: knowledge about diseases and their presentation in a patient, and heuristic problem solving knowledge. In this section we will discuss the representation of medical and patient specific knowledge and develop a patient specific model. As the two representations share general descriptive schemes, we will deal with them concurrently. The representation being discussed will be specifically indicated if it is not clear from the context.

Illness can be described as a change in the normal state or function in a patient. To describe an illness, we need a formalism to represent the states, the state changes, the normal and the abnormal functions and their interactions in a patient in terms of the primitives known to the

system. It is also important to recognize various composite situations in order to get a global perspective of the patient's illness. Recognition of these situations or compound descriptions in a diagnostic system is important because it provides us with the ability to reason at a high level of abstraction, organizing a large number of seemingly unrelated facts; more importantly, we may use clinical and diagnostic knowledge which is generally organized around high level concepts. Therefore, the diagnostic system should allow descriptions to be in terms of both the high level concepts such as diseases and the detailed physiological states and processes underlying the presentation of the illness in a patient.

3.4.1.2 State

Following Forrester, we say that the states in the system describe two kinds of variables: levels and rates [32]. The levels are the accumulations within the system. They represent the present values of those variables that have resulted from the accumulated differences between the inflows and outflows. Rates define the present, instantaneous flows between the levels in the system. The rates correspond to activity, while the levels measure the resulting state to which the system has been brought by the activity. A state in the system is described by its temporal characteristics, severity and other aspects relevant to the state. A state is a "primitive-state" if it does not contain internal structure and is a "composite-state" if it can be defined in terms of other states.

One important function of diagnostic reasoning is to relate causally the diseases and symptoms observed in a patient. These causal relations play a central role in identifying clusters that can be meaningfully aggregated and in developing coherent diagnoses. The presence or absence of a causal relation between a pair of states can change their diagnostic, prognostic and etiologic interpretations. Therefore, the system should have the capability of hypothesizing about the presence or absence of a causal link. We also note that, in a manner similar to state descriptions, a causal relation can be described at various levels of abstraction.

3.4.1.4 Link

A link specifies the relation between two states. In the past (PIP, INTERNIST, GLAUCOMA), links were used to describe causal relations between states. From our study, we have come to the conclusion that this single representation of the interaction between states is inadequate, because this representation forces us to assume that every interaction between states is causal (or statistical) in nature. We believe that the interaction between states occurs at various qualitatively distinct levels. For example, two states may be causally related to one another, or they may be associated with one another in a statistical sense, without any known causal relation between them, or the presence of one of the states may alter the interpretation of the other state without changing the likelihood of the other state in any significant, predictable way. To capture these differences, we will use three types of links described below. Further, we believe that a link between a pair of states in the medical data base may or may not be present in a given patient (for example, although hypotension and acute tubular necrosis are causally related in general, they may not be related in a given patient.). Therefore, in order to reason with relations between a pair of states in the patient specific model, we must instantiate links and incorporate them in the model.

Causal link: A causal link specifies the "cause-effect" relation between the "cause" (the antecedent) and the "effect" (the consequent) states. In the past (PIP, INTERNIST), causal links were described by specifying the type of causality (may-be-caused-by, complication-of, etc.), and a number

representing in some form the likelihood (conditional probability) of observing the effect given the cause. From our study, we have noted that the conditional probability of observing an effect given the cause or vice versa, depends upon various aspects of the cause, such as severity, duration etc. as well as other factors in the context in which the link is invoked (such as the age, sex, weight, etc. and the current hypothesis about the patient). For the effective use of a causal relation, we need to take these conditions into consideration. To illustrate this, let us consider (a simplified) causal relation between diarrhea and metabolic-acidosis. In terms of conditional likelihood, we could state the relation as P(Metabolic-Acidosis/Diarrhea) = 0.7 or the probability of observing Metabolic-Acidosis given Diarrhea is 0.7. A rule-based description of the causal relation can be specified as follows;

```
IF DIARRHEA IS SEVERE AND ITS DURATION IS GREATER THAN TWO DAYS
   THEN
     IF THE PATIENT HAS NOT RECEIVED BICARBONATE-THERAPY RECENTLY
        THEN THE PATIENT MAY HAVE MODERATELY SEVERE METABOLIC-ACIDOSIS WITH NORMAL ANION-GAP
        ELSE THE PATIENT MAY HAVE A MILD METABOLIC-ACIDOSIS WITH NORMAL ANION-GAP.
```

From the above example it is apparent that the conditional probability of observing metabolic-acidosis and its severity and duration depend on the severity and duration of diarrhea and bicarbonate-therapy. Generalizing, it appears reasonable to expect that causal links between the cause and effect nodes should contain the information on how an instance of the cause relates to an instance of effect and other factors influencing the relation.

A causal link in the system is an object denoting the causal relation between a cause-effect pair. It specifies a multivariate relation between various aspects of the cause and the effect, taking into account the context and the assumptions under which the causal relation is being instantiated. A schematic description of a causal link is presented in figure 3.2.

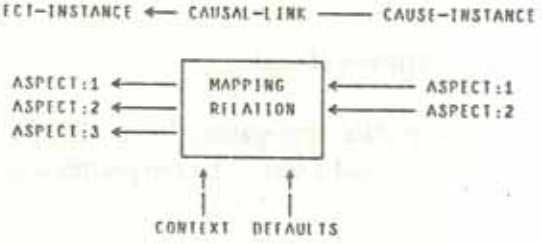


Fig. 3.2

Associational-links: This link states that the presence of one state influences the expectation about the presence or absence of the other state. It suggests that the two states are correlated, but does not specify the reason for the correlation or association between the states. For example:

HYPONATREMIA — ASSOCIATED-WITH → CENTRAL-NERVOUS-SYSTEM-SYMPTOMS

Fig. 3.3

it is well known that severe hyponatremia is associated with central nervous system disorders such as coma, stupor and confusion. But the specific physiology for the causation of these disorders is not well understood.

Grouping-links: Grouping links state that the presence of two states simultaneously (in the patient specific model) represents a situation which is recognized as a part of some useful abstraction. This

link does not imply any correlation or causal connection between the states connected by the link. In other words this link is used to group together states without any commitment to their mutual cause-effect relation. For example, let us consider the following group of symptoms, severe hyponatremia, low creatinine (less than 0.6) and normal bicarbonate.

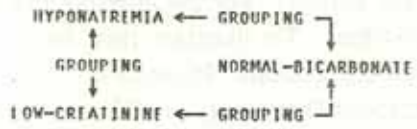


Fig. 3.4

This group of findings is strongly suggestive of SIADH (Inappropriate secretion of antidiuretic hormone). They are grouped together in the program, which can then be used to suggest SIADH at higher levels of aggregation.

3.4.1.6 Hierarchic Description of States and Link

In this section we will discuss a hierarchic representation scheme for the description of the patient specific knowledge. In this representation, the low end of the hierarchy describes the physiological knowledge about the patient in terms of primitive physiological concepts and relations known to the system. This physiological description of the patient is then successively aggregated into higher level concepts and relations, gradually shifting the emphasis from physiologic to syndromic description or, in other words, from causal to phenomenological descriptions. The phenomenological nature of the aggregate description allows us to use efficiently the hypothesize and test paradigm [77] for global problem solving. The causal nature of the detailed description allows us to use causal reasoning and to restrict the number of hypotheses generated to a small number by imposing causal and physiological consistency requirements. We will first introduce the state and link aggregations with the help of examples.

3.4.1.6.1 Example I: Hierarchic description of states

To illustrate the concept of a state aggregation, let us consider the condition of excessive loss of lower gastrointestinal fluid (Lower-GI-Loss). The compositions of the lower-GI-fluid and the plasma fluid are as follows.[1]

	LOWER GI FLUID	PLASMA FLUID	
Na	100 – 110	138 – 145	mEq/l
K	30 – 40	4 – 5	mEq/l
Cl	60 – 90	100 – 110	mEq/l
HCO3	30 – 60	24 – 28	mEq/l

Fig. 3.5

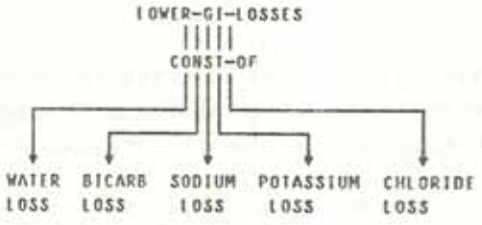


Fig. 3.6

In comparison with plasma fluid, the lower GI fluid is rich in bicarbonate and potassium and is

---

1. In this section the following abbreviations are used. CBY (caused by), SBY (succeeded by or followed by), CONST-OF (constituent of), and COMP-OF (component of).

deficient in sodium and chloride (except in case of hyperchloremic diarrhea). This information is represented in the knowledge base by decomposing lower-GI-fluid into its constituents (and associating appropriate quantitative information with the decomposition) as shown in fig. 3.5. Now the loss of lower-GI-fluid (lower-gi-losses) would result in the loss of corresponding quantities of its constituents (in relation to the total quantity of fluid lost) as shown in figure 3.6 (the quantitative relation is not shown for reasons of clarity). Therefore, an excessive loss of lower GI fluid without proper compensation of electrolytes will result in hypobicarbonatemia, hypokalemia, hyperchloremia, hypernatremia and volume-loss as shown in fig. 3.7.

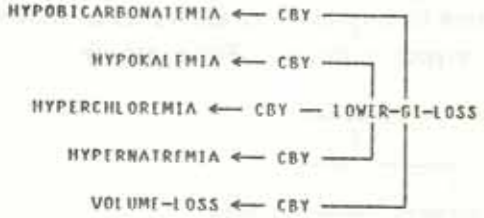


Fig. 3.7

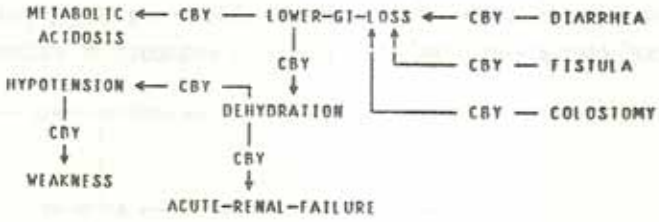


Fig. 3.8

At the next level of detail we can describe the causes and consequences of lower GI loss as shown in Fig. 3.8.

3.4.1.6.2 Example 2: Hierarchic description of causal relations

To illustrate the concept of a causal link abstraction, let us consider the following aggregate causal assertion.

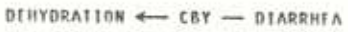


Fig. 3.9

The causal mechanism of diarrheal-dehydration can be explained as follows; diarrhea causes lower gastrointestinal fluid loss (Lower-Gi-Loss) which causes dehydration.

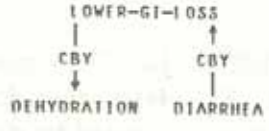


Fig. 3.10

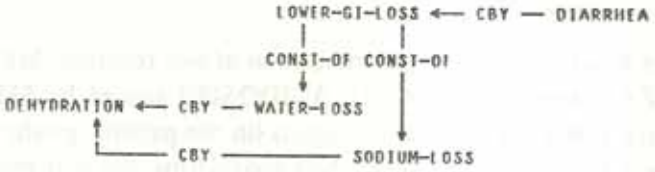


Fig. 3.11

In greater detail, the lower gastrointestinal fluid loss consists of water and sodium loss (along with other electrolytes not included here for the simplicity of presentation) which results in dehydration under either of two conditions: a) the water loss is not appropriately compensated by fluid intake; b) the sodium loss is not appropriately compensated, resulting in an inability to retain the water in the fluid intake as shown in fig. 3.11.

3.4.1.8 Types of Aggregations

The hierarchic patient-description structure is built by aggregating low level concepts into higher level concepts or by elaborating high level concepts into their constituents at lower levels. Aggregation allows us to summarize the network of cause and effect nodes describing the patient's

illness by substituting aggregate nodes describing them for large chunks of the causal networks and by retaining only the important nodes (analogous to landmarks in a conceptual map [59]). The aggregations and elaborations can be divided into five classes, described below.

Temporal-Aggregation: An observation provides us (clinician or the program) with a snap-shot of the state of some process. During the course of evolution of a patient's illness, a sequence of snap-shots becomes available to the program. These snap-shots describe the progress of a process through time. Temporal aggregation allows us to resynthesize the process and describe it as a single concept in the patient specific model. Further, temporal aggregation allows us to summarize a sequence of values of a state variable into a single qualitative description and allows us to make hypotheses about temporal patterns or sequences of values we expect to observe. For example:

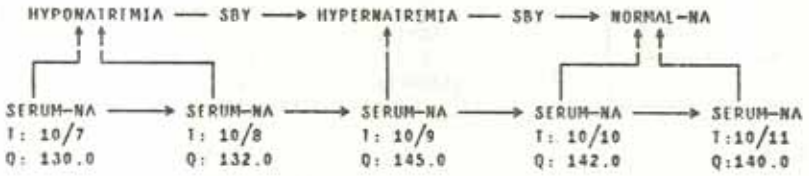


Fig. 3.12

Component-Summation: Component summation becomes necessary in our program because the causal links in the system contain specific quantitative information on how the various aspects of cause and effect instances (in a given patient) relate to one another. Therefore, the matching of the cause-effect relation will be unsuccessful or undesirably affected if the aspects of the effect instance are distorted by other nodes. In order to prevent this from happening we must instantiate the effect of each cause exclusively and then combine them together. This is illustrated by the following example,

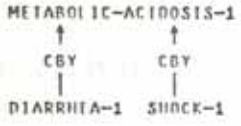


Fig. 3.13(a)

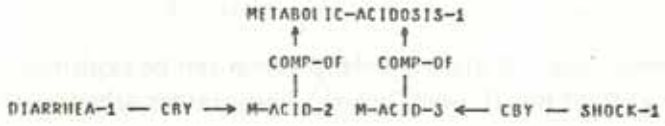


Fig. 3.13(b)

The above description is a superimposition of two relations: METABOLIC-ACIDOSIS-1 caused by DIARRHEA-1 and METABOLIC-ACIDOSIS-1 caused by SHOCK-1. The description does not explicitly state that the metabolic-acidosis (in the patient specific model) is partly caused by diarrhea and partly by shock. Therefore while considering the first relation, the program will assume that the entire metabolic-acidosis is being caused by diarrhea. This will result in the cause-effect relation being mismatched (because the severity of acidosis and anion-gap associated with it are larger than can possibly be attributed to diarrhea). A similar mismatch would also occur while considering the relation between the shock and the metabolic-acidosis. To avoid this problem let us consider the description of the example as shown in fig. 3.13(b). The state description of fig 3.13(a) can be considered as an abstraction of fig 3.13(b), where METABOLIC-ACIDOSIS-2 and METABOLIC-ACIDOSIS-3 are respectively the components of METABOLIC-ACIDOSIS-1 caused by DIARRHEA-1 and SHOCK-1. The decomposition of an effect with multiple causes into its causal components provides us with valuable information in evaluating prognosis and in formulating therapeutic interventions.

Taking another example, metabolic-acidosis could be considered to be hypobicarbonatemia causing a reduction in pH, which causes hyperventilation and reduced pCO2 which in turn causes an increase in pH, an example of negative feedback. The increase is less than the initial reduction, causing a net reduction in pH.

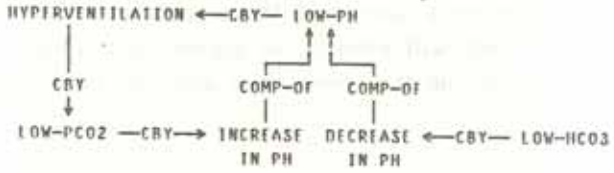


Fig. 3.14

The component summation mechanism allows us to separate the primary component of the change from the secondary feedback components in feedback loops and allows us to fold causal chains in feedback loops that represent continuous processes. On the other hand, decomposition of a state into its contributing factors raises a new problem, that of combining the factors together. The combined effect of two components may not be additive and may depend on their causes and on the physiological mechanisms involved in the particular case. This problem needs further in-depth study, but for the time being let us assume that there is some (in all likelihood a local) mechanism that allows us to combine the contributing factors together satisfactorily.

Constituent-Aggregation: Many diseases and conditions have some internal physiological structure, or a set of physiological conditions that can be considered as definitional for the disease or condition. This knowledge is represented with the help of constituent-aggregation or disaggregation. An example of constituent aggregation is shown in fig. 3.6.

Causal-Aggregation: Some of the possible groups of states are conceptualized and have names describing them as conceptual objects. In some sense, these concepts provide the structure for organizing different situations that are commonly encountered during diagnosis. Because of historic evolution in the recognition of situations and their nonstandard use, terms such as disease, syndrome, physiological states etc. have overlapping definitions. But the existence of these different terms signifies the existence of different types of clusters. We will try to identify the basic structure of these clusters without differentiating between the specific mechanisms responsible for their differences, mostly because we do not have sufficient knowledge in the problem solving domain to exploit the subtle differences between them. An example of causal aggregation is shown in fig. 3.8 .

Link-Aggregation: Links are used to represent causal relations between states. The causal relations are understood and can be described at different degrees of specificity and detail. This has been illustrated in example 3.2. The link aggregation hierarchy represents an alternate hierarchy to the state aggregation hierarchy, and provides us with a different view point in summarizing the patient description by allowing us to identify and eliminate intermediate states from causal chains of reasoning. They also help us in evaluating the belief in the causal link at higher level of aggregation using their detail descriptions.

3.4.2 Representation of Uncertainty

In previous sections we have discussed the representation for describing illness in a patient assuming that the true state of the patient's illness is known. But in reality the knowledge about the patient is incomplete and uncertain. Its incompleteness gives rise to more than one possible (hypothesized) diagnosis for the patient's illness. In this section we will study two major ways in which this uncertainty appears and will extend the patient description using the concept of hypothesized description to account for the uncertainty and the possibility of more than one hypothesized diagnosis.

Uncertainty in the patient description appears in two related but distinct ways. First, because of incompleteness in our knowledge about the true patient description (ignorance) and second, because of the inherent chance nature of the disease process. To illustrate the differences between the two, let us consider a patient with severe diarrhea and vomiting. The above findings are sufficient to assume that the patient may have metabolic disturbance, but not enough to specify the type of disturbance (which depends upon the relative amounts of fluid lost due to diarrhea and vomiting). This can be represented by specifying a high level of belief for metabolic disturbance and not specifying any belief value for metabolic acidosis or metabolic alkalosis, thus preventing the program from making any inference based upon the relative likelihoods of metabolic acidosis and alkalosis. Thus we note that a given level of ignorance of the program can be represented by asserting concepts at an appropriate level in the classification hierarchy.

The problem of evaluating the measure of belief in a given hypothesis is a difficult one. Various people have worked on this problem [25, 113, 114, 124, 141], but no good practical solution has been found because the simplifying assumptions made in the most often used probability theory (e.g., those of completeness of the universe of discourse, conditional independence of events etc.) are not valid here. In addition the issues of the incompleteness and uncertainty of information have often been confused. We wish to separate these two issues in order to exploit the capabilities of AI techniques, which are much better at handling incomplete information (by hierarchic description) than dealing with uncertainty. This would allow us to reduce our dependence on less than adequate schemes available for dealing with uncertainty.

3.4.2.2 Hypothesis

A hypothesis is a supposition or conjecture put forth to account for known facts [131]. Stated differently, a hypothesis can be considered to be a meta level concept describing a relation between two objects and specifying the type of relation (being hypothesized between the two objects) and our belief in this relation. To illustrate the various types of relations possible, let us consider the following three hypothesized relations:

Type-1: the patient has hyponatremia,

Type-2: the hypotension of the patient is caused by hyponatremia, and

Type-3: the symptoms observed correspond to acute glomerulonephritis.

The type-1 hypothesis expresses our belief in a particular state being the true state in the patient. It is used to describe the degree of belief in a state-description (or diagnosis) at any level of abstraction. The type-2 hypothesis describes the degree of belief in a link between two states. Note that the degree of belief in the link is independent of our belief in the states associated with the link. Therefore, it is possible to construct a situation where the belief in the link is high while the belief in the states being linked may be quite low. This would correspond to holding the point of

view that neither of the two states are likely, but that if they are the true states, then it is very likely that they are related by the link represented in the hypothesis. The type-3 hypothesis expresses our belief in the appropriateness of an abstraction. In other words, it expresses our belief in the matching between a state-description and its abstraction, i.e. diagnosis. From type-1 and type-3 hypotheses it becomes apparent that two different belief values can be associated with a given diagnosis: these are, our belief in the appropriateness of a diagnosis given the patient facts and our belief in the patient facts.

In this section we have developed a descriptive mechanism to represent a hypothesis and uncertainty about the hypothesis. Ignorance about the patient facts is represented structurally by selecting the level at which a hypothesis is made. Uncertainty about the hypothesis is described explicitly by associating a belief measure with the hypothesis. In the next section we will extend this to the patient description structure in generating a set of competing hypotheses to explain the patient's illness.

3.4.2.4 Complete Hypothesis

At any point in the diagnosis the clinician (program) has a partial understanding of the illness of the patient. This understanding can be described by a set of hypothesized diagnoses, where each hypothesized diagnosis accounts for all the observed manifestations of the illness. In the program, a diagnosis is represented by a patient description structure (described in Section 3.4.1). Each such patient description structure with its associated hypothesis structure is called a "complete hypothesis" and abbreviated as "CH". Note that each complete hypothesis provides an alternate explanation for the disease process in the patient and only one of these complete hypotheses can be correct. Thus we have a set of alternate diagnoses which are mutually exclusive and competing; therefore, they can be rank-ordered according to their likelihood.

At any point in the diagnosis there are only a few significantly distinct explanations for the patient's illness, although each such explanation can have a substantial number of variations. This problem can be contained by selecting an appropriate level of detail for representing the patient description. This should allow us to represent small differences in the hypotheses implicitly (by professing ignorance about them) while focusing on the major differences. Again, during problem solving we will need to compare the different alternatives (complete hypotheses) to identify the differences between them. If each complete hypothesis is represented separately, this task can become substantial. This problem can be overcome by allowing different hypotheses to share common sub-hypothesis structures, thus producing a single structure to represent the set of alternate CH's in which the important differences move up the structure, while the smaller differences tend to be buried deep inside the structure.

3.4.3 Description Building Operations

During the design of the program, we have tried to separate the data structures required for the representation of a patient description and those required for efficient problem solving. This allows us to represent the patient description, incorporating all the available findings and derived facts about the patient with the objective of semantic clarity, coherence, completeness and explainability. In the preceding section we have studied the structural organization of the patient specific model. In this section we will outline the operations used to create and maintain the patient specific model as new information is added to it.

In this section we will study the operations for creating and augmenting the patient description. These operations are INITIAL-FORMULATION to create an initial patient description from the presenting complaints, initial findings and lab-results; AGGREGATION to combine various findings into causal clusters representing different disease hypotheses (moving up); ELABORATION to decompose aggregate findings and hypotheses into their components or specific subclasses (moving down); and PROJECTION to hypothesize associated findings and diseases suggested by states in the patient description at the same level of abstraction (moving sideways).

Initial-Formulation: From the observation of the clinical behavior of doctors, we have noticed that the initial response of a physician to the chief complaints is predetermined and depends upon the "clinical situation". On further pursuing this observation, we noted that the number of initial situations (i.e., the way in which the patient is initially presented to the physician) is not large. For example, in the domain of acid-base disturbances there are about 30 such situations. Therefore, it is feasible to compile a set of initial patient descriptions with situation specific information such as expectations, causes, consequences, probable diagnosis and initial set of exploratory questions (similar to playing the initial chess game using book moves). This will allow the program to set up a specific framework around which the incoming information can be organized and thus avoid a serious difficulty encountered by programs using hypothesize and test paradigm. That is, when there is very little information or when the available information is nonspecific, a large number of possible hypotheses are activated (for example, in PIP, the first few facts entered could trigger as many as half of all the possible hypotheses)[83].

Aggregation: The aggregation operation is used to combine causally related states into clusters representing states at a higher level of abstraction as follows. When a set of new findings is entered in the program, it groups these findings into causally related clusters about which it is fairly certain. Next, it tries to group these clusters in alternate ways. If the number of alternate groupings is small (possibly two or three) the program builds alternate structures describing these possibilities. If the number of possibilities is large, the program tries to abstract already formed clusters into semantically larger concepts and tries again. This process is continued until it reaches a level of abstraction where most of the objects are connected to one another or when most of the objects are etiologies or diseases, at which point structural abstraction is no longer useful. Each abstraction generated by this process provides us with a coherent partial diagnosis for the patient. Within each abstraction pyramid all the diseases, findings etc. are mutually complementary, while the alternate abstraction pyramids provide us with competing diagnoses.

Elaboration: The process of abstraction in the system is complemented by the process of elaboration. When the program is provided with information about some disease or it generates some hypothesis at a high level of abstraction, it must assimilate this information in the patient description. This can only be done if sufficient structure exists at lower levels of hierarchy to structurally support this hypothesis. If not, the supporting structure must be constructed by decomposing this hypothesis into a more detailed presentation. Here again, we are faced with the situation that there may be many possible elaborations (presentations) for the same abstract condition. But as this hypothesis must be consistent with already known facts about the patient, parts of the elaboration will already be present in the patient model and the rest of the elaboration must be consistent with the model. This should reduce the number of possible elaborations greatly. Even then, the lack of information at lower levels of detail may cause a potentially large number of alternate elaborations to be possible and thus prevent us from elaborating any abstract hypothesis

to the lowest level of detail without further discriminating information.

Projection: So far we have studied the data operations that allow us to move up (abstraction) or down (elaboration) to build the hierarchic description of the patient model. Here we will discuss the third major type of data operation that will allow us to broaden the cross-section of the model at any given level of abstraction. The basic process of projection can be described as follows. Suppose at any given instance we are considering some hypothesis H. If H is present then some of its antecedents (causes) or consequents (effects) must be present. Therefore, if hypothesis H is assumed, it is reasonable to assume that at least one of its antecedents and most of its consequents will also be present in the patient. The process of projection allows us to suggest new hypotheses about the causes and the consequences of a given node at the same level of abstraction. These new hypotheses can then be used to group different nodes into causally antecedent-consequent pairs allowing further abstraction or elaboration. Note that abstraction and elaboration operations do not suggest new hypotheses.

3.4.4 Diagnostic Problem Solving

The patient description developed above was designed to provide the program with the capability of expressing its understanding about the patient's illness. In the patient description we are interested in assimilating all the available information in a coherent form. On the other hand, for any specific diagnostic problem, a large portion of this information is not directly relevant. A diagnostic problem statement should focus on that part of the patient description where the understanding is uncertain. It should describe different alternatives (which can be differentiated), and it should be easily decomposable into smaller problems. In this section we will discuss the process of identification and formulation of the diagnostic problem, its representation and decomposition into subproblems, the problem solving strategies, and the use of expectations in identifying discrepant information and in directing the flow of control.

3.4.4.2 Problem Identification and Formulation

At any point during the diagnosis the program has a set of complete hypotheses vying to explain the patient's illness. The reason for this is the lack of discriminatory information needed to resolve the differences between the CH's. The addition of this information to the patient description should result in resolution of these differences. The specific places where this information is lacking can be identified by identifying places where two or more hypotheses differ from each other in interpreting the known findings. Each place so identified represents a potential diagnostic problem. All the diagnostic problems identified above are collected in a list (problem-set) and are used in problem formulation.

Viewed differently, the problem-set identified above describes the set of problems all of which need to be solved in order to differentiate between the competing Complete Hypotheses. The availability of a set of problems to work on simultaneously provides the problem solver with the ability to minimize the sum total effort needed in solving all the problems by abstracting common aspects of problems and by selecting an efficient order in which the problems are solved. This can be done using either a rank-ordering heuristic or a problem-abstraction heuristic.

i. Rank-Ordering Heuristic: This represents a first cut heuristic. Here the problem set is rank ordered according to some criterion such as the need and urgency of diagnosis, the therapeutic

importance, global usefulness in patient understanding etc. and the problem with the maximum score is selected for problem solving.

ii. Problem-Abstraction Heuristic: Quite often, many of the problems in the problem set share some common feature, such as the duration and severity of the illness, the specific organ system or the disease mechanism involved. In such cases it is useful for the problem solver to abstract the common aspect of the problems or to partition the problem set in different classes and then differentiate between them, thus attacking a group of alternatives simultaneously. In short the problem formulator either formulates or selects one of the problems from the problem set for further problem solving.

3.4.4.4 Problem Description and Goal Structure

The problem formulated above is set up as the top level diagnostic goal for the diagnostic problem solver. A diagnostic goal consists of the following components; a) a primary goal which describes the main problem to be solved by the problem solver, b) a context which describes the reason for solving the problem and c) an expectation which describes the programs prior expectation about the outcome of the problem solving activity based on the knowledge already available to the program. These expectations are used in determining the consistency of the incoming information with the patient description and in directing the flow of control [71]. An example of a goal statement is given in the figure 3.17.

Once the top level diagnostic goal is identified the problem solver sets up a goal structure (a plan for problem solving) by decomposing this goal into subgoals recursively until we reach subgoals that can be solved using primitives known to the system. The subgoal generation is accomplished using elaboration and projection operations (described in Section 3.4.3) in the context of the patient description. Each subgoal is associated with some part of the patient description relevant to the the problem being decomposed. Therefore, we can view the goal structure as a representation of problem specific information extracted from the patient description. This interaction between the two allows us to assimilate the information gathered during problem solving with the patient description efficiently (as the interpretation and context in which the information is relevant to the patient model is known a priori). It also allows us to associate semantically meaningful expectations with goal statements to check the incoming information with the patient description for apparent and real contradictions.

3.4.4.6 Problem Solving Strategies

For a long time one type of strategy has dominated the thinking of the medical profession -- the differential diagnosis. The codification of this approach in a book such as French's "Index of Differential Diagnosis" was considered an important step forward in the systematic organization of diagnostic procedures. On the other hand, most of the computer programs using the hypothesize and test paradigm for diagnosis have generally emphasized the confirmation strategy confirmation is an important strategy in itself, its effectiveness is limited to situations where only one hypothesis is considered or one hypothesis dominates the competing hypotheses. In a study of strategy selection in medical diagnosis using protocol analysis, Miller [71] has identified various different strategies and situations where these strategies can be used for efficient diagnostic problem solving. In this program we propose to extend strategies identified by Miller using our experience with Internist [94, 95, 96, 97] and PIP [85, 87, 122] and adapt them for use in conjunction with the

patient description developed in previous sections. These strategies can be broadly classified as follows: Confirm, Rule-Out, Differentiate, Group-and-Differentiate, Refine and Explore. The selection of an appropriate strategy is based upon the syntactic structure of the diagnostic problem (e.g., the number of alternate hypotheses being considered and their beliefs relative to one another) as described below. The confirmation strategy is used when we have only one hypothesis under consideration, or when among a group of hypotheses, one hypothesis is much more likely than all other alternatives under consideration. The rule-out strategy is used to eliminate some hypothesis, which is substantially less likely than all other alternatives under consideration. The differentiation strategy is used to discriminate between two hypotheses with similar belief factors. The group-and-differentiate strategy is used when we have a large number of alternate hypotheses with similar belief factors. Here we need to discard a large number of hypotheses rapidly in order to focus our attention on a small number of alternatives. This is done by partitioning the alternatives into a small number of groups according to some common characterization and then applying a differentiation strategy to rule-out (or confirm) one of the groups, thus narrowing the hypothesis set. The refinement strategy is used to refine a hypothesis about a general class of diseases into more specific hypothesis. Note that the refinement of a hypothesis into more specific hypotheses generally results in a disjunctive set of hypotheses. Therefore, the refinement strategy is generally followed by differentiation. Finally, the explore strategy is used when the patient description does not have any well defined diagnostic problems to solve. In such a situation we explore the findings systematically, to gather sufficient relevant evidence to formulate a specific diagnostic problem.

3.4.4.8 An Example

Let us consider a situation with two possible patient descriptions shown in figures 3.15 and 16.

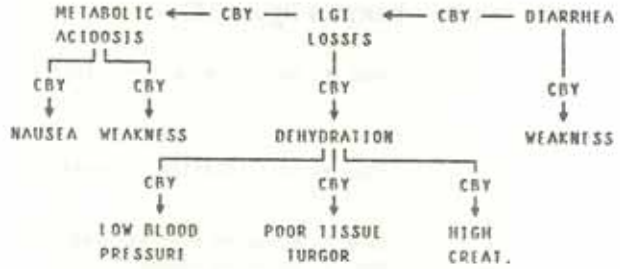


Fig. 3.15

Fig. 3.16

The top level goal representation for the above patient is shown in figure 3.17.

```
GOAL 1
 PRIMARY-GOAL: (DIFFERENTIATE DIARRHEA ACUTE-RENAL-FAILURE)
 CONTEXT: (CAUSE-OF METABOLIC-ACIDOSIS)
 EXPECTATIONS:
    (OR
       LIKELY    [DIARRHEA
                      SEVERITY: SEVERE
                      DURATION: GREATER-THAN TWO DAYS
       POSSIBLE: [ACUTE-RENAL-FAILURE
                      SEVERITY: MODERATE
                      DURATION: GREATER-THAN ONE WEEK])
       SUBGOALS: GOAL 2
```

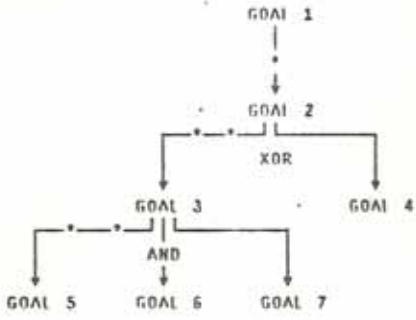
Fig. 3.17

We can differentiate between the diarrhea and acute-renal-failure by finding out the state of hydration of the patient.

```
GOAL 2
 PRIMARY-GOAL: (DIFFERENTIATE DEHYDRATION EDEMA)
 CONTEXT: (STATE-OF HYDRATION)
 EXPECTATION: (OR
                 LIKELY   [DEHYDRATION
                               SEVERITY: SEVERE]
                 POSSIBLE [EDEMA
                               NATURE: (OR GENERAL PEDAL)
                               SEVERITY: MODERATE])
SUBGOALS: (XOR GOAL 3 GOAL 4)
```

Fig. 3.18

This goal can be achieved by either confirming dehydration or edema. A graphic representation of complete goal structure for this situation is shown in figure 3.19



GOAL 1: (DIFFERENTIATE DIARRHEA ACUTE-RENAL-FAILURE)

GOAL 2: (DIFFERENTIATE DEHYDRATION EDEMA)

GOAL 3: (CONFIRM DEHYDRATION)
GOAL 4: (CONFIRM EDEMA)

GOAL 5: (CONFIRM LOW-BLOOD-PRESSURE)
GOAL 6: (CONFIRM POOR-TISSUE-TURGOR)
GOAL 7: (CONFIRM MODERATELY-ELEVATED-CREATININE)

Fig 3.19

3.4.4.10 Control Flow

In the section above we have noted that at the time some information is requested, the problem solver has a hierarchy of goals or a goal stack representing the downward (depth-first) locus of control flow (shown with "☆" in fig. 5.6). We have also noted that each subgoal in the goal stack is associated with expectations. The expectations associated with a goal can be viewed as a predicate which must be satisfied when the goal is attained and fails if the goal is no longer attainable. More specifically the expectation predicate may evaluate to one of four values; (i) Success, (ii) Partial Success, (iii) Failure or (iv) Contradiction.

After every question the expectation of the immediate goal is evaluated. If the result is a success, then control is returned to the superior, with an indication of success. The evaluation results in partial success if the information gathered is not certain enough to satisfy the expectations but can be interpreted in a way that will lend support to the immediate goal. On the other hand if the evaluation results in a failure, the problem solver tries to look for an excuse or a caveat that can cause the particular goal to be negated without contradiction. This mechanism allows us to distinguish apparent contradiction from real contradiction, providing the information acquisition process some robustness. More importantly, it prevents erroneous evaluation of hypotheses by preventing apparently contradictory information from corrupting the patient description. For example, let us consider a patient suffering from a Urinary Tract Infection (UTI) who has recently received antibiotic therapy for some unrelated ailment. Now let us look at the program at a point where UTI is its leading hypothesis, it has requested information about the result of the urine culture and it is told that the urine culture is negative. Let us also assume that the program is not aware of the antibiotic therapy. If the program now evaluates the UTI-Hypothesis, the negative urine culture would almost always cause the hypothesis to be rejected. Now as UTI is no longer its leading hypothesis, the next question, about the recent use of antibiotics, which could explain the negative finding, will not be asked. To avoid such situations the program should delay the evaluation of the current hypothesis whenever an unexpected finding is encountered and should explore this finding in greater detail. If an excuse is found, the current goal may result in success or partial success. If no excuse is found, but the discrepancy is not very serious, it results in failure. In case of failure, the expectations of the immediate superior goal are evaluated and the above-described process is repeated from there. On the other hand, if a major discrepancy is found, there is some serious error in the patient description or in the problem formulation. This situation is handled with the help of a special contradiction handler, which identifies the scope of the contradiction by backtracking through the goal stack to find a point where the contradiction does not affect the formulation of superior goals. It also invokes a debugging program that modifies the patient description in the light of the new finding. If the resultant change in the patient model does not change the top level objective of the problem solver significantly, the program continues with the same problem structure; otherwise, a new problem is formulated.

The process of back-tracking is computationally expensive. Therefore, commonly occurring contradictions and heuristics for recovering from them are generally precompiled in the knowledge base of the program. Availability of this information allows clinicians (and the program) to recover from anticipated contradictory situations effectively without resorting to back-tracking explicitly.

The new information gathered during problem solving must continually be added to the patient description. This process is considerably simplified because the incoming information is consistent with the patient profile and its context is well defined. The use of the goal structure provides a strong focus to the question-asking behavior of the program and provides us with a mechanism to shift from "Global" to "Local" problem solving (a shift observed in the behavior of clinicians [103, 71, 55]). It also makes the reason for question asking and the context in which the question is being asked explicit, allowing us to explain if necessary why a particular question is asked and the effects of various answers to the question.

3.5 Therapy Recommendation

In this section we will outline the therapy recommendation module. The task of therapeutic interventions can be classified along two axes: acute vs. chronic and symptomatic vs. etiologic. By an acute therapy we mean treatment given over a short period of time (from a few hours to a couple of days) such as giving glucose and insulin to combat hyperkalemia, while by chronic therapy we mean one that is given over a long period of time (e.g., from a few days to years) and may require a continuous long term management of some incurable disease such as chronic renal failure, rheumatoid arthritis or diabetes. Therapies can also be classified along a symptomatic to etiologic scale based on whether the therapy is given to alleviate the manifestations of some disease (such as Aspirin to combat headache and pain in influenza) or to eliminate the cause of the disease itself such as insulin therapy for diabetic ketoacidosis. Although we treat these classifications as categorical, we fully understand that each scale is a continuum and that categorization is somewhat unnatural.

One reason why physicians can give therapy without being very precise about it is the fact that the patient state can be evaluated during the administration of therapy. If the patient is not responding according to the expectations then the difference between the expectation and the actual patient response provides the most useful information in the patient evaluation and in correcting and adapting the therapy for the patient. This is the prime way in which therapy is adapted for individual patients. For example, to administer bicarbonate therapy, a physician must estimate the space of distribution of bicarbonate in a given patient (bicarbonate distributes throughout the total body water). The total body water varies significantly from patient to patient (e.g., the patient may be obese, emaciated, dehydrated or edematous) and is often difficult to assess precisely. If the electrolytes are checked after the initial dose of bicarbonate administration, the physician can easily estimate by extrapolation how much more would be required to achieve the therapeutic goals. This represents a very important aspect of therapy administration and we intend to use it by providing a therapeutic plan with expectations at intermediate points where the therapy should be evaluated and mid-way corrections done to achieve the therapeutic goals adequately.

The task of therapy can be divided into (a) initial therapy recommendation, (b) therapeutic evaluation, and (c) adaptive reformulation of the therapeutic regimen. This proposal will outline our first approach to the formulation of an initial therapeutic plan for a patient with acid/base and electrolyte disturbance. We are deferring consideration of the evaluation of past therapeutic efforts (except that this forms part of the diagnosis task), and we also leave to later investigation the feed-back control of the therapeutic plan.

Even the initial treatment formulation is a multi-component and multi-step process. Quartering this problem area further, we will start considering acute symptomatic treatment, and extending our work to the other three categories (e.g., chronic etiologic) later. In addition, we assume that no diagnostic problem solving is to be performed by the treatment formulator -- any information it needs will be provided by the diagnostic module upon request. For example, if the degree of dehydration is needed in formulating a recommendation for fluid loading, the program may request this information from the diagnostic program or some other interface.

3.5.1 Treatment Formulation

The treatment formulation task is divided in two steps. The first step is to characterize each of the individual abnormal conditions of the patient and to establish for each a set of treatment goals and methods which would be appropriate for that condition in isolation. The second step is to integrate this set of individual treatment plans into a coherent initial treatment recommendation, wherein the components are rationally fitted together.

A representative set of individual patient variables to be affected by this treatment are:

volume
serum sodium
serum potassium
serum bicarbonate
pCO_2
O_2
serum glucose.

These will be called the chemical state variables, and their values will define the patient's chemical state. In addition to this chemical state, another set of characterizations is needed to determine appropriate individual therapies for each of these levels:

renal function
pulmonary function
cardiac function (+ status of congestive heart failure)
liver function
central nervous system function (+ status of convulsions)

and state factors which also predict possible future disequilibrating effects,

presence and status of diabetes
presence of ileal fistula

The first part of this list defines the patient's functional state and the latter part his etiological state. These plus the previously defined chemical state make up what we refer to as the patient's basic therapy state, which will be abbreviated as state here.

From our informal study of the symptomatic therapy protocols we believe that individual corrective treatment recommendations can be determined by considering only the patient's basic therapy state . Ordinarily, we expect that in fact only the particular chemical state variable to be adjusted and a few of the functional or etiologic state variables will be of use in formulating an individual recommendation. Thus, for example, although the best therapeutic recommendation for a patient whose acid/base and electrolyte problems result from renal failure caused by nephritis might be an appropriate antimicrobial agent to control the infection, in this program we are examining only those short-term symptomatic responses which might be taken to stabilize the patient's chemical status (taking into account that the therapeutic plan cannot depend upon the renal-function in the patient to regenerate the bicarbonate in the patient or to excrete the uric acid).

Each of the chemical state variables in our model is numeric; thus, the principal thrust of every recommendation is to specify an appropriate level for this variable value to reach after an appropriate time. Other, perhaps non-numeric characterizations of these desired shifts may also be appropriate, perhaps to indicate primary and secondary goals. For example, the recommendation to increase serum sodium from a value of 110 mEq/L to 120 mEq/L in six hours in a patient with convulsions may be said to have the primary goal of improving CNS function to stop convulsions and the secondary goal of raising the serum sodium out of a critical region. The recommendation must also include a set of expectations about the future state of the patient at appropriate monitoring points of the therapy. For example, we might specify that in the above case, the serum sodium after the passage of three hours should be near 115 mEq/L. In addition, the recommendation must include components which define the nature and composition of the corrective action to be used to achieve the goal state. Continuing this example, we might recommend the use of IV hypertonic saline as the mechanism of achieving the desired goal, giving perhaps 2 liters over the six hours. (These values would, of course, depend on the other state variables of the patient.) So far, we have the impression that the correct way to specify the quantity may need to be fairly general and may be specified as a specific amount, some amount per unit time, some relatively simple function which computes the amount from a small set of parameters, etc. A treatment recommendation for the above example can be described as shown below:

```
TREATMENT RECOMMENDATION
  QUANTITATIVE GOAL       serum Na+ = 120 mEq/l
  TIME TILL GOAL          6 hours
  PRIMARY PURPOSE         stop convulsions
  SECONDARY PURPOSE       raise serum Na+ out of critical range
  EXPECTATION             achievement of quantitative goals and both purposes
  REVIEW ADVICE           expect serum Na+ = 115 mEq/l after 3 hours

  SPECIFIC TREATMENT
    AGENT                 IV hypertonic saline
    QUANTITY              2 l.
    DURATION              6 hours.
    RATE                  cc/hour.
```

The treatment recommendations for all the chemical states are then sent to the second component of the program. This combines all the recommendations into one treatment plan. For example, if along with the treatment plan for IV saline we also have a recommendation to administer 100 cc of bicarbonate and one liter of 5% dextrose, this program then combines all the three into one solution and recommends administration of the entire fluid through one IV running at the rate of 500 cc/hour. Along with each such recommendation the program also generates a check list of items that need to be checked before the therapy can be administered safely. For example, before administering the fluid, the program may need to know that the patient is capable of tolerating the fluid load (is not in danger of getting pulmonary congestion etc.).

At many instances during the program, the patient specific model contains sufficient uncertainty that the therapy recommendation program cannot generate a single set of recommendations and must generate individual recommendations for each complete hypothesis separately. In such cases the therapy program also identifies the reasons for not being able to generate a single set of therapeutic recommendations. These reasons are then used to drive the

diagnosis program further. (The meta level goal of the diagnostic program is to generate a unique therapeutic plan.) Thus the diagnostic program selects its next problem to be solved based upon the information provided by the therapy selection program.

3.6 Significance

For the intermediate term future, the significance of this research will be in terms of its contribution to the methodology of artificial intelligence based medical programs. It will ultimately enable us to build a program in the area of acid/base and electrolyte diagnosis and therapy which will include much of the knowledge of the best experts in that domain, but that stage of accomplishment probably lies beyond the end of the projected grant period. At the moment, however, it is the development of appropriate methods and technical tools that is important if truly expert programs are eventually to be built.

We have described some of the deficiencies of the best programs which have been implemented with the current technology. The proposed AI techniques will allow us to address these problems by making available in the program explicit representations of all those components of the expert physician's decision making which we believe to be important to him. We have proposed a hierarchic representation of medical knowledge, permitting the description of causal and associational connections at several levels of detail, for expressing both general medical knowledge and specific information about the particular patient under consideration. We have introduced the notions of hypothesis and complete hypothesis and described how the physician's state of mind may be representable in the machine via these mechanisms. We have developed techniques for reformulating hypotheses based on the acquisition of additional information, and for selecting new information to seek based on the current hypothesis. We have also suggested a potentially useful method of deferring the need for detailed probabilistic computations while general diagnostic problem solving is in progress. We have also outlined the beginning of a therapeutic module to complement the diagnostic program we described.

Each of the above components represents one solution to important AI problems facing all researchers. Because the complexity of the application domain we have chosen is sufficiently high that we are forced to develop deep reasoning techniques to solve them, the results of our work should also be applicable to other efforts. For example, our notion of explicit hypotheses and the central role of their reformulation should be a significant addition to current diagnostic programs such as the Present Illness Program, INTERNIST, and the Birth Defects Program we are developing.

The process of eliciting form expert physicians a formal version of their knowledge often yields a significant benefit in addition to its utility in the development of computer programs and representation theories. We discover time after time that for many of the questions which the naive computer programmer must answer there are no written answers in the medical literature. Yet the medical student, who begins his education at a level of naivete similar to the programmer's, must learn those answers because they define how he practices medicine. Medical education tends to emphasize the factual over the procedural aspects of knowledge, with the assumption that clinical exposure will provide ample opportunity to learn the processes. In our work, we are forcing physicians to make more explicit what processes they employ in their own practice. They have felt that these more explicit models are of great value in how they teach their students, interns and residents. Thus, the elucidation of medical knowledge can have an immediate benefit in medical

education.

4. A Behavioral Analysis Study of Clinical Problem Solving by Experts

Personnel:

Jerome P. Kassirer, M.D., Principal Investigator
G. Anthony Gorry, Ph.D., Investigator

4.1 Purpose

a) to study the methodology of experiments directed at analyzing clinical problem solving tactics.

b) to study the diagnostic and therapeutic problem solving behavior of expert clinicians.

c) to study the development of clinical problem solving prowess.

d) to study the mechanisms by which expert clinicians revise diagnostic probabilities.

4.2 Background

A. Introduction - Despite the impressive expansion of the scientific basis for clinical practice in the 20th century, ambiguity rather than certainty remains prevalent in the clinical setting where crucial tasks involve problem solving, decision making and clinical judgment. Unfortunately, little attention has been paid to the cognitive processes that underlie the problem solving methods of physicians: knowledge about how experts solve clinical problems has been derived principally from the introspection of experienced clinicians. This introspective method has been useful as a framework to teach students and house officers how to approach complex clinical problems and has also been employed to design computer programs that closely simulate the diagnostic behavior of experts [42, 85, 112].

B. Problems with the Introspective Approach - The introspection of experienced clinicians has yielded valuable principles and pragmatically useful practices but because the clinical problem solving process is so complex, even highly intelligent physicians trying to describe their own tactics may omit key elements. In addition, the introspective method may be faulty because the experienced clinician may describe his diagnostic and therapeutic methodology in one fashion when in fact he is following an entirely different technique.

C. Rationale of Behavioral Studies - Because of the limitations of the introspective approach, the emphasis of modern research on human problem solving has turned from examining what problem solvers say they do to behavioral studies that analyze what they actually do. This trend is apparent in studies in the field of artificial intelligence [72, 138]. Similar studies of problem solving have been carried out in general domains, such as logic [79], chess [76], and cryptarithmetic [77]. More recently, behavioral studies of human problem solving have been carried out in the medical domain. These studies have emphasized the careful observation of experienced physicians as they have approached a diagnostic problem [28, 55].

D. Our Previous Observations - Our recent study of clinical problem solving focused on the problem solving behavior of experts involved in "taking the present illness" from a simulated patient [55]. In this study we identified several important aspects of the problem solving strategies of experienced clinicians. These strategies included the following observations: 1) Specific diagnostic hypotheses are often formulated when little clinical data is available. 2) Only a small number of active hypotheses are considered at any one time (4 to 11 hypotheses). 3) The hypotheses that are generated then provide a context for further diagnostic refining. These hypotheses also form the basis for expectation for further information gathering and they provide a context for identifying potentially life threatening and treatable complications. 4) The process of hypothesis evaluation consists of case building strategies to refine or reject a diagnostic hypothesis. 5) The diagnostic style differed among experts; some experts used a highly focused or directed approach whereas non-experts used a more structured approach analogous to a "review of systems". 6) Differences could be identified between experts and non-experts (in this case, several nephrologists were measured against the performance of a gastroenterologist and a cardiologist). The experts asked fewer questions, mentioned the correct diagnosis earlier, made a firm diagnosis earlier and maintained a smaller number of active hypotheses than the non-experts. 7) Experienced clinicians did not use a purely inductive method of data gathering which persisted until a solution to the problem spontaneously emerged, in contrast to that methodology recommended by others [112].

In other studies, we showed that the process of hypothesis evaluation ("case building") consists of repeated revisions of a probability structure [42, 112]. We also showed that after all "cost-free" tests were exhausted, the resultant diagnostic probabilities could be utilized to make subsequent management decisions [42]. This probability structure is clearly one of the most important elements in the process of patient management because all subsequent decisions involving risk and value of tests and treatments are based on this diagnostic pattern.

E. Observations by Others - The other report comparable to our study of clinical problem solving is the study of Elstein, Shulman, and Sprafka [28]. In this extensive study of clinical problem solving, Elstein and his colleagues found, as we did, that early hypothesis generation was a powerful tendency. They also found that an inductive method of data gathering was not used. However, in their study no difference could be discerned between experts and non-experts in the various categories of diagnostic ability. We have reservations about these observations on expert behavior. First, the retrospective method they used to ask the clinicians why certain questions were used can produce substantial distortion [140]. Second the method they used to identify experts (i.e., by an election process) may be faulty. Third, as described before, we were able to find differences between experts and non-experts; and although our study was rather limited in scope, such a difference is intuitively sound.

F. Disagreement Among Experts - In any study designed to assess the opinions of experts, cognizance must be taken of the fact that expert opinion cannot be considered uniform. The problem of how experts behave and how they differ from one another is important not only in understanding the quality of clinical decision making but also in comparing the behavior of computer programs. It is widely known that experts only agree part of the time in making even relatively simple judgments. Thus, in interpreting chest x-rays and coronary arteriograms, considerable variation in expert opinion has been noted [23, 48]. The problem of expert disagreement is of critical importance in designing experiments to evaluate computer programs in our own study. We have learned, for example, that physicians at the Tufts-New England Medical Center are far more conservative when prescribing digitalis than comparable experts at Baylor

Medical Center. This discrepancy has made it difficult to assess our digitalis therapy program [43]. Similar difficulties arise comparing the performance of other computer programs (MYCIN) to the performance of experts [115]. Thus, we believe that it is highly desirable to develop improved methodologies to assess expert behavior.

G. The Methodology of Behavioral Research - Early studies of behavior focused primarily on observations of lower animals (for example, mice going through a maze) and the results of these experiments on lower animals were then extrapolated to humans. More recently, as pointed out above, the concept of behavior-oriented studies is to carry out detailed studies of humans solving a limited problem, for example, cryptarithmetic [77]. Although the potential values of this kind of study in clinical medicine are great, some concerns must be raised about the pitfalls of available methods:

1) Problems of experimental design - In our study of clinical problem solving [55] we noted that the selection of a case could influence the results, and that the presence of physicians could also influence the results. The participants may view the diagnostic exercise as a game rather than a true-to-life experience, and their responses could thus be altered. In addition, the process of having the participants articulate their problem-solving approach could also alter their tactics.

2) Problems with analysis of the data - Analysis of the data generated from research efforts such as Elstein's study and our study [28, 55] is difficult because the data generated are selected by judgment of the investigator and are thus "soft". Examples of these soft data include: whether or not a new hypothesis has been generated, whether the diagnosis has been established, and what the object of the given question may be. More accurate scoring methods for this kind of "soft" data must be developed.

3) Bias of results - Multiple studies show humans to be fallible in making accurate probability judgments and making a variety of decisions. Humans often derive conclusions from small quantities of data, they often depend on stereotypes of characteristics, they ignore concepts such as statistical regression, they are poor judges of probability, they stick tenaciously to their judgments once they make them, even if these judgments are erroneous [27, 54, 61, 64, 128, 129]. Although these studies identify the limitations of humans as problem solvers, they do not abrogate the importance of the research suggested here. In the diagnostic process, data on exact probabilities can be obtained only for the simplest clinical problems; they cannot be obtained for complex clinical problems. For example, in a 50-year-old man with crushing chest pain, a statistical probability can be derived which defines the likelihood that the patient is suffering from a myocardial infarction. Such statistical likelihoods cannot be readily assessed, however, for a 50-year-old man with crushing chest pain who also has a discrepancy in pulses between both femoral arteries, constipation for 10 days and whose history discloses a blow to the sternum one week before in an automobile accident. In the latter case, only a subjective probability assessment can define the likelihood that the patient is suffering from a myocardial infarction. For this reason, subjective assessments must be used as the basis for the assessment of probabilities and for later decision making.

4.3 The Experimental Plan

4.3.1 Experimental Design and Analysis

With a relatively specific experimental plan in hand (see below), we will investigate, with the aid of a cognitive psychologist, methods for avoiding problems of experimental design. In addition, we hope to develop methods for analyzing the data in a more objective and quantifiable form. In the process of designing the experiment, we plan to consult with others involved in the project (Drs. Pauker and Szolovits) in developing these methodologies.

4.3.2 Study of Expert Clinical Problem Solving Behavior (tentative plan)

1) Aspect of problem solving behavior to be studied

a) Examination of evolving probabilities - In our previous study of problem solving behavior we examined how physicians established diagnostic hypotheses and assembled a coherent and adequate diagnosis, but we did not assess how these physicians viewed the probability of each of their diagnostic hypotheses as the information available to them increased. In the proposed experiment we anticipate studying how new data alters the perceptions of posterior probabilities and the extent of the variations in these probabilities as the diagnostic hypotheses evolve.

b) Variability of the prior probability among experts -- as noted before, the probabilities of disease after exhaustion of all cost-free tests are the "critical probabilities" upon which all subsequent management decisions are based and they are the probabilities that we used in a computer generated decision analysis study to make a therapeutic decision [42]. In a subsequent study, we showed how sensitivity analysis on these probabilities could affect the management decision [92]. We hope to define how variable these critical probabilities are among experts and the reasons for this variation. In addition, we hope to assess whether this variability influences medical management decisions.

c) The variability of management decisions - After cost-free tests have been exhausted, the management decision is based on the value and risk of test and treatments. We plan to study the decisions made by experts and the reasons for the variability of these decisions. We also plan to study whether experts can assess diagnostic and therapeutic thresholds [84, 89].

2) Subjects - The study will focus on the problem solving behavior of experts. For the purpose of this experiment experts will be defined as geographic full time or strict full time specialists in gastroenterology. We plan to study approximately 20-30 such individuals. These experts may be recruited from the ranks of Gastroenterology services at Boston teaching hospitals, Baylor Medical Center, Washington University (St. Louis), Yale, The University of Pennsylvania, and the University of Chicago, institutions at which the investigators have personal ties.

3) Selection of case material - With the cooperation of a gastroenterologist and an abdominal surgeon at the New England Medical Center, we will select two cases for detailed analysis. Cases will be selected to conform to the following criteria:

a) Only real patients will be selected -- the actual patient data will be used for the case descriptions.

b) The patients selected will have a clinical presentation and history suggestive of and consistent with the final diagnosis. No effort will be made to find a "classic" case of a particular disease.

c) Cases to be selected will be rich in history, physical findings and laboratory data.

d) Cases will be selected so that after all cost-free tests have been exhausted, considerable diagnostic uncertainty will still exist.

e) Cases will be selected which require a decision involving assessment of risks of tests or treatments, the values of tests or treatments and an integration of all these facts into a final decision. We have called the point in the cognitive consideration of diagnostic and management problem solving at which all cost-free tests have been exhausted, the critical decision node. This is the same point at which computer oriented decision analysis for patients with acute renal failure was described in one of our earlier studies [42].

Tentatively, we have decided to use the following kinds of cases:

1) suspected appendicitis

2) diarrhea of unknown cause

The rationale for the choice of these cases is as follows: Suspected appendicitis represents a rather simple diagnostic problem in which the plan of approach is fairly well defined. This initial case will probably serve as the standard for other analyses. We are planning to select a rare cause of diarrhea as the second case, such as pancreatic tumor secreting vasoactive intestinal peptides, in order to see if the approach differs from the approach used for a simple case of suspected appendicitis. A case of this type is already in hand.

4) Method of obtaining data

1) We will request the participation of the physicians at the various medical centers by a personal phone call. The request to participate in the study will be made standard by the investigator reading the request from a prepared text. We also will standardize the instructions to the participants. The participants will be visited in their own office, will be given instructions from a prepared text. The session will be recorded on tape for later transcription.

The cases will be prepared so that "packets" of information can be made available sequentially to the participants. We will ask the participant after each packet is presented to describe which diagnoses he/she is considering, how likely he/she considers each diagnosis to be, and the reasoning behind the choices. The clinical information will be divided into approximately 5-6 packets, possibly conforming to traditional groupings of these data; all pertinent "cost-free" information will be exhausted when the last packet is provided.

Next the participant will be asked to make a decision regarding further management and to describe the rationale for this decision.

Finally, the participant will be asked to make an assessment of the diagnostic or therapeutic threshold [84, 89]. By obtaining these data we will be able to assess the consistency of the implicit management decision given earlier.

We will ask the investigators not to discuss the case with others.

5) Pilot Study--Prior to embarking on the extensive study described above, we will first develop methods for presenting the case material. Selection of a case will be made with the help of a gastroenterologist and abdominal surgeon and tests will be made of the suitability of the case and the format of the presentation, using as subjects the gastroenterologists at the New England Medical Center Hospitals. The pilot study will continue until we have selected appropriate cases and until we have standardized the material to be presented and the mode by which the clinical data is to be presented. The results of the pilot study will be reviewed with the cognitive psychologist and alterations of the format will be made if appropriate.

6) Analysis of data--From each analysis of a case by one of the participating physicians, we shall develop a structured record of pertinent data for subsequent analysis. In brief, the data record for an individual doctor dealing with one case can be described as follows.

As noted above, case material will be presented to the subject in packets and the subject's analysis of each packet will be recorded. From the information contained in the transcriptions of that analysis, we shall determine the new hypotheses considered and the previously held hypotheses that have been rejected based on the packet of information received. Further, grouping of hypotheses into more comprehensive ones or refinements of hypotheses into more specific ones will be noted. Finally, the assessment of diagnostic probabilities will be recorded. After all packets of information have been provided, the final listing of diagnostic probabilities will be taken as the "critical probabilities" described above. The resulting record of the physician's analysis of the case then can be organized in a matrix, the columns of which represent the sections of the case presented, and the rows the hypotheses considered. Entries in the matrix will indicate the state of the hypotheses (under consideration, rejected, etc.) during each section. Special columns between sections will contain the assessments of diagnostic probabilities made at those points. Note that although the number of columns (sections) for a given case will be the same for all physicians, the number of rows in the matrix will depend upon the number of hypotheses mentioned by each physician subject.

The data clerk will be trained to create such records from the data gathered from the subjects. As part of the first phase of the study, a number of cases will be abstracted in this way by the data clerk and by several physicians experienced in this type of research. Results will be compared in order to identify problems and to instruct the data clerk. When the latter's performance is judged satisfactory, the second phase of the data collection will proceed. In that phase, however, the data clerk will meet regularly with Dr. Kassirer to review progress and to resolve ambiguities. Periodically the data clerk will abstract cases a second time, so that we can detect any drift in the interpretation of case materials.

Before embarking on extensive data analysis, we will consult again with the cognitive psychologist regarding the methods of data analysis. At the present time, we assume that the analysis will proceed in several stages. First, simple descriptive statistics will be computed from the matrices. Comparison of the number of hypotheses by section across the physicians studied is an example of such a measure of interest in a given case. Further, we shall assess the correlation of the total number of hypotheses considered by a physician in the first case with that number in the second case. Additionally, we shall use rank-order statistics to assess the extent to which the subjects concur as to the likely diagnosis.

By turning to the subjective probabilities directly, we can gain further insight. Direct comparison of these probabilities by section by case is a useful beginning. More complex analysis of the way in which probabilities change from one section to another will provide further information of interest.

Many exploratory analyses of the data can be conceived, and some undoubtedly will yield results which will suggest specific subsequent analyses. For example, using a simple measure of dispersion, such as the number of hypotheses which account for a given percent of the probability, we can rank physicians after each section of a case. It is of interest to know if those rankings would change significantly from section to section. Perhaps physicians who focus sharply on a few hypotheses at the outset hold to them throughout the case; in any event, we can investigate the relationship between such a measure and diagnostic accuracy or action proposed at the critical decision point.

Another aspect of the performance of the physicians in the study is less quantifiable at this point, but is clearly important. This is the overall process of hypothesis development. Aspects of this process have been alluded to above (for example, refinement versus aggregation of hypotheses), but the question of diagnostic style will remain. We have discussed this question in a recent publication [55], so here we shall note only that some interesting ideas may result from the informal review of the transcripts by experienced clinicians who can identify the outlines of general approaches to diagnosis in the recorded materials. For this reason, our approach to data analysis will be both quantitative and qualitative, with the former perhaps serving as a major stimulus for the latter.

4.3.3 Study of Development of Clinical Problem-Solving Prowess

Assuming that the protocol described above (or one similar to it, modified during the course of the study) provides data concerning the problem-solving behavior of expert clinicians, the same techniques will be used to study the problem solving capabilities of those presumably less expert in gastroenterology. We shall assume that the expert hierarchy (in inverse order) to be as follows:

1) academic gastroenterologist

2) clinical fellow in gastroenterology

3) resident in Internal Medicine

4) intern in Internal Medicine

5) junior or senior medical student

6) first and second year medical student

By studying some (possibly all) of these groups and comparing the results to the experts, we should get a better notion of the development of diagnostic problem solving strategies, the interaction between the knowledge base of individual and their problem solving skills, and development of management decision capability.

4.4 Significance of the Research

A deeper understanding of the problem-solving behavior of experts should have multiple benefits. 1) It may provide a mechanism for understanding why experts disagree when important decisions are made. 2) It may improve the ways in which diagnoses and complex management decisions are made. 3) It may make it possible to develop better computer-oriented approaches to diagnosis and management. 4) It may improve our capacity to teach diagnostic and management processes.

5. Tools for Clinical Consultation with Decision Analytic Techniques

Personnel:

Stephen G. Pauker, M.D., Principal Investigator
Jerome P. Kassirer, M.D., Investigator
Byron Davies, Research Assistant

5.1 Introduction

5.1.1 Objective

To develop a set of computer programs to assist the "naive" physician in carrying out decision analysis on a particular clinical case involving a patient with a difficult management problem.

5.1.2 Background

For the past decade, the techniques of decision analysis and utility theory [2, 56, 98] have been refined sufficiently so that they can now be applied in clinical medicine. Most published studies on decision analysis in medicine have been concerned with general classes of clinical problems rather than the complex problems of individual patients. In 1973, however, we published an example of how decision analysis could be applied to an individual patient management decision and in that publication we considered in some detail the methodologies employed [112]. Since that time, the Clinical Decision Making Group at the New England Medical Center Hospital has been actively involved in the bedside application of these tools. At first, we directed our efforts toward teaching the medical house staff and a group of attending physicians to use the complete methodology of decision analysis, but these efforts met increasing resistance because of the huge time costs involved and because of the complexity facing a novice in applying the techniques. Because of these problems, we terminated this task after a trial of six to eight months. In the past several years, however, we have continued to develop and refine decision analytic techniques for use in clinical practice.

This developmental phase has gone in several directions. First, we developed prototypical analyses which can be readily applied by physicians in day-to-day practice [84, 89]. Examples of such prototypes are the "treat-no treat" and "test-no test" models which center around the concept of diagnostic and therapeutic "thresholds," concepts which we developed. Second, we have been developing improved mechanisms for assessing utilities or values from both physicians and patients. We have been concentrating a good deal of effort in this latter area because it represents the crux of many medical decisions, but is an area poorly understood in either formal or informal analyses. Our first attempts at assessing patient preferences dealt with the issue of coronary artery by-pass surgery for disabling angina [86, 90]. We were able to demonstrate how variations in patient's utilities could have major effects on the clinical choice between surgery and medical therapy. To a limited extent, this approach has been used in complex cases by members of the

Cardiology Division at the New England Medical Center Hospital. We have also done extensive work on utility assessment in prospective parents facing the decision of whether or not to seek amniocentesis for the pre-natal diagnosis of trisomy 21 [88, 93]. More recently, we have begun to assess the attitudes of patients and members of the general population toward survival for varying lengths of time and have demonstrated, in patients with lung cancer, that the classic measure of therapeutic efficacy -- the five year survival rate -- may not provide an optimal basis of decision making [69]. In analyzing that problem, we began to develop some crude computer programs to combine utility data acquired from patients with survival data from the literature in order to develop measures of therapeutic efficacy which reflect the attitudes and desires of the individual patient. Third, we have been developing computer programs to deal with the diagnostic and therapeutic decisions involved in staging and treating patients with Hodgkin's disease [108, 109]. Those programs develop a decision tree representing the problem, project probabilities of each stage of disease based on the patient's clinical findings, and calculate the expected utility for a variety of diagnostic and therapeutic plans, based on several alternative measures of utility.

We have also developed a methodology for dealing with difficult management decisions in individual patients with a wide range of disorders and have published a detailed example of such an analysis [92]. At first, such efforts were crude and very time-consuming, but as we have gained experience, we have become more efficient and have been able to provide consultations which have impacted on individual management decisions. At the present time, we offer such consultations on a regular basis to physicians at the New England Medical Center Hospital. Simultaneously, this service also provides elective rotations for House Staff and medical students. We are in the process of applying for funds for clinical fellowship training in this area. In the past nine months, we have carried out more than fifty consultations at the New England Medical Center Hospital and have recently begun to receive requests from other institutions.

The format of these consultations is variable, depending on the specifics of the clinical problem, but varies from structuring clinical problems to data interpretation, utility assessment, and sensitivity analysis; many include all of these issues. Many of the problems we have analyzed are repetitive: they often involve extensive computation of expected outcomes, survival curves, and sensitivity analysis on a variety of parameters. Most of the computations are done on hand calculators. Graphs showing relationships between variables must be produced by hand, a time-consuming task. In fact, a single consultation requires the close attention of an expert to structure the problem, gather relevant probabilities from the literature or from expert consultants, carry out the complex task of utility assessment, make the appropriate calculations, decide which of the many variables to test by sensitivity analysis, and then display relationships between variables. The expertise to carry out such potentially important clinical analyses is localized to but a few institutions in the country and the time required by such individuals to do a careful, thoughtful analysis is prohibitive.

5.1.3 Rationale

In an effort to make consultations of this type available widely, we propose to develop a computer program to reproduce, as nearly as possible, the clinical decision making capability of an expert in decision analysis. Our concept is that such a program would not be data-intensive -- we do not, at this time, envision the development of a huge data base of facts needed for decision analyses (although such data bases will undoubtedly be needed and developed in the future).

Rather, we see this project as knowledge-intensive, that is, we hope to incorporate into this program the expertise and heuristics which we have developed in applying decision analysis to clinical medicine. We envision that such a program could be used first to assist physicians in performing actual clinical decision analyses and, second to educate students and physicians in the techniques. Any program developed as a. part of this project can be extensively tested by a comparison to the routinely-prepared consultations by our group.

We plan to build extensively on prior work in this area [57, 60, 102, 110, 142]. However, we feel that the special nature and constraints of the problem domain of clinical medicine will necessitate a basically new implementation. Also, since this developmental effort will largely be in the MACLISP language, it will be simpler to design a new system than to adapt currently available software.

5.2 Specific Aims

We plan to develop these computer programs in a sequential fashion, first developing purely computational support for the calculation of expected utilities, survival curves, Bayes' rule, and sensitivity analyses. We will then develop the more complicated heuristic programs designed to assist the physician in structuring the clinical problem and in matching the problem to the appropriate prototype. The basic organization of this research will be to develop a layered set of programs. The central kernel will be computational -- a program for calculating the expected value of a given plan, given a description of the probabilities and utilities of the various potential outcomes of that plan. Based on that kernel, programs will be developed to perform sensitivity analyses and to make graphic portrayals of the results of analysis. The next layer of complexity will involve programs for calculating and simulating outcome probabilities and utilities. Finally, based on all the lower layers, programs will be developed to assist the physician in structuring the decision problem and identifying similar analyses which have already been performed and which could form the basis for the new analysis. The development will probably follow this sequence:

1. Develop a computer representation which will capture the structure of a decision tree, including the associated probabilities and utilities, and allow for the symbolic representation of the various factors entering into the clinical decision (outcomes of tests and treatments, risks of tests and treatments, values of outcomes, probabilities, etc.).

2. Develop a computer program which will calculate the expected utilities for any decision tree.

3. Develop a graphics package for displaying decision trees and analyses of those trees, including plotting expected utility as a function of any parameter or combination of parameters, and plotting threshold diagrams of simultaneous sensitivity analyses of two or more parameters.

4. Develop programs to do sensitivity analyses for any parameter or combination of parameters.

5. Develop computer programs to assist in the assessment of utilities based on techniques such as the lottery, certainty equivalents, conjoint measures, and categorical scales.

6. Develop computer programs to calculate and represent survival curves.

7. Develop computer programs to help the physician construct and simplify clinical decision trees.

8. Develop computer programs to help the physician "match" clinical problems against well-known and previously analyzed prototypical situations.

5.3 Methods of Procedure

The computer programs which we plan to develop for this project fall into two general classes: 1) Computational production programs which carry out the mechanics of decision analysis and, 2) heuristic, knowledge-based programs which help the physician structure and analyze the problem. Since the computational program will be the central algorithm of the analysis, we plan to develop that program first. (This central algorithm can be viewed in a role similar to that of the pharmacokinetic model of the digitalis therapy advisor [43].) The center layers of these programs will assume that the physician is able to structure his problem in the form of a decision tree and only provides support for tree analysis. The outermost layers will help the more naive physician structure his problem domain. This bottom up approach is necessary because the computational support of the kernel-algorithm may well be necessary to help the physician simplify and structure his problem domain effectively.

We hope to implement that program on two different computers. First, we shall develop the program on the PDP-10 in MACLISP because the expertise in artificial intelligence at the Massachusetts Institute of Technology will be required for research into the more difficult heuristic program. Second, we shall also implement the program, in a brief "running" form, on a small computer (PDP-11/V03) which is situated at the New England Medical Center Hospital. We will probably implement this version in FORTRAN, making the working part of the program easily exportable to other institutions, and also making it accessible to clinical fellows, house staff, students, and other individuals who will be using and testing the system. In both systems, display will be accomplished on a Digital Equipment Corporation VT52 terminal. In the case of the running system (11/V03), a VT-XX-KA interface will allow transfer of the screen contents to a dot-matrix, impact printer. Program and data storage will be on standard discs for the PDP-10 system and on floppy discs for the 11/V03 system.

5.3.1 Tree Representation

In the MACLISP environment, decision trees will be represented as a nested list, with properties on appropriate atoms pointing to either value cells or functions for determination of probabilities and utilities. This representation was successful in a small program used for the calculations of expected utility for coronary artery by-pass surgery. Each node of the tree will be represented as a list of branches. For chance nodes, each branch will point to the next downstream node and contain a measure of the likelihood of that branch's occurrence. That measure of likelihood may be either a maximum likelihood estimate, a range or distribution of likelihoods, or a function which when called will return a value for the probability. The importance of such

functional representation is that, when run in interpretive mode, they will allow modification of any parameter of the tree *or* any parameter which can affect an outcome probability. For example, consider a diagnostic test with various conditional probabilities in diseased and non-diseased populations, depending on the decision criterion chosen to define a positive test result. If the form of the distributions can be expressed functionally (e.g. overlapping normal distributions of unequal variance) then the effect of changes in the decision criterion on the expected value calculation can be straightforwardly found.

For decision nodes, each branch will point to the next downstream node but will contain no measure of likelihood. Rather, a maximization function will be applied which will select the downstream node with the highest expected value in its value cell. If no value is available, recursive application of this procedure will generate such an expected value.

Each final outcome of the decision tree will be represented as a node without downstream branch pointers. It will contain the utility of the outcome either as a value in its value cell or as a function which can return the utility based on prespecified parameters. This representation of outcomes will allow the simple expression of "functional sub-trees." In our clinical experience, we have found that complex decision trees can be greatly simplified by recognizing common sub-structures which differ only in probabilities or utilities, but which are structurally identical. In such cases all common sections are represented as a single sub-tree with the variable probabilities and utilities being functional parameters to the sub-tree. With the proposed outcome representation given above, such sub-trees would be naturally represented as functions which in their evaluation could easily contain entire trees. This representation scheme is quite natural in MACLISP but some modifications are required for FORTRAN implementation in the PDP-11/V03 system which has no list processing facilities. The simplest approach will be to implement a list-processing environment using FORTRAN arrays. The major reason for maintaining each node as a separate atom (as opposed to making the tree into a single nested list) is to allow for simple mechanisms of tree modification (adding and deleting nodes and branches) and to allow simple correspondence between the FORTRAN and MACLISP versions of the program.

5.3.2 Calculation of Expectation

In classic decision analyses, calculation of expectation is from outcome to choice, i.e., from right to left; hence the term "folding back the decision tree." Such an approach can be computationally complex since the computation must begin from each terminal branch. We shall exploit the structure of the decision tree and the recursive nature of MACLISP by expanding the tree from left to right, following the actual problem in evolution. This recursive technique has been successfully used in the simple angina decision program mentioned above.

Implementation on the 11/V03 in FORTRAN will be somewhat more difficult because of the lack of built-in recursive bookkeeping. Since our trees will tend to be rather simple, with rarely over twenty outcomes and virtually never over 100 outcomes, we should be able to employ the FORTRAN array structure to do explicit bookkeeping for recursion. Such bookkeeping will impose a significant computational overhead, but since our problems will be small and our computations will be on a dedicated machine, we do not feel that this burden will be unmanageable. The alternatives will be to implement recursion in machine code or to calculate from right to left (folding backwards) by keeping a threaded list of outcomes and backward links. The former

approach will limit the machine independence of our developed programs, while the latter may offer no real computational advantages. It certainly would require more core and we are planning to use PDP-11/V03, with a limit of 28K words of addressable core.

Returning to the MACLISP version, we hope to provide an interface between our program and the MACSYMA program for symbolic mathematical analysis which has been developed at the MIT Laboratory for Computer Science. That program is capable of symbolic processing such as equation solution and integration. For certain analyses, it would be very useful to symbolically solve for the threshold value for a variety of parameters, that is, to do sensitivity analyses symbolically. In simple problems we have used this approach before, e.g., the therapeutic [84] and diagnostic [89] thresholds and the utility threshold of our genetic counseling model [88, 93]. This symbolic approach would be far too complex for low level implementation on a PDP-11/V03, but for certain recurrent problems it would provide a new tool for decision analyses.

5.3.3 Graphics for Decision Analyses

The graphics package will serve two main functions: 1) It will provide hard-copy representation of decision trees in standardized format, and 2) It will display the results of various analyses. Such displays are often graphic. They may be plots of expected utility as a function of one or more parameters as a function of another. An example of the former plot might be expected utility as a function of the probability of disease; an example of the latter plot might represent the threshold operative risk against either the patient's utility for some level of disability or against the operative success rate. The level of detail required for such plots will be easily met by using the graphics capability (limited though it is) of the VT52 terminal. The 24x80 character screen of the VT52 can be easily represented as an array either in FORTRAN or MACLISP. The array will be of the byte-size variety to minimize space, since each array position will hold a single character for printing. The graphics program will first fill the graph array and then simply print its contents as an alphanumeric string.

This simple graphics scheme will allow us to display decision trees and sensitivity analyses. For applications requiring more than 24 lines of graph in the smallest dimension, we can concatenate several graphs together, essentially providing unlimited size in one dimension and 80 characters in the other. Trees or graphs of higher complexity will rarely be needed since it is the basic principle of clinical decision analysis to express the problem in the simplest possible terms, breaking it into sub-problems if necessary. One could argue that any tree or analysis which is too complex to be represented by such simple graphics is probably too complex to be presented to the clinician and should therefore be simplified into small sub-problems.

5.3.4 Sensitivity Analyses

The basic principle of a sensitivity analysis is to examine the stability or robustness of a given decision when the probabilities or utilities which form its basis are varied over reasonable ranges. In fact, there is another kind of sensitivity analysis which is less commonly performed because it is so complex and time consuming, that is, to vary the structure of the problem domain, such as changing the time horizon or adding neglected factors. The more classic type of sensitivity analysis will be fairly simply accomplished by the system described above. A given probability or utility can be varied and the resulting expected utility displayed as a function of that parameter. For most probabilities, the decision tree representation make the expected utility a linear function of

the probability so displays and threshold calculations are fairly simple. For those situations where a given parameter affects an outcome at more than one level of node (e.g., an annual mortality rate which is applied each year to survivors), the expected utility is no longer a linear function of the parameter.

In classic sensitivity analyses, each factor is varied, one at a time, while all other factors are held constant. In clinical problems, it is often of great interest to examine the effect of changes in more than a single parameter. The approach which we have developed for such analyses is the double-threshold diagram, where a graph is made plotting the threshold value of one parameter against the other parameter expressed as an independent variable. What is meant by threshold value is the value of the dependent parameter at which two courses of action have the same expected utility. Such a series of points describes a decision iso-preference curve which separates the decision space into 2 or more areas, corresponding to the various alternative actions. Any pair of values for the two parameters corresponds to a unique point in the decision space. Depending on which area of the decision space such a point is contained within, the optimum decision is read.

Another kind of sensitivity analysis corresponds to plotting a loss function based on a given parameter. Often it is of interest to note not only what the optimum decision is, but also to consider the magnitude of loss which would be expected if a sub-optimal choice were made. This is particularly important near the threshold point where expected losses are usually small and hence where mistakes cost the least.

Finally, decision analyses based on problem structure could also be simply accomplished by this system, although some expertise and guidance would need to be provided in examining the kind of structural modifications which would be fruitful analyses.

5.3.5 Utility Assessment

One of the least well understood areas of clinical decision making centers around the issues of utility scales, that is, some means of ordering the relative worth of various outcomes. Scales such as survival, life expectancy and dollar cost have been used traditionally. Our group has had some experience in another approach to this problem, that is, involving the patient in the decision making process as a direct source of utility data. Most of our experience has been with using the lottery technique [69, 88, 91, 93] for such assessments in both discrete problems (i.e., genetic counseling) and in continuous variables like survival in lung cancer. We have recently had extensive collaboration with cognitive psychologists (Duncan Luce, David Birnbaum, Amos Tversky), utility theorists (Howard Raiffa, Richard Meyer, Donald Shepard, David Bell) and other workers in the medical decision analysis area (Barbara McNeil, Milton Weinstein). These discussions have allowed us to develop some further ideas about utility assessment. These ideas are still developing actively, but include approaches such as 1) realistic scenario presentation, 2) separation of utility and value by approach to risk, 3) using marginal value function to build up total value functions, 4) using iso-preference functions to reduce multi-dimensional problems to equivalent single utility representations, and many more.

As these ideas become further developed, we shall incorporate them into the layer of utility assessment programs. For now, such programs will deal largely with the lottery technique and simple discounting. We plan to develop graphic means of displaying utility scales and of showing the expected utility of given courses of action on the utility scale of certain outcomes. To the extent

that the calculated expected utilities may tend to bunch near the center of broad utility scales, appropriate graphic devices for expanding and magnifying portions of the utility scale will be developed.

Our purpose, at least initially, is not to develop computer programs which will interact directly with patients in the utility assessment task. Rater, we hope to provide an approach to utility assessment which the physician might carry to the bedside. The programs will help the physician pose meaningful questions about the particular problem being analyzed and will show graphically the implications of various potential patient responses on a utility scale.

5.3.6 The Calculation Of Outcome Probabilities

The simplest approach to outcome probability calculation will be the implementation of a simple program to perform Bayes Rule calculation. Such a program is trivial and already in hand [44]. However, the major problem with the clinical application of Bayes Rule is understanding its limitations in terms of both the common assumptions of conditional independence and the deeper limitations of completeness and mutual exclusivity [124]. We shall develop a program which will help the physician examine his assumptions about independence, completeness and exclusivity and develop a formulation of Bayes Rule which is clinically appropriate. Directly relevant to this area of the program is the clinical cognition sub-project of this grant in which the ability of expert physicians to correctly assess probabilities will be examined. Also relevant is a small project now under way at the New England Center Hospital in which the ability of physicians at different levels of training to assess the clinical likelihood of myocardial necrosis in patients who present with possible myocardial infarctions is being assessed. The section of this proposal concerning clinical cognition is also relevant to this work.

A second and more complex area of outcome probability assessment centers around the issue of survival curves. It is rare that such curves are known for a given patient with a particular disease. What is often known is the details of a small portion of the curve, the survival curve for the general population, the relative burden of a given disease relative to the normal population or the annual likelihood of certain morbid events in various regions of the survival curve. In our experience doing clinical decision analyses we have found several techniques of frequent use. First is the construction of a survival curve based on the normal populations survival behavior and some knowledge of the excess death rates of disease, both treated and untreated. It was this approach which allowed us to model the survival of patients with bronchogenic carcinoma [69]. Another technique used frequently is the development of a Markov state transition model with treatments. Such Markov models pose great computational burdens, particularly when sensitivity analyses are performed on the transition probabilities which contribute to the model. Computational representation of such models is quite simple, however. We plan to implement both of these approaches to help the clinician model the survival and health status of patients under various plans.

5.3.7 The Construction of Decision Trees

The most complex aspect of the project will involve the design of heuristic programs to help physicians and students design and simplify decision trees and decide which prototypical analyses might be appropriate for a given clinical problem. These programs will rely heavily on the techniques of artificial intelligence and may well involve examination of the structure of

preliminary trees. Our prior experience in teaching these techniques to students and house staff has demonstrated some rather characteristic errors which can be explicitly looked for. The beginner often creates trees that are too complex and bushy. Such trees often contain repetitive sub-trees, or subsidiary decision problems which can be simplified considerably. In addition, the beginner often forgets to consider the potential errors of certain kinds of test results. For example, when the pathologic report of a biopsy is reported as inflammation, the novice may not consider that this result could be erroneous and that tumor might still be present. Thus, by creating models of common errors and common classes of clinical problems, we hope to use heuristic programming techniques to help the clinical analyst avoid such mistakes.

Our experience suggests that the simplest approach to tree construction is to express the problem in its cleanest, starkest characteristics, eliminating all but the most important aspects. It is far simpler to add complexity to a simple analysis than it is to simplify an overgrown bushy tree. The approach of the Hodgkin's Disease Project [108, 109] at this hospital has been quite the opposite -- they grow the tree in its bushiest detail, letting the computer "automatically" neglect those branches which contribute little to the expected utility. Such a computationally heavy approach depends on the existence of any extensive data base, such as the one which they developed over the past four years for Hodgkin's disease. The problem facing the individual clinicians with an individual patient is quite different -- his data base is sparse and any problem structure which requires detailed knowledge of may different outcome possibilities would be unmanageable. The watchword must be simplicity.

We will attempt to model two approaches to tree generation -- building up from a simple tree and beginning with a known tree for an analogous problem. We hope to develop programs which will encourage the user to begin with the simplest possible tree and slowly add complexity. We plan to include heuristic advice about common errors as those situations develop. In addition, the program will continually search an ever enlarging data-base of problem structures (to which each completed tree will be added). When the problem or a sub-tree of the problem seems structurally similar to a prior tree, the program might alert the user to that similarity. Two purposes will be served. First, the user may be able to draw on the experience of the known problem structure, saving time and avoiding certain common errors. Second, as structural correlations among problems are recognized, we will become alerted to certain recurrent themes which occur sufficiently frequently to require extensive theoretical analysis, both for the sake of the analysis itself and to allow the addition of heuristics for that type of problem to the tree construction package. Obviously, this domain structure matching process will alert the user when certain already analyzed prototypes (like the 'treat-no treat' prototype) are appropriate and would expedite the analysis.

5.4 Human Subjects

In the first several years of development of these programs, applications to actual consultations are not anticipated. In the later years of this project, these programs may be used by members of the Clinical Decision Making Group in the course of providing clinical consultation. All interaction will be with the consultant physician and not the physician responsible for the primary care of the patient. These consultations will be, as is now the practice, overviewed directly by an attending physician who is a member of the clinical decision making group. Thus these programs will assist physicians in established decision making roles. When the programs are ready for such clinical use, a protocol for such use will be submitted to the Human Investigations Review Committee of the

Hospital. All patient information entered into the computer will bear no patient identification data. Since these programs will be assisting physicians in their established roles and will provide computational support but will not alter such roles and since all output will be checked by hand, it is not anticipated that informed consent of the patient would be required. If, however, such consent is felt necessary by the Review Committee in later years, then such consent will be obtained.

5.5 Significance

Clinical decision making forms the backbone of the practice of medicine. The development of new, explicit, logical approaches to such decisions offers opportunities both for improved patient care and for improved education of students and physicians, but we are concerned that these explicit methods have required the presence of experts in decision analysis and that the graduate physicians are resistant to learning the techniques. We hope, by developing a computer-based decision analysis program, that the methods of decision analysis could be brought to the bedside without mounting a massive educational program. As the clinical application of decision analysis has found increasing acceptance in medicine and as both society and the medical community have become more concerned with the effectiveness of medical decision making and medical care, the dissemination of the these approaches will become more important. The development of a "clinical decision analyst's apprentice," as described here, could have an important impact on the quality and effectiveness of medical decisions and such educational goals might be quite appropriate for the National Library of Medicine to support.

6. The Diagnosis of Birth Defects

Personnel:

Stephen G. Pauker, M.D., Co-Principal Investigator
Peter Szolovits, Ph.D., Co-Principal Investigator
Marylou Buyse, Collaborator
James P. Donohue, Collaborator
Howard B. Sherman, Research Assistant

6.1 Introduction

6.1.1 Objective

To develop an improved diagnostic algorithm for use with the Birth Defects Information System being developed by the National Foundation.

6.1.2 Medical Background

In the United States alone, more than a quarter of a million children are born with birth defects each year. In addition to the children's lives, the emotional, financial, and spiritual lives of their immediate families are greatly affected. Birth defects have an impact on the daily lives of more than 25 million Americans. The Birth Defects Information System is directed toward helping children, their families, the community, and ultimately, future parents and their relatives.

In the past, it has been extremely difficult to locate complete, up-to-date information on any given birth defect. This information usually has been scattered among journals and textbooks in different medical specialties, with no centralized source.

The National Foundation-March of Dimes initially attempted to resolve this problem by compiling the Atlas & Compendium of Birth Defects, which contains known information on birth defects in the standardized format for efficient reference. The first edition of the Compendium appeared in 1972, with more than 300 physicians from various medical specialties as contributors. Soon it became obvious that printed editions of the Compendium could not keep abreast of rapidly increasing information in the vast field of birth defects. This led to the conclusion that only computer technology could provide instantaneous update of information as required.

The Birth Defects Information System is a computer-based system that can assist physicians, patients and their parents and relatives in obtaining information about birth defects for clinical and research uses. BDIS is operated by the Center for Birth Defects Information Services, a division of Tufts-New England Medical Center, funded by the National Foundation-March of Dimes.

The basic data for the system consists of information on most known birth defects and has been gathered and assembled by The National Foundation-March of Dimes. The computer implementation is based on research executed by the Massachusetts Institute of Technology Sloan

School of Management and Tufts-New England Medical Center.

6.1.2.2 Use of the System

Providing information about birth defects. Every day more than 600 babies are born with birth defects. Some are immediately apparent, while others are hidden and may not reveal themselves until later in life. Some are rare. Many defy diagnosis. Some are genetic, while other have obscure origins.

Parents want all available information. Some common questions are: What is the risk of occurrence of the birth defects in other members of the existing family? What would be the effects on future generations? What is the prognosis for the afflicted child's recovery, or what are the child's special needs? Parents usually consult local physicians. Due to the specialized nature of the birth defects, the physician often refers them to a birth defect center. Until now, the birth defects center performed the slow and arduous task of research to attempt to answer the parents' questions.

Information also may be sought by prospective parents after learning about birth defect in the family. What are the chances that one of their children will be similarly afflicted?

The BDIS provides information through queries via computer terminals. A summary of all known medical information about more than 1,000 of the known birth defects is presently in the computer and can provide many types of medical information, ranging from the minimal diagnostic criteria for a birth defect to other known references and experiences relating to it. Work to gather data is continuing on the remaining known birth defects.

6.1.2.4 Potential Future Uses

Assisting in recognition of new birth defects. Attaching a name (hence a history, prognosis, etc.) to an abnormality exhibited in a patient is important since it allows a physician to recommend treatment and to predict the future course of events for the afflicted child and the child's family. This name allows a pool of information that has been accumulated over the years to be brought to bear on the patient's condition. Before the computer system was developed, however, a physician, medical team, or birth defects center in one area might observe a particular abnormality, but be unable to match it with an existing known birth defect. This case might then be filed away as an unknown syndrome. At some later date, a physician in another community may recognize the same abnormality, but since he also realizes it is not one of the known birth defects, he also has no choice but to file it away. Only by coincidence would those two physicians ever get together. Further, with the number of such unknown birth defects being observed every day (over 40 per cent of birth defects patients at Tufts-New England Medical Center have been classified as unknown), it is impossible for a team of physicians to recognize groupings and patterns.

With the aid of the computer, the problem of classifying new (or previously unknown) birth defects will be greatly alleviated, and the system itself assists in grouping similar unknowns. For example, a physician inputs the signs and symptoms observed in a patient, and the computer then suggests possible syndromes that match those signs and symptoms. If the physician determines that these suggested syndromes are not applicable, he or she declares to the computer that those signs and symptoms are associated with an unknown syndrome. The computer stores this information in its memory. At some later date, when similar signs and symptoms are recorded by

another physician, the computer can recognize the similar pattern and outputs a message to the center, noting that the two unknowns are similar. It keeps track of all physicians who have observed similar unknowns so that they may contact each other and discuss their cases.

In the future, the BDIS will assist the March of Dimes in publishing Syndrome Identification, its periodical on unknown birth defects. The National Foundation-March of Dimes will be able to publish similar "unknowns" that appear to have been recognized by several people.

Printing of updated Birth Defects Compendium. Over the past several years, the March of Dimes has drawn together an enormous amount of information on birth defects, in the Birth Defects Atlas and Compendium. Since the BDIS has all this information, as well as more updated information stored in its memory, subsequent versions of the Compendium can be produced by the computer, using computer-based phototypesetting techniques. This greatly reduces the cost of producing updated versions of the Compendium. The March of Dimes will continue to sponsor the ongoing data-gathering to maintain the updated Compendium.

Training medical students. The information system may be used to assist in the teaching of diagnostic procedures and medical terminology and to provide information on birth defects to future physicians.

Acting as an early warning system. Since this information system can gather information on birth defects throughout the entire country and receive information on unknowns throughout the country and perhaps the world, it has the potential to recognize clusters of abnormalities, either high occurrences of known birth defects or new birth defects, occurring in a particular geographic area.

Assisting in research of causes of birth defects. The programming tools used to construct this information system may also be used to construct other databases and statistical analyses. That is, if a pool of data exists on a number of patients then various statistical tests can be performed to trace possible causes of that birth defect.

6.1.2.6 Assisting in Diagnosing Birth Defects

One of the most important projected uses of the Birth Defects Information System is in assisting in diagnosing birth defects. It is extremely important to associate a name, pertinent historical information, and all other available information with an observed abnormality in a child. Once the abnormality is so identified, meaningful diagnostic treatment, prognosis for the future, and preventive mechanisms can be undertaken in the child as well as in relatives to protect future generations. However, many pediatricians do not encounter birth defects as frequently as they see more common childhood diseases. Therefore, the pediatrician often refers the child to a birth defects center.

But even in a birth defects center, it is difficult and often unproductive to try to identify a particular combination of abnormalities in a child and associate it with a known syndrome. Since the computer system can look at many combinations of facts related to the patient in question, it might offer some assistance to the physician attempting to diagnose an unknown birth defect in an individual child. This work is quite important and is being pursued both by the National Foundation and by an independent group based in Belgium.

The work done to date by the group sponsored by the National Foundation has centered on data base creation. They have compiled a relatively complete data base of birth defects and now hope to turn their attention to the problem of diagnosis. The work done so far has only been of limited success. The current system is based on a hierarchical network of signs and symptoms, such that the mention of a sign or symptom is automatically expanded into a set of synonyms, related findings, and super-set findings. The user is questioned about the finding to generate sub-sets of more specific findings. All this manipulation is necessary since the data base was created from numerous separate reports using variable terminologies and classifications of findings. Rather than stylize the data base into standard terms, those researchers have chosen to expand the input case description into a set of synonyms. The current matching algorithm contains substantial logic to develop appropriate matches when certain modifiers and specializations of findings are not specified or not known. Two scoring algorithms are used: the match of the case findings to each diagnosis is calculated and the highest ranking candidates are considered the best matches; alternatively, each finding is assigned a score inversely related to the frequency of that finding in the data base and, for each diagnosis, the sum of the scores for matching findings is normalized by the sum of the frequency scores for all findings. The first algorithm ignores specificity and incompleteness of data base description, while the second assumes that all diagnoses have equal prior likelihood.

The National Foundation Birth Defects Data Base is currently resident on an IBM System/370 under the VM/370 operating system. There are approximately 10 million characters of storage. A subset of the complete data base is also available which deals exclusively with skeletal birth defects (that is, anomalies of the bony skeleton) and can be used as a testing environment. The data base consists of two parts -- a hierarchical list of findings and a book of diagnoses. The finding list, in its current form, gives each finding a unique identifying number and provides links to all diagnoses which mention that finding. The identifying numbers provide the hierarchical order to the finding list. For each diagnosis, findings are grouped into sets. Each set is provided with a weight, on a scale of 1 to 5, which describes the importance of that set of findings in making the diagnosis. Credit is not given for the set unless the criterion of set presence is met. The presence criteria are quite simple -- for each set, a number is assigned and that number of findings in the set must be present to determine that the set is indeed present.

The work developed by the Birth Defects Group to date has not been based on work in similar diagnostic domains done by workers in the field of Artificial Intelligence in Medicine (AIM).

6.1.3 Computer Science Background

The kind of pattern matching diagnostic task to be done here has been approached by two groups in the AIM community -- the Pittsburgh group working on the INTERNIST system [94, 95, 96, 97] and the Boston Group working on the Present Illness Program [85, 122]. Both of these systems have been described in detail in a recent publication [124], but in summary: the INTERNIST system creates a hierarchy of diseases (not of signs and symptoms as the Birth Defects Group has done) and then describes each lowest level (most specific) disease in terms of atomic findings. Each finding for each diagnosis is assigned two numbers -- a frequency relating to the conditional probability of the findings occurring in the disease and an evoking strength relating to the posterior probability of the diagnosis given the finding. Based on these numbers, a complex partitioning algorithm creates some simple alternative hypotheses and proceeds to evaluate them by gathering more data. The Present Illness Program uses a network data structure with diagnoses

separated into diseases, clinical states and physiological states. Each set of finding constitutes a frame and frames are linked together both causally and associatively. Certain findings are given privileged trigger status which allows them to evoke the active consideration of certain diagnostic possibilities. Scoring is accomplished both locally within each frame and cumulatively by propagating scores from related frames. Detailed logic is included to allow the expansion of findings as they are reported and to provide appropriate matching logic for inadequately specified findings either in the patient data base or in the frames of diagnostic knowledge.

6.1.4 Rationale

The Birth Defects Project has been limited by a sub-optimal diagnostic algorithm. Our Present Illness Project has been limited by the lack of resources for extensive data base creation. Therefore, a collaboration between our groups seems quite logical. We hope to apply the new computer science techniques developed in the Present Illness and INTERNIST projects to the Birth Defects data base and provide a more effective diagnostic algorithm. Conversely, we hope to use their rather complete data base to explore the diagnostic theories we have developed. In addition, the need of the Birth Defect Project might be met by either the INTERNIST or the Present Illness approach. We hope to implement their data base in a fashion which will allow the operation of both diagnostic algorithms so they can be contrasted, compared, and hopefully melded into a single more powerful approach.

6.2 Specific Aims

i. To convert the National Foundation Birth Defects Data Base into a form which can be used by either the INTERNIST or Present Illness program algorithms.

ii. To implement a version of INTERNIST in MACLISP so it can be tested and modified in accordance with the needs of the Birth Defects Project.

iii. To implement two separate diagnostic algorithms, an INTERNIST based one and a Present Illness based one, both of which can operate on the Birth Defects Data Base.

iv. To compare and contrast the performance of the two algorithms.

v. To form a new combined algorithm which incorporates the best performance features of each but which is tailored to the Birth Defects Problem Domain.

6.3 Methods of Procedure

Our first task will be to convert the Birth Defects data base into a form compatible with both the INTERNIST and Present Illness systems. We will then implement the two separate diagnostic algorithms and compare their performances. Finally we shall develop a new, combined approach and test that algorithm in the Birth Defects network of diagnostic centers.

The birth defects problem domain is somewhat restricted relative to the more general domains being explored by the INTERNIST and Present Illness systems. The time course of the presentation of the birth defect is relatively compressed, so issues of time might be neglected at first pass. This is not strictly true since some defects evolve over time and in that evolution the findings

change somewhat. However, in its current form, the birth defects data base ignores most of these transitions. The set of findings used by the data base consist largely of physical examination descriptors, with some historical data and some data from routine x-ray examination.

Both the Present Illness and INTERNIST domains necessitated consideration of costly tests and complex laboratory findings. The birth defects domain will be of particular interest because the diagnosis of many birth defects is based largely on dysmorphology and hence appropriate organization and matching of physical findings will be of paramount importance. Both the Present Illness and INTERNIST systems emphasize the sequential nature of the clinical data gathering task. In general, consultations provided by the birth defects diagnostic system will be based on data already in hand and issues concerning the ordering of data gathering (the question selection task) should not be a major problem. We anticipate that this problem domain will be helpful in developing better hypothesis evaluation techniques (that is, scoring algorithms) since the evaluation and matching task is the crux of the diagnostic problem in this domain.

6.3.1 Data Base Implementation

Our first task will be to implement the Birth Defects Data Base into a form processible by MACLISP [73] for the two diagnostic systems. Physical transfer of files will be easily accomplished since the IBM machine now containing the data base is at MIT and similar transfers to the research system used by the clinical decision making group (a 512K word PDP-10) are a common practice. For the INTERNIST system, the data base will need to be reduced to atomic form. Since unique English identifiers already exist for each finding, this transition should be straightforward. We shall also need to implement a synonym dictionary and to provide importance weights to each finding. Implementing the system using the Present Illness program will be slightly more complex. In that program, each finding is represented as a nested list of properties and values. Thus, the eight hundred odd findings in the data base might collapse into a more compact set. for example, cafe-au-lait spots with irregular outlines and cafe-au-lait spots would be represented as

```
(CAFE-AU-LAIT-SPOTS (STATUS PRESENT) (OUTLINE IRREGULAR))
```

and

```
(CAFE-AU-LAIT-SPOTS (STATUS PRESENT))
```

The property list shown here raises an important point -- that diagnostic information is carried by both the presence and the absence of findings. The INTERNIST formalism can only convey that information if separate atoms are created to note the presence or absence of a finding. The fact that

```
(CAFE-AU-LAIT-SPOTS (STATUS ABSENT))
```

automatically excludes

```
(CAFE-AU-LAIT-SPOTS (OUTLINE IRREGULAR))
```

from being present is easy to represent in the Present Illness formalism but much more cumbersome in INTERNIST. In fact, the entire finding array with appropriate logic of matching supersets and partial descriptions has been implemented in a discrimination net formalism in the most recent version of the Present Illness Program. Extension of that mechanism to include synonyms should be straightforward.

Once the finding domain has been represented, our attention will be turned to representation of the descriptions of the diagnostic entities -- that is, the birth defect syndromes. The present data base separates findings into sets and contains matching criteria for each set and a single weighting factor for each set. This representation is not dissimilar from the INTERNIST mechanism in that the weight factor seems to be analogous to the frequency score. We will have to expand the data base to include evoking strengths for each set of findings in each syndrome. To the extent that the data base is complete (which is unlikely but which is the assumption of the current diagnostic algorithm being used by the National Foundation program), a first pass at specifying evoking strengths may be done automatically from the data base by noting the relative frequency of findings in various syndromes, assuming equal prior probabilities. That calculated estimate will then be reviewed and modified by the pediatricians and clinical geneticists who are responsible for data base generation.

Syndrome representation in the Present Illness system will be slightly different. Each syndrome will be denoted by a frame and "differential diagnosis links" will be made to related frames. The sets of findings within each syndrome might be represented either as another frame (a finding complex type frame) or as a scoring clause within the syndrome frame. The latter route will be simpler but will make scoring somewhat more complex, requiring a detailed consideration of score propagation. The latter mechanism may require a fair amount of duplication between frames.

6.3.2 Program Implementation

The Present Illness program is already implemented directly in MACLISP [73] and is a functioning program, albeit on a toy-sized data base. Few additions to that program will be required to achieve initial operational status. The INTERNIST program is now implemented in INTERLISP on the SUMEX-AIM computer. Details of the program have been only sketchily reported by the Pittsburgh group, but the diagnostic algorithm is described in sufficient detail to allow us to implement it with little difficult. The major effort and complexity of the INTERNIST program has been concentrated on the creation of a huge data base covering virtually all of internal medicine. For the birth defects problem domain, we do not plan to share any of that data base. Hence we shall only need to implement the core algorithms. Compared to the internal medicine data base, the birth defects problem domain is quite small and we do not anticipate any problems in fitting the data base or the program core image into our PDP-10 computer. We shall seek the counsel and cooperation of the Pittsburgh group in implementing their central algorithm in MACLISP.

6.3.3 Algorithm Comparison

Once the Birth Defects Data Base has been implemented in both the Present Illness and INTERNIST formalisms, we plan to run a series of known cases through each to compare performance. We expect that each program will appear better in certain kinds of cases, although neither will provide clearly superior performance. We shall try to classify the strengths and weaknesses of each algorithm based on performance, types of failures, and the nature of the cases which cause success and failure for each.

In a later stage of comparison, the two diagnostic algorithms will be given a set of actual unknown cases being entered into the Birth Defects consultation system. Performance of each algorithm will be compared to performance of a clinical geneticist actually seeing the patient in

consultation. When comparison is made to a clinical geneticist examining the patient, the appropriate contrast would not be to the program alone but to the program and a primary care physician acting in consort. The purpose is not to provide performance which is superior to the geneticist, but rather to bring the skills of the nonspecialists up to a higher level than he could accomplish through unaided decision making.

6.3.4 Algorithm Development

The next phase of this project will be to examine the three diagnostic algorithms (National Foundation, INTERNIST, and Present Illness) and develop a combined algorithm which incorporates the best features of each when addressed to the birth defects problem domain. We suspect that the major strength of INTERNIST will lie in its partitioning algorithm and its evaluation of both frequency and selectivity (evoking strength) of a finding. We expect that the strength of the Present Illness program will lie in its network organization of frames and its logic for matching findings to frames. The Present Illness program has also shown strength, in other problem domains, in the areas of context formation and hypothesis generation. The strength of the National Foundation algorithm may well lie in its close association with the data base and the extent that it reflects the current style of reasoning of clinical geneticists.

We plan to implement a combined algorithm in MACLISP and to test that algorithm in an extensive array of cases, including all cases used to test the separate algorithms. Once performance is ascertained to be at an acceptable level, we hope to implement a streamlined version of the diagnostic algorithm in a form that can be efficiently run on the IBM 370 system which the National Foundation plans to use as the major hardware support for this project when it reaches the level of production.

6.4 Human Subjects

This project will only develop a diagnostic algorithm for use on the Birth Defects Data Base. In the first several project years it will not be used in actual cases. In the later years of this project, that algorithm may be used with the Birth Defects Program. In that circumstance, the Human Subjects protections already in place for that clinical research project will stand. We will simply be replacing one program segment with another and will not change the design of any experiments or the use of any human subject data. Any cases entered into either the developing data base or tested with our diagnostic algorithm will contain no identifying information capable of linking that case to a named individual.

6.5 Significance

The diagnosis of birth defects is an important task to which the National Foundation has already devoted significant resources. We hope to be able to combine the technology developed by the Artificial Intelligence in Medicine community with that data base and thereby improve the effectiveness of that project in diagnosing birth defects. In addition, the data base will afford us the opportunity to apply our technology to a new domain and thereby make it somewhat more general in its application. Finally, we hope to compare our approach to that of another AIM group and eventually combine the technologies into a stronger diagnostic algorithm.

7. Multi-Modality Therapy Design: The Protocol Writer

Personnel:

Robert H. Friedman, M.D., Principal Investigator
John G. Krikorian, M.D., Investigator
Peter Szolovits, Ph.D., Investigator
Research Assistant, to be appointed

7.1 Introduction

7.1.1 Objectives

Our objective is to apply AIM techniques to the process of therapy planning, particularly in the design and implementation of cancer treatment protocols. We propose to study the structure of ideal and currently used protocols and to develop a computer program, the Protocol Writer, to aid physicians to design new protocols based on our models of their structure. We propose to study the use of formal protocol models to direct and monitor the execution of protocols. We also plan to investigate the possibility that partial knowledge of the kinetics of cell growth, the actions of therapeutic agents, and the abstracted results of previous studies can be used to aid the designer of new protocols.

7.1.2 Background: Cancer Treatment Protocols

At any moment, there are hundreds of separate cancer treatment protocols being used actively by physicians to treat cancer in this country and abroad. In the United States, approximately 20% of all cancer patients are treated under the aegis of a formal cancer treatment protocol. In some cancer types, namely acute leukemias in children, more than 80% of all patients in the United States are treated by physicians utilizing a formal protocol. Cancer protocols are designed either to answer clinical research questions such as the relative efficacy of different drugs or are solely used as guidelines to treat patients. In the natural history of protocol development, a treatment regimen shown to be beneficial in a clinical research study may become the "standard" non-experimental treatment regimen for that disease (until it is supplanted by a "better" approach). A new regimen may be tried because it incorporates a new therapeutic agent thought to be beneficial, it uses previously-known agents in new combinations, or it uses a previous combination of agents in a modified sequence or dose schedule [16, 19, 33, 53].

At the present time, the cancer treatment world is extremely active with multiple clinical research studies on every type of cancer occurring simultaneously [16]. In order to rapidly evaluate a number of treatment regimens, institutions have banded together to form cooperative groups in order to undertake multiple protocol studies. By this means, it is possible to enter an adequate number of patients for statistical analysis within a short period of time, usually within a year. A number of cooperative groups have been set up in the United States over the past decade. These include the Acute Leukemia Group B (ALGB), the Southeastern Cancer Study Group (SECSG), the Southwest Oncology Group (SWOG), the Radiotherapy Oncology Group (RTOG), National Surgical Adjuvant Breast Program (NSABP), and the largest, the Eastern Oncology Group

(ECOG). In smaller groupings, institutions or physicians in oncology may also conduct studies. These typically involve institutions in the same region, in the same medical center complex, or occasionally within a single hospital. It is true that most patients treated under the aegis of an experimental treatment protocol will be cared for in an academic teaching hospital with an active clinical and basic science research program. However, with the prodding of the National Cancer Institute (Division of Cancer Control and Rehabilitation), academic centers have begun to involve community hospitals in the experimental protocol process. This has resulted in an even larger number of patients treated under such protocols and an increasing number being treated in their community institutions. This situation has been accelerated by the self-interest of the teaching hospital which is able to accrue patients more rapidly for experimental studies if it involves community institutions. These contacts may also result in greater referral of some patients to the tertiary institution. Furthermore, it is in the self-interest of the community hospital to enroll its patients in some cancer treatment protocols if they are conducted within its walls. They are thus able to attract patients seeking the best in therapy who otherwise might go to another community hospital, or more likely, to a regional academic teaching hospital. This growth in the numbers and spread of experimental treatment protocols has also resulted in the increasing use of "standard" protocols both at community and teaching hospitals. In many settings, oncologists will treat almost all of their patients on either research or non-research protocols, with each physician using as many as fifty different treatment protocols.

7.1.2.2 The Structure of Cancer Treatment Protocols

Experimental cancer treatment protocols are designed by individual cancer clinicians or groups interested in a particular type of cancer and/or modality of treatment. Large cooperative groups frequently have formal mechanisms for review of treatment protocols through committees of clinicians and statisticians. Smaller groups may have the same degree of review or the review may be quite cursory. In any case, protocols tend to be written in a similar format [75]. In the beginning, there is an introduction section in which the background and rationale for initiating the study is presented. This is usually followed by an objectives section which states the research questions the protocol is attempting to resolve. Frequently, several treatment regimens are compared and assessed by objective measures of patient response. Another frequent objective is to compare the response of different types of patients (by histology, anatomic stage of disease, symptomatic stage of disease, presence of other biologic factors) to the same therapeutic regimen. Next, the selection of patients is defined by what categories of patients are eligible and which are ineligible for entrance into the study. Criteria frequently employed include type of cancer, its histology, and the extent of disease. The presence or absence of specific previous treatment may be considered. Patients with certain co-morbid diseases or organ system dysfunctions (frequently hepatic, renal, and bone marrow function) may be excluded. There may be considerations of age and sex and other demographic characteristics of the patient. Finally, most studies require that the patient have some measurable evidence of disease by which to assess the success or failure of the treatment. All studies define a minimum set of tests or procedures required at or near the date the treatment commences, to establish baseline information. These include blood tests, radiologic and nuclear medicine procedures and biopsies or other surgical procedures. Next is a section on randomization which outlines the procedures statisticians will employ in allocating treatment to specific patients. The treatment regimen section follows, detailing the treatments to be employed within the protocol. These may include surgery, radiotherapy, and medical therapy. This section is the heart of the protocol and is usually the lengthiest and most complex. Of particular importance are the specifications of the treatment time and dose schedules with rules for time and dose modifications.

Since these modifications frequently depend upon tumor growth or response and treatment toxicity, methods for defining these parameters are specified in the protocol. Since other host factors unrelated to the growth of the tumor or the toxicity of the treatment may effect the treatment, these must also be defined. These requirements necessitate the selection of specific symptoms, physical examination items, and tests to be performed and assessed on a regular schedule. Finally, there is a protocol section on supportive therapy describing allowable palliative and other supportive measures that can be used without interfering with the conduct of the research study. These include the administration of blood, antibiotics, and occasionally localized radiotherapy as long as they do not interfere with the interpretation of the experimental study.

Non-research cancer protocols are similar to experimental protocols but are briefer. The introduction section provides the rationale for using the treatment. The objectives section (frequently combined with the introduction) indicates the expected response, either extended survival or a longer period free of disease symptoms (compared to no treatment). Patient selection is more loosely defined, usually limited to disease entities, such as "metastatic breast cancer". There is no randomization section. The treatment and supportive therapy sections are similar but usually do not require as extensive data collection as experimental protocols.

7.1.2.4 The Treatment Regimen

The heart of the cancer protocol is the treatment regimen section, containing a precise description of the treatments to be initiated, including a schedule and a set of treatment decision rules. Usually for chemotherapy and radiotherapy, treatment is defined in terms of a "cycle". A cycle is a basic sequence of therapy that is repeated. For example, a 28 day cycle may include the administration of two agents with different toxicities and different tumor effects, with agent A given on day 1 only, and agent B on day 15 only. Beginning on the 29th day, the treatment cycle (Cycle 2) begins again with agent A. Treatment is cycled primarily to deal with the inherent toxicities of the agents allowing the patient to recover. In some protocols, a cycle is set up in order to permit the sequencing of agents to kill tumor cells maximally. In the example, each agent is given every 28 days in order for the patient to recover from its unique toxic effects. In addition, agent B is given 14 days after agent A on each cycle because it kills tumor cells that agent A fails to kill. A treatment "course" is a series of cycles which complete a therapeutic regimen. The number of cycles in a course may be precisely defined by the protocol or in some cases may be variable depending on some outcome parameter such as tumor response. Within the treatment cycles and the course of therapy, the treatment may be modified by rules depending on the tumor growth or response and treatment toxicity. The protocol thus details the basic treatment regimen including therapy cycle and course, and any required modifications based on response to therapy or toxicity. The use of dose modification strategies is almost universal in treatment protocols. Frequently evidence of toxicity dictates the reduction, withholding or temporary postponement of some of the therapeutic agents.

7.1.2.6 Deviations from Protocol Rules

In experimental treatment studies, the participating oncologist extensively records details of the treatment process for each patient subject including specified examination and laboratory information, a therapy log, and evidence of response to therapy or toxicity. This information is reviewed by the Study Chairman's office, first for compliance to protocol rules and then for outcome. In recently conducted protocol studies there has been alarmingly high incidence of major

deviations from protocol rules [18, 70]. There are a number of reasons for this. When each oncologist may use up to fifty different protocols at one time in his practice, mistakes are obviously made. This has become particularly true in recent years with lengthier and more detailed protocols. Dr. Friedman and his colleagues at Boston University-University Hospital are developing a computerized Cancer Care Data Management System (CCDMS) which will facilitate the execution of cancer treatment protocols, and hopefully reduce the practical problems of using them. The protocol will be embedded in the operation of the Data System. Each time information is required by the protocol and each time a decision on cancer therapy is made, the Data Management System will prompt clinical personnel concerning information required and treatment decision rules they should use. We expect that protocol rules will be followed more uniformly. However, this system will not eliminate other problems inherent with currently written cancer protocols, namely, problems with (1) protocol ambiguity, and (2) insufficient "intelligence".

7.1.2.8 Cancer Treatment Protocols: Problems in Ambiguity

In the process of reviewing a large number of oncology treatment protocols written over the past five years, it appears that the more recent ones are being written in a less ambiguous manner. Nevertheless, there continue to be problems of definition. These problems divide themselves into three general types: (1) imprecise definition of terms, (2) ambiguity concerning time and dates, and, (3) inconsistent and conflicting dose determination rules. An example of an imprecise term appears in ECOG Protocol 3274 where the criterion "stomatitis judged as clinically severe" serves to modify the dosage of therapy during treatment. Another ECOG protocol written four years later, number 1578, uses the criterion "with less than three separate oral ulcers and eating: one-half dose MTX [methotrexate]. With equal or greater than three separate oral ulcers, or unable to eat, hold MTX until ulcers clear, after which MTX is to given at 75% of previous dose." This is clearly an improvement. Protocol 2878, uses the parameter "dangerous arrhythmia" which is too general and ambiguous. Protocol 3274 provides a more precise definition of "active heart disease": (1) recent (less than three months) myocardial infarction, (2) symptomatic coronary insufficiency, (3) symptomatic heart block, and (4) present evidence or past history of congestive heart failure. However, this same protocol states that "no hope of surgical cure" is reason to disqualify a patient for entering the protocol. Some definition of the extent of disease (which anatomical structures contain tumors) would be less ambiguous. Protocol 1377 uses a similarly vague term, "felt to be uncontrollable locally by either or both modalities."

Dates and times may also be imprecisely specified. Protocols frequently do not indicate when baseline information needs to be acquired. Can "baseline tests" be obtained after the initiation of therapy? Can tests performed six months previously be considered baseline? As mentioned above, all protocols require that specific information including symptoms, examinations, and laboratory tests be collected on regular intervals to assess tumor growth and toxicity of therapy. This is expressed in terms such as "obtain X every month," where X is some piece of information. The level of precision of these schedules is never well defined. If a test is to be obtained every three months as some x-rays are, what happens if it is obtained in two months? Or four months? Or four-and-a-half months? Are these violations of the protocol? Radiotherapy and drug treatment schedules are usually specified by particular treatment days. However, ambiguous situations exist when therapy is postponed because of toxicity. Few protocols define allowable limits of delay. At what point is the integrity of the treatment cycle interfered with by postponing therapy?

A third and major source of ambiguity exists with conflicting and imprecise dose determination rules. Protocols frequently have dose reduction rules for evidence of renal and hepatic insufficiency. Most protocols fail to comment when there is combined renal and hepatic dysfunction, with the implication that the more "severe" rule will apply. Protocol 1578 actually specifies that the clinician "use the lower of two doses". Frequently, the rules are very imprecise such as in Protocol 1377, in which "MTX may be stopped if the attending physicians think this is in the best interest of the patient." Protocol 1578 states that "pleuritic, erythematous, macular rashes [flat, red rashes that itch] of the trunk and extremities" call for the "reduction of drug dose [which] leads to normalization in a few days." The schedule for dose reduction is not specified. Protocol 3175 indicates that if WBC < 2,500, or platelets are < 75,000, no drugs will be given. The tests will be repeated weekly and the treatment will resume once the counts exceed these defined limits. What doses are given on resuming therapy (which depends on definition of cycle day) is unclear.

What is clear is that currently written protocols contain many ambiguities. Oncologists using a protocol have several options in resolving ambiguity. First, they can contact the author and ask him to resolve the question. Secondly, they can provide their own interpretation. Unfortunately, the latter approach is usually taken. However, even if the Study Chairman is contacted, there is no guarantee he will be consistent in his resolution of the ambiguity. Finally, he does not, as a rule, communicate his interpretation to other oncologists using the same protocol.

7.1.3 Cancer Treatment: Underlying Models

Since cancer treatment attempts to eliminate every tumor cell, agents are administered in ways that inevitably produce toxic effects and frequently produce severe and life-threatening complications. Authors of protocols must use strategies based on an understanding of basic biological processes. To do so, they draw on their understanding of the actions of the agents in cancer and non-cancer cells, the natural growth characteristics of cancer cells, and interactions of tumor, treatment, and patient.

In general, data from experimental laboratory studies are not directly translated into clinical treatment regimens. More frequently, the general principles illustrated in laboratory studies are adapted in the clinical studies [1, 33, 45, 58, 62, 80, 82, 136]. Interestingly, treatment protocols are frequently developed over time based on the results of previous studies. As each previous study is completed, the new experimental protocol will include a modification based on prior results. For example, in the treatment of lymphomas, a standard protocol called BCHOP was modified by shifting the administration of two of the agents, bleomycin and prednisone, from the first part of the therapy cycle to the middle of the cycle when it became clear that some patients developed recurrent disease during the latter half of the cycle. This change was supported by principles of cell kinetics related to the doubling time of the lymphoma cells [74].

Protocol authors commonly utilize several principles in attempting to mute the toxic effects of the therapeutic agents. One common strategy is to select agents with non-overlapping organ toxicities. For instance, a standard treatment protocol, CVP, employs cyclophosphamide which causes myelosuppression (reduction of the leukocytes) and cystitis (bladder inflammation) as its major toxicities. The second agent, vincristine, causes neither of those problems, but has frequent neurotoxicity. Prednisone, the third agent, causes none of the toxicities noted above. Thus CVP uses three agents that do not produce additive toxicity to any single organ system. A modification of this strategy is to select agents and a schedule of administration that results in non-overlapping

organ toxicities over time. Several treatment protocols include Adriamycin (doxorubicin), cyclophosphamide, and a nitrosourea. Each of these have myelosuppressive effects. However, the peak effect for cyclophosphamide is seven to ten days after administration, for Adriamycin it is seven to fourteen days, and for the nitrosourea it is three to five weeks. Thus, the peak effect of each agent falls at different times and thus a treatment cycle can be designed using all three agents administered simultaneously. Treatment protocols also take advantage of known pharmacokinetic actions of single agents. For instance, the chemotherapeutic agent 5-fluorouracil produces less toxicity if it is given in single doses over five days as compared to the administration of the same total dose on a single day. Moreover, if the five day administration is performed by continuous intravenous infusion, it is better tolerated than if each daily dose is given by rapid infusion over several minutes. These phenomena are not true for other chemotherapeutic agents.

A common strategy to limit toxicity is to reduce the dose of an agent when there is evidence of toxicity. This dose modification strategy is almost universal in protocols. Strategies for dose modification are related to the agents used. For instance, cyclophosphamide, Adriamycin, and vinblastine produce "sharp" toxic effects on the bone marrow. The time courses of their effects are readily predictable and the duration of the severe effect is quite limited in time, with patients almost universally recovering quickly following peak effect. Other agents such as the nitrosourceas, nitrogen musturds, and mitomycin C are very unpredictable in terms of the time course of their toxic bone marrow effects. With the former group of agents, it is possible to design treatment schedules in which dose reduction is not necessary when bone marrow toxicity is noted. With the latter group of agents, most protocols will require the reduction of drug dose when any evidence of bone marrow toxicity is present (see Figure 2). The dose of these agents is only returned to the baseline level when there is no evidence of toxicity. For example, a protocol using "sharp nadir" agents, the Dartmouth Oatcell Protocol, specifies that the methotrexate dose on day 21 of the cycle is given regardless of the leukocyte and platelet count on that same day because the bone marrow suppression from prior cyclophosphamide is predicted to disappear promptly. Protocols occasionally consider prior sensitivity of the patient to the regimen. For instance, ECOG protocol 1278 [26] requires that the dose of three agents given for "all subsequent courses of therapy" will be 75% of the calculated dose if the leukocyte count during any prior cycle of therapy is <2500 or the platelet count is <50.000. Protocols may also take into consideration other patient factors in modifying the dose of therapy. The CVP Protocol for non-Hodgkin's Lymphoma calls for the reduction of the initial dose of cyclophosphamide from 400 mg/m2 to 250 mg/m2 if there is extensive prior radiotherapy. Other protocols call for reduced doses of therapy for patients above a certain age.

Fig. 2. Example of a Dose Reduction Schedule

PERCENT FULL DOSAGE TO GIVE

		PLATELET COUNT		
		>100,000	75-100,000	<75,000
WBC	>4,000	100%	50%	0
	2500-3999	50%	50%	0
	<2,500	0	0	0

Protocol authors also draw on a number of principles in order to maximize the tumor cell kill. As mentioned previously, the modified BCHOP protocol changes the administration of bleomycin from day 1 to day 15 and 22 and the dose of prednisone from days 1-5 to days 15-28. This is combined with day 1 and day 8 administration of vincristine, cyclophosphamide, and Adriamycin. Cancer cells that escape death from the three agents given initially will be more vulnerable to bleomycin and prednisone when they are given in the latter half of the cycle. The protocol COMA also utilizes multiple agent sequencing to maximize cell kill by two cell kinetic principles. Cyclophosphamide is given on day 1 and vincristine is given on days 1, 8, and 15. These agents are active against cells which are dormant (not in the growth cycle). Initially, when the tumor mass is large, a greater percentage of the cells are dormant. When the size of the tumor mass decreases, the percentage of cells in a growth phase increases. Cyclophosphamide and vincristine, active against dormant cells, function to reduce the tumor mass, resulting in a greater percentage of cells entering cell cycle growth. These cells are then vulnerable to drugs that specifically attack growing and dividing cells. Thus, the protocol includes such agents, methotrexate and ara-C, given once a week beginning on day 22. Secondly, vincristine interrupts spindle formation in the cancer cell which tends to synchronize the multiplying cell population so that all cancer cells enter each phase of the cell cycle simultaneously. This results in a higher percentage kill when cycle-specific antimetabolites such as methotrexate and ara-C are used. Protocol authors need to be aware of the great cell kinetic specificity of various chemotherapeutic agents. For instance, ara-C is totally ineffective in killing tumor cells when given in a single intravenous dose. If it is given over 5 to 7 days, whether by pulse or continuous intravenous administration it is very effective. This occurs because the drug kills tumor cells during a very specific phase of the growth cycle. Since the doubling time of a tumor cell population may be as long as 100 hours, there may be a period of only several hours in which the agent is effective. In some instances, tumor cells become "resistant" to the effects of chemotherapeutic agents. For this reason, protocols commonly alternate cycles of therapy. For instance, a standard lymphoma protocol uses four drugs for one cycle followed by four different drugs for the second cycle. These drug combinations are repeated in alternating cycles throughout the course of therapy. Another strategy used to maximally kill tumor cells is to localize the site of therapy. Chemotherapy may be perfused into involved organs such as the liver or instilled into confined spaces such as the subarachnoid space in instances of central nervous system cancer. Radiotherapy may be administered to selective organ sites. For example, in acute lymphoblastic leukemia, the brain is selectively irradiated to prevent the delayed occurrence of leukemia in the central nervous system. In some protocols, agents are combined because they have selective combined organ effect. In the protocol M-BACOD, dexamethasone and methotrexate are included because they have central brain effects whereas the other agents have none.

These strategies are some of the fundamental ones employed by protocol authors in order to maximize cancer cell kill and minimize host toxicity. Over time, these principles have become more refined and we can certainly expect that they will continue to be improved and added upon as clinical experience increases and as the fruits of basic laboratory research are translated into clinical research studies.

7.1.3.2 Cancer Treatment Protocols: Insufficient Intelligence

Although a great deal is known about optimal strategies for writing protocols (see above), this knowledge is not applied uniformly. This is not surprising since there are a large number of independent strategies potentially useful in minimizing toxicity and increasing tumor kill and a large number of protocol authors. There are manuals available to protocol writers which outline

the desired content of a protocol [75]. These help insure uniformity in structure and completeness, but do not help the author with the process of defining protocol rules. They deal more with form than with substance. This is not surprising for it is not a simple task to guide an author through efficient steps to create an optimal protocol. At the present time, writing protocols is more an art than a science.

For these reasons, protocols differ in their degree of sophistication. In the above section, an example of utilizing prior results in a decision rule was demonstrated from ECOG protocol 1278. In this protocol, the dose of three drugs given in future cycles of therapy is reduced based on drug toxicity in the present cycle of therapy. This is combined with a second independent decision rule stipulating reduction of therapy in the current cycle if toxicity is present. Thus, it is possible to "double reduce" the dose of therapy based on current and previous toxic effects. In most protocols utilizing the same agents, there is only the rule dealing with dose reduction for current toxicity.

The generation of protocols in treatment of the lymphomas (discussed in the previous section) is an example of increasing sophistication and intelligent use of biologic principles. The modified BCHOP protocol better reflects the lymphoma tumor cell kinetics than did the previous (BCHOP) protocol. Of interest, it also resulted in better treatment outcomes.

Insufficient "intelligence" in treatment protocols can cause practical problems, particularly in experimental treatment studies. A major tension in treating patients under the aegis of experimental protocols occur because the physician caring for the patient is at once the clinician whose interest is the patient's welfare and the scientist whose interest is the integrity of the experimental study. If he feels a treatment rule specified by the protocol is not "intelligent" he will likely violate the rule. Since protocols usually do not prescribe conditions for removing patients from study, the patient may continue in the study. Retrospectively, the case may have to be removed from analysis because of protocol deviations. If the clinician perceives that the protocol has many unintelligent rules, he most likely will not use it or will more frequently violate its stipulations.

7.1.4 Rationale

We believe that the domain of multi-modality therapy design provides an important and challenging area for further developing AIM techniques. Sophisticated computer aids in clinical medicine have made significant strides in the past few years; however, most of the work has concentrated on diagnostic problems rather than therapy. Although some computer aids to therapy have been successfully implemented [70], they tend to emphasize the performance of a therapeutic regimen rather than the planning which enters into the formulation of the regimen. Recently, several Artificial Intelligence programs have been built which include aspects of therapeutic decision making for digitalis [43], glaucoma [133, 134], and infectious disease (MYCIN) [115]. These programs provide a base of experience for extending the use of computerized aids to multi-modality therapy planning, which builds on each of the above decision techniques. For example, the digitalis program's notions of clinical feedback and consequent adjustment of treatment, the glaucoma program's relationship between disease progression and recommended therapy, and MYCIN's drug selection methods for choosing the smallest drug set to cover all suspected infections, are all needed components of a general therapy planning program. Thus far, no program has attempted to put together into a single framework the totality of techniques needed in the general therapeutic setting.

Other AI work promises to be applicable for developing the Protocol Writer system to assist physicians in designing improved cancer protocols. In the area of business applications, an experimental program which helps a manager to design a procurement system for his company addresses many similar problems [8]. During the design process, many decisions may be made tentatively. It is the program's responsibility to demonstrate the consequences of that choice to the designer, and to permit him to change that choice without rerunning the whole design process. Record-keeping techniques to facilitate this task are a current focus of AI research [121, 24].

Our own group has made significant progress in addressing questions of test sequencing and therapy selection for Hodgkin's Disease and non-Hodgkin's lymphomas [104, 105, 108, 109], and another section of this proposal describes plans to extend the decision analytic techniques developed there to other hematologic diseases. Conklin, *et al* developed a prototype protocol selection program in radiotherapy based on a simple decision tree [18]. Those efforts concentrate on decision making up to (and including) the point of initial selection of a therapy but leave the design and execution of the entire therapeutic plan over time outside their scope. In this subproject we focus on these next steps.

7.2 Specific Aims

1. Develop a Protocol Writer program based on formal models of the structure of protocols to assist authors in writing protocols that are unambiguous and complete. Initially we will consider protocol design for one class of cancers, the lymphomas with particular attention to the diffuse histiocytic lymphomas.

2. Develop an interface program between protocols such as those produced by the Protocol Writer and the Cancer Care Data Management System (CCDMS) which is under development at Boston University-University Hospital. This interface will permit the dynamic monitoring and direction of the CCDMS to assure that the protocol is being followed accurately.

3. Investigate the possibility of creating formal models of tumor cell growth and the action of therapeutic agents and using these to enrich the Protocol Writer Program.

7.3 Methods of Procedure

7.3.1 Selection of a Small Domain

We plan to build programs which incorporate detailed knowledge of current cancer treatment protocols and their data requirements and dependencies. Further, we wish to investigate whether more basic biologic models of cancer care can be applied fruitfully with the aid of a computer in developing new protocols. Because these problems are conceptually difficult and in large part addressable in a specific subdomain of cancer, we have decided to narrow our focus to a small subset of cancers. A sensible subset is a specific cancer site with the following characteristics:

(1) Treatment of this cancer must be a field of active clinical investigation with many treatment protocols. Ideally, there should be several generations of protocols extending back in the past with the expectation of active clinical therapeutic investigation in the near future. Examples of both experimental and non-experimental protocols should exist.

(2) Treatment should be multi-modal, including radiotherapy and chemotherapy at a minimum and ideally surgical and supportive therapy as well. Within chemotherapy and radiotherapy the major subclasses of agents and modalities should be represented.

(3) Currently used protocols should employ "advanced" protocol strategies utilizing sophisticated concepts in design. These strategies should be as inclusive as possible of the strategies used in other types of cancer.

(4) There should be some systematic understanding of the mode of action of the agents used in therapy and the course of related toxicity.

(5) The cancer should be common enough for practical evaluation. The presence of unique sub-subtypes within this cancer would be desirable for preliminary analysis.

(6) The basic biological principles governing the growth and spread of the cancer are comparatively well elucidated. This requires an active biomedical literature including basic laboratory and clinical research studies.

(7) These basic biological principles should be common to other cancers to a high degree.

(8) An oncologist knowledgeable in this field of cancer should be available.

Based on the above criteria, we have selected the lymphomas in general and diffuse histiocytic lymphomas in particular as our study cancer [74]. Dr. John Krikorian, a faculty member in the Department of Medical Oncology at Boston University School of Medicine, will be the principal oncologist for the study. He has major clinical research background and interest in this field.

7.3.2 The Protocol Writer

We plan to design and implement an interactive computer program, the Protocol Writer, to assist the author of a new cancer treatment protocol to create a protocol that is complete, internally consistent, and free of ambiguity.

Initially we plan to adopt the methods employed in a program to assist a manager with the design of a procurement system for his organization. This program, called PROCTOR, was developed at the MIT Laboratory for Computer Science [8]. It includes a simplified model of the typical firm and its inventory characteristics, of the structure of a procurement system, of constraints of information availability and size, and of preferred strategies of eliciting the needed information from the user. We plan to define a set of analogous models for the Protocol Writer.

The model of what a protocol should include will be based on the "Outline for Writing a Multimodality Protocol" [75]. The significant parts of the protocol are reviewed in section 7.1.2.2 and include:

1. Objectives -- a concise description of why this protocol is being implemented.

2. Patient Selection -- conditions for patient eligibility and ineligibility.

3. Randomization -- what data will be used as the basis for forming subgroups of the patient group for experimental purposes.

4. Treatment Regimen -- specification of surgical procedures; the type, site, and dose of radiotherapy; the schedule for drug treatment and time and dose modifications for chemotherapy and radiotherapy.

5. Supportive Therapy -- definition of allowable therapy.

6. Response Criteria -- outcome measures.

7. Records to be kept -- information to be kept for patients on the protocol.

The basic Protocol Writer will guide the user through a much more detailed version of the above outline, prompting for each design component to be described. Most of the data may be obtained by multiple choice questions, supplemented by a free-text editing subsystem for cases where none of the program's choices are appropriate. Actually, each time the user needs to resort to free text, it indicates a deficiency in our model of the protocol, which we will then repair. Even this initial Protocol Writer program will be written in such a way that its model of the protocol is a separate data structure from the interpreter program which asks the questions. In this way, modifications to the protocol structure may be made to allow new categories or individual entries without reprogramming the interpreter.

The knowledge of protocol structure included in the basic Protocol Writer is actually a codification of expert opinion about how to structure protocols. This level of program is already useful to the protocol writer because it helps him to make his protocol explicit and complete. Building the program at this level provides us with a foundation to further expand the Protocol Writer.

The next level of development for the Protocol Writer is to include internal constraints in the model which tie together separate components of the protocol. These range from simple examples (e.g., all the data used for dose reduction calculations must be collected each time a dose is considered), to more complex ones (e.g., criteria for taking a patient off a protocol should be consistent with the initial patient selection criteria). These constraints also provide the technical mechanism to help control ambiguity in the protocol. For example, they can be used to assure that, e.g., no contradictory dose modification rules are simultaneously satisfiable. A great deal of experiential knowledge of protocol design can be encoded in such constraints. We plan to include known relationships between specific agents, their usual dosage levels and toxicities. The addition of constraints does not alter the basic nature of the program, which attempts to guide the user through the complete specification of the protocol. However, the program now monitors that individual answers made by the user are consistent with his previous choices; it can point out incompleteness and contradictions during protocol development.

A third step in developing the Protocol Writer will be to add the ability for the user to change his previous decisions and have the program automatically recompute other changes which this alteration requires. For this, we will employ the record-keeping techniques developed by Swartout [121] and being further advanced as part of the core research component of this proposal. This seemingly simple problem is in fact difficult because after a number of changes have had their effects propagated throughout the protocol model, a concise description of these must be made available to the user. For example, if a drug is moved from day 1 to day 15 of a cycle, the data collection constraints may now force certain laboratory values to be obtained at both days 1 and 15. The user must be informed, because he may instead choose to obtain laboratory values only at day 1 but change the dose modification rules so that they do not depend on current values.

The Protocol Writer will be implemented in the OWL language on the MIT-ML computer, and it will be accessed by direct and dial-up lines from MIT and University Hospital. Using OWL will permit us to build directly on techniques and programs developed under other parts of this proposal to aid us in representing models of protocols and providing explanations of the deductions of the Protocol Writer.

7.3.3 The Protocol Monitor

Having built a protocol with the aid of the Protocol Writer, the user should use the computer to help monitor and direct the execution of the protocol as well. In an information system, it is possible to embed the protocol rules so that their execution is better insured than in the usual manual record settings. Each time information is required by the protocol and each time a decision on cancer therapy is made, the physician can be prompted and his actions guided. Dr. Robert Friedman is building a Cancer Care Data Management System (CCDMS) at Boston University-University Hospital including programs to direct protocol execution. The translation from written protocol to programs will be done manually, separately for each protocol. Clearly, mechanizing this process becomes sensible and useful if the protocols are already modeled in the computer, as they would be by the Protocol Writer. We plan to develop a Protocol Monitor program to do this.

We see two possible approaches to this task, corresponding to the classical computer science notions of interpretation and compilation. In the interpretive mode, the full model of the protocol is present as decisions are made regarding an individual patient. This is advantageous because no transformation of the protocol model is needed and because the full model can be used by the physician to check his understanding of the program's recommendations. However, the computing support needed by the Protocol Monitor is large and may not be available in the execution system. In the compiled mode, the protocol model is translated into a small execution program, much as is envisioned now, only the translation is by computer, not by hand. This program is fast, but loses the capabilities mentioned above.

Initially, we plan to develop the Protocol Monitor at MIT, using the interpretive mode and building on the models created for the Protocol Writer. Principally, an addition is needed to those models representing the time sequence in which various actions and observations need to be made. In that environment, we will work out the needed methods using simulated patients. We will then investigate the coupling of our computer to the CCDMS to monitor protocol execution in a clinical setting but still using the interpretive model and our large computer. Although we have not yet designed the software or selected the hardware to achieve this coupling, our previous successful

experience in networking and an active group of investigators on network techniques at MIT gives us confidence that we can achieve it.

Alternatively, we will investigate the development of a compiler to translate our protocol models into programs to be run directly on the CCDMS. This compiler must generate code for every potential action that the interpreter could take. We believe that relatively straightforward compilation techniques developed for conventional programming languages coupled with recent work by members of our group in automatic programming [63] will enable us to develop such a compiler.

Another potential application of the Protocol Monitor is to review retrospectively data collected in the typical cooperative cancer treatment studies to identify violations of protocol to aid the Study Chairman to scrutinize his data. Technical violations could be categorized by type, severity (vis-a-vis the scientific study), and etiology. This would help the Chairman assess whether cases should be analyzed and also identify where problems existed in the protocol. For instance, if a particular protocol rule is frequently violated, he might infer that it should be changed. The fact that data from individual cases is automated in most cooperative study groups means that rapid and inexpensive Protocol Monitor review of cases is feasible.

7.3.4 Deeper Models of Cancer Therapy

A strategy found in almost all protocols is the desire to maximize the amount of treatment (to kill as many tumor cells as possible) while minimizing the toxicities of therapy to normal cells. Using knowledge of the kinetic properties of cell growth of normal and malignant tissues [81, 119] and the actions of therapeutic agents [80], more optimal treatment regimens have been developed and evaluated, to maximize tumor kill and minimize toxicity [1, 17, 34, 45, 58, 62, 82, 130, 136].

In section 7.1.3, we described a number of cancer therapy strategies arising from deeper physiological and pharmacokinetic models. In the domain of therapy with cardiac glycosides, such models coupled with models of expert clinicians' heuristic rules for setting therapeutic goals and adjusting for toxicities have been successful as the basis of a demonstrable potential clinical aid [43]. Although cancer therapy is a more complex problem and its underlying processes are less well understood, we believe that the attempt to include formal models of what is known can enrich the Protocol Writer.

This task is to formulate explicit models of (1) tumor cell growth and (2) the action of therapeutic agents against tumor cells and normal tissues. To accomplish these objectives, we will review the biomedical literature pertaining to principles of cancer growth in general and the lymphomas (diffuse histiocytic lymphoma) in particular. We will develop strategies to translate these laboratory and clinical studies into explicit models of tumor cell growth. The models will be referenced to the medical literature so that they have explanatory power. Gaps in the models will be supplemented by expert opinion. It will then be possible to review any aspect of the tumor growth (cell kinetic) model and determine what information, whether experimental or expert opinion, supports the model. A similar process will be employed to develop models of the action of therapeutic agents. The relevant medical literature will be culled, supplemented by expert opinion from Dr. Krikorian, consultants and standard text books in the field. We expect that a substantial part of the clinical pharmacologic literature will be based on studies in non-lymphoma patients. All studies, whether basic laboratory, clinical non-lymphoma or clinical lymphoma, will be interpreted

by our staff before inclusion in the models. These models of therapy action will contain explicit rationale, similar to the tumor growth models.

We then propose to expand the Protocol Writer by incorporating these models. The augmented Protocol Writer could then predict interactions among therapies as suggested by the models, and could use those interactions as possible new constraints to be applied to the protocol development process. For example, the program will be able to compute the anticipated time course of toxicities expected from various agents and suggest changes of agent or timing based on the principle of non-overlapping organ toxicities (see section 7.1.3).

With the growth of understanding of cancer therapy there should be no ultimate reason to block the automatic generation of optimal treatment plans, although we are far from that level of understanding. We will, however, build additions to the Protocol Writer which can perform a suggestive application of known principles to aid the protocol designer. Ultimately, the lack of "sufficient intelligence" in protocols will be overcome as more of the scientific basis of cancer treatment is folded into the protocol development and execution process.

7.4 Human Subjects

The Protocol Writer program will be used by physicians to develop new cancer care protocols. These protocols will not be used in any actual clinical setting at least during the first year of the proposed project. When they are ready for clinical use, they will be subject to the same review procedures now in effect for the creation of new protocols. The Protocol Monitor program, in directing the actions of the CCDMS, will ultimately impact on clinical therapy. During the first several years, this program will be used only on hypothetical cases or retrospectively on real cases from which all identifying information will have been removed. When clinical use of the program is contemplated, it will be subject to the human subjects review procedures applicable for the CCDMS itself. In addition, before any clinical use of either program is undertaken, the programs will be reviewed by the appropriate MIT committee.

7.5 Significance

If successful, this project is an important contribution to the fields of computers in medicine and clinical oncology. Therapy planning is an important problem in AIM, and the clinical cancer environment is a particularly good environment in which to study therapy. Cancer therapy is a field of rapid growth at this time. It has become extremely complex with multiple modalities used together or in sequence, determined by increasingly sophisticated decision rules, and frequently in clinical research settings. This environment offers a rich fabric of planning strategies and loosely connected fragments of knowledge which must be assembled by the designer of a protocol and which will form the basis for the Protocol Writer. Our proposed method is to fill in the outlines of a prototypical model of the treatment protocol under many constraints imposed by models of cancer therapy. Similar techniques have been applied in other, simpler domains [6, 8], suggesting that these methods are generally useful. The scale of the protocol design problem, the additional difficulties introduced by the multiplicity of simultaneously usable therapeutic agents, and the development of both disease and treatment over time introduce significant new research issues. The conceptual underpinnings of contemporary cancer therapy rest on a number of individual models of disease and treatment action. Despite considerable knowledge in the field, the models are quite incomplete, and treatment plans cannot easily be translated from them. The existence of

complex and incomplete models presents a domain perfectly suited for the application of the "science of weak methods" [78], artificial intelligence.

The existence of cancer information systems such as the Cancer Care Data Management System being developed by Dr. Friedman at Boston University-University Hospital opens up the possibility of the practical application of the logic of the cancer treatment model to daily patient care, using the Protocol Monitor. The data system significantly reduces the practical difficulties in executing detailed decision rules. Protocol authors will not be limited to creating treatment protocols easily presented in a readable written format and easily executed. Much more complex and detailed treatment rules will be possible. Moreover, each time treatment is planned, the physician caring for the patient can be presented with backup data from the protocol treatment model to substantiate the treatment decision.

Implementing the project in a computerized information system represents an important advantage for AIM. Usually, an AIM application requires interaction with the user in order to input data, thereby impeding practical use of the system and forcing possibly undesirable limits on the data available to it. In this application, any data needed will already be available from the CCDMS, and interaction can be limited to the substance of the task. Moreover, validation of AIM systems usually requires the tedious abstraction of clinical records. Since a data system can contain the relevant primary data as well as a description of the actual decisions, some major practical limitations to AIM applications are eliminated.

The project is of importance to clinical oncology as well. Any techniques that can improve treatment decision making offer the possibility of helping patients either by increasing the effectiveness of therapy or decreasing the unintended complications of therapy or both. As therapeutic regimens become increasingly complicated, authors of protocols need help in designing treatment plans that accomplish what the authors intend. Experimental studies require the specification of rules to ensure the integrity of the study while protecting the human subject. The Protocol Writer is planned to address the practical needs of the cancer specialist while presenting important basic problems in AIM for solution.

References

1. Alberts, D. S., Davie, B. G. M., and Salmon, S. E., Treatment of multiple myeloma in remission with anticancer drugs having cell cycle specific characteristics, *Cancer Treat. Rep.* 61:381-388 (1977).

2. Bell, D. E., Keeney, R. L., and Raiffa, H., *Conflicting Objectives in Decisions*, John Wiley and Sons, New York (1977).

3. Betaque, N. E., and Gorry, G. A., Automating Judgmental Decision Making for a Serious Medical Problem, *Management Science* 17:8: B-421 - B-434 (April 1971).

4. Bleich, H. L., Computer-Based Consultation: Electrolyte and Acid-Base Disorders, *American Journal of Medicine* 53:285 (1972).

5. Bobrow, D. G., and Winograd, T., An Overview of KRL, a Knowledge Representation Language, Artificial Intelligence Laboratory, Stanford University, Technical Report AIM-293 (1976).

6. Bobrow, D. G., Kaplan, R. M., Kay, M., Norman, D. A., Thompson, H., and Winograd, T., GUS, A Frame-Driven Dialog System, *Artificial Intelligence* 8:155-173 (1977).

7. Bobrow, D. G., and Winograd, T., Experience With KRL-0, One Cycle of a Knowledge Representation Language, *Proceedings of the Fifth International Joint Conference on Artificial Intelligence, Vol. 2* pp. 213-222, M.I.T., Cambridge, Ma., August 1977. Pittsburgh, Pa.: Carnegie-Mellon University, 1977.

8. Bosyj, M, A Program for the Design of Procurement Systems, MIT Laboratory for Computer Science, Technical Report 160 (1976).

9. Brachman, R. J., Theoretical Studies in Natural Language Understanding, Bolt, Beranek, and Newman Inc., Technical Report 3888 (1978).

10. Brown, J. S., Burton, R. R., Hausmann, C., Goldstein, I., Huggins, B., and Miller, M., Aspects of a Theory for Automated Student Modelling, Bolt, Beranek and Newman, Inc. BBN Report No. 3549, ICAI Report No. 4, May 1977

11. Buchanan, B. G., and Lederberg, J., "The Heuristic DENDRAL Program for Explaining Empirical Data, *Proceedings of the IFIP Congress 1971*, Ljubljana, Yugoslavia (1971). Also Artificial Intelligence Laboratory, Stanford University, Technical Report AIM-141 (1971).

12. Burton, R. R., and Brown, J. S., A Tutoring and Student Modelling Paradigm for Gaming Environments, in R. Colman and P. Lorton Jr. (eds.), *Computer Science and Education* (Advance Proceedings of the Association for Computing Machinery Special Interest Groups on Computer Science Education and Computer Uses in Education Joint Symposium, Anaheim, CA), SIGCSE Bulletin, Volume 8, Number 1

(SIGCUE Topics Volume 2), February 1976, pp. 236-246

13. Cage, P. E., Carson, E. R., and Britton, K. E., Simulation of Renal Function in Man, *MEDINFO 77* IFIP, North-Holland Publishing Company (1977).

14. Carr, B., and Goldstein, I. P., Overlays: a Theory of Modelling for Computer Aided Instruction, MIT Artificial Intelligence Laboratory Memo 406, February 1977

15. Carr, B., Wusor II: A Computer Aided Instruction Program with Student Modelling Capabilities, MIT Artificial Intelligence Laboratory Memo 417, May 1977

16. Carter, S. K., Clinical Trials in Cancer Chemotherapy, *Cancer* 40(suppl.):544-557 (1977).

17. Cohen, L., Cell population kinetics in radiation therapy: Relationships of cellular surviving fraction to RBE and OFR. *Radiology* 114:213-217 (1975).

18. Conklin, J. M., Munzenrider, J., Neurath, P. W., and Ross, W. M., Computer-Aided Medical Decision Making in Radiotherapy, *Radiology* 123:411-466 (1977)

19. Cornfield, J., Recent Methodological Contributions to Clinical Trials, *Am. J. Epid.* 104:408-421 (1976).

20. Davis, R., Applications of Meta Level Knowledge to the Construction, Maintenance and Use of Large Knowledge Bases, Ph.D. thesis, Stanford Artificial Intelligence Laboratory Memo 283 (1976).

21. Davis, R., Buchanan B., and Shortliffe, E., Production Rules as a Representation for a Knowledge-Based Consultation Program, *Artificial Intelligence* 8:15-45 (1977).

22. Davis, R., Generalized Procedure Calling and Content-Directed Invocation, *SIGART Newsletter Proceedings of Symposium on Artificial Intelligence and Programming Languages*, August (1977).

23. Detre, K. M., Wright, E., Murphy, M. L., and Takaro, T., Observer Agreement in Evaluating Coronary Angiograms, *Circulation* 52:979, (1975).

24. Doyle, J., A Glimpse of Truth Maintenance, MIT Artificial Intelligence Laboratory Memo 461a (1978).

25. Duda, R. O., Hart, P. E., and Nilsson, N. J., Subjective Bayesean methods for rule-based inference systems, *Proceedings of the 1976 National Computer Conference*, AFIPS Press (June 1976).

26. Eastern Cooperative Oncology Group Meeting, Nov. 12-14, 1978.

27. Einhorn, H. J., and Hogarth, R. M., Confidence in Judgment: Persistence of the Illusion of Validity, *Psychol. Rev.* 85:395, (1978).

28. Elstein, A. S., Shulman, L. A., and Sprafka S. A., *Medical Problem Solving*, Harvard University Press (1978).

29. Engelfinger, J. A., and Goldman, P., The serum digitalis concentration--Does it diagnose digitalis toxicity?, *N Engl J Med* 294, 867-870 (April 15, 1976).

30. Feinstein, A. R., Clinical Biostatistics XXXIX. The haze of Bayes, the aerial palaces of decision analysis, and the computerized Ouija board, *Clinical Pharmacology and Therapeutics* 21:4:482-496 (April 1977).

31. Flehinger, B. J., and Engle, R. L., Jr., HEME: A self-Improving Computer Program for Diagnosis-Oriented Analysis of Hematologic Diseases, *IBM J. Res. Develop.*, pp. 557-564 (November 1975).

32. Forrester, J. W., *Industrial Dynamics*, M.I.T. Press, Cambridge (1961).

33. Frei, E. 3d, Rationale for Combined Therapy, *Cancer* 40(suppl.):569-573 (1977).

34. Fu, K., Phillips, T. L., Kane, L. I., and Smith, V., Tumor and normal tissue response to irradiation in vivo, *Radiology* 114:709-716 (1975).

35. Gaines, B. R., Foundations of fuzzy reasoning, *Int. J. Man-Machine Studies* 8:623-668 (1976).

36. Goldberg, M., Green S., Moss, M. L., Marbach, C. B., and Garfinkel, D., Computer-based Instruction and Diagnosis of Acid-Base Disorders, *JAMA* 223:3:269-275 (1973).

37. Goldberg, A., and Kay, A. (eds.), Smalltalk-72 Instruction Manual, Xerox Palo Alto Research Center, Technical report SSL 76-6 (1976).

38. Goldstein, I., The Computer as Coach: An Athletic Paradigm for Intellectual Education, MIT Artificial Intelligence Laboratory Memo 389, December, 1976

39. Goldstein, I. P., and Roberts, R. B., NUDGE, A Knowledge-Based Scheduling Program, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Technical Report AI Memo 405 (1977).

40. Gorry, G. A., A System for Computer-Aided Diagnosis, Project MAC, Massachusetts Institute of Technology, Technical Report TR-44 (September 1967).

41. Gorry, G. A., and Barnett, G. O., Sequential Diagnosis by Computer, *JAMA* 205:12:141-146 (1968).

42. Gorry, G. A., Kassirer, J. P., Essig, A., and Schwartz, W. B., Decision Analysis as the Basis for Computer-Aided Management of Acute Renal Failure, *The American Journal of Medicine* 55:473-484 (1973).

43. Gorry, G. A., Silverman, H., and Pauker, S. G., Capturing Clinical Expertise: A Computer Program that Considers Clinical Responses to Digitalis, *American Journal of Medicine* 64:452-460, (March 1978).

44. Gorry, G. A., Pauker, S. G., and Schwartz, W. B., The Diagnostic Significance of the Normal Finding, *N Engl J Med* 298:486 (1978).

45. Hart, J. S., Livingston, R. B., Barlogie, B., Murphy, W. K., Gehan, E. A., and Bodey, G. P., Kinetic determinants in the design of chemotherapeutic trials, in *Grouth Kinetics and biochemical regulation of normal and malignant cells,* ed. by Drewinko, B., and Humphrey r. M., Baltimore: Williams and Wilkins (1977).

46. Hawkinson, L. B., The Representation of Concepts in OWL, *Proceedings of the Fourth International Joint Conference on Artificial Intelligence,* MIT Artificial Intelligence Laboratory (1975).

47. Hendrix, G. G., Expanding the Utility of Semantic Networks Through Partitioning, *Proceedings of the Fourth International Joint Conference on Artificial Intelligence,* Artificial Intelligence Laboratory, Massachusetts Institute of Technology (i975).

48. Herman, P. G., Geson, D. E., Hessel, S. J., Mayer, B. S., Watnick, M., Blesser, B., and Ozonoff, D., Disagreements in Chest Roentgen Interpretation, *Chest* 68:278, (1975).

49. Hewitt, C., Description and Theoretical Analysis (Using Schemata) of PLANNER: A Language for Proving Theorems and Manipulating Models in a Robot, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Technical Report AI-TR-258 (1972).

50. Hewitt, C., Bishop, P., and Steiger, R., A Universal Modular ACTOR Formalism for Artificial Intelligence, *Proceedings of the Third International Joint Conference of Artificial Intelligence* Stanford Research Institute, Menlo Park, California (1973).

51. Hewitt, C. E., and Smith, B. C., Towards a Programming Apprentice, *IEEE Transactions on Software Engineering* SE-1:26-45 (1975).

52. Jelliffe, R. W., Buell, J., Kalaba, R., Sridhar, R., and Rockwell, R., A Computer Program for Digitalis Dosage Regimens, *Mathematical Biosciences* 9:179-193 (1970).

53. Johnson, R. E., Symposium Theme: A Conceptual Approach to Integrated Cancer Therapy, *Cancer* 35:1 (1975).

54. Kahneman, D., and Tversky, A. , On the Psychology of Prediction, *Psychol. Rev.* 80:237-251, (1973).

55. Kassirer, J. P., and Gorry, G. A., Clinical Problem Solving: A Behavioral Analysis, *Ann. Int. Med.* 89:245, (1978).

56. Keeney R. L., and Raiffa H., *Decisions with multiple objectives: Preferences and value tradeoffs*, John Wiley & Sons, (1976).

57. Kirkwood, C. W., and Ureta, H., Calculator and Computer Programs for Elementary Multi-Objective Decision Analysis, TR No. 77-11, Dept. of Industrial and Operations Engineering, U. of Michigan, Ann Arbor (1977).

58. Kramer, S. (ed), Computer Augmented Radiation Oncology, *Cancer* 4(suppl.):2129-2139 (1976).

59. Kuipers, B. J., Representing Knowledge of Large-Scale Space, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Technical Report AI-TR-418 (1978).

60. Leal, A., and Pearl, J., An Interactive Program for Conversational Elicitation of Decision Structures, *IEEE Transactions on Systems, Man and Cybernetics* SMC-7:368-376 (1977).

61. Leaper, D. J., Horrocks, J. C., Staniland, J. R., and DeDombal, F. T., Computer Assisted Diagnosis of Abdominal Pain Using "Estimates" Provided by Clinicians, *British Medical Journal* 4:350-354 (1972).

62. Lincoln, T., Morrison, P., Aroesty, J., and Carter, G., Computer Simulation of Leukemia Therapy: Combined Pharmacokinetics, Intracellular Enzyme Kinetics, and Cell Kinetics of the Treatment of L1210 Leukemia by Cytosine Arabinoside, *Cancer Treat. Rep.* 60:1723-1739 (1976).

63. Long, W. J. A Program Writer, MIT Laboratory for Computer Science, Technical Report TR-187 (1977)

64. Lusted, L. B., A Study of the Efficacy of Diagnostic Radiologic Procedures: Final Report on Diagnostic Efficacy, American College of Radiology and the National Center for Health Resource Administration, (1977).

65. Martin, W. A., and Fateman, R. J., The MACSYMA System, *Proceedings of the Second Symposium on SYmbolic Manipulation*, Petrick, S. R., (ed),. Los Angeles (1971)

66. Martin, W. A., Descriptions and the Specialization of Concepts, *Proceedings of the Fifth International Joint Conference on Artificial Intelligence, Vol.* 2, M.I.T., Cambridge, Ma., August 1977. Pittsburgh, Pa.: Carnegie-Mellon University, 1977. Also available as M.I.T., Laboratory for Computer Science, M.I.T./LCS/TM-101, Cambridge, Ma., March 1978 (1977).

67. McCarthy, J., Abrahams, P. W., Edwards, D. J., Hart, T. P., and Levin, M. I., *LISP 1.5 Programmer's Manual*, The M.I.T. Press. (1962).

68. McNeil, B. J. *et al.*, Primer on certain elements of medical decision making, *New Engl J Med* 293:211-215, (1975).

69. NcNeil, B. J., Weischelbaum, R., and Pauker, S. G., The Fallacy of the Five Year Survival in Lung Cancer, *New Engl J Med* 299:1397-1401 (1978).

70. Mesel, E., and Wirtschafter, D., *et al*, Clinical algorithms for cancer chemotherapy--systems for community based consultant-extenders and oncology centers, *Meth. Inform. Med.* 15:168-173 (1976).

71. Miller, P. B., Strategy Selection in Medical Diagnosis, Project MAC, Massachusetts Institute of Technology, Technical Report TR-153 (September 1975).

72. Minsky, M., Semantic Information Processing, (Ed.) MIT Press, Cambridge, (1968).

73. Moon D., A., MACLISP Reference Manual, Laboratory for Computer Science, Massachusetts Institute of Technology, (1974).

74. Muggea, F. M., Davis, H. L., Jr., and Rosencweig, M., Current Cooperative Clinical Trials in the Non-Hodgkin's Lymphomas, *Cancer Treat. Rep.* 61:1191-1197 (1977).

75. Northern California Oncology Group, Outline for Writing a Multimodality Protocol (1978).

76. Newell, A., Shaw, J. C., and Simson, H. A., Chess - Playing Programs and the Problem of Complexity, *IBM J. Res. Devel.* 2:330-335, (1958).

77. Newell, A., and Simon, H. A., *Human Problem Solving*, Prentice Hall (1972).

78. Newell, A., Artificial Intelligence and the Concept of Mind, in Schank, R. C., and Colby, K. M, eds., *Computer Models of Thought and Language*, W. H. Freeman and Co. (1973).

79. Newell, A., and Simon, H. A., The Logic Theory Machine: A Complex Information Processing System, *IBM Trans. on Information Theory*, 2(3):61-79, (1976).

80. Nichol, C. A., Pharmacokinetics: Selectivity of Action Related to Physiochemical Properties and Kinetic Patterns of Anticancer Drugs, *Cancer* 40(suppl.):519-528 (1977).

81. Nicolini, C., The Principles and Methods of Cell Synchronization in Cancer Chemotherapy, *Biochem. Biophys. Acta.* 458:243-252 (1976).

82. Norton, L., and Simon, R., Tumor size, sensitivity to therapy, and design of treatment schedules, *Cancer Treat. Rep.* 61:1307-1317 (1977).

83. Patil, R. S., A Program for Expert Diagnosis of Acid Base and Electrolyte Disturbances, Ph. D. Thesis Proposal, Laboratory for Computer Science, Massachusetts Institute of Technology (1979).

84. Pauker, S. G., and Kassirer, J. P., Therapeutic Decision Making: A Cost-Benefit Analysis, *N Engl J Med* 293:229-234 (July, 1975).

85. Pauker, S. G., Gorry, G. A., and Kassirer, J. P., and Schwartz, W. B., Toward the Simulation of Clinical Cognition: Taking a Present Illness by Computer, *The American Journal of Medicine* 60:981-995 (June 1976).

86. Pauker, S. G., Coronary Artery Surgery: The Use of Decision Analysis, *Annals of Internal Medicine* 85:8:∞∞, (1976).

87. Pauker, S. G., and Szolovits, P., Analyzing and Simulating Taking the History of the Present Illness: Context Formation, in Schneider/Sagvall Hein, eds., *Computational Linguistics in Medicine*, North-Holland (1977).

88. Pauker, S. P., and Pauker S. G., Prenatal Diagnosis: A Directive Approach to Genetic Counseling Using Decision Analysis, *Yale Journal of Medicine* 50:275 (1977).

89. Pauker S. G., and Kassirer, J. P., The Diagnostic Threshold: a Cost Benefit Analysis, *Clinical Research* 25:330A, (1977).

90. Pauker, S. G., Surgical Therapy for Coronary Artery Disease: Caveat Emptor, *American Heart Journal* 93:543-546 (1977).

91. Pauker, S. G., Acquisition and Use of Patient Attitudes in Clinical Decision Making, *Proceedings of the Fourth Illinois Conference on Medical Information Systems*, University of Illinois (May 1978).

92. Pauker, S. G., and Kassirer, J. P., Clinical Applications of Decision Analysis: A detailed illustration, *Seminars in Nuclear Medicine* (Oct. 1978).

93. Pauker, S. P., and Pauker, S. G., The Amniocentesis Decision: An explicit guide for parents, *Annual Review of Birth Defects* 1978 (in press).

94. Pople, H. E., Jr., Artificial-Intelligence Approaches to Computer-Based Medical Consultation, IEEE Intercon Conference (1975).

95. Pople, H. E., Jr., Myers, J. D., and Miller, R. A., DIALOG: A Model of Diagnostic Logic for Internal Medicine, *Advance Papers of the Fourth International Joint Conference on Artificial Intelligence*, available from the Artificial Intelligence Laboratory, Massachusetts Institute of Technology (1975).

96. Pople, H. E., Jr., Presentation of the INTERNIST System, Proceedings of the A.I.M Workshop, Rutgers Research Resource on Computers in Biomedicine, Rutgers University (June 1976).

97. Pople, H. E., Jr., The Formation of Composite Hypotheses in Diagnostic Problem Solving: an Exercise in Synthetic Reasoning, *Proceedings of the Fifth International Joint Conference on Artificial Intelligence*, available from the Department of Computer Science, Carnegie-Mellon University, Pittsburgh, PA 15213. (1977).

98. Raiffa, H., *Decision analysis.* Addison-Wesley Publishing Company, Reading, Massachusetts. (1970).

99. Richards, B., and Goh, A. E. S., Computer Assistance in the Treatment of Patients with Acid-Base and Electrolyte Disturbances, *MEDINFO 77* IFIP, North-Holland Publishing Company (1977).

100. Rieger, C., The Commonsense Algorithm as a Basis for Computer Models of Human Memory, Inference, Belief and Contextual Language Comprehension, in R. Schank and B. L. Nash-Webber, eds., Proceedings of a Workshop on Theoretical Issues in Natural Language Processing, Massachusetts Institute of Technology (June 1975).

101. Rifkin, R. D., and Hood, W. B., Bayesian analysis of electrocardiographic exercise stress testing *New Engl J Med* 297:681-686, (1977).

102. Rousseau W. F., and North D. W., User's Manual for TREE, SRI Technical Memorandum (1971).

103. Rubin, A. D., Hypothesis Formation and Evaluation in Medical Diagnosis, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Technical Report AI-TR-316 (January 1975).

104. Rutherford, C. J., Desforges, J. F., Davies, B. L., Indications for Laparotomy in Symptomatic Hodgkin's Disease Patients. Abstract 399, American Society of Hematology Meeting, 1977. *Blood* 50:207, (1977).

105. Rutherford, C. J., Safran, C., Desforges, J. F., Davies, B., and Barnett, A. I., The decision to Use Combined or Single Modality Therapy for Hodgkin's Disease (HD) in Adults, Abstract, American Society Hematology Meeting 1978, *Blood* 52:274, (1978).

106. Sacerdoti, E. D., A Structure for Plans and Behavior, Stanford Research Institute Technical Note 109, (1975).

107. Safran C. et al., Diagnostic planning and cancer management, Massachusetts Institute of Technology, Technical Report LCS-TR-169, (1976).

108. Safran, C., Desforges, J. F., Tsichlis, P. N., and Bluming, A. Z., Decision Analysis to Evaluate Lymphangiography in the Management of Patients with Hodgkin's Disease, *N Engl J Med* 296:1088-1092 (May 12, 1977).

109. Safran, C., Tsichlis, P. N., Bluming, A. Z., and Desforges, J. F., Diagnostic Planning using Computer Assisted Decision-Making for Patients with Hodgkin's Disease, *Cancer* 39:2426-2434 (June 1977).

110. Schlaifer, R., Computer Programs for Elementary Decision Analysis, Harvard University Press, Boston (1971).

111. Schwartz, W. B., Medicine and the Computer: The Promise and Problems of Change, *N Engl J Med* 283:1257-1264 (1970).

112. Schwartz, W. B., Gorry, G. A., Kassirer, J. P, and Essig, A., Decision Analysis and Clinical Judgment, *The American Journal of Medicine* 55:459-472 (October 1973).

113. Shafer, G., *A Mathematical Theory of Evidence*, Princeton University Press (1976).

114. Shortliffe, E. H., and Buchanan, B. G., A Model of inexact reasoning in medicine, *Mathematical Biosciences* 23:351-379 (1975).

115. Shortliffe, E. H., *Computer Based Medical Consultations: MYCIN*, Elsevier North Holland Inc. (1976).

116. Silverman, H., A Digitalis Therapy Advisor, Project MAC, Massachusetts Institute of Technology, Technical Report TR-143 (1975).

117. Smith, B. C., A Proposal for a Computational Model of Anatomical and Physiological Reasoning, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, AI-Memo 493 (1978).

118. Steele, G. L., and Sussman, G. J., The Revised Report on SCHEME, a Dialect of LISP, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Technical Report AI Memo 452 (1978).

119. Sullivan, P. W., and Salmon, S. E., Kinetics of tumor grouth and regression in IgG multiple myeloma, *J. Clin. Invest.* 51:1697-1706 (1976).

120. Sussman, G. J., and McDermott, D. V., From PLANNER to CONNIVER - A Genetic Approach, *Proceedings of the 1976 Fall Joint Computer Conference*, pp. 1171-1179, AFIPS Press (1976).

121. Swartout, W. R., A Digitalis Therapy Advisor with Explanations, Laboratory for Computer Science, Massachusetts Institute of Technology, Technical Report TR-176 (February 1977).

122. Szolovits, P., and Pauker, S. G., Research on a Medical Consultation System for Taking the Present Illness, *Proceedings of the Third Illinois Conference on Medical Information Systems*, University of Illinois at Chicago Circle (November 1976).

123. Szolovits, P., Hawkinson, L, and W. A. Martin, An Overview of the OWL Language for Knowledge Representation, *Proceedings of the Workshop on Natural Language Interaction with Databases*, International Institute for Applied Systems Analysis, Schloss Laxenburg, Austria, Jan. 10, 1977 (forthcoming). Also appeared as MIT/LCS/TM-86 (June 1977).

124. Szolovits, P., Pauker, S. G., Categorical and Probabilistic Reasoning in Medical Diagnosis, *Artificial Intelligence* 11,:115-144 (1978).

125. Szolovits, P., Coordinating the Use of Categorical and Probabilistic Reasoning, *Proceeding of the 1978 Computer Laboratory Health Care Resources Workshop*, Texas Technical University School of Medicine, Lubbock, Texas (January 1978).

126. Szolovits, P., The Lure of Numbers--How to Live With and Without Them in Medical Diagnosis, *Proceeding of the Colloquium in Computer-Assisted Decision Making using Clinical and Paraclinical (Laboratory) Data*, 1978.

127. Teitelman, W., *INTERLISP Reference Manual*, Xerox Palo Alto Research Center (1978).

128. Tversky, A., and Kahneman, D., Belief in the Law of Small Numbers, *Psychol. Bull.* 76:105-110, (1971).

129. Tversky, A., and Kahneman, D., Judgment under Uncertainty: Heuristics and Biases, *Science* 185:1124-1131 (September 1974).

130. Valleron A. J., Mathematic Models of Cell Synchrony and Drug Scheduling, *Cancer Treat. Rep.* 60:1899-1911 (1976).

131. *Websters New Collegiate Dictionary*, A Merriam-Webster (1977)

132. Weiss, S. M., A System for Model-Based Computer-Aided Diagnosis and Therapy, Ph.D. Thesis, Computers in Biomedicine, Department of Computer Science, Rutgers University, CBM-TR-27-Thesis (June 1974).

133. Weiss, S. M., Presentation at the Third Illinois Conference on Medical Information Systems, University of Illinois at Chicago Circle (November 1976).

134. Weiss, S. M., Kulikowski, C. A., Amarel, S., and Safir, A., A Model-Based Method for Computer-Aided Medical Decision-Making, *Artificial Intelligence* 11: 145-172 (1978).

135. Wescourt, K. T., Beard, M., and Gould, L., Knowledge-based Adaptive Curriculum Sequencing for CAI: Application of a Network Representation, *Proceedings of the National Association of Computing Machinery* (1977).

136. Wheldon, T. E., and Kirk, J., Mathematical derivation of optimum treatment schedules for the radiotherapy of human tumors, *Br. J. Radiology* 49:441-449 (1976).

137. Winograd, T., A Computer Program for Understanding Natural Language, MIT Artificial Intelligence Laboratory TR-17, 1971

138. Winston, P. H., New Progress in Artificial Intelligence, The Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Technical Report AI-TR-310 (1974).

139. Winston, P. W., *Artificial Intelligence*, Addison-Wesley Publishing Company, Reading, Massachusetts, (1972).

140. Wood, G., The Knew-It-All-Along Effect, *J. Exper. Psychol.* 4:345, (1978).

141. Zadeh, L. A., Fuzzy sets, *Information and Control* 8:338-353 (1965).

142. Zamora R. M., Leaf E. E., Tutorial on the Use of the SRI Tree Language, SRI Technical Memorandum (1974).