LABORATORY FOR
COMPUTER SCIENCE

MASSACHUSETTS
INSTITUTE OF
TECHNOLOGY

MIT/LCS/TM-310

AN ARTIFICIAL INTELLIGENCE APPROACH TO CLINICAL DECISION MAKING

PETER SZOLOVITS
JEROME P. KASSIRER
WILLIAM J. LONG
ALAN J. MOSKOWITZ
STEPHEN G. PAUKER
RAMESH S. PATIL
MICHAEL P. WELLMAN

SEPTEMBER 1986

545 TECHNOLOGY SQUARE, CAMBRIDGE, MASSACHUSETTS 02139

An Artificial Intelligence Approach to Clinical Decision Making

Peter Szolovits Jerome P. Kassirer William J. Long Alan J. Moskowitz
Stephen G. Pauker Ramesh S. Patil Michael P. Wellman
M.I.T Laboratory for Computer Science, Cambridge, Massachusetts
and
Tufts-New England Medical Center, Boston, Massachusetts

June 30, 1986

Abstract:
This memo is the text of a proposal from the MIT Laboratory for Computer Science Clinical Decision Making group to the National Library of Medicine, requesting support for a five-year program of research. The proposal is to develop an integrated program for the representation of complex knowledge and reasoning processes in medicine, using the techniques of artificial intelligence (AI) and decision analysis (DA). The project will address problems in both diagnostic and therapeutic reasoning in two medical domains: coronary disease, and fluid and electrolyte disturbances. The planned large-scale AI system will incorporate a common knowledge representation formalism and alternative reasoning programs to do data interpretation, hypothesis generation, hypothesis testing and revision, diagnostic query planning, therapeutic planning and automatically-generated explanations. DA reasoning will be integrated into the system, especially to improve the test and treatment planning components. A uniform knowledge base will be developed in the two identified medical domains, utilizing knowledge acquired from the analysis of protocols, expert opinion, case-based learning, hand-coded knowledge from the literature, and strictly limited attempts to automatically acquire structured knowledge from English free-text. A panel of specific cases will be developed that can be used to validate additions to the knowledge base and changes in the reasoning components of the system. A series of protocol analyses will be done, involving both diagnostic and therapeutic decision making in clinical settings where uncertainty and risk are predominant factors. Specific attention will focus on the degree to which experienced clinicians rely on case-based reasoning and means will be developed to represent, index and utilize such case-based experience in the reasoning program. A new generation of DA tools will be built to aid the construction of DA models for use in actual clinical settings. These tools will utilize the same knowledge bases as the AI models, and will share with them the ability to look in varying depth of detail at problems and to generate explanations of their workings. A set of template analyses dealing with rapidly progressive glomerulonephritis, renal transplantation, use of anticoagulant therapy and the timing of surgery for valvular heart disease, and a knowledge base of risk and prognosis estimates in these domains will also be constructed. An AI based system for quantitative and qualitative reasoning about patient preferences will augment a system to help health professionals and patients to establish individual patient utility structures.

Keywords: artificial intelligence, medical reasoning, clinical decision-making, knowledge representation, diagnosis, therapy planning, hypothesis generation, protocol analysis, uncertainty, preference.

The research outlined in this report is supported by the National Library of Medicine through NIH Grant R01 LM04493.

1 Introduction

This report is the text of a proposal to the National Library of Medicine for a five-year research grant that was awarded to MIT/LCS and Tufts/NEMCH to pursue the collaborative research described herein. The project period is from Fall 1985 through Fall 1990. In line with our tradition of making our preliminary ideas, even in the form of research proposals, available to our working colleagues, we present the text of the technical part of the proposal. Readers should keep in mind that this is a description of work to be done, not accomplishments already made. Of course, we plan to bring our accomplishments up to the standards of our dreams.

In addition to the listed authors, we wish to thank other members of the Clinical Decision Making Group at the MIT Laboratory for Computer Science and of the Division of Clinical Decision Making, Tufts University/New England Medical Center Hospital. We also wish to thank the National Library of Medicine for undertaking to support this work as well as many other excellent projects around the country that are aiming at the same general goals. We also acknowledge our grant, No. R01 LM04493, which supported the publication of this report.

2 Specific Aims

Our overall aim is to develop a framework for the representation of complex knowledge and reasoning processes in medicine, using the techniques of artificial intelligence and decision analysis. We shall address problems in both diagnostic and therapeutic reasoning and shall develop methodologies for interconnecting both techniques. Realizing that research in both domains has not yet reached a "critical mass" of sufficient magnitude to promote general use, we shall focus on a limited set of clinical problems and to develop a shared, relatively uniform structured knowledge base capable of serving each project. Specifically, we propose to:

1) Design and implement a modular, integrated system for medical reasoning, in some ways analogous to the Macsyma program for symbolic mathematics. The system will provide interfaces among a variety of approaches and will operate on a knowledge base that includes empiric knowledge, physiologic knowledge, case-based experience and heuristics.

2) Develop new and adapt established components of medical reasoning (e.g., data acquisition; hypothesis generation, evaluation, revision and restructuring; bayesian reasoning; explanation; multi-level reasoning; planning; expected value calculation; sensitivity analysis; reasoning about temporal events and causality; anatomic and physiologic reasoning) to conform to the uniform system.

3) Develop a uniform technique for representing both the state of a patient (e.g., history, physical examination and laboratory data) and the set of working hypotheses and expectations generated by the reasoning module. The patient-specific model will enforce consistency using both rules and constraints and will be capable of describing an evolving sequence of snapshots over time by providing a formal representation for reasoning about time series events.

4) Develop a uniform knowledge base in several medical domains, including coronary artery disease and fluid and electrolyte balance. Such knowledge bases will conform to an incrementally evolving representation scheme and will utilize knowledge acquired by a broad variety of techniques: protocol analysis, expert opinion, case-based learning, hand-coded knowledge from the literature and limited attempts to automatically acquire machine readable free text (natural language) in highly-restricted domains. The maintenance of the uniform knowledge base will identify areas of inconsistency for either automatic or manual resolution.

5) Perform a series of protocol analyses involving both diagnostic and therapeutic decision making in clinical settings where uncertainty, risk and temporal evolution are the predominant factors. These analyses will help identify heuristics and strategies used by physicians in planning in settings of competing goals and will help structure areas of the knowledge base.

6) Develop and maintain a panel of specific cases (patient specific models) that can be used to validate any additions to the knowledge base.

7) Develop a decision analysis system that will function in a variety of medical domains (including coronary artery disease, valvular heart disease, and acute and chronic renal insufficiency). The system will use artificial intelligence techniques to assist both experienced and naive analysts in constructing, debugging, and interpreting decision analytic models, sharing the same knowledge bases as the models developed for diagnostic and physiologic reasoning.

8) Develop an artificial intelligence based system for quantitative and qualitative reasoning about utilities (outcome descriptors) that will explore short-term morbidities and other transient disutilities, and allow direct assessment of utilities using a micro-computer based "feedback system."

9) Create a set of programs for use in medical practice and education that promote early application of subsets of our evolving integrated system and knowledge base.

3 Significance

Over the past decade the "classic" practices of medicine and medical education continue to approach a brink of intertwined crises—an explosion of information, an explosion of procedures, an explosion of costs, and an impending explosion of regulation. Traditional approaches to medical practice and medical education are, and likely will continue to be, unable to address this new set of challenges. The GPEP report of the American Association of Medical Colleges (Graduate Professional Education for Physicians) identifies a crisis in education and boldly points to the field of medical informatics as a centerpiece for education and practice in the twenty-first century. Indeed, in addition to the educational benefits of such an approach, the group of leading physicians that formulated those recommendations properly recognized that the small, isolated and presently underfunded field of medical information science offers one of the few hopes for physicians to maintain control of their profession.

Unfortunately, in 1985 Medical Informatics is scarcely ready for that challenge. Many separate technologies have been used to approach many separate problems. There has been virtually no integration of technologic advances in one medical domain by investigators using other technologies in other domains. Truly, it is a cottage industry of independent "artists." Of course, such diversity is quite appropriate in a young field and might continue to be appropriate if we had the luxury of a leisurely pace of development. Sadly, we do not. We must begin to draw together the separate threads of our research into a coherent carpet upon which medicine can ride into the future.

The Clinical Decision Making Groups at MIT and NEMC have certainly been major contributors to this diversity. Evolving from initial work in flow charts and decision theory, we have developed two main—but quite separate—streams of research. We have used artificial intelligence techniques to approach the generation and structuring of hypotheses, limited amounts of therapeutic planning, explanation, and pathophysiologic reasoning. We have used classic decision theory to approach management dilemmas facing individual patients and generic problems facing many patients. We have developed macrocomputer programs that make diagnoses, explain their reasoning, and serve as intellectual blackboards for framing therapeutic ideas. We have developed microcomputer programs that explore decision trees and utility instruments that explore patients attitudes toward their illness. We have focused on isolated domains in nephrology and cardiology. But we have not integrated these closely related domains and problems into a coherent approach. Each new project developed a new data base, usually employing a new knowledge representation scheme. Only infrequently were knowledge bases from one project reformulated into the representations of a new project. Thus, we have developed many exciting pieces and ideas but have not fit them into an enlarging tapestry of knowledge and process.

We propose a large coherent project that will address this problem. We believe that medical practice requires the use of both categorical and probabilistic reasoning and that many tasks should be addressed by combinations of these approaches. We propose to develop a single coherent framework for representing both kinds of knowledge in a single knowledge base and for using alternative processing techniques for a given task. In such a framework, each sub-project will add knowledge in a form that can be used by investigators working in other sub-projects. Furthermore, we shall deliberately cross-fertilize artificial intelligence and decision theory in several ways—we shall use probabilistic and utility based reasoning to address certain hypothesis evaluation tasks and shall use categorical reasoning to help structure, analyze, debug and explain decision tree models. We shall utilize protocol analysis of experienced clinicians to identify both categorical and probabilistic rationales; we shall also use these analyses to help develop knowledge for the growing coherent knowledge base. Finally, we shall be utilizing both mainframe computers and personal workstations operating in a coherent set of languages to optimize the dissemination of intermediate products

among colleagues and former trainees.

Our detailed specific aims fall into three major categories, each intended to advance the goals of integration introduced above. First, we propose to build a large-scale artificial-intelligence based system that will ultimately incorporate a common knowledge representation formalism for expressing all forms of medical knowledge, including empirical knowledge derived from judgmental clinical experience and statistical data collections, as well as deeper medical models such as pathophysiologic, physiologic, and biological knowledge. Within this system we intend to incorporate the large variety of reasoning mechanisms that developed over the years in a number of separate research projects, to provide a common interface so that different components can be used together, and ultimately to use the common knowledge representation to enable the construction of new reasoning and action modules whose internal structures are mutually consistent. The great advantage of this organization is that individual researchers can focus on the development of a single module to explore a new research idea about aspect of the medical reasoning task (e.g., how to generate initial diagnostic hypotheses, or how to plan a sequence of therapeutic interventions) and rely on already available components to help test out their ideas. Indeed, we expect that competing modules for several tasks will be constructed, thus enabling us to compare directly the capabilities of alternative approaches to some problem (e.g., two modules concerned with verifying the correctness of a hypothesized disease, one based on reasoning from underlying causal models, the other from probabilistic clinical associations). Of course, a typical project will likely use one module or the other. If more than one is used and if those modules conflict in their conclusions (a likely occurrence), then the specific project will have to resolve such conflicts. In other words, the modules will be passively available but will not automatically become active participants, unless asked.

Second, we plan to unify our work on AI decision-making models and our work on decision-analytic reasoning by integrating decision analysis as a component of the general medical reasoning system (above) and by extending the use of AI techniques to helping to structure, analyze and debug new decision trees and capture, analyze and reason about the nature of individual patients' preferences. We have long recognized that in any medical AI program, whether diagnostic or therapeutic, ultimately decisions must be made in the face of unresolvable uncertainty. For example, a diagnostic program must be able to decide whether to stop further investigation because the value of additional information is exceeded by the risk (or cost) of obtaining it; a therapeutic program must decide whether a secondary disorder should be aggressively treated or deferred until the patient's primary problem is under control, where there is a serious trade-off between the risk of complications from the combined therapies, on the one hand, and the risk of serious morbidity from the (yet-) untreated secondary disorder on the other. Although we have developed decision aids to help physicians make just such decisions, neither we nor our colleagues have yet incorporated such capabilities into AI programs. Instead, such programs have relied on crude numerical scoring criteria to make these decisions, often poorly. For example, neither the Present Illness Program (built by our group) nor Internist-I (from Pittsburgh) had any good criteria for when to stop diagnostic investigation, short of exhaustively tracking down possible explanations for every finding discovered. Incorporating explicit decision analytic models in our AI programs should significantly improve their performance in the face of uncertainty and risk.

Conversely, we believe that we are now ready to try to develop a new generation of decision analytic tools that make it easier to construct, apply and interpret decision analysis in actual clinical settings. Although the use of clinical decision analysis is rapidly increasing, it remains difficult to build new decision models and to interpret their results. The decision analyst commences with a blank sheet of paper and must laboriously think through in each case what are the serious alternatives, risks, etc. In this project, we plan to generate a large body of medical knowledge, encoded in a common representational framework. Because much of the knowledge of clinical

medicine (at least in narrow domains) will be encoded therein, we will be able to use this knowledge to help generate decision trees automatically, and thus to ease the decision analyst's task and to make it likely that his product is more complete. We have also conducted much research on the problem of automatically generating explanations in AI programs, but have not yet applied these methods to decision analysis problems. This should also be very fruitful, as the problems of explaining the results of an analysis are one of the greatest difficulties of the technique and should be amenable to AI methods. In addition, recent AI work on the qualitative understanding of mathematical models—and of preference models in particular—promises to allow us to introduce much more flexibility in decision analysis, using partial information about a patient's preferences to constrain the decision model.

Thirdly, we intend to use protocol analysis to explicate the methods that human clinicians use to reason about uncertain, risky, and evolving cases in two specific areas of medicine: coronary artery disease and fluid and electrolyte balance. We shall develop and maintain an extensive knowledge base covering these areas and design a formalism for maintaining a panel of specific patient cases, collected to test and validate the operation of our evolving programs. The problem of knowledge acquisition is thought to be the next major bottleneck in the development of "expert systems". We plan to push on the methodology of knowledge collection and encoding, to develop formal methods that can be used by trainees, as well as expert clinicians, so that the problem of collecting and maintaining a large body of up-to-date knowledge about specific medical disciplines should no longer require the super-human efforts of a senior clinician (we think here of Dr. Jack Myers in the Internist project) but can be factored into more sharable formal parts. We have used protocol analyses, both formal and informal, to suggest new computational techniques. Such analyses have given us insights into the clinical decision process we are seeking to model in the computer. We plan to extend our protocol studies, and also to integrate their use into the larger system to help define the structure of the general medical knowledge to be represented, suggest appropriate ways to represent and index past cases, and (perhaps) develop new case-based reasoning methods.

The above discussion has concentrated on the significance of our proposed research within the context of medical reasoning systems. If the promise of medical informatics (or medical artificial intelligence, as we have been referring to it) is to be fulfilled, a diversity of interesting research ideas must be pulled together into a common system wherein detailed interaction and sharing of conceptual models and stored knowledge is supported. Such a system will produce an environment, not now available, in which new experiments can be performed within a framework of other components, an existing knowledge base, and an existing panel of carefully collected cases useful for testing and verification of the ideas. In addition, this system will enable us to extend our research significantly, merging the symbolic and probabilistic reasoning methods we have developed, and continuing our studies of the behavior of human decision-makers to help understand and capture both their knowledge and methods of thought.

Ultimately, of course, the goal of this research is to create computer systems that incorporate the best expertise of human clinicians and the "distilled wisdom" of probabilistic data, systems that are widely available and cheaply reproducible, systems that can help improve the quality and efficiency of medical care by providing consultation on demand, monitoring possible errors, permitting the investigation of diagnostic and therapeutic alternatives, explaining the application of general medical knowledge to specific cases, and helping to educate both existing and new physicians.

4 Preliminary Studies

Beginnings of the MIT/NEMC collaborative research group. The Clinical Decision Making Group traces its beginnings to projects at MIT and NEMC in the late 60's, investigating the use of computers for diagnostic and therapeutic decision-making. At NEMC, a group of investigators pursued the notion that a straightforward encoding of the questioning behavior of a medical expert could lead to programs that behaved as experts in the corresponding medical domain [Blei72,Schw70]. Such flowcharts did in fact capture some of the style and substance of medical reasoning, but were found inapplicable to large problems because of their lack of explicitness in expressing the underlying medical knowledge [Szol78]. At the same time, G. Anthony Gorry's doctoral thesis presented a technique for diagnostic reasoning based on a sequential Bayesian probabilistic model [Gorry77] which extended earlier work by Warner [Warner61]. In this approach, the impact of new evidence is evaluated by applying Bayes' rule to compute new likelihoods for all possible diagnostic hypotheses after each piece of evidence is introduced. The principal advance of this work was to add an active questioning component that would determine what was the most appropriate next question to ask, given what was already known about a patient. The key idea was to ask that question that was most likely to be most informative (in the information-theory sense) by choosing that question whose likely answers would lead to the minimum expected entropy in the resulting posterior probability distribution. This model was successfully applied in a number of domains [Beta71] but is limited by simplifying assumptions that must be made to avoid an unsatisfiable appetite for conditional probability estimates [Szol78].

These two groups joined forces in the beginning of the seventies to build the Acute Renal Failure program, which incorporated the sequential Bayesian model for diagnostic reasoning and a decision analytic model for therapy selection [Gorry73]. The decision analytic approach also appeared as an attractive method to apply to human (not just computerized) decision making in clinical medicine [Schw73], and our group has become one of the strongest advocates of the clinical use of decision analytic techniques. A landmark pair of papers on these ideas appeared in 1973 in the American Journal of Medicine, marking the first appearance in a major medical journal of clinical decision analysis. It is with this project that our group began an active effort to apply decision analysis clinically, leading to the establishment of the Division of Clinical Decision Making at NEMC, the development of microcomputer tools to support the use of decision analysis in the clinical setting, and the training program at NEMC to teach physicians to use such methods.

The first of our Artificial Intelligence in Medicine programs: The Present Illness Program. As the limitations of flowcharts and decision analytic programs for large domains of medical reasoning became apparent, the group began to experiment with explicit attempts to emulate the reasoning process of expert human clinicians as the basis for building intelligent medical consultation programs. The Present Illness Program (PIP) developed a representation of knowledge based on *frames*, expressing typical findings associated with various diseases, physiologic states and clinical states found in the domain of nephrology, with additional knowledge indicating which findings were particularly strong indications for the disease hypothesis, flexible scoring functions to help evaluate the fit of observed findings to hypotheses and the degree that hypotheses under consideration accounted for the total set of known abnormalities, and links among frames that represented causal and complicating relationships as well as differential diagnostic lists. PIP used a strategy of *triggering* to evoke hypotheses when they were suggested by strongly-indicative findings, and an activation strategy that permitted it to pay attention only to such triggered hypotheses and their causally linked relatives. The program's performance on selected cases was impressive [Pauk76], though a number of fundamental problems remained unsolved [Szol78, PauSz77]. A later version

of the program also introduced the first explicit temporal model in an AI medical diagnostic program, allowing PIP to do a simple form of reasoning about diseases in the past and their present consequences, as well as to make limited prognostic predictions about the future course of present diseases [Szol76].

The digitalis therapy program. At the time PIP was exploring diagnostic reasoning, our group also began a project in therapeutic management, selecting the administration of digitalis glycosides as a problem domain. The resulting Digitalis program [Gorr78] was the first AIM program with the explicit understanding that it would encounter a patient multiple times. This meant that the program was involved in repeated management decisions, having to generate expectations about what it expected to see on the next encounter, and having to revise its understanding of the patient's case when those expectations were violated. The program used a pharmacokinetic model of digitalis to compute dosage schedules from a goal body stores level and a knowledge of renal function, and used heuristic rules derived from clinical knowledge to select the appropriate goal level based on the patient's underlying disease, factors predisposing the patient to digitalis toxicity, and past experience with the drug in that patient. The notion of the *patient-specific model* (PSM), a data structure that keeps track of all that is known about the patient—all reported data, interpretations of those data, hypotheses under consideration, planned tests and treatments, etc.—developed from this program and continues to play a central role in many of our later projects. The Digitalis program's performance on retrospective cases was comparable to that of the house staff actually caring for the patients, and there was some indication that the program would avoid life-threatening errors by never overlooking relevant factors in its systematic exploration of each case [Long80]. However, the narrow focus of this program and the fact that the management of both heart failure and arrhythmias has altered dramatically (with the introduction of new therapeutic agents) has reduced the value of a program that focuses on the single drug digitalis. Indeed, we have begun work on two independent follow-up projects that explore the patient management problems in both ventricular arrhythmias and congestive heart failure.

The Hodgkin's disease program. In the mid-70's another collaborative project between our groups produced a decision-analytic program for diagnostic test planning and therapy selection for patients with Hodgkin's disease. In addition to its value in bringing together disparate sets of data about Hodgkin's patients, suggesting some general guidelines on diagnostic test selection and providing a common decision model for this class of patients, the program also introduced several interesting new concepts [Safr76, Ruth81]. First, the complete decision tree was not explicitly stored in the program; instead, the sets of possible tests and treatments were represented and the program itself generated those branches of the tree whose expected value seemed significant. Thus, it could work with reasonable-sized decision trees in a domain where the total tree would have thousands of nodes. Second, a sophisticated user interface permitted the user to explore limited "what-if" scenarios and to get an assessment of which facts contributed most strongly to the program's conclusions and which weighed most heavily in the opposition (by ranked likelihood ratios). In later extensions of this formalism to non-Hodgkin's lymphomas and other domains, the underlying computational model was extended to introduce likelihoods and utilities that could be expressed in symbolic forms, not fixed on a numeric scale. Expected values in such a tree are expressions, then, not numbers; we explored the use of a symbolic mathematical manipulation system (Macsyma) to manipulate and simplify these expressions, and their use in performing sensitivity analyses by symbolic solution of the equations.

Clinical use of decision analysis, and the Division of Clinical Decision Making at NEMC. The joint efforts of our group in applying decision analysis to medicine have been centered at the New England Medical Center. The initial work on sequential diagnosis by Gorry was extended

by Schwartz, Kassirer, Gorry and Essig in the domain of acute renal failure. Their more general paper on clinical judgment was one of the first introductions of formal decision theory to the medical literature. Our group developed an interest in the development of simple bedside models of therapeutic and then diagnostic settings. That work introduced the concept of treatment and diagnostic thresholds. Our interest in simplification led us to promulgate the tabular form of Bayes rule and, more recently, the declining exponential approximation for life expectancy. Our interest in more complex models led to work in Markov process representation and the DECISION MAKER computer program for sensitivity analysis. Another program, in very preliminary form, examines a limited frame-based medical knowledge base and generates plausible decision trees using depth first search. No heuristics for model development are supported.

Our interest in the application of decision theory to generic situations led to reports on problems such as coronary surgery (which included the first application of decision theory to prospective decision making in actual individual patients), prenatal diagnosis (which has led to a model applied to over 1500 patients), brain biopsy in herpes encephalitis, the workup of a cold thyroid nodule, the workup of nephrotic syndrome, the workup of Hodgkin's lymphoma, and the necessity for cerebral arteriography in patients with polycystic kidney disease.

Our interest in the assessment of patient utilities and their implications for medical decision making originated in patients with coronary disease and pregnant couples concerned about trisomy 21. We have identified the fallacy of arbitrary indices such as the five year survival rate and the implications of risk averse attitudes among patients. We have examined tradeoffs between quantity and quality of life. We developed a standardized instrument for utility assessment in patients being treated for arthritis and are now applying the same instrument to coronary disease, renal transplantation, peripheral vascular disease and breast cancer.

The Division of Clinical Decision making was formed in 1980 when our group received their first training grant from the National Library of Medicine. The Clinical Decision Making Consultation Service began receiving consultation requests two years before that time after its formation by Drs. Kassirer and Pauker. It remains one of only two such units available, the other being at Dartmouth-Hitchcock Medical Center, formed by J. Robert Beck, MD, one of our first trainees. Our interest in the application of decision theory to individual patients has led to a series of published clinical decision conferences and provides an on-going stream on clinical material that guides our research interests. We are now in the first year of our second five-year training grant. Our program has graduated 12 physicians and currently has another 4 in training. We have also established a sabbatical program that allows academicians from other institution to visit our unit.

Protocol analyses to study human clinical problem solving. Starting with the investigations that led to PIP, we have been continually involved in the formal and informal study of how human clinicians actually make medical decisions. We have found such studies to be a critically important part of our ability to generate new ideas for how to build computer methods of reasoning, and the more formal studies have also generated results of direct interest to cognitive science.

Our early innovations in the representation of knowledge—including applications of frames, triggering relations between findings and hypotheses, and the use of causal and differential diagnosis links between hypotheses—were inspired by two types of empirical observation of clinicians. First, "informal" but very carefully documented observations of physicians in action, collected by a researcher highly trained in AI knowledge representation techniques and supported by physician collaborators, provided an overview of different types of knowledge and problem-solving processes to be investigated [Rubi75]. Second, more formal protocol collection experiments were designed that asked a physician to "think aloud" while taking the present illness of a patient simulated by a researcher thoroughly familiar with the case [Kass78]. The verbatim transcript of these encounters

provided valuable data about diagnostic strategies. This type of protocol collection is sensitive to the natural control structures of the physician's knowledge, and provided strong evidence of the frame-oriented knowledge base, with hypothesis-directed questioning and triggering of new hypotheses. These observations led to the design of the Present Illness Program.

Several years later, motivated by other empirical research on clinical cognition [Elst78] and developments in the methodology of protocol analysis [Eric80], we undertook a review of the literature on clinical cognition and the methodologies for investigating it empirically [Kass82]. We augmented the "thinking aloud" protocol with the "cross examination" protocol, developing the notion of designing a variety of protocol collection techniques sensitive to different aspects of the subject's knowledge representation. Using these methods, we collected data about causal reasoning and explanation by expert physicians, culminating in the development of a new model for causal reasoning by qualitative simulation [Kuip84a; Kuip84b]. Our analysis determined that much of the content of a causal explanation centered around the qualitative description of continuous parameters, describing them in terms of their ordinal relations with a set of "landmark values" and their directions of change. We were able to construct a representation that is an abstraction of differential equations, and provides a qualitative description of the behavior of a mechanism, given a qualitative description of its structure. Thus, as with PIP, we have developed important knowledge representation techniques based on our observations in verbal protocols.

Under other funding, we are currently pursuing similar investigations of physicians' strategies for making decisions among alternatives in critical situations involving substantial risk and uncertainty.

Automatic generation of explanations and justifications. We have long recognized that it is critical for any medical expert system to be able to explain what it knows about a particular area of medicine and how it has brought that knowledge to bear in reasoning about a specific case. Without this ability, a user is left to take the program's advice or warnings on faith, and there is little indication that even good advice would often be heeded if not backed up by justification. Indeed, a recent survey of what physicians demand from potential computerized expert systems places the need for explanation at the top of the list of desiderata, even more important than consistently-correct performance [Teac81].

Underlying our approach to explanation has been the the explicit representation of as much knowledge as possible. After all, it is virtually impossible to explain what isn't known explicitly. Our specific research on automatically generating explanations began with the Digitalis program, which was re-implemented in the knowledge-representation language OWL in the mid-70's to support an ability to translate the program's procedures into readable English text. This first version of an explanation capability could tell how some procedure was done (e.g., how the program determined the patient's sensitivities to digitalis), in what ways any data would be used by the program (e.g., what effect does the patient's age have on the program's reasoning), and how the program's general procedures had applied in the case of a particular consultation. In addition, the program had included a form of sensitivity analysis because it could explain what difference an altered response to any question would make in the program's reasoning chain, using a form of dependency-maintenance [Swar77].

Later studies of how medical students reacted to this program's explanations suggested a fundamental defect in this simple form of explanation (which was also shared with other widely-used explanation methods such as the ones in Winograd's SHRDLU and Shortliffe's Mycin). The difficulty was that the knowledge the program needed to perform its (in this case) therapy selection task was more limited than the knowledge needed to understand why the procedures it used were reasonable. To provide effective explanations, it is not enough to state how something should be done, but to justify from additional background knowledge why that is the reasonable way to do it. The

final version of the Digitalis program, again coded in a general-purpose knowledge-representation language, used methods of automatic programming to *derive* the performance program from an underlying model of the medical domain (e.g., hypokalemia in the presence of digitalis raises the risk of ventricular fibrillation) and from underlying performance principles (e.g., if a drug is being given that, together with a correctable abnormality, raises the risk of a dangerous condition, then recommend correcting the abnormality and in the meantime reduce the level of the drug). Explanations could then be generated that highlighted how the principles and specific relationships in the model interacted to yield the program's behavior [Swart83]. The English generation routines developed in this project were then also successfully employed in the ABEL program because both had been developed in the same knowledge-representation language.

Our more recent studies on explanation have concentrated on questions of how to generate more cohesive English text, to improve the readability of the program's output [Gran82] and on how to generate explanations of complex physiological systems [Asbe84].

Comparative studies of different diagnostic methods. In the late 70's, as we were trying to understand the shortcomings of the first generation of AI in Medicine systems [Szol82], we undertook a comparative study of the PIP and Internist I diagnostic algorithms. Versions of both programs were applied with a new data base constructed for the domain of birth defects diagnosis, and we were able to complete detailed comparisons of detailed differences between the two programs, whose overall approach is similar, but whose details differ significantly [Sher81]. We found, for example, that the triggering scheme of hypothesis activation in PIP was indeed quite effective in reducing the number of simultaneous hypotheses that the program had to consider, compared to Internist's broader "think of anything supported by the findings" approach, without often overlooking the correct hypothesis. On the other hand, Internist's criterion for confirming a hypothesis, which relies only on its relative ranking compared to competing hypotheses, could on several occasions correctly sort out ambiguous cases that left PIP's demand for an absolute level of match between predicted and observed findings unsatisfied. We believe that such comparative studies have not been done often enough, and their realization is one of the motivations for our proposal to bring numerous AI reasoning methods into a coherent and comparable framework within the program we propose.

Introducing causal models, hypotheses at multiple levels of detail and quantitative reasoning. In reviewing the inadequacies of the early AIM programs based on seeking a match of patient facts to expected findings of prototype disorders (whether in an explicit frame-based approach like PIP or based on production rules like Mycin), we focused on one central problem: When a prediction and an observation fail to match, the program has no recourse but to lower its confidence in the hypothesis that made the prediction, and perhaps raise its confidence in another. Yet when we observed human clinicians faced with such problems, a disagreement between fact and expectation was often the occasion for much more active, interesting diagnostic reasoning. Indeed, it appears that just such discrepancies are the key that allows a human reasoner to recognize that a second disorder is present, that this patient is exhibiting some subtle variant of a disorder, that the expression of a disease is partly masked by treatment in progress, etc.

In the ABEL (Acid/Base and Electrolyte) project, we explored the use of causal models of medical relationships at multiple levels of detail to capture in a single system both the typical associational facts known in medicine (e.g., diarrhea causes acidosis) and successively deeper expressions of the mechanism of that association, reaching down to fundamental issues of the conservation of hydrogen ions, the composition of gastro-intestinal fluids, the homeostatic mechanisms of respiration and renal excretion, etc. In addition, this program introduced a notion of quantitative consistency, giving it the ability to reason about whether the degree of one disturbance was sufficiently ex-

plained by the magnitude of its purported causes(s), and thus the ability to hypothesize additional disturbances if the simplest relationship appeared inadequate [Patil81]. Such an expressive knowledge representation also made possible new, more sophisticated diagnostic strategies and plans for optimal information-gathering [Patil82]. Both these issues are further subjects of investigation in the current proposed study. We have also implemented an initial therapy program, ABET, that proposes acute symptomatic therapy for acid/base and electrolyte disorders [Brom83].

Recent and Current work. At present our group is working on a number of problems that form an important part of the background for the current proposal, because we propose to integrate the various methods developed in these other projects into a single computer system. In that environment, the parts can be used together, can call on each other's services as solvers of subproblems, and ultimately form the basis for a new common representation and reasoning scheme in which each of these methods (as well as those derived from earlier, simpler reasoning schemes) is available as interrelated facilities.

Congestive Heart Failure (CHF) project. In the CHF project we are exploring the idea of an "intelligent blackboard" to help physicians think through difficult cases of patients with heart failure [Long82]. The program we are building is to have at its heart a large causal model of nodes representing different qualitative states of different portions of the circulatory system, with links representing the causal relations among these states and constraints that are maintained by a truth maintenance system. The program also has facilities for accepting data and hypotheses about the patient, for interpreting those data within the current clinical context, for propagating the consequences of input data and intermediate conclusions to other nodes in the network, for allowing assumptions to be made by either the user or a program and for maintaining logical dependencies between those assumptions and their consequences so that assumptions and their results can be retracted and alternatives considered (via a TMS). This capability will permit the program to be used not only for diagnosis but also to evaluate possible therapeutic interventions: the user can assume some interaction and explore what consequences are propagated. The program also will eventually incorporate a reasoning module that employs our knowledge of temporal relationships associated with causal links in the network to calculate possible sequences of events and states that must have held in the past to account for what is currently known. For example, consider a program that is reasoning about a future time. In that context, "now" is the future and "the past" of that future is now, which should match the actual present (for diagnostic confirmation) or the present should be changed to match it (for therapeutic reasoning).

The CHF project will continue as an independent research project, with a significant (but far from total) overlap of personnel and interests, but we plan to incorporate the general-purpose reasoning mechanisms such as those described here into the overall system we propose here, to make these capabilities available as part of the research system and to permit their use in our planned focus on the coronary disease and fluid and electrolyte balance problem domains.

The ventricular arrhythmia management project. For the past four years we have pursued a collaborative project with a group of physicians at Boston University/University Hospital to develop a comprehensive system for ventricular arrhythmia management in the setting of a cardiac intensive care unit [Long83]. This system is to incorporate (1) a real-time monitoring module that oversees the outputs from a standard arrhythmia monitoring system; (2) a disease-assessment module that tracks the clinical state of the patient as evidenced by changes in patterns of events from the monitoring system as well as clinical inputs from doctors and nursing staff, reports of therapies undertaken, laboratory data, etc; (3) a therapy module that generates a treatment plan based on (the disease-assessment module's view of) the patient's underlying disease, modifies this plan based on

the patient's response to treatment, and predicts expected results from pharmacokinetic and pharmacodynamic models of the therapeutic agents utilized; and (4) a user-interface and explanation module that attempts to keep the user informed and in control of the complex management process. The overall goal of the Arrhythmia project is to replace the "knee-jerk" reactions sometimes seen in overwhelmed CCU staff with informed decisions based on a computer-aided careful consideration of therapeutic goals, expectations, and incremental successes. This large project, which is seeking separate funding and is currently under review, is in collaboration with a different group of investigators than we propose here, and addresses many concerns (real-time data interpretation, for example) that are not central to the present proposal. There is, however, one major reasoning concept developed in the Arrhythmia project that we plan to incorporate into our program. The control structure of the Arrhythmia program focuses special concern on the problem of being able to change some input data, especially data about a past event, and to have the system determine the present consequences of that change. This is difficult for two reasons: First, one cannot simply roll back the clock and start reasoning again from the changed data because the program will have taken actions based on what it had believed to be true, and these generally cannot actually be undone if they affected the real world (not just some internal model). Second, it is impractical to include in every knowledge source in the program an explicit component to deal with possible changes of its inputs, and how to respond. Therefore, a mechanism is needed that effectively allows any knowledge source to act as if its input data were unchangeable, and then provides a general facility to re-do that reasoning that does in fact have to change. Such a facility has been designed [Long84], and we plan to incorporate its ideas in the proposed system.

Qualitative simulation. Beginning with some early work in our group on the relationship between anatomical and physiological function [Smith78] and later general work in AI on endowing computers with reasoning capabilities about simple processes in the physical world [Hayes79], we have been interested in extending the causal reasoning models of the ABEL and CHF programs to the level of physiological models. At the same time, Ben Kuipers and Jerome Kassirer, as part of their investigation of the role of causal reasoning in human clinical thought, have developed detailed qualitative physiological models from explanations given by human doctors. These models differ from classical differential-equation models in that they can (and typically do) severely underspecify the precise quantitative relations involved. For example, a functional relationship between two parameters may not be known exactly, but it is simply known to be monotonically increasing. A value may be unknown, but it may be known that it lies between two particular landmark values and is decreasing toward the lower one.

The QSIM (Qualitative Simulator) program takes such a qualitative description of a physiological process and simulates the behavior of the system forward in time, determining significant future time points (e.g., when that falling value reaches its next landmark) and the possible qualitative behaviors of the values of all parameters at those points and intermediate intervals. Clearly, sometimes this system cannot determine exactly what will happen and must explore alternative futures. For example, when two quantities are changing at the same time, it is in general impossible to tell which will reach its next landmark first; similarly, if a quantity is being increased by one effect and decreased by another, its direction of change cannot in general be told without knowing the relative magnitudes of the influences. QSIM extends the the abilities of programs to use causal relationships in their reasoning by allowing those relationships to be computed from a deeper model of the processes that underlie them. Thus, for example, rather than simply encoding primitively that an increase in one physiological parameter causes another one to decrease, QSIM would run the underlying model to determine what happens to the second parameter. QSIM is being used to model and reproduce the observed reasoning of human subjects. Alan Moskowitz and

Ben Kuipers are in the process of assembling a small diagnostic program based on QSIM, which uses hypothesis evocation strategies similar to PIP's, predicts what should happen under those hypothetical circumstances by qualitative simulation, and then evaluates hypotheses by trying to verify the predictions of the QSIM model. We intend to incorporate this qualitative simulation capability in our proposed program, and to develop our integrated program and knowledge base to support just the sort of experiments represented by this combination of different components.

Qualitative mathematical reasoning. In a recent Master's thesis, Elisha Sacks has extended the qualitative reasoning notion by introducing an ability to reason with the mathematical form of the equations describing a system. Previous qualitative reasoning systems typically limit what can be said about the values of parameters to identifying which range of landmark values bound them and the sign of their derivatives. Functional relations are also usually limited to rough characterizations of their basic form (e.g., monotonic or not, increasing or decreasing). In Sacks' QMR program [Sacks84], by contrast, any function can be defined by an exact closed form on a set of intervals, or by successively weaker parameterized forms, or by the rough characterizations of other qualitative reasoners. QMR makes use of more precise specifications when they are available. In addition, because the mathematical form that characterizes some behavior is explicitly represented, QMR can often describe the behavior of a system by analysis of the mathematical form of its total (though perhaps under-specified) time behavior rather than by step-by-step simulation of its evolution in time. This makes it much simpler to identify asymptotic or cyclic behavior than in programs that generate a trace of the predicted future of a system and then have to extract appropriate characterizations of that behavior from the trace. Although QMR promises to be a powerful analysis tool, thus far it has been applied only to quite simple physical systems (though ones typical of other qualitative simulation exercises), and requires further extension to deal with larger and more complex systems such as we are likely to encounter in physiological models.

Reasoning about preferences. One domain in which QMR is already finding some application is a project being pursued by Michael Wellman, to develop a program that will reason about the partially-specified preferences of individuals. The Utility Reasoning Package (URP) encodes a large body of knowledge from the utility theory literature and attempts to answer questions about a subject's preferences given some information about them by applying theorems of utility theory. Much of this reasoning also must be done qualitatively because precise quantitation of individual utilities is difficult to obtain and errorful.

5 Experimental Design and Methods

5.1 An Integrated AI System for Medical Reasoning

Despite the record of success of medical AI research programs during the past decade, the sad fact is that only two rather small programs based on AI methods (for pulmonary function test evaluation and for interpretation of serum electrophoresis results) are in routine (not developmental) clinical use. In fact, the field is strewn with the carcasses of "successful" projects. Most of the techniques that have been developed in one research project are reflected, at best, in ideas transported into others, with very little opportunity to make use of previously-developed knowledge bases or reasoning modules in the service of new systems. Therefore, the researcher interested in implementing a program for exploring the use of detailed physiological reasoning based on qualitative simulation to evaluate diagnostic hypotheses,[1] for example, must create some form of knowledge representation in which to express the program's underlying knowledge, must find an active medical collaborator to help select a medical domain in which the techniques may be refined and tested, must participate in the collection and debugging of at least a small knowledge base in the selected domain, and must then choose some existing hypothesis-generation method as described in the literature, implement a version of it that fits with the other components listed above, debug it, and perhaps make some refinements just to set the stage for the research problem originally posed. Note that although hypothesis generation can be an interesting domain of research, that is not what the exemplified researcher is interested in here; indeed, there is perhaps no need to advance the state of the art in hypothesis generation in order to make substantial progress in the use of qualitative simulation as a hypothesis-validation tool. Yet all the other components of work must still be done, and even when they are done, the resulting system will still lack some capabilities of other systems that have been built to explore different but similar objectives. For example, another system that addresses hypothesis generation may have useful and extensive capabilities to generate explanations (not critical in this example project, but of some significant value) but these will be unavailable unless the list of pre-conditions of the desired research is extended by yet another component to build.

The consequence of this state of affairs is inefficiency in research progress, and great difficulty in integrating the results of diverse research efforts toward a comprehensive system. Feigenbaum had recognized this difficulty already by 1977, when he called for new generations of AI/Medicine researchers to "stand on their predecessor's shoulders, not their toes." [Feig77] The codification of one simple knowledge-representation and inference scheme, EMYCIN, did in fact serve as a vehicle for this admonition, and much of the commercial success of "expert systems" can trace its lineage to such simple rule-based systems. The technical means to support advanced research programs, however, were limited in the late 1970's, and most researchers who were attempting to advance new ideas about medical reasoning based on such innovative ideas as the use of causality, multiple levels of detail, time, qualitative simulation, etc., found the available general-purpose tools inadequate.

Within our own research group, of the many interesting research projects reviewed in section C of this proposal, only a very few examples of actual sharing of code or knowledge can be found. The comparative study of the PIP and Internist-I diagnostic strategies actually made use of the code of the original PIP, outfitted with a new knowledge base about birth defects, and ABEL made use of the language-generation parts of the explanation-oriented version of the Digitalis program. This means that many opportunities for sharing of code and data have been missed, and many researchers have been distracted from their primary interests by the need to build enough of a infrastructure to test their newly-contributed code and ideas.

This state of affairs is reminiscent of the symbolic mathematical computation community of

[1] This example, reflecting the experience of Dr. Benjamin Kuipers in our group, is an actual one, not invented.

twenty years ago, wherein one project might develop an excellent program for factoring of expressions, another for integration, a third for power-series expansions, and a fourth for expression simplification. Not until the late 60's, however, with the development of such comprehensive systems as Macsyma, were all these capabilities integrated to provide a better environment both for developing new subsystems and for users with complex, multi-part problems.

Two important lessons should be emphasized from the Macsyma experience: First, a large degree of integration led to the selection of a small number of systematically-supported representations, general translation facilities among those representations, and a requirement that any function in the system must be able to accept any validly represented input without error.[2] Second, integration did not go so far as to demand a single, uniform representation for expressions or a single approach to functional tasks. Thus, Macsyma recognized that an expanded polynomial representation may be best for addition, but it is terribly inefficient for computing the integral of $(x+1)^{1000}$.

Our proposal here is to undertake an effort akin to that of the Macsyma project: to build a coherent, integrated system for medical reasoning that is based on a uniform underlying representation of knowledge and to provide modular reasoning facilities that address alternative means of diagnostic and therapeutic reasoning within a circumscribed medical domain. The principal reasons for undertaking such a project are (1) to improve the efficiency of research, as discussed above, (2) to encourage efforts in particular toward the successful integration of probabilistic and categorical reasoning, and (3) to face the issues of how to capture, represent and reason with the comprehensive medical knowledge of some limited area of medical expertise and how to capture, represent and use a panel of cases within that limited domain.

We propose to approach the desired integration we seek in two stages: First, we shall build interface facilities among modules of several of the existing reasoning systems that we now have, to permit experimentation with their joint use. This will involve characterizing precisely what combinations of programs we wish to try to use together, defining what information needs be passed between them, and building special-purpose, *ad hoc* translation facilities to support their common use. Examples of possible such interactions are: the triggering machinery of PIP for hypothesis formation coupled to an hypothesis evaluation scheme based on QSIM (to carry further the example from above), and the URP program that performs a qualitative analysis of preference structures with a decision-tree evaluator. Of course, before such integration can be achieved, we must dissociate each existing program into its components, some central and some supporting.

The second, and larger, step toward our goal is to define a common set of knowledge representation formalisms and to build new reasoning programs that are fully based on these, so that they will be coupled together by virtue of their underlying representations, not only through translations between their inputs and outputs.

5.1.1 Representation Language

The first requirement for an effective close integration of a large number of reasoning components is that they share some common underlying language within which to communicate their requirements and results. Although we propose initially to foster communication among various existing reasoning modules by special-purpose translation, that is clearly not the long-term solution. Instead, we ought to have a language in which any component of any internal "thought" of any of the

[2]This did not require that every algorithm must work with any representation. The function could explicitly ask for conversion to a representation it was capable of processing, or it could even simply return a trivial symbolic answer: e.g., asking for the integral of a form the system could not integrate would return the represented equivalent of "integral(...)".

reasoning modules can be stated. This means that the representation must be able to express such typical associations as "diarrhea causes acidosis," "a liquid stool is usually basic," "patient A has lost 10 pounds in the past 24 hours," "it is unlikely that any patient loses 10 pounds in 24 hours," "although patient A is reported to have lost 10 pounds in 24 hours, this fact is disbelieved because of the previous principle," however "in a series of patients with cholera (ref ...) the mean daily weight loss was ten pounds." Indeed, virtually any of the technical issues that we are interested in advancing finds a counterpart in the required knowledge representation. Thus, we need to develop a uniform representation language in which we can make statements about causality, temporal relations, preferences and values, plans, the behavior of (physiological) systems, likelihoods and distributions of probabilities, facts of biochemistry, anatomy and physiology, records of previous cases, summaries of data from the literature, records of what a program has been told, what it has inferred, and what it has recommended, and internal states of the reasoning system itself, such as its goals, assumptions, and problems.

This comprehensive list defines a difficult and perhaps overly-ambitious research project. We are saved from the implications of such a difficult task to some extent by the fact that we are not attempting to solve these representation problems in their total generality, but with respect to the particular representational needs of the medical domains we are focusing on and the reasoning components we are developing. [3] Therefore, the representation we shall develop will evolve during the course of the project as we identify additional capabilities needed from it. There will, however, be a major early effort to design a representation that will serve the needs of those modules to be included from the beginning.

It is important to note here that what we need to develop is not another knowledge representation language of the form ordinarily thought of by artificial intelligence researchers, such as KL/ONE, FRL, OWL, etc. Indeed, we plan to use such a representation language as the implementation medium for what we need. To understand the distinction, it is useful to consider the contrast that Alan Newell makes between the *symbol* level and the *knowledge* level of understanding of an AI program. At the symbol level, a medical diagnosis program is seen to be performing activities such as calculating the values of certain parameters, applying inference rules, asking questions of the user, etc. At the knowledge level, it is making hypotheses, testing them, predicting the outcome of some therapy, etc. Naturally, its knowledge-level processing is implemented in terms of its symbol-level capabilities. Thus, at the symbol level, the program's making a new hypothesis may appear as the addition of a symbol such as `hypothesis-12` to the list that is the value of the parameter `current-hypotheses`. There is indeed often a fairly deep hierarchy of such levels, for the symbol level may itself be implemented in some other programming language, which in turn runs in the machine language of the computer, which ultimately is pushing electrons around wires. In general, it is preferable to implement the knowledge level needed for an application in that symbol level language which comes closest to directly supporting the needs of the user at the knowledge level. As a very simple example, consider that if sequences are to be an important element of reasoning at the knowledge level, then a symbol-level implementation language that has strong support for manipulating lists would be a good choice.

Choosing a symbol-level language in which to implement our common representation is difficult because even the best well-developed, well-supported flexible languages such as Lisp are at too low a level to provide much of the support we will need. Therefore, we are looking at more advanced representation languages such as NIKL, KL/ONE, KRYPTON, FRL/HPRL, OWL, etc.

[3] Lest the reader conclude that this task is impossible, we already have a well-developed prototype for such representations—English. The only problem is that that system is difficult to manipulate within the computer. We are *not* proposing to deal with natural language in general, but do anticipate that we shall require a very general representation scheme.

as possible choices, and have tentatively settled on NIKL as our working choice. All of these languages supply important capabilities we shall need in building our own representations, such as taxonomic hierarchies, automatic inheritance of attributes and values from hierarchic superiors, methods of categorizing new instances into their appropriate place in the taxonomy based on their characteristics, etc. In contrast to lower-level programming languages, however, many of the design issues of these knowledge representation languages are not well fixed, and languages tend to change often in response to new demands on them or a newly-developed understanding of their mechanisms. There is the risk, then, that in using such a knowledge representation language, one will be drawn into the arguments at what should (for this project) be the symbol level, and may have to ride atop a moving base as that language continues to change. Despite this danger, we have no real alternative to building on top of such a language, because the kinds of facilities provided are needed; if we were to begin with Lisp, for example, we would simply have to build up our own version of some representation language (which we have indeed done in the past [Szol77]).

Among the available languages, NIKL [Mose84] appears to have the best current compromise between elegance of design and practical usability. NIKL is the New Implementation of KL/ONE, and falls into the current tradition of languages that maintain a distinction between a *definitional* and an *assertional* component. The definitional component encodes, not surprisingly, definitions of terms and their hierarchic relationships. What is important about NIKL and the best of the other languages in this tradition is that the hierarchic relationships are present by virtue of other knowledge in the system, not simply because of arbitrary declarations by the user. Thus, if `anemia` is defined as a `disease` with a role `hematocrit` that is restricted to be less than 38%, then any other disease with such a role and a restriction on its value that is more restricted (e.g., less than 25%) will automatically be *classified* under `anemia`.[4] NIKL also supports multiple inheritance, so that a single concept may be classified under several others. The definitional knowledge expressed is *monotonic* (i.e., not subject to retraction during reasoning). NIKL also provides facilities for *instantiating* its knowledge to form specific instances of concepts that represent not generic conditions but their occurrence in a specific patient; this is important for representing the patient-specific models. NIKL's assertional component, the means it has of stating arbitrary facts about the world and deriving conclusions from them, is not precisely defined; it can use virtually any inference mechanism, from a general-purpose theorem-prover (practically intractable) to arbitrary computations (which can be fast, but not necessarily correct).

The task for us, then, in using such a representation language, is to determine how to express the kinds of concepts relevant to medical reasoning in the representation language in such a way that the power of the underlying language is exploited. For example, if the knowledge is appropriately structured, NIKL can automatically derive that the causal relationship between bacterial endocarditis and renal infarct is a specialization of the more general causal relationship that some heart diseases cause some kidney diseases. Furthermore, additional concepts, such as "diseases that cause renal diseases" and "diseases that cause renal infarct" are all classified appropriately to form the generalization hierarchy needed by a reasoner. Making this happen, though, is often difficult, and achieving a representational design that causes the right relationships to be manifest in the organ system hierarchy, etiologic hierarchy, temporal hierarchy, etc., will be a major task of our design.

[4] There is some technical difficulty here with how ranges of numbers are represented, which is not fully demonstrated in this example.

5.1.2 Components of the Integrated Reasoning System

What are the components of an integrated medical reasoning system? From our retrospective analysis of the parts of the medical systems whose capabilities we would like to bring together, we have identified the following components that we intend to implement in the proposed system:

1. A data interpretation module transforms externally-reported data into the form used by the rest of the system in its reasoning. Although every program must do this, our most extensive experience with data interpretation as an explicit module is in the heart failure program, which will serve as our initial model.

2. A module for hypothesis formation takes a set of initially-presented facts about a case and produces the initial hypothesis structure that will drive the program's reasoning. Examples of such a module are the triggering mechanism in PIP, the EVOKER program currently being developed, the special-purpose acid/base nomogram interpreter of ABEL, and Davidoff's GENERALIST. This one area in which there are many competing methods.

3. Hypothesis evaluation is the process of validating the predictions that arise from a hypothesis. The two principal approaches to date have been to use a measure of likelihood derived from a Bayesian or pseudo-probabilistic scoring function, or a measure of conceptual completeness and consistency, implementing a version of Occam's Razor—the simplest hypothesis that accounts for the facts is best. How to figure out what hypotheses account for what facts is in turn an interesting problem, that can be addressed by simply listing manifestations associated with each known disease, calculating the list from a chain of causal possibilities, or simulating the mechanism of the disease to predict its consequences. Categorical methods may also be applicable, such as the confirmation of an hypothesis by verification of its one pathognomonic finding.

4. Hypothesis revision suggests appropriate modifications of the hypotheses under consideration. These may be based on purely internal reasoning (e.g., deciding that there is now enough specific information known in a case so that aggregate hypotheses should be disaggregated to allow differentiation among the more detailed alternatives), or on discrepancies arising in hypothesis evaluation when new information fails to match what was predicted. Revision of a set of hypotheses may involve restructuring of individual hypotheses, the aggregation of several into one, the introduction of completely new hypotheses (by the methods of hypothesis formation) and the deletion of existing hypotheses.

5. Data gathering is the active part of the hypothesis evaluation/revision cycle, suggesting new facts that would be useful to help resolve remaining uncertainties. Strategies here can range from the most simple—ask about the first piece of supporting evidence in favor of the leading hypothesis—to much more sophisticated—plan a sequence of tests that will yield the most information at the least risk. This issue becomes even more complex when competing hypotheses lead to competing and conflicting goals (see section D.5).

6. Explanation and justification concerns ways of describing what the program believes to its users and convincing them that those beliefs are supported by valid underlying knowledge. Strategies here range from means of translating the procedures and knowledge structures of the program into English to the much more difficult task of fitting an explanation to what the user is thought to know already.

7. A causal reasoning module propagates logical causal and temporal consequences of what is known within a network of cause/effect relationships. This is a sufficiently common component of other modules that it should be separately broken out. Dependency maintenance is also a component of this module. This capability will be useful in hypothesis formation, evaluation, and explanation.

8. Multi-level causal reasoning permits a system to deal with problems at various levels of detail, from a "shallow" or associational to a "deep" physiological one. Only the ABEL program (and, in design, Caduceus [Popl81]) has successfully exploited this capability thus far. We propose to distill this out into a separate facility.

9. Therapy planning and management involves the repeated evaluation of the patient's state and the recommendation of some action (e.g., provide therapy, re-think the diagnosis, simply wait for further developments). This may be driven by the degree to which therapeutic expectations are met (as in the Digitalis and Arrhythmia programs), by following carefully laid-out contingency plans, or by decision analysis.

This list is not meant to be exhaustive, but represents currently-identified reasoning components.

The next section discusses further details of various approaches that we propose to investigate for each of these modules. To return to our hypothetical researcher at the beginning of this section, our intent is that he or she will be able to interface with a number of these components to form a complete, testable program. Further, comparative studies of alternative versions of any such module will become easier, and this valuable form of critical research will be encouraged. Later sections also discuss the codified medical knowledge base that should also be available to the researcher and the panel of cases that will facilitate the testing of such new components.

5.2 Components of Medical Reasoning

As introduced above, an integrated medical reasoning system must have a number of components to deal with the various tasks that comprise the overall diagnostic or therapeutic task. This section describes several approaches for these components, and the extensions we plan to investigate within the framework of this project.

5.2.1 Data Interpretation

Data interpretation is the process of assessing the validity and meaning of raw input data in the context of the individual case. This process functions as a separation of labor between the tasks of assigning meaning to data and the tasks of using the data for diagnosis and management. The data interpretation task can be divided into two parts. The first is assessing the validity of the data. Consider some examples: Laboratory values are subject to both random variation and not infrequent errors from mislabeling, contamination, and so forth. By knowing the possible changes over time of the parameters being measured and the constraints on relationships between parameters, a program can recognize many of the values as in error or likely to be so. Similarly, the validity of a weight change can quickly be clarified by asking if the weight was taken on the same scale as the previous weight. Physical examination findings, such as those from auscultation are greatly dependent on the skill of the observer and the nature of the physical environment. For example, the occurrence or lack of a gallop should have different significance if it comes from a skilled observer listening in a quiet room rather than an intern listening in a noisy setting to a patient on a respirator.

The second task of data interpretation is assigning meaning to the findings in the context of the patient. Comparing findings to "normal" is not sufficient. Normal ranges are determined from the statistics of a population. The population information about a parameter only tells the percentage of the population for which a particular value would be abnormal, not whether it is abnormal for the individual. Furthermore, disease states can change the appropriate values for a parameter in the individual. For example, for a patient with a history of chronic hypertension, a blood pressure of 140/85 may be too low. However, if the same individual suffers a severe myocardial infarct, that same pressure of 140/85 may be too high. The problems of validity and meaning are addressed to some extent for specific domains by ABEL and the heart failure programs. For example, ABEL interprets electrolyte values in terms of the likelihood they are abnormal as well as in the context of the other electrolytes. If one electrolyte value is changed, it may change the assessment of another electrolyte because the overall picture is now more consistent with a different interpretation. Similarly, the heart failure program uses the distance of a value from the normal range to determine the strength of the evidence for an abnormal value and combines that with the evidence of causes or effects from the physiological model to assess the value.

The exact boundary between data interpretation and medical reasoning is domain specific and dependent on what form the data must be in for the medical reasoning. For example, the reasoning module in the heart failure program requires the truth of the qualitative parameter values in the model with associated time intervals, a certainty assessment, and severity. Until the data are in that form, they cannot be interpreted. Thus, the interface between the data interpretation module and the reasoning modules may require communication in both directions. Typical information will consist of a *priori* likelihoods for values (determined from established causes or effects, for example), and requests for data assessments in situations where many possible assessments could be made.

The mechanisms mentioned above are only partial answers to the problem of data interpretation.

For the general framework, we shall develop a more extensive set of mechanisms to handle the needs that arise in medical domains. A partial list of the mechanisms that should be available for data interpretation is as follows:

1. Mechanisms to handle validity constraints from physiology or other fixed bounds, constraints from other parameter values, and the constraints of parameter value changes across time.

2. Mechanisms to handle possibly erroneous data.

3. Mechanisms to handle the correction of data (see [Long83a] in the appendix for a mechanism for propagating the changes in data to those interpretations that depend on the value).

4. Mechanisms for providing derived values from the entered data. For example, the Henderson-Hasselbalch Equation gives the pH in terms of the ratio of the HCO_3 and the pCO_2. The user should be able to enter any two of these values and have the program reason with all three.

5. Mechanisms for interpreting descriptive information. These would be domain specific interpretation programs, such as a program to assign the likelihood of angina and other types of pain from a description of chest pain.

6. Mechanisms for deciding what "otherwise normal" and other forms of missing information mean in the context of the patient.

7. Parsimony, probability, certainty factor (etc.) based interpretation mechanisms incorporating constraints on the likelihood measures [Yeh85].

Thus, the modules we will develop will support reasoning about the validity of data and about the meaning of data, starting from the techniques we have used in other projects, reimplemented to conform to the general framework, and extending to cover the other data interpretation requirements we have identified.

5.2.2 Hypothesis Formulation

One of the most fascinating observations from studies of clinical cognition was the discovery that cognitive processes involved in diagnosis are very similar to those involved in scientific fields in general, namely, that they are hypothetico-deductive. Clinicians generate a small number of specific hypotheses, very early in the diagnostic process, which are continually evaluated, revised, and elaborated during the process of diagnosis until adequate diagnostic understanding is achieved.

The experience in the AI in Medicine field also suggests that the hypothesis formulation activity plays a central role in effective management of the large space of possible diagnoses that a clinician (or a successful AI program) must deal with. A number of strategies have been tried in the past such as the "triggering heuristic" in the Present Illness Program, the "evoking strength calculation" in INTERNIST-I. To develop a better understanding of these techniques, an experimental study was performed in which several of these strategies were implemented in the same domain (using the same knowledge base). Based on the results of this study [Sher82], a number of potentially useful new strategies and combination of existing strategies were identified. In order to explore these new strategies effectively, Kuipers has implemented a new program called EVOKER, which provides meta-level support for rapid prototyping of these heuristic strategies. We propose to use the EVOKER program in two ways: first, to evaluate triggering strategies to be used for

program development, and second, to assist in protocol analysis by providing tool for simulating the observations.

Although use of triggers and other "associative recall" mechanisms provide us with considerable advantage by identifying only those pieces of information potentially relevant to the case, they alone can not be relied on to reduce the space of possible hypotheses sufficiently for effective diagnostic reasoning. Considerable thought must be given to structuring the space of possible hypotheses and developing algorithms which can be used to group and reduce the number of these hypotheses to a small and manageable size for effective diagnostic reasoning.

A number of programs, most notably INTERNIST-I, have attempted to solve this problem through the use of a hierarchy of diagnostic hypotheses. However, the use of hierarchies in these programs has failed to achieve the expected results. There are two main reasons underlying these inadequacies. First, the inadequate and often overly restrictive semantic basis for defining the hierarchy. For example, the disease hierarchy in INTERNIST-I requires that the manifestations associated with an aggregate node must be those and only those which are common to all diseases under it. As a result, the number of manifestations associated with an aggregate node decrease rapidly as we move up the hierarchy, so much so, that aggregate nodes such as "liver disease" do not have even the most characteristic finding such as "jaundice". Thus, INTERNIST-I is deprived from making effective use of the hierarchic organization. The second problem results from mixing a number of different commonalities (or criteria) in organizing a hierarchy and the resulting a *priori* selection of one of these commonalities in decomposing each of the aggregate node into its sub-nodes. As a result, when this hierarchy is used during the process of hypothesis formulation, either for grouping a number of similar hypotheses or for refining a hypotheses into a set of more specific hypotheses, only commonality that was selected a *priori* in forming the hierarchy can be used, which may or may not be suitable in the context of a given case.

We propose a new organization for structuring of the possible hypotheses which takes a much broader view of the task. We believe that a substantially richer and more principled organization of this knowledge is essential for advancing the state of art in medical diagnosis. Such a representation will contain a number of hierarchies (in a spirit similar to that proposed in ABEL and CADUCEUS). These hierarchies, however, would be used to organize different components of medical knowledge where each hierarchy deals with only a single component. For example, one such hierarchy would describe the functional anatomy, another to organize the specific etiologies, and yet others to describe homeostatic mechanisms, temporal characteristics etc. Because of the choice of a single theme for each hierarchy, these hierarchies can be organized in a coherent and systematic fashion such that they provide a smooth and organized progression of concepts as we move from one level in the hierarchy to the next.

Each of these hierarchies will then be used to characterize the space of disease hypotheses. Thus, for example, the disease of Acute Glomerulonephritis would be characterized as an "acute" disease (temporal characteristic) involving "glomerulus" (anatomical site) caused by an "immune reaction" (specific etiology), etc. Such explicit characterization of each disease hypothesis will permit us to choose, at run time, one of the commonalities which best suits the needs of the problem solver.

Furthermore, in the early phases of diagnostic process, when sufficient information is not available for effective generation of a small number of hypotheses, the same structure can also be used to characterize the patient's condition along each of these dimensions, and the problem solving directed towards refining these characterizations, until sufficient information has been obtained to clearly identify a small set of hypotheses. A program for formulating the differentiation problem for the presenting complaint of pain based on detailed characterization of various components of the pain (GENERALIST) was implemented by Frank Davidoff (while visiting our group). The program was able to begin its questioning with only one or two presenting complaints and with

only a few more questions was able to characterize the patient's illness sufficiently to identify a small and well defined differentiation problem. The mechanisms embodied in this program will be useful for a variety of medical domains when packaged as part of a complete framework, although the same form of analysis that it includes for pain will need to be undertaken for other common presenting complaints.

One of the most difficult problems faced by diagnostic systems is the decision whether the patient under consideration is suffering from single or multiple disorders. Making a single disease assumption (as is done implicitly in PIP) considerably simplifies the diagnostic task by simplifying the problem of properly accounting for findings. That is, under this assumption, a finding is either explained by a disease or is not explained by a disease. A program based on these techniques may, however, be misled in the presence of multiple disorders when the diseases interact with each other by either aggregating some symptom or canceling the effects on others.

To deal with such difficult situations, a new approach must be taken which incorporates explicit notions of severity, quantity, duration and onset of symptoms, and that has sufficient pathophysiologic knowledge to help sort out the potential interactions among co-occurring diseases, when such interactions are suspected. Our ABEL program addresses these issues using three key new ideas. First, a significantly more complex representation of the program's hypotheses called "composite hypotheses" are used. This new representation of a diagnostic alternative includes hypothesized diseases, clinical states and relations which together form a complex that accounts for observed findings from all aspects of the case. Furthermore, they clearly identify those that can not be adequately accounted for and thus form a focus for further diagnostic problem-solving.

Second, it allows a disease to account for only that part of a finding that is justifiable, requiring the remainder of the finding to be accounted for by other complicating factors in the complex. Furthermore, when a complicating factor is suspected but not known, the same mechanism can be used to determine the component of the finding that "remains to be accounted for" to direct its search for the complicating factor.

Finally, ABEL tries to approach the problem of diagnosis at multiple levels of detail, from causal knowledge of common clinical associations at the top to detailed knowledge of pathophysiology at the bottom level. Such an approach allows us to exploit the clinical knowledge in proposing different ways of extending a hypothesis (i.e. exploring the space of diagnostic alternatives) and proposing possible relationships among the observed findings and clinical states, while the detailed knowledge of pathophysiology allows us to verify the physiologic validity of each proposed relation, and to determine the effects of co-occurring diseases in a systematic manner.

5.2.3 Hypothesis Evaluation

On of the major tasks in any diagnostic system is to evaluate competing hypotheses in terms of the relative strength of our belief in the truth or perhaps in terms of the relative importance of their pursuit. Two basic streams of hypothesis evaluation can be considered—probabilistic and categorical.

Probabilistic evaluation The probabilistic evaluation of competing hypotheses is most often based on the use of Bayes rule (when full knowledge of relevant conditional probability data is available) or weaker scoring rules when information is incomplete. Formal Bayesian scoring, which would involve specifying a full matrix of condition probabilities, is often replaced by a two state world in which is diagnostic hypothesis is present of absent and in which the condition probability of a finding with hypothesis absent is taken as a fixed number, independent of the distribution of

other diseases within the disease category. We have examined that situation [Szol78] and delineated the potential errors that such an approach might produce.

Clearly the Bayesian and pseudo-Bayesian approaches are best applied when the domain of consideration has been sufficiently constrained. We shall experiment with the idea of using categorical techniques to restructure the domain of consideration into a form in which such techniques might be applied. When such applications force the program into the "two state fallacy," pointers in the knowledge base will make the inference program aware of the potential problems and will appropriately constrain the propagation of weak conclusions.

Causal consistency, minimum unexplained findings: coherence and parsimony An alternative to calculating some approximation of the likelihood of hypotheses is to compare them instead according to a different criterion, the quality of explanation provided by the hypothesis. The quality of an hypothesis is based on the number of different causal explanations that are necessary to explain observed findings, the number of findings that cannot be completely explained and the number of possible etiologies that must be hypothesized to completely explain the observed findings. A hypothesis which uses the smallest number of independent etiologies and leaves the least number of unexplained findings is judged to be better that all the others. In the narrow domain of the ABEL program, we have found that this approach works remarkably well in narrowing the field of hypotheses because it requires that each hypothesis be a physiologically and clinically valid explanation of the patient's illness.

One special circumstance that permits relatively easy categorical decisions to be reached is when a pathognomonic finding definitely indicates the presence of a disease. In addition, a program can blur the distinction between truly pathognomonic and simply strongly indicative findings so long as the resulting occasional errors can later be recognized and corrected. The use of underlying reasoning mechanisms based on constraint propagation and dependency maintenance should permit such aggressive evaluations to be made, recalling that they may need to be retracted if they fail to hold up in further investigations of the case.

In conjunction with the hypothesis formation and evaluation process based on causal modeling of the patient's illness, we have also been working on reasoning techniques for projecting qualitative causal models forward for identifying the prognosis based on each model (QSIM). Such qualitative models can not only be used to predict the consequences of possible treatment/management alternatives for the patient, but can also be used to evaluate the validity of a given hypothesis by matching its predictions with the observations. Thus, if an hypothesis is correct, then the consequences predicted by simulating the mechanism of the hypothesized disease should match those seen in the patient.

Ultimately, in situations where there are two or more acceptable explanations of the patient's illness, the program will be faced with the difficult problem of making decisions about its future course of recommended action. In such situations, a probabilistic evaluation of the likelihood of each alternative can provide an important bridge to the use decision analytic methods in selection of appropriate management actions, and we shall need to develop some approximate methods for making a *posteriori* probability estimates of complex hypotheses even if such probabilistic values are not used earlier in the evaluation process.

5.2.4 Hypothesis Revision

The result of evaluating a set of current hypotheses is typically used either to revise the set of hypotheses or to revise individual hypotheses within the set. The set is revised by adding new hypotheses, deleting old ones or aggregating subsets. Individual hypotheses are revised by intro-

ducing extensions to account for new data or newly-discovered discrepancies, or by some internal re-organization to move toward simplicity or completeness.

The most interesting current work on hypothesis revision is that being pursued in the Caduceus project [Popl82], where a uniform space of causal and taxonomic links is matched against a small number of heuristic operators that dynamically re-organize the program's knowledge into new problem sets. Although the combinatorics of such an approach may make it too costly to apply universally, we plan to incorporate that basic insight, limiting its application only to situations where the evaluation module reports some serious difficulties with the currently-formulated hypotheses.

One dramatic form of hypothesis revision, which has been one focus of our own work recently, is the decision whether the patient under consideration is suffering from a single disease or a multiplicity of disorders. Although making the assumption of a single disease is an attractive simplifying assumption made by many earlier programs, it is inappropriate for the types of serious cases that would be most likely to come up for consultation. One key to recognizing that a single disorder fails to adequately explain a case is if either no single disorder accounts for all the known abnormal findings. Thus, it is not enough to downrate a hypothesis during evaluation for leaving important facts unexplained; this may be just the clue that the hypothesis must be extended with additional components. A second key is that many findings have quantitative and temporal characteristics, and that although some hypothesis may be able to explain some form of that finding, it may not be able to account for its degree of severity or its precise time of occurrence. Clearly, the representation must support the statement of quantitative and temporal characteristics to make this hypothesis revision strategy applicable. Given this ability, deficiencies identified in hypothesis evaluation can be the clues that lead directly to the revision of hypotheses to account for those additional findings or portions of findings. More subtly, this form of reasoning also applies to the case of "the dog that didn't bark," in that failure to observe an expected finding may indicate that it is being masked by some other abnormal process. We plan to include the types of quantitative representations developed in ABEL in our knowledge representation, and to develop further revision strategies based on the notion of hypotheses partially accounting for quantitative evidence.

5.2.5 Data Gathering

The diagnostic task is made up of two parts: one consists of hypothesis formation and revision to interpret the known data and the other deals with data gathering to clarify difficulties remaining with the hypotheses under consideration. Much of the research in AI in Medicine in the past has focused on efficient means for hypothesis formation and evaluation with only a small fraction of efforts going to improving data gathering strategies. For example, at one extreme, rule-based programs such as MYCIN use a predetermined order in which to ask questions, and at the other extreme, frame-based programs such as Internist and PIP, maintain very little control over the data gathering process from one pre-packaged group of questions to another. In more recent years programs such as ABEL have attempted to introduce the notion of planning to generate a data-gathering plan which can be used to provide more efficient and coherent questioning of the user.

Another problem, quite common in areas such as clinical medicine where enormous amounts of information must be handled routinely, is the possibility of errors in the data. Therefore, an ability to identify questionable information and to challenge and to correct it quickly is an important ingredient of clinical expertise. [See also the discussion in D.2.1 from the standpoint of data interpretation.] This problem is handled in ABEL with the use of expectations about the possible outcomes of the questioning process, and using these expectations to check whether the outcome of the questioning (i.e., the data presented to the program) is consistent with currently held beliefs

about the patient's condition. If the data is inconsistent, an excuse finding mechanism is activated which allows ABEL to pursue the questionable findings further before accepting them.

One of the limitations with the method developed in ABEL arises because ABEL alternately generates information gathering plans and executes them. Thus, although the program can respond to unexpected findings to a limited extent, the plan once generated is essentially executed to completion before the process is repeated. However, based on the observation of clinicians, we have noticed that the plan generated by the physicians differs from those generated by ABEL in the following ways:

Physicians do not alternate between plan generation and evaluation. Rather, they seem continuously to be augmenting and modify their plans in response to incoming information. Furthermore, based on the expectations of possible outcomes to some questions, they often develop plans for each of the likely alternatives (contingency planning). Finally, while complete plans are generated by the program, physicians generate only a partial plan for questioning, requiring that they have a clear idea of only a limited future, choosing to leave the rest of the plan in partial development, so the steps are only refined if and when that part of the plan is deployed. They also consciously identify points in their plan where they would stop to re-evaluate the line of questioning, possible hypotheses, etc., based on how the questioning process corresponds to the expectations of the clinicians.

For this project, we propose to conduct formal clinical protocol analysis to get a better understanding of this "expectation driven continuous contingency planning" process [the protocol analysis process is discussed in depth in section D.5] used by the clinicians. Using the results of this analysis, we will develop analogous approaches for the use of diagnostic programs.

5.2.6 Explanation and Justification

The Clinical Decision Making Group has provided many innovative ideas in the AI community in research to give programs effective explanation capabilities. The work on explanations started with methods for generating English explanations for the program code in the Digitalis Therapy Advisor and has progressed to methods for explaining the underlying medical knowledge that justify the procedural methods (as discussed in the Preliminary Studies section). An effective explanation facility needs to be an integral part of any medical expert system for a number of reasons. First, the information or advice provided by the system must be justified for the user in the same way that an expert consultant provides not only a recommendation but, more importantly, the rational basis for making the decision. Since the physician is ultimately responsible for the case, the reasons are as important as the conclusion. Second, the knowledge in an expert system needs to be open to review both for analyzing the reasoning process in individual cases, and also for critical assessment by experts in the field. The explanation facility serves as a mechanism to make the knowledge understandable to the computer naive expert and serves as a good debugging tool for the system developers. Finally, in a complex medical domain many of the kinds of conclusions that should be provided for the user represent complexes of interaction in the medical domain. For example, an appropriate therapy recommendation in a coronary artery disease case may be:

> "A beta-blocker is the first choice, but monitor the patient for the development of bradycardia, CHF, or bronchospasm."

Such a recommendation includes both the therapy and the plan to follow to make sure the therapy will meet the expectations. Such plans in general will have to be generated in English from the internal representation of the therapy plan from the PSM. That capability requires the same technology as the explanations.

The most important characteristic of an explanation facility is that it reflect the true state of the knowledge base and the PSM. Thus, the explanations need to be generated from the structures they are purporting to explain [Swar81]. The use of a flexible knowledge representation language makes this possible. The language allows the designer to include with the knowledge structures not only the information necessary for the performance aspects of reasoning but also information to generate the appropriate English constructions as well as any other information that may prove useful for other purposes in the expert system. The best example of the power of using a suitable knowledge representation system is ABEL's explanation system. It was actually developed for the XPLAIN system, but both systems used the same language and the explanation system was transferable.

For this project we will start with the basic English generation algorithms already developed in the group, translated to operate on the target semantic representation. We will then continue the development of these techniques by incorporating the ideas of Granville for condensing and improving the English generation. There are a number important research issues in this area that need to be addressed to make the explanations more effective. One is the tailoring of explanations to the user's needs. That is, producing explanations with detail where the user wants or needs detail. This requires producing and using a model of the user's needs [Clan83]. Another area of research is the effective explanation of different kinds of processes. Humans have techniques for explaining complex structures that provide the appropriate cues and emphasize the important aspects to make the explanation understandable. It is important that we collect and analyze such techniques to determine their applicability.

Thus, the goal is to produce an English generation module that operates on the representation system and that has the flexibility to be used by a variety of expert systems to produce the kinds of expressions of structure necessary for explanations, justifications, and recommendations.

5.2.7 Causal Reasoning

Reasoning from causal relationships is important in any domain where determining the mechanisms underlying the patient's condition is useful. This is the case whenever associational mechanisms are insufficient to adequately characterize the situation. Thus, domains where there may be multiple interacting diseases, multiple relationships between diseases, or where the simple clustering of findings is inadequate to differentiate among the meaningful diagnostic and therapeutic entities are all candidates for causal reasoning. Two domains we have worked on are prime examples. The acid-base and electrolyte domain requires causal reasoning because of the interactions among the physiological parameters. It is possible that two mechanisms operating at the same time will affect a physiological parameter such as potassium in opposite directions resulting in a normal level that would be inconsistent with either mechanism considered alone. In the heart failure domain, the cardiovascular compensatory mechanisms result in a very similar patient presentation regardless of the cause for the failure. That is, when the cardiac output does not meet the demand for output the patient experiences fatigue, peripheral vasoconstriction, increased pressure and fluid accumulation in the lungs and systemic circulation — the picture of heart failure. Thus, it is necessary to look for the more subtle findings that indicate what the limiting factor or factors might be in the system and hence to trace back to the primary reasons for the heart failure.

In developing medical consultation programs for these and other domains that require reasoning from causality, we have developed mechanisms that will be useful in many domains. For example, the heart failure program makes use of a representation of causality in the logical formalism of a Truth Maintenance System (TMS) that provides automatic propagation of the implications of causal relationships. That is, a fact such as "myocardial ischemia is caused by inadequate myocar-

dial oxygen supply or excess myocardial demand" is captured by representing the implications of the causality as a logical statement. In this case, if the effect is known to be true, then one or both of the possible causes must be true. Thus, the following logical statement is in the data base:

```
myocardial ischemia => inadequate myocardial oxygen supply
                       or excess myocardial demand
```

The TMS maintains that statement by automatically asserting the logical consequences whenever any of the terms are asserted to be true, false, or unknown. In the Heart Failure program we have extended this mechanism to allow the system to reason about causal mechanisms for which there is a significant delay between cause and effect. For example, fluid accumulation in the body happens over a period of days. Thus, a cause for fluid accumulation could be present without any evidence of excess fluid if insufficient time had passed or there could be fluid accumulation without a presently existing cause. The extension of the causal reasoning mechanisms to account for such problems of time of causation involve including time bounds for the causation and time bounds for the disappearance of the effect [Long83b] (see the Appendix).

The development of ABEL has also contributed much to our understanding of the process of reasoning with causal relationships. In that domain mechanisms were developed to reason with quantitative relationships between parameters as well as a first order mechanism to reason about the gradual effects of the homeostatic mechanisms operating in that domain.

These mechanisms for handling aspects of causal reasoning are independent of the medical domains and represent inroads into the understanding and reasoning about causal mechanisms, but there are many more mechanisms that need to be explored. For example, we need to develop a taxonomy of the types of links between causal entities as well as the representation and reasoning mechanisms for supporting the distinctions. These link types should support the types of causal relations [Reig77] including continuous relations, one-shot events, threshold events, non-reversible relations and so forth. The nature of these relations also needs to be studied more formally. That is, some combination of the severity and duration of the cause or causes and the presence of predisposing or precipitating factors determines the likelihood and severity of the effects, but these mappings follow patterns and have properties that need study. There is also a hierarchy of relations among the causal relationships with each relationship representing a summary of relationships below it. To complicate matters, there is no ultimate base level of description and medical knowledge is not consistently available at any one level. Thus, there need to be links representing associations between entities or entity groupings for which precise causal relations are unknown. Also, there need to be relations whose existence is dependent on the state of the patient to capture mechanisms that are only important under certain conditions.

Given a more complete understanding of the nature of causal relationships, we also need a better understanding of the appropriate ways to use those relationships in reasoning. This includes mechanisms for highlighting the important mechanisms in the patient situation and ignoring the unimportant relations. Similarly, we need better mechanisms for separating primary causal mechanisms from secondary mechanisms and worsening factors. These issues are starting to be addressed in the Heart Failure program, but research on these issues is needed in other domains.

To provide these methods for the general framework outlined in this proposal, it will be necessary to integrate the representations of the causal nodes and links in the domain model with the consistency and implicational support to be provided by such mechanisms as the TMS. Appropriate logical relationships for the causal representation in the Heart Failure program have been developed, but each additional refinement of representation carries additional implications for the relationships among nodes. Furthermore, it will be useful to provide mechanisms for maintaining

context between the parts of the PSM and the medical reasoning mechanisms so high level processes such as reasoning about hypotheses have adequate control over the propagation of causal implication.

There are other kinds of reasoning, both about causality and about time, that will need to be developed and integrated into the system. For example, temporal aggregation is the process of taking multiple parameter values gathered over time and inferring time intervals over which the parameter has values or trends meaningful to the reasoning mechanisms. This process is important to data interpretation as well as being an important technique for identifying appropriate time intervals for causal reasoning and matching the cause-effect relations mentioned above. As the overall framework develops the addition of such mechanisms applicable in multiple parts of the reasoning process will greatly enhance the flexibility of the system.

5.2.8 Multi-level Causal Descriptions

Medical knowledge about different diseases and their pathophysiology is understood to varying degrees of detail. Our understanding of medical expert reasoning suggests that an expert physician may use several levels of reasoning within a single case when dealing with a difficult and complex case. For our program to reason at a sophisticated level of competence, it will also need to share such a range of representations. For example, in order to be effective in exploring the space of diagnostic alternatives the program must be able to describe the case in a brief and succinct way, and yet be able to reason at great detail when such reasoning is necessary to sort out the effects of various complicating factors. We have begun to address this problem by representing medical and case-specific knowledge at a number of different levels of detail. Each level of the description in this system is a network of causal relations between diseases and findings (and intermediate clinical/pathophysiologic states). Associated with each node representing the state of a disease or a finding is a set of attributes describing the severity, duration, the time of onset, etc. In addition each node is associated with a causal network of nodes and links at the next more detailed level, providing a more detailed description of the medical entity represented by that node. In an analogous manner, each causal relation at some level is described using a chain of causal relations that represent the next more detailed description of the mechanism responsible for the causal relation among the cause and the effect node. Furthermore, the description of a node or a link at some level is related to that of the next more detailed level through the use of focal links, which serve as markers that help align the causal description at some level of detail to levels adjacent to it. Furthermore, this formalism allows us to encode in a single system the clinical level experiential knowledge necessary for effective diagnostic search, and detailed knowledge of pathophysiology which is essential for dealing with multiple concomitant diseases. Finally, we have also developed reasoning strategies that can move the inferences and conclusions drawn at one level of detail to other levels in the context of the ABEL program [Pati81] (for further details see appendix).

Our intention is to generalize the multi-level operators we have developed to make them applicable in other domains. The primary advance needed to make the operators more easily applicable in other domains is the ability to define the levels according to the needs in each patient case, rather than relying on fixed levels. To add this capability will require additional operators that will plan the levels for the case by selecting from the knowledge base hierarchy the appropriate specializations of the physiological terms corresponding to the required reasoning level and connecting these together to maintain consistency.

5.2.9 Therapy Planning and Management

Therapy planning and management is at the forefront of our research and at the same time one of the least explored areas of medical expert systems. Because of the strong interactions between diagnostic reasoning and management reasoning, an integrated system, such as the one we are proposing is needed to adequately face many of the issues of management reasoning. For example, the work on the Digitalis Therapy Advisor, the first AI program to model iterative therapy management behavior, lead to our research on a program for the diagnosis and management of heart failure because of the need to track the response of the patient's heart failure to the drug.

Our experience in therapy management comes from four programs. The Digitalis Therapy Advisor loaded, adjusted, and maintained a patient on digitalis for either atrial arrhythmia or heart failure over the course of a number of sessions. Thus, it encapsulated an iterative approach to finding the patient specific dosage of a drug and made use of a mathematical model of drug behavior. The Ventricular Arrhythmia Advisor extends this model to include deciding when a drug has failed and drug effect assessment under uncertainty. ABET, a therapy program associated with ABEL, approaches the problem of symptomatic therapy in electrolyte disorders. It provides a simple mechanism for dealing with severity and urgency and shows how physiological interactions and the expectations of therapy effect can be used to determine appropriate therapy levels. Finally, the Heart Failure program integrates therapy management with the diagnostic reasoning. It uses a causal physiologic model to find and assess therapies as well as diagnose the patient. (Most of the decision analysis activity is also patient management, but we are concerned here with the more mundane issues of therapy selection, planning, evaluation, and assessment that do not require formal decision analysis.)

While it is clear that therapy planning depends to a large extent on the nature of the diagnostic representation, there are aspects of the problem that are almost independent. We have thought for a long time about the nature of general models for therapy management that would capture the standard strategies for finding the patient specific therapies and dosages. The determination of a therapy plan requires an assessment of the need for the therapy in terms of the risks of the disease state or anticipated disease state, the risks and probability of the toxic potential of the therapy, and the probability and potential for benefit from the therapy. It also requires a plan to meet that need utilizing knowledge about the properties of the therapy and the available tools for determining the level and effect of therapy. We have extracted this model from the examples above, but need to provide the mechanism in the general framework we are proposing. This general mechanism will have a number of components available for new domains including flexible and informative implementations of standard pharmacokinetic models. That is, implementations that answer questions about the likelihood of therapeutic and toxic levels, times of high and low drug levels and so forth. Also, included will be mechanisms implementing typical strategies for balancing the therapeutic and toxic risks, and mechanisms for planning adjustments to the therapy. There are also issues of long term management that have not yet been addressed, such as anticipating and testing for possible future problems by conducting drug trials, that need research in the context of overall patient management strategy.

Many of the therapy planning issues are tied to the diagnostic representation. For example, the selection of an appropriate therapy depends primarily on the causes of the patient state and the conditions determining the response to therapies in the patient. We have been addressing these problems in the Heart Failure program in ways that will be of more general applicability. To determine a candidate therapy, the causal chains leading to the effects needing treatment are examined. The therapies are selected to break those causal chains as near to the primary cause as possible. The next step is to reason about the propagation of the expected changes from the therapy

through the causal model. That is, adding the therapy will cause a change in some parameters which will in turn cause changes in others. To handle this process properly, it is necessary to deal with the addition of effects through different pathways, feedback through the homeostatic mechanisms, uncertainty of response across causal pathways, and the variety of time delays in causal relations. We have developed methods for most of these problems by modifying the results of signal flow graph analysis to allow incremental reasoning along the pathways from cause to effect, accounting for feedback loops appropriately, utilizing the typical kinds of knowledge available about relationships between pathways. These methods require the use of links between the cause and effect parameters rather than the causal relationships between parameter values used in diagnosis, but the relationships between the alternate representations are clear.

In addition to mechanisms such as these, the therapy management tools need to support reasoning about the changes occurring after therapy is applied and the implications of those changes to the diagnosis and to future therapy. This is an area that will require much more research before general mechanisms can be developed for the overall framework. Thus, even though therapy is sometimes viewed as the last piece to be added to the program, there are many aspects of the problem for which mechanisms can be generalized to provide tools for building therapy planning tools for other domains.

5.3 Patient-Specific Model

One of the major difficulties in integrating various components of medical reasoning systems has been diversity of the pertinent knowledge versus the limited scope of the representations used in these systems. This limited approach to representation of patient specific knowledge is understandable given the exploratory nature of these systems and the limited scope of their expertise. This limitation however has resulted in systems which, although they perform like an expert in one aspect of medical activity, have, unlike experts, a complete lack of understanding of other aspects of medical care. Therefore, although they can give excellent advice on an isolated problem, they cannot be expected to provide the best advice from the point of view of the patient management taken in its entirety. Thus, for example, a diagnostic program which does not have an understanding of the overall state of health of a patient is likely to treat an otherwise healthy patient with chronic urinary tract infection in the same manner as an extremely sick patient with the same complication.

To overcome this difficulty, we propose to develop an extensive representational capability for the case specific information in a Patient Specific Model (PSM), further developing the idea that has evolved through the Digitalis Therapy Advisor PSM and the ABEL PSM. The PSM we are developing will act as the central repository for everything known to the system about the patient including the raw description of the case material, the facts derived from that material, conclusions drawn by the various mechanisms that make up the program, the set of possible composite diagnoses, possible treatment plans, internal states of the reasoning program itself (goals, assumptions, plans, etc.) and whatever human commentary is available about the case and the program's performance on it. By being in the central repository, the PSM, this information will be accessible from every module in the program. Thus, the diagnostic program could examine the therapeutic implications of each diagnostic alternative before selecting a specific differentiation problem. A therapy evaluation module may look at the current set of diagnostic alternatives in evaluating the overall merit of a treatment plan. Furthermore, the PSM organizes the case such that procedures can analyze it and characterize it in whatever ways are needed to learn from it. The PSM also constitutes a complete record of the past analysis of the case for each new session.

In the previous section we have presented a number of components of medical reasoning. Each of these must operate from and contribute to the PSM. Thus, the PSM structure must be well defined in the sense that each new requirement from a newly designed reasoning mechanism must be considered an addition, rather than a change — upward compatibility analogous to that in the knowledge base (see section D.2). The PSM structure can be considered a complex of instantiations of the appropriate parts of the knowledge base. Each term in the PSM is a term in the KB specialized by the patient and the role it plays in the PSM. Thus, the various structures in the PSM are virtual copies of the corresponding parts of the KB, giving meaning to structures. Because of this relation to the KB, all of the case independent knowledge is available directly from the case specific structures and such functions as explanation cause no additional overhead. Many of the reasoning mechanisms will also make heavy use of this relationship between the PSM and the KB. One way to think about the structure is to consider the KB as supplying the medical templates for the reasoning mechanisms and the PSM as supplying the slot fillers. This also implies that the PSM and the KB are highly interrelated. As a result, the PSM must be constructed from a given KB. Since all of the information is there, new PSMs can be constructed given a new KB, but it must be recomputed.

There are a number of dimensions along which the PSM can be viewed. First, it contains the raw description of the case as supplied by the user. It may be entered by the user interacting with the program or be provided by protocol or other method of case entry. From this description it will

always be possible to recreate the case. Therefore, the description must be sufficiently complete to give the program essentially the same information about the patient as the physician would have. It's not possible to record everything because the machine is not able to see, touch, listen to, and feel the patient and the environment. It is possible to record much of the significant detail of the patient interactions (often left out of the charts) such as the times of actions and times that information becomes available (e.g. the blood was drawn at 9:00AM and test results were reported at 4:00PM). It would also include records of interactions with the user including requests for data from the user, response of the user to each query as well as other information such as explanations and possible recommendations.

Each level beyond the raw data involves some kind of processing of the case, either by the reasoning mechanisms or by humans. One kind of interpretation is that by human observers. These will include commentary on the case from expert opinion and from protocol analysis (see section D.5). The protocol analysis will include both the raw text and the analyzed text integrated into the KB in the same ways as the case description itself. There will also be commentary on the program's performance on the case. One of the important methods of determining incompleteness and inconsistencies in the KB as well as overall system performance is by expert critique. These critiques correspond to the case and can be linked to the case in such a way that decisions to change the KB will have justifications from the critiques (see section D.6).

The parts of the PSM constructed by the reasoning mechanisms correspond to several dimensions. For each session of the case, there will be complete state descriptions of the patient reflecting the program's conclusions from the patient data at the time of the session. In addition, there may also be intermediate partial descriptions of the patient state reflecting statements in the case such as "three days ago the patient was seen in the emergency room ...". Not only are the states important but the changes between sessions and significant time points are important and will be represented in the PSM. Thus, along the time dimension there will be a complete picture of the evolving case as determined from the case information. It should be noted that this does not necessarily correspond to the actual evolution of the case. In fact parts of the state descriptions at each session point will be causal reconstructions of how the disease state may have arrived at the observed point (such reconstructions are a part of the causal reasoning process discussed in section D.2.7).

Another component of the PSM is the set of disease hypotheses at each session. Each disease hypothesis is a complex of the instantiated descriptors specifying the constituents, severity, temporal relations and so forth (as discussed in section D.2.2). Each disease hypothesis is made up of the diagnosis and all of the reasoning conclusions that the various mechanisms attach to it, including data collection plans to refine the diagnosis, treatment plans, expectations for disease course and therapy results, contingency plans, findings accounted for and not accounted for, discovered inconsistencies, decision analyses, and so forth. Even with only a few diagnoses, this amount of information would be overwhelming to determine for each one. However, not all diagnoses will require all of the analysis and also there will be a great deal of overlap between diagnoses and the common information will be available to each diagnosis that shares that portion of the PSM.

The treatment plan will contain pointers back to the disease hypotheses which it is intended for, expectations about the possible response of the patient to the treatment, plans for routine adjustment to treatment based on patient response and contingency plans for dealing with rare but anticipatable reactions to the treatment.

The data gathering plan would plan the program's interaction with the patient. This plan would contain a set of questions to be asked and expected responses to these questions. It would also contain additional hooks that would allow us to locally explore input data if discrepancies are detected and finally for extending and modifying the data-gathering plan if unanticipated data are

received and the discrepancies cannot be explained away easily.

The PSM at each session point will also include a plan describing overall patient management. This plan would bring together the various diagnoses at that point and allow global decision making about various clinical activities, e.g.., *test, treat or wait* (stated in a very simplified terms). A range of alternative formalisms for representing and analyzing these patient management plans would be investigated. In emergency situations or other situations that require immediate actions, a set of data driven rules (demons) can be used to initiate an appropriate shift in the focus of the program's activity. In other situations, categorical rules may be sufficient to evaluate the alternatives and in truly uncertain situations, a full-blown decision analysis may be necessary before ordering some expensive test or initiating therapy in the face of significant uncertainty.

Finally, there is the dimension of the causal levels within the description of the diagnoses. These range from the physiological level causal description of the case as cast in the diagnosis to the clinical level description with as many levels in between as are appropriate for characterizing the relationship between the case facts and the diagnosis. Again, these parts of the PSM have a lot of overlap with the corresponding parts of other diagnoses.

The data structures for the representation of the PSM will be designed to promote as much overlap as possible among the various structures to keep the search spaces relatively small as well as to decrease the number of structures that must be build by the programs. One of the open questions is how much copying of structures is actually necessary. A second important question is how to provide mechanisms for enforcing consistency within the PSM. For example, the causal relations at the various levels should be consistent with each other. While the mechanisms may compute one level from analysis of the next level, there will also be constraints with other parts of the PSM that the builder mechanisms may not check. This problem is actually closely related to the problem of sharing structure because they both depend on the optimum use of the inherent constraints in the structure. Part of the answer will be to include in the substrate facilities for truth maintenance and dependency tracking, but the questions do not have easy answers.

Thus, the PSM is the structure that maintains the entire picture of the case. In the same way as a Problem Oriented Medical Record [Weed71], the PSM is indexed by the semantic content of the case, only with the PSM the indexing is along all of the dimensions that are available in the medical knowledge base. As a result, the PSM also serves as the structure that other mechanisms will use to extract the case characterizations for learning from the case and allowing recognition of similarities between the case and others — case based learning.

5.4 Knowledge Base for Coronary Artery Disease and Fluid and Electrolyte Balance

At the center of the system we are proposing to build is a knowledge base (KB) which will be the repository for all of the domain facts known by the system. We do not propose a KB to cover the breadth of medicine, of internal medicine, or even of some subspecialty areas like cardiology or nephrology in toto. Rather, we shall be developing a KB that will describe in depth the knowledge necessary for handling a significant subset of cases and problems in the areas of coronary artery disease and fluid and electrolyte balance. We have selected these topics because of their common occurrence in medicine, the requirements for a diverse array of reasoning processes in dealing with patients having these problems and reasoning in caring for patients with these problems and because the are foci of interest and expertise among both the physicians and computer scientists that constitute the group.

The purposes of the knowledge base are many. It will act as the source of medical knowledge for all of the programs built from the mechanisms discussed and planned in section D.2.2, including both the knowledge for performing the reasoning and the knowledge for explaining conclusions. It will act as a repository for the case material to be collected in section D.6 in both the analyzed and raw forms as well as the protocols and their analyses described in section D.5. Finally, it will support our experiments in case-based learning and knowledge acquisition from literature.

The fundamental notion of a knowledge base is a highly structured, richly indexed representation of all of the domain information available to any of the programs using the knowledge base. This project, in which there will truly be multiple programs using the KB and multiple programs entering new information into the KB, will test the concept of a meaningful flexible representation of the knowledge. The driving motivation for such a KB (and therefore a test for success) is the ability to use the knowledge in multiple ways, especially ways that were not anticipated when the knowledge was entered in the KB.

The philosophy underlying much of this project is the incremental evolution of an integrated and coherent system. This philosophy places additional constraints on the KB. First, the coverage and complexity of the representation will be evolving as the mechanisms that use the KB evolve. At the same time, the information being entered into the KB from the various sources (experts, protocols, texts, cases, etc.) will gradually be accumulating. To ensure that the KB can always be brought into a consistent state when new detail is added to the representation as well as preserving a trace of the source of knowledge, the information will always be kept in a form from which all of the content can be recovered as well as integrating the derived facts needed by the various mechanisms. Second, the refinements of the representation must be viewed as additions and therefore upwardly compatible with the previous forms of the KB. The reason is that all of the mechanisms that ran in the simpler KB must continue to run. This statement is made with the understanding that as the representations become more complex, properties of medical terms that were directly recorded as properties may become derived or inherited properties of the term and associated terms accessible through more complicated mechanisms. Thus, the process of retrieving the desired information may change as the KB develops but not its accessibility.

Form of the Knowledge Base The knowledge base is constructed from structured terms and roles with inheritance from multiple hierarchies and rules for semantics as discussed in section D.1. The basic entities in the knowledge base are terms in the representation language with connections into each of the hierarchies that provide generalizations of the terms and with properties and characterizations that provide the information needed by different modules. For example, the concept angina pectoris will be represented by a term in the KB. It is a specialization of

the pain term, with the location specialized to indicate it is typically felt in the chest and with specific temporal, severity, and descriptive properties. Pain in turn is a specialization of feeling and therefore both angina pectoris and pain inherit all of the caveats of feelings — dependence on the consciousness of the patient and so forth. Thus, angina pectoris fits into the hierarchy under pain with role restrictions on the appropriate pain properties. It also is a symptom of myocardial ischemia and therefore fills a symptom role to that physiological parameter with various properties of likelihood and specificity. In addition it will have many other properties and serve in other roles to encapsulate the available knowledge about angina pectoris. The various properties, likelihoods, and other information will also have links to the sources of the knowledge.

The KB is made up of those kinds of information needed to do medical reasoning including diseases, physiological parameters, causal links, therapies, anatomical relations, findings, descriptive statements, abstractions, probabilities, common sense information, as well as the case data, protocols, text book information and so forth. Each of these entities has appropriate properties to hold the needed information. For example, physiological parameters as KB terms have *values* to represent the specific properties of each meaningful value type, *bounds* to record the limits of the parameter, *measurements* to provide links to the findings that give evidence of the value of the parameter, *volatility* for information about how fast the parameter can change in the patient, *summarizes* to specify the set of parameters and links for which this parameter acts as a summary if this is a summary parameter, and so forth. In addition, the parameter is attached as part of the KB to indicate what kind of physiologic entity it is (pressure, flow rate, sound, etc.), what anatomic entities it is associated with, how it should be explained to the user, and so forth. Not all of these properties will be useful to all programs, but most will be useful to some.

Another kind of term is a causal link. From our experience with ABEL, Heart Failure, and other programs, causal links have at least the following properties: *cause parameter value* the parameter and value or range of values that activate the causal link, *effect parameter value* the parameter value that may be caused through this link, *cause conditions* any conditions needed to enable the causal relationship, *causation type* the nature of the causation (probabilistic, threshold, continuous, precipitating, etc.), *causation strength* the mapping of severity of the cause to severity of the effect, *time to cause* the range of time required for the cause to produce the effect, *time to normalize* the range of time for the effect to dissipate once the cause ends, *probability* the probability of the effect occurring, and *summarizes* to provide indexing to the nodes and links that this link summarizes (if any).

The most difficult information to represent, but also some of the most important, is the summary information. Often, the information available in the medical domain is information that is not at the same level of detail as the KB. For example, the prevalence of MI with two-vessel coronary artery disease is important information, but it is at a different level than the causal links between coronary restriction, reduced coronary flow, decreased myocardial supply, myocardial ischemia, and myocardial infarct. This information is a property of a summary link from the two-vessel coronary restriction to the myocardial infarct. Similarly, there will be summarizations of pathways of change among parameters to indicate known overall effects, summarizations of feedback loops to name them and provide the appropriate characterizations, summary nodes and links to suppress unneeded details (as a generalization of the multilevel descriptions used in ABEL), and nodes that represent characterizations of other nodes. The problems of generating such summary terms so they properly and completely identify the portion of the KB summarized and providing the mechanisms needed to find them in the KB when they provide needed facts for the programs will provide a significant research challenge. Since at the same time it is necessary to be able to enter all kinds of information into the KB, our approach will be to start with less restrictive representations capable of at least holding everything that needs to be entered and as we gain understanding in

how to consistently represent the details of the information, we will add those extensions to the representation and the corresponding capabilities to the knowledge acquisition routines.

Many of the kinds of information normally included in a medical text are examples of summary information. The clinical features of a disease are links from the summary node representing the node or nodes that constitute the disease to the observable ultimate effects that have a significant probability of occurring under some assumptions about the disease. The clinical course of the disease is the summary of the causal links with delays that represent significant changes in the observables caused by the disease. The prognosis is the summary of alternate clinical courses with characterizations of their outcomes. Similarly, differential diagnosis, etiology, treatment, and so forth provide summary information. Thus, each of these will undergo an evolution in representation as the KB becomes more sophisticated.

Additional information about various possible tests will be provided. This knowledge-base would describe the effectiveness of the test procedure in terms of its specificity and selectivity, its error rates, its cost in terms of pain and morbidity, possible complications, dollar cost, the time delay before the results of the test are available, clinical situations when this test is possibly relevant and situations in which it is generally recommended.

The information included for treatments would be information similar to that developed for describing diseases, in that various physiologic processes through which a treatment cures a disease or blocks the undesirable conditions resulting from diseases can be described using the anatomic and physiologic knowledge developed for describing disease processes. Similarly, the cost and effectiveness of a treatment can be described using representations developed for describing treatment. Additional information describing the cost/utility of treatment and its possible adverse side-effects and their probabilities will also be stored for the purposes of decision analysis.

There will also have to be some special kinds of KB representations developed for particular problem solving mechanisms. An example is the knowledge needed to determine the probable location of an MI from the pattern of ECG changes it produces. This is in fact a pattern recognition problem similar to recognizing a shape from intensity information but simpler. Thus the KB needs to represent the information to drive a pattern recognition algorithm, but in a form that is also usable for explanation when the user wants to know what the ST changes in V4 say about the location of the MI. Another example is the representation of decision analysis trees. Not only must the decision points and links be represented, but the characteristics of the curves (symmetry, abrupt changes, slope, etc.) and the properties of the model must be represented (see discussion in section D.7).

Scope of the Medical Coverage The KB will cover a subset of two rather narrow medical areas: fluid and electrolyte balance and coronary artery disease. In both of these areas we will start by acquiring general knowledge from text material in the area and supplement this with case material and the additional general material to cover the cases.

For the general material in fluid and electrolytes, we propose to begin by incorporating the knowledge about salt and water balance, the acid-base balance, and regulation of potassium and calcium ions. In each of these cases, the knowledge base would contain the homeostatic mechanism responsible for the regulation of these electrolytes, and pathophysiology of errors of regulation of them. We would also develop a framework for describing general medical knowledge of anatomy, physiology, specific etiologies etc. In particular, when describing anatomical knowledge, we would include knowledge of various components of kidney and their functions, knowledge of different fluid compartments such as ECF, ICF and various joint spaces etc., knowledge about pathways for fluid flow, i.e., the glomerule is connected to the tubule which is connected to the collecting duct etc., knowledge about the relative locations of each organ in relation to other body organs, e.g., cranial

cavity is above cardio-thoracic cavity. Such relational information could be used, for example, in reasoning about the presentation of pedal edema in a comatose patient. Knowing that comatose patients are unlikely to be in erect posture for any significant amount of time to allow the edematous fluid to accumulate in the legs of the patient, the program should itself suggest that the observation of pedal edema is not likely in patient even if he has severe nephrotic syndrome.

The knowledge for coronary artery disease will be similar to that discussed above. It will include chronic coronary disease, the various treatments (medical, surgical, angioplasty, and so forth), and more acute results of coronary artery such as angina, unstable angina, and myocardial infarct. We do not plan to go past the diagnosis phase of infarct management, however, the management of those cases that involve some degree of heart failure are already within the scope of the Heart Failure grant.

The knowledge of diseases, tests and treatments will be limited to only those which are relevant to the area of application namely, fluids and electrolyte and coronary artery diseases. For example, the disease knowledge in the knowledge base would be limited to those diseases that are directly related to this field and those aspects of diseases from other specialties which influence considerations in this area e.g., effects of brain injury on respiratory gases, the effects of portal hypertension in causing cardiomegaly.

Knowledge Base Acquisition The first step in building the knowledge base is to refine the framework described above, work out the details of the representation for the kinds of knowledge we know we will need and start to develop the tools needed to transform information into the required representation and tools for editing the knowledge base. This effort will require approximately the first six months of the project.

The second step is to develop the initial taxonomies of medical terms and concepts for the data base. We have already had considerable experience in this effort in ABEL and the Digitalis Therapy Advisor. We plan to use available resources in this endeavor as much as possible. For example, we would investigate the possibility of using the computer readable text of SNOMED, to develop an initial data base of medical terms. Development of such an extensive structured terminology will help us establish a common vocabulary which can be shared by all the modules in the knowledge base. It will also be useful for the process of integrating case material into the knowledge base.

We will then hand craft small portions of this knowledge base and start using it to develop knowledge editing tools which will allow moderately sophisticated residents and fellows to edit the knowledge base without the aid of their knowledge engineer counterparts. These editing tools will include explanation routines which will be used to rephrase the facts entered and explain their consequences and relations to other facts in the knowledge base. This hand crafted knowledge base will also allow us to begin our projects in medical reasoning and decision analysis using this integrated (but small) knowledge base and provide early input in the evolution of the knowledge representation.

We will continue to explore and develop knowledge acquisition tools, which will attempt to acquire knowledge in the context of a case. If for example, while running the case a clinician finds some error in the program, he will be able to examine the logic of the program's inferences and based on it be able to augment or modify the programs knowledge base (without having to know the syntax of the programming language or the knowledge representation language).

In addition to these relatively straight-forward approaches to building the knowledge base, we will explore several more visionary approaches. These include expert opinion, protocol analysis, case based reasoning, and literature. The simplest is the use of expert opinion. That is, we will enter the algorithms, rules, relationships, and reasoning mechanisms of experts with references to the expert so reasoning mechanisms that take into account alternate opinions can be developed.

One of the classic techniques for probing the knowledge base of the expert is to present specific cases in a fact by fact format and to ask the expert to introspect and reflect about his current case formulation and what he would do next. Although the knowledge derived from such activities can be used to generate rules, relations, rubrics and facts, the format in which they are provided can be quite different. Therefore, we plan to represent both the physician's protocol response and the translation of that emission into a rule in our data base. Of course, appropriate links for backtracking and stylistic analysis will be created and the facts will be segregated into a private database for the particular physician subject until the knowledge has been reviewed and deemed to be of relevance to the broader data base and sufficiently correct for release to that data base. The rules so derived will also be linked to the particular case from which they were derived because when a rule is found to malfunction it will be important to know why that rule was created so that the concepts and the particular situation it tried to address are not lost in the revision.

One of the relatively new ideas we hope to explore in this research is something we have called "case based reasoning," something akin to reasoning by analogy. As we have reflected on our own behavior and those of colleagues, we are struck with their regular reliance on the similarity of a new patient to as specific previous patient. They often anchor their reasoning about diagnosis and therapy for the new patient to the actions they took and the results achieved in this specific prior encounter. Although this could be a human foible necessitated by the weak indexing techniques available to human reasoning and perhaps it leads to errors (e.g., the classic excess reliance on the last case seen or the errors generated by using an availability heuristic [Tver74] to estimate likelihoods. On the other hand, the technique is so broadly used in expert reasoning that we must wonder whether its representation in our knowledge base will be important. Even if it is not important in antegrade reasoning (and we doubt this), it may be quite important in providing the coherent retrograde reasoning used in explanation.

The central issues for the representation of case based reasoning will be the development of a formalism for structuring the information, indexing the cases and parts of cases according to the features that make them "interesting", and the identification of the parts of the case that must be mapped for analogical reasoning. One of the purposes of the protocol analyses we shall be describing in section D.5 will be to identify how often this type of reasoning is used and how it is integrated into expert behavior.

Another important mechanism for incremental knowledge base and process development is to observe performance to a variety of specific cases (i.e., probes), to identify instances and patterns of poor performance and then to modify or add to the process or the knowledge base or both. This is much of the rationale for Section D.6 (see below) of our proposal in which we shall be building a panel of such probes, somewhat analogous to the panels of antibodies and antigens used by our immunologically oriented colleagues. We therefore need to represent both the cases themselves (see Section D.6) and the performance of the system when presented with the cases. The analysis of that performance will be done manually by the investigators, but as specific instances of poor performance lead to modifications in the knowledge base, we need to index those additions to the cases and the performances upon which they were based. This indexing will allow us to be sure that such corrections are maintained when the knowledge base is expanded and modified because we shall then know what situations must be covered. This indexing scheme may also help the system evolve in a more coherent manner. For example, if "bell-and-whistle F21" was required to provide correct performance because "fact Q25" was in the data base but then "fact Q25" was modified or removed, then "bell-and-whistle F21" might no longer be required. If the indexing scheme did not identify this situation, then there would be a tendency to add "bell-and-whistle J5367" do undo the effects of "bell-and-whistle F21" which was no longer required.

Once a system has been constructed and the major task becomes updating it as new knowledge

is acquired, the medical literature will become a major source of new knowledge. In the earliest phases of creation of a knowledge base, the expert is used to create, overview, and to interpret the literature. Soon thereafter, however, when the sheer burden of the quantity of knowledge becomes an important factor (as the domain of interest expands even slightly as a result of the case material), it will become convenient to use the literature, either short summaries in textbooks or more specific summaries in journal article abstracts. In certain situations, well chosen review articles from the literature might be a better approach to a specific area then would be textbook descriptions. Thus, we plan to use the literature, even early in this project.

We would also like to be able to read medical text directly and extract the knowledge from it. This experiment will take place in phases. First, we will identify a small amount of textual material and rely on the one-to-one collaboration between a computer scientist and a physician to code that textual material (all of it, not some of it) in the knowledge base. Next we will use a parser (such as Martin's) to process similar text and develop methods for relating the new facts to the facts in the KB. We should at least be able to determine the feasibility of developing automated knowledge extraction methodology in a medical domain where there already exists a detailed KB.

Within the narrow subdomain for which we codify very detailed knowledge, we hope to generate a standardized MEDLINE search strategy to identify relevant new articles in the literature. Those articles will be reviewed by the physicians to insure their relevance to the knowledge base and will then be either hand coded or translated into stilted pseudotext for automatic parsing. It is conceivable, if all goes extremely well, that both a parser and limited areas of the knowledge base will be sufficiently well developed to allow articles from certain on-line full text journals to be incorporated into the knowledge base.

Completeness and Consistency Given all of the different kinds of properties and relationships among terms in the KB and the variety of methods of knowledge acquisition, there is a need to determine whether the KB is complete and consistent with respect to the information needs of the various mechanisms to be used in programs. To meet these needs, we will develop tools to compute derivable facts from the KB, to check the consistency of a node with the information on its summary nodes and the nodes that it summarizes, and to check the completeness of the needed information across the medical domain. In most cases that is not as big a job as it sounds. For example, in the Heart Failure program, the causal links between nodes are only explicitly stated in the KB in one direction. A simple procedure goes over the KB before the program runs to add the missing properties to the nodes and verify that are consistent for the computational requirements of the program. Similarly in this KB, it will be the responsibility of the developers of the reasoning mechanisms to establish the criteria for the information their mechanism needs, the conditions for consistency, and the procedures to enforce those conditions from the KB structure.

Another type of consistency that is important in a KB such as this is the consistency of the facts. For example, the facts from the texts may not be consistent with derived facts from protocols or entered expert opinions. The function of the KB support is to identify those kinds of inconsistencies so they can be resolved manually (or flagged as alternate opinions with appropriate attribution). The representation language of the KB should make it possible to uncover many inconsistencies of this type by entering the facts into the data base in such a way that the procedure entering the second fact will discover the first and notice the inconsistency. Of course the limitation is the ability to determine the logical implications of facts. The second approach is simply providing the explanation facilities to generate English descriptions of the facts and relations in the KB for humans to search for inconsistencies.

Within a large complex general KB, the search for the information needed by the particular mechanisms in a consultation program may be time consuming. In addition, the desired medical

bounds of individual medical programs may be more restricted than the KB as a whole. For these reasons, we expect that the KB for particular medical programs will, in a sense, be derived from the whole KB. That is, the properties needed by the mechanisms in the program will be precomputed from the general KB and added if they are not already explicitly on the KB terms. This will speed the program while still giving it full access to the KB for such purposes as explanation. Also, the domain bounds for the program will be added as properties in the KB. For example, the program may want to consider lesser causes of something as "other causes" or some disease entity as a primary cause even though the KB knows about further causes. The ability to do this will also be useful for spinning off programs for other purposes (medical education and so forth) as described in section D.9.

5.5 Planning Patient Management in Complex Medical Problems

5.5.1 Introduction

Purpose Protocol analysis is a powerful data gathering tool that we have used to garner insights into a physician's diagnostic and therapeutic processes. In this subproject we propose to use protocol analysis in a directed fashion to gather information about how physicians plan patient management. In particular, we will investigate how physicians reason about clinical situations that evolve over time and what strategies they use to deal with the tradeoffs of competing goals in planning clinical management. Clearly, the ultimate formulation of goal-directed planning is to make the patient well, usually requiring a diagnosis. However, this global view of the physician's activities is really quite removed from the details of his activities. His reasoning and planning really occur from a more local viewpoint, where the global objectives are not always explicitly considered.

We propose to examine a small set of clinical situations in which the physician is attempting to establish a diagnosis while simultaneously providing therapy (curative or supportive), a circumstance in which the immediate goals may be in direct competition and where the priority of actions is not always clear. Even if we accept (in a decision analytic framework) a consistent set of goals (e.g., establish diagnosis in order to optimize therapy in order to provide the best outcome), in the real clinical arena the physician must often move back and forth among separate goal streams because they rarely occur in neat sequential packages. Theoretically, the physician would like to establish the diagnosis before initiating therapy, because the therapy is then more likely to be correct and efficacious (e.g., find the source of the infection and identify the organism before beginning therapy), or because the therapy may interfere with the diagnosis (e.g., it may be hard to interpret a renal biopsy if the renal disease is partially treated by steroids). But the urgency of a life-threatening situation, the practicality of patient or physician convenience, limited time and resources, or concerns about costs often push the physician toward early intervention.

Thus, planning a simultaneous diagnostic and therapeutic workup must involve a set of partially satisfied goals in which the best approach for each goal changes with time and the evolving clinical picture. Such partial satisfactions of a changing array of goals develops because the evolving health state of the patient changes the relative importance or feasibility of achieving each goal and its associated subgoals. For example, at first, diagnosis is paramount but then enough is known about the diagnosis that an emergent situation requiring therapy has been identified. At that point, diagnosis gets put on hold (but of course, the patient's response to therapy provides additional diagnostic clues) and a therapeutic plan is initiated. In addition to conflicts between diagnostic pursuits and therapy, conflicts can occur between different therapeutic goals when several disease processes coexist and similarly between different diagnostic goals when the manifestations of several disease entities are present.

One of the most interesting aspects of efficient planning in the medical domain is that it may be possible to select among different subgoal paths leading toward a common goal. In this context, one might almost view each goal to be a separate attribute of a multiattribute utility function. Furthermore, it may be the case that different, or even conflicting goals may share a common set of subgoals if certain paths (not necessarily the most efficient path for achieving the subgoals in isolation) allowed movement toward the simultaneous satisfaction of several goals. The strategy of planning a sequence of actions in such settings requires a far richer set of planning tools than found in any AI planners currently implemented in other domains. Separate goals must be maintained and the evaluation function (or other heuristic criterion) for choosing among various "next steps" must be capable of multiattribute evaluation. Furthermore, as the clinical picture evolves over time the physician planner must consider abandoning certain goals or perhaps putting them on hold because other problems become more immediate. For example, working up the etiology of

chest pain may need to be deferred if the patient becomes hypotensive and his hemodynamic status must be assessed immediately to allow the institution of life-saving therapy.

In this project, we propose to use protocol analysis to examine a small array of such complex, commonly encountered and probably acute clinical situations. Our plan here is not to evolve a computer program capable of practicing acute care medicine, but rather to use the rapid changes and urgency of such clinical situations to highlight the physician's ability to reason about an evolving process and to organize (and reorganize) conflicting goals.

We shall develop a set of programs capable of planning among conflicting objectives. Such plans will clearly involve multiattribute utility evaluation or heuristics, whether numeric or symbolic. We shall then examine the behavior of these limited planners in narrow contexts to develop a better understanding of their behavior. These planning modules will be constructed in a fashion to allow their ready interface with the integrated medical reasoning system.

Patient Management and Planning Over the last year and a half, we have been studying the reasoning processes of physicians making decisions under uncertainty. Our methods involve a formal analysis of the verbatim transcripts of physicians solving difficult patient management problems aloud [Kuip84b]. This work has allowed us to make a detailed comparison between formal decision analysis and the behavior of expert physicians in selecting among alternatives in a difficult case. In the course of our studies we have observed evidence of the planning process used by expert physicians to formulate a plan combining invasive and noninvasive testing with therapeutic measures. This planning process has characteristics well beyond the scope of current planners in the AI literature. For example, we have observed clinical experts to use contingency planning in which the later stages of plans were conditioned on specific test results, patient progress, or responses to intervention.

Another planning feature demonstrated by clinical experts is the use of embedded planning operations, in which provisions are made to reconsider a strategy at a specific time in the future, estimating that the state of knowledge at that point in time will allow the construction of a better plan. Such "deferred planning" is an essential requirement for planning medical management. For example a strategy often encountered in the care of patients with the acute onset of abdominal pain is to re-evaluate the patient's status after several hours and to perform surgery if the signs and symptoms are significantly worse. The ability to reason about changes of the state of the patient over time is essential to patient management. Without this capability, systems will be unable to capture the sense of urgency that underlies the decision to seek more invasive interventions in compromised patients.

Physicians use a variety of techniques to deal with the uncertainty about the state of the patient in planning their care. One common strategy is to construct planning actions that will cover a broad range of hypotheses. The strategy of broad spectrum antibiotic coverage for sepsis of undetermined etiology is one such strategy.

Physicians routinely select a plan or approach to a patient based on their experiences with other patients who presented in a similar fashion. While this "case based" reasoning can introduce a systematic bias when estimating prevalence or performing probabilistic reasoning, it nonetheless serves an important role in planning patient care. These experiences can serve as fragments for building specific patient management strategies.

Expert clinicians do construct sophisticated clinical management plans, many involving competing goals and plan modification over time, and we know very little about how they do it. It is this gap in our understanding that we propose to investigate.

Project Goals We are proposing an empirically based study of medical management in difficult medical problems. The project goals are:

1. To gain further insight into the methods employed by physicians in planning patient management
2. To build on these insights in developing the planning methodology, terminology and utility assessment techniques required for an artificial intelligence system to perform the complex task of planning clinical management.
3. To design a set of computer programs for planning among conflicting objectives and allow for their interface with an integrated medical reasoning system.

The two prongs of the study are:

1. Formal protocol analysis of verbatim transcripts of physicians planning the management of complex patients
2. Design and implementation of a general planning program that will use a variety of planning strategies and will be based on empirical observations from the protocol analysis and on our on-going work in utility assessment.

5.5.2 Protocol Analysis

Methodologic Foundations for Protocol Analysis We recently published a detailed review on the research methodology in clinical cognition [Kuip84b]. In that article we discuss the knowledge-based nature of expertise, the need for extremely rich experimental data to investigate the structure of knowledge, the collection of verbatim "thinking aloud" transcripts to collect such data and the need to avoid retrospective theorizing on the part of the subject. Recent theoretical work on the analysis of verbal data [Eric80] supports our approach to the study of clinical problem-solving. The rational for use of protocol analysis is discussed in detail earlier in this proposal (preliminary studies).

Case Material For this project we shall select a small array of acutely evolving clinical situations (e.g., unstable angina, possible pulmonary embolus, gastro-intestinal bleeding, acidosis, sepsis) and shall develop protocols for presentation to experienced clinicians. These protocols will emphasize changing situations, conflicting diagnostic and therapeutic goals, possibly conflicts between short term and long term goals, tradeoffs among morbidities and mortalities, and resource constraints. Physicians will be presented with these cases broken into small sections and asked to formulate plans. We shall develop a notation for expressing which goals and subgoal paths have been specified and which are active.

The two cases that we have selected (see appendix) were drawn from patients seen by members of our division at the New England Medical Center. Both are difficult but not uncommon examples that illustrate the management of acute medical problems. The first case involves the management of a patient with bacterial endocarditis complicated by cerebral emboli. The second case is a patient with unstable angina and severe cerebrovascular disease. Both cases evolve over a relatively short period of time (1 - 2 weeks) and require prompt, decisive and life saving actions. Subsequent cases will be selected to deal with specific topics in cardiovascular disease and possibly other medical domains in the later years of this project.

Through our experience in preparing cases for problem solving sessions, we will follow these general criteria in selecting cases:

- The case should be rich in history, physical findings and laboratory data.
- The case should not involve the interpretation of image data or physiologic signals but should include the report of the data when applicable.
- The case should not demand obscure information or virtuoso performance by the clinician, but rather competent application of knowledge, experience and skill.

Specific case material prerequisites for the present study are

- The cases must involve multiple conflicting goals.
- Clinical insights and management plans must evolve over time.

Other possible clinical problems to be selected include:

- Acute atrial flutter/fibrillation. Management considerations: 1) Extent of work-up 2) Need for cardioversion 3) Need for anticoagulation.
- Syncope. Management considerations: 1) Extent of work-up (e.g. electrophysiologic studies [EP and EEG] and cerebrovascular studies) 2) Whether to initiate treatment in light of the possibility of obscuring the diagnosis (e.g., anticonvulsants for seizures and antiarrhythmics for arrhythmias).
- Acute Dyspnea. Management considerations: 1) Extent of work-up (e.g. bronchoscopy and pulmonary angiography) 2) Therapeutic considerations: Anticoagulation (possibly thrombolytic agents) for possible pulmonary embolus may interfere with invasive work-up.

The Selection of Subjects Recognizing that expert performance is highly dependent on domain knowledge, we will be studying the behavior of academic subspecialists in cardiology and possibly in other subspecialty areas of internal medicine, as they solve patient management problems of coronary artery disease and other acute medical problems respectively.

Informed Consent See section E (Human Subjects) of this grant proposal.

Collection of the Data Thinking Aloud Experiment. After the subject has been given a preliminary explanation and instructions, the case is presented on a series of cards. Each card holds one paragraph from the case description (see appendix). The subject is asked to think aloud about his decision-making process after each piece of information is presented. Once presented, cards are available for the subject to refer back to. The interviewer intervenes only to prompt the subject to keep thinking aloud.

Active probing: After the subject responds to a case without active probing, the interviewer follows a prepared interview schedule. Questioning begins with gentle, undirective probing of the decision, progressing to a more directive probe with specific questions about the management plans formulated by the subject. The segments are ordered so that the directive questions are last and do not influence the response to earlier undirective probing.

The interviews are tape-recorded and the analysis is performed from verbatim transcripts. The transcriptions do not include any identification of the subject.

L651 One is:
L652 what do you think the patient has?
L653 If you think the patient has a disease
L654 which the pathologist can properly diagnose on a fragment of tissue
L655 because the pathology is classical
L656 and readily recognizable.
L657 Or you think it will be diagnosable
L658 because a microorganism will be identifiable
L659 which will have readily identifiable morphological characteristics
L660 and one would recommend a transbronchoscopic lung biopsy
L661 as the first approach.

Figure 1: A sample transcript segment. Phrases the refer to the domain objects and their attributes are underlined.

Analysis of the Data (protocols) The steps in the analysis are outlined below. Detailed examples of each stage of this process can be found in [Kuip85] and the appendix.

Step 1: Segmenting the transcript. The transcript is segmented into line sized fragments and paragraph-sized chunks which facilitate the subsequent content analysis. A one-hour interview yields approximately 1000 segmented lines. Figure 1 depicts a sample transcript excerpt.

Step 2: Characterizing the types of reasoning seen in the transcript. In this step of the analysis, the transcript is reviewed to identify the knowledge referred to or reasoning taking place in each chunk. The types of reasoning and categories of knowledge to be recognized are drawn from the wide range of knowledge representations and inference mechanisms developed in artificial intelligence research [Wins84, Nils80]. The list below is by no means complete, but characterize some of the more commonly found types of medical reasoning.

- Reasoning in a hypothetical context
- Deducing causal antecedents
- Deducing causal consequences
- Describing a dependency
- Reasoning with quantities
- Enumerating a list
- Absence of knowledge

From this analysis, we hope to determine what types of knowledge are actually used by the clinician in making his decisions, when he has abstracted away from the full detail available, when he has "compiled" his decision in advance to focus on only a few factors, what heuristics he applies to manipulate complex information, and how he reasons within hypothetical contexts.

Step 3: Referring phrase analysis

The purpose of this step in the analysis is to capture both the content of the knowledge being used by the subject and the reasoning process that is employed.

```
objects = Disease, Patient, Pathology, Microorganism, Morphology, Biopsy-Sample
Patient.Disease-hypotheses          ::= set of Diseases
Disease.Pathology                   ::= Pathology
Classical(Pathology)                ::= true — false                (1-place predicate)
Recognizable(Pathology,Biopsy-Sample) ::= true — false              (2-place predicate)
Biopsy-Sample                       ::= Tissue-Fragment — ...
Disease.Microorganism               ::= Microorganism
Microorganism.Morphology            ::= Morphology
Recognizable(Morphology)            ::= true — false                (1-place predicate)
recommend(Action1,Action2)          ::= assertion of preference between Actions

Rule1:      L653-656
   IF:      for-all D in Patient.Disease-hypotheses,
               Classical(D.Pathology)
               and
               Recognizable(D.Pathology,Tissue-Fragment)
   THEN:    recommend(TBBx,any)

Rule2:      L657-660
   IF:      for-all D in Patient.Disease-hypotheses,
               Recognizable(D.Microorganism.Morphology)
   THEN:    recommend(TBBx,any)
```

Figure 2: The particular domain objects and their attributes referred to in this section of the transcript. Included are two rules which capture the physician's reasoning about those objects (out of context).

Each line of the transcript is examined to identify the domain object being referenced. These object phrases are distinct from the wording used to refer to them. Each line of the transcript is then reviewed to identify the assertions made about each domain object. We assume that the content of these assertions constitutes at least some of the knowledge employed by the expert physician. With the transcript analyzed in terms of domain objects, their attributes and relationships, the progress of the decisions that the subject focused on are then analyzed. Figure 2 shows the objects identified by the subject in the transcript segment of figure 1, some of their associated attributes, and rules that capture the reasoning observed. The products of this analysis can then be structured in an appropriate knowledge representation.

Step 4: Developing a script of the argument structure of the subject's explanations. This part of the analysis looks at the progress of the management plan and the overall control structure of the planning process.

With reasoning characterized by types, each of the paragraph-sized chunks are examined on a sentence-by-sentence basis, addressing the following questions:

- What points are being raised?
- Why is each point raised and what claim does it support?
- How is each point justified?

Patient has unspecified infection (L001-L003)
pulmonary infiltrates
⇒ consider congestive heart failure (L004-L014)
⇒ generalize to consider other non-infectious processes (L015-L016)
if can't find non-infectious etiology,
return to problem of identity of infection (L017-L019)

Need to decide whether to do invasive test (L020-L024)
appearance of sputum unknown, but probably not helpful (L025-L030)
do wet prep of sputum before more invasive test
because occasionally it might yield helpful information (L031-L042)

Urgency of decision could be worse: (L043-L049)
patient has been stable for 48 hours;
if patient had been deteriorating, urgency would be worse.

Decision: bronchoscopy vs open lung biopsy (L050-L056, L067-L069)
percutaneous needle biopsy is not an option (L057-L059)
(risk of pneumothoracies)
transtracheal aspiration is not an option (L060-L066)
(not productive)

Figure 3: Script analysis of selected section of the transcript

- What triggered each consideration in the argument?
- What are the dangling threads of the argument?
- What underlying common knowledge does the explanation presume?
- What change is made to the evolving plan?
- What type of action, if any, are added to the plan?

The results of this analysis are a description of the types of actions that can be incorporated into a plan, the types of criticisms that an evolving plan is subjected to, the way possible plan elements and critics are triggered, and the overall control structure of the planning process. Figure 3 shows a small fragment of a script analysis taken from a protocol (not the same one used for the previous examples).

Validation of the protocol analysis We will assess the protocol analysis on two levels:

- Internal Consistency - Is there a single consistent descriptive scheme that handles everything in the extract under focus? Does the analysis as a whole cover all that is seen in the protocol?
- External Validation - Can our observations about planning patient management be formulated into the specifications for a computer program that will run and accomplish the same planning tasks?

5.5.3 Developing the Planners

We will be testing our observations about patient management expertise by constructing computer programs to perform the same planning task. Our programs will be organized within the AI planning paradigm, where the task is to specify a sequence of actions intended to achieve a goal. As we have already noted, the patient management task is considerably more complex than the usual planning domains explored by traditional AI programs, and therefore will force us to explore several issues that have not been adequately addressed by current AI planning technology.

State of the Art in Planning The greatest part of traditional AI planning research has focused on simple goal satisfaction in deterministic domains. Consequently, planners to date have illustrated approaches to these central issues, and provide a starting point for looking at some of the tougher problems that arise in patient management (as well as most other realistic planning situations).

Over the last 15 years, planners have improved over the original STRIPS [Fike71] implementation, while remaining basically within the STRIPS planning paradigm. Some of the main innovations have been hierarchic planning (ABSTRIPS [Sace74]), constraint posting (NOAH, MOLGEN [Stef81a]), along with a variety of methods for satisfying constraints and dealing with interactions among conjunctive goals (NOAH, MOLGEN, SIPE [Wilk84]). Other research has added to the body of tactics employed by state-of-the-art planners. Recently, Chapman [Chap85] has examined these nonlinear planners and has developed a simple algorithm (TWEAK) that purports to completely capture the workings of this class of planning programs. His work serves to define the realm of applicability of existing planners. The planning behaviors described in the introduction above are among those activities that are fundamentally beyond the state-of-the-art. In particular, current planners have no means to reason about partial or uncertain satisfaction of goals, much less to explicitly consider tradeoffs in choosing planning steps. Such issues clearly dominate decisions arising in the management of a sick patient, where diagnosis and the effects of therapy are uncertain, and the health state is constantly evolving. The following is the outline of a research program aimed at developing planners that can address these issues.

The Proposed Planning Tool The first step of our planning research will be to develop a tool for defining planning programs. This tool will consist of facilities for manipulating generic plan objects (operators, states, domain objects), provided in a metarule language for defining planning strategies. The planning program will be a simple metarule interpreter, operating on a knowledge base of domain-independent planning strategy rules (such as the sort described by Wilensky [Wile80] or Stefik [Stef81b]), and another knowledge base of domain-specific plan object specifications. The idea is that we should be able to implement a wide range of different planners by defining different bodies of domain-independent plan strategy rules. An early test of the tool will be to define an existing planning program (perhaps TWEAK) and try it on a simple problem domain.

The flexibility of this tool is a great asset in experimenting with different planning ideas. This flexibility can only be achieved by including powerful primitives in the metarule language, including low-level access to plan objects and parts of the planning process. Primitives of the plan representation (such as the split and join operators of NOAH) will be considered first-class plan objects. In addition, we must provide facilities for changing the representation of plan objects without requiring radical changes to plan strategy rules. Flexibility should be enhanced by our use of a high-level knowledge representation language (NIKL) and the medical knowledge base shared with the rest of the project.

Incrementally Evolving Planners Our research strategy will be to evolve planners incrementally, based on insights into planning behavior gleaned from protocol analysis. Development will be part of an iterative process, cycling between computer implementation and protocol studies. New planning mechanisms and representations will be incorporated one at a time, to isolate specific behavior issues.

The first few iterations will be geared toward the planning behaviors we have already observed in our previous analyses. The planning tool will include facilities for representing contingencies, uncertain multiattributed outcomes, embedded planning operations, and tradeoff formulations, though the form of these objects will be subject to change throughout the project (we suspect that the first pass representation will not be very specific or powerful).

Initially, uncertainty will be represented in purely qualitative terms. That is, it will be possible to state gross relations among events (e.g., A supports B, A and B are exclusive, A is irrelevant to B, A is more likely than B, A is certain), but no numeric likelihoods will be employed. While this representation will clearly not support precise probabilistic reasoning, it is quite useful for tradeoff formulation and simple dominance testing. Yeh's work [Yeh83] on partial specification of probabilistic knowledge may provide the basis for a qualitative probability reasoner.

The ability to embed an operator to perform further planning is an advance over traditional execution monitoring approaches (as described by Wilkins, for example). In execution monitoring, planners constantly make observations about the actual results of their operators and revise their plans when certain distinguished events occur, such as the negation of a precondition to some action. We will try to take advantage of the possibility that domain knowledge will indicate the specific points and/or unexpected events that would trigger plan revision (and perhaps what sorts of revisions to consider) at the time of original plan construction. Because in patient management events rarely turn out exactly as expected (indeed, we may not expect any precise course), complete execution monitoring may not be feasible. Embedded planning operators provide a valuable focusing mechanism.

To support case-based reasoning (use of experiential knowledge), the planning tool must provide access to plan fragments as first-class objects. These structures may resemble skeletal plans (MOLGEN) or Schank's scripts.

Preferences and Tradeoff Formulations One major emphasis of the planning work will be to develop a mechanism for representing and reasoning about choices in situations of multiple, competing objectives. This is an extension of our previous (and ongoing) work on URP (the Utility Reasoning Package), a program that reasons about utility-theoretic preference models [Well85a]. URP employs utility-theoretic knowledge and qualitative reasoning techniques to determine the mathematical structure of a multiattribute utility model based on qualitative assertions about an individual's preferences for the different attributes. Based on this structure, it is often possible to determine preferences for specific multiattribute outcomes, or to focus further questions to resolve the problem.

From the point of view of this project, URP is a vehicle for exploring the set of useful assertions to make about preferences and how to reason about them. Currently, URP supports assertions about the interactions of preferences for different attributes (independence conditions), and about the gross behavior of the utility for a single attribute (for example, monotonicity or risk aversion). Different combinations of these types of assertions result in a wide range of different preference model structures.

In the planner, it will be necessary to define a vocabulary (or calculus) for describing preferences for multiattributed outcomes. This vocabulary will be a subset of URP's (evolving as URP does), made up of the most useful qualitative preference properties. We will also develop a separate

knowledge base of health-preference knowledge, mapping concepts from the planning domain to concepts in URP's technical vocabulary. The benefits of this extra level of mapping are described in a recent paper [Well85b].

The planner will combine the qualitative preference and likelihood knowledge to formulate tradeoffs which arise in selecting actions (e.g., invasive testing procedures or treatments). Because this formulation step is based only on facts of likelihood and preference we are not committed to any particular resolution technique. We hope to employ powerful dominance testing procedures whenever possible; in other cases we would resort to heuristics, experiential knowledge, or decision-analytic techniques requiring more precise data. In any case, the ability for formulation alone would be a substantial advance in AI planning technology.

5.6 Panel of Specific Cases

As any decision support system develops, it is important to test incrementally any extensions of that system to insure consistency and validity. In the context of the systems we shall be developing, this incremental testing will take two forms: checking consistency within the knowledge base and validating performance against a panel of cases.

The first check for consistency relates to examining the updated knowledge base for syntactic and semantic errors. The second phase is to insure that the new knowledge is consistent with other knowledge in the knowledge base. The next phase is to examine the implications, both consequent and antecedent, of the new knowledge and insure that inconsistencies are not implied. In both of the last two phases, the identification of an inconsistency does not imply whether the new knowledge or whether the existing knowledge base is in error. In fact one might look at the addition of knowledge as a probe of the existing knowledge base.

Since each addition of knowledge is actually a refinement of the knowledge, one might view the process as introducing a refinement in a consistent manner. The first step in refining a knowledge base is to compare the assumptions of the knowledge base with the assumptions of the new knowledge. A common situation is the introduction of knowledge that uses specializations of entities in the data base. For example, in the coronary artery disease domain if the existing knowledge base does not differentiate the areas of the heart where the coronary insufficiency exists, adding new knowledge that is specific to an area will require refinement of the existing knowledge to appropriately specialize the knowledge in situations where the areas of the heart make a difference. The knowledge base, no matter how complex, makes simplifying assumptions and encodes the implications of those assumptions in ways such as the likelihood measures on associations. As refinements are introduced to the knowledge base, some of those encodings are no longer appropriate and the underlying knowledge needs to be distributed among the refined structures. But, of course, the previous level of summarization was useful when it was generated and may well be useful as a summarization for reasoning or explanation at a later time. Thus, we plan to keep both representations (appropriately linked) in our knowledge base, rather than choosing to generate such summarization *de novo* each time it is needed. Of course, detailed consistency checks will be required because then specific organization of medical hypotheses that were active when the summary was generated may not hold at a specific instance when the knowledge need be applied.

The second kind of consistency check is to examine the knowledge base or appropriate subset from perspectives that were not used in the design of the new knowledge to look for unexpected implications. Often the implications are entirely appropriate, but these views give a good indication of relative consistency between old and new knowledge. An example of this approach would be computing the lists of primary diseases (or therapies or tests or whatever is appropriate) that could or could not cause a newly introduced entity. A third approach is to design "cases" that will exercise the new knowledge. This approach is useful because it shows how the knowledge will fit into the existing case reasoning. It is not a complete test because the implementors tend to design cases that are similar to the examples that were used to develop the new knowledge in the first place.

Thus, another kind of check we shall perform after updating the knowledge base will involve reprocessing a panel of interactions with the data base. That panel will be comprised of cases submitted for diagnosis, cases submitted for therapeutic management and physician requests for explanation of program and model performance. For such incremental verification to be feasible we shall need to develop a formalism for representing this panel of cases and physician requests. The advantage of this approach is that the cases, since they are collected independent of the new knowledge development, are likely to test uses of the knowledge that were not considered.

To as great an extent as possible, we shall try to express these cases as instantiations of portions of the knowledge base. The rationale for using this type of representation is straightforward: it should minimize the effort required to interpret case specific data because those data should already have been expressed in the language and syntax of the knowledge base. In addition, the description of these specific cases should provide additional opportunity to expand the knowledge base because the need for additions to the knowledge base will surely arise as that base is instantiated with specific cases.

We shall continue our basic philosophy of using actual clinical cases—complete with all their warts and wrinkles. If we were to use generically (hyper-)simplified cases, the temptation would be all too great to make the simplifications in a manner most consistent with the existing knowledge base. Such bias in case development would tend to make the programs appear to perform better but would not foster the rapid expansion of the knowledge base.

One advantage of a knowledge representation language is that the cases can be represented in more detail than the reasoning knowledge is able to utilize and the knowledge can still operate on the generalizations, characterizations, or attributes that are needed by the knowledge. As the knowledge base is developed, the program can use the aspects of the description that are appropriate. For example, a typical chest pain description might be:

> "Chest pain, brought on by exertion, had occurred during the past 3 months, and was relieved by rest and nitroglycerin."

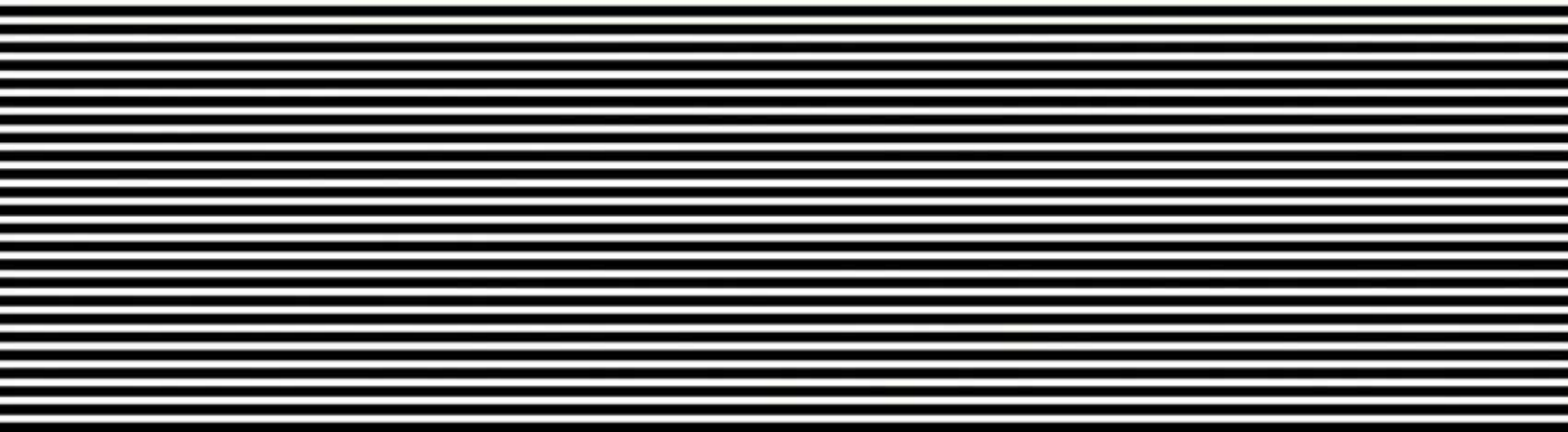
This would be represented in the knowledge representation by a form resembling the following:

```
[((chest pain) patient-1) specializes (chest pain)
  roles: initiation (typically exertion)
         time-interval (end (at-least <now>))
                       (begin <3 months before now>)
         relief (typically (or (rest therapy)(nitroglycerin therapy)))
  characterizations: ((anginal pain) history)]
```

At one level, this description is adequately captured by the characterization "history of angina" (added by some interpretation process), yet there is much more information in the description that might be needed by a reasoning strategy that utilizes the consistency of the patient's history, or by knowledge that assesses the likelihood of different causes of chest pain. As long as the whole description is represented, the various knowledge bases can use the aspects of the description that are needed.

We shall develop a pseudo-natural language interface, using the same technologies originally developed by Martin in the OWL system, and applied to medical decision making by Swartout and Long in their implementations of the digitalis therapy advisor. Our purpose in developing this parsing scheme will not be to accept arbitrary case descriptions. Rather we shall be using this procedure to encourage additions to the growing knowledge base. Thus, we shall initially adopt a rather stilted grammatical form that will rely heavily on nouns, adjectives and verbs but which will deal largely with simple declarative sentences. We shall eschew complex sentence structure, dependent clauses and the extensive use of pronouns. In fact we may even develop a stilted form of declaration. For example, we shall employ a form like

> Chest pain (CP-2) occurred after exertion (EXERT-1). Either that exertion (EXERT-1) was walking on level ground for 2 blocks or EXERT-1 was climbing stairs for one flight. Chest pain (CP-2) had duration of 10 minutes without therapy. CP-2 had duration 2 minutes after sublingual nitroglycerin.



rather than a form like

> The patient experienced exertional chest pain, provoked by walking several blocks on level ground or climbing a single flight of stairs. That pain lasted about 10 minutes unless the patient took a sublingual nitroglycerin. In that case, the pain was relieved after 2 minutes.

Note that the stilted form avoids many problems in reference ("patient experienced pain"), in semi-quantitative description ("several blocks"), in conditioning ("unless ..."), and in clause reference ("in that case"). We have had experience in using this simplified language in the heart failure project. The use of real cases will rapidly develop an expanding vocabulary and set of modifiers used to describe concepts.

We shall use the same simplified syntax when physicians request explanations about specific cases, regions of the knowledge base, or rational of inference. In this way, physicians requests for explanation will be stored in a form quite analogous to the representation of patients in the panel. This consistency should make explanation models more responsive to growth in the knowledge base and will allow the questions that physicians ask of the program to most efficiently push knowledge base expansion.

The test panel will be stored primarily in this stilted language, but that language will be used to generate machine representations of specific cases as instantiations of connected regions of the knowledge network. Each time the knowledge base is modified, the entire panel of test cases will be reprocessed and any differences in internal representation generated will be flagged for examination by the investigators.

With this uniform mechanism for case representation available, we shall go one step further. We shall ask the experienced clinicians in the research team to develop their own diagnoses, explanations, and answers to questions posed by users. These expert but biased and unblinded responses will be phrased in the same simplified natural language and will be parsed by an only slightly modified front end. The purpose of this exercise will be both to compare program output to expert behavior and, more importantly, to insure that the concepts used by experts in their own explanations are at least available in the knowledge network.

Underlying this experiment is an interesting assumption. We believe that the reasoning that experts expose when they are asked for explanations (hindsight explanation, if you will) is often quite different than the reasoning the use in prospective inference. We further believe that this alternative form of reasoning is an important part of their expertise and should be available in the knowledge base. Explanations generated by this approach will sometimes be analogous to the backward chaining explanations generated by rule based production systems because those rules of inference are often generated by experts operating in their explanation mode.

One of the more difficult problems in developing our knowledge base will be the representation of disease processes that evolve or at least change over time. We believe that by using specific patients we shall push the representation of such temporal relations. We believe that inference mechanisms sometimes drive representation, but that more often specific representations both allow and lead to certain natural mechanisms of reasoning. The language used by experienced physicians in describing patients that change over time is likely a product of their mechanisms for temporal reasoning. Thus we believe that by developing a scheme for representing specific patient descriptions with temporal disease evolution we shall be provide the substrate used by experienced physicians in their temporal reasoning.

5.7 Decision Analysis System

To this point in the application we have proposed to develop an integrated set of patient management modules, employing a framework based largely on categorical reasoning, with the ability to use probabilistic reasoning, quantitative modeling and techniques of sensitivity analysis, when appropriate. In the remainder of the application we take the inverse view of the integration of these two streams of work. We shall be dealing primarily with medical domains in which the Bayesian reasoning of the formal decision theorist has seemed the most natural approach to decision support. These domains will typically be addressing well-focused questions of patient management—topics closer to the narrow end of the funnel view of medicine proposed by Blois [Bloi80].

Although the formalism of decision theory offers a comfortable framework for reasoning in narrow domains, its successful applications have been limited to the relatively few groups that are quite experienced with the technology. Ever since the technique was proposed for application to medicine (by Lusted in the 1960s and more extensively by Gorry, *et. al*), criticism has focused on certain obvious problems—the lack of available data, problems in quantifying utilities, the possibility of over-simplification of complex problems, the computational burden of a complex technique, the interpretation of basically quantitative results in a basically qualitative world and the related issue of deciding when a result is significant. These complaints have evoked both philosophic and practical responses from the clinical decision analysis community. The data requirements are no greater than for any form of reasoning: the physician must only admit where knowledge stops and where subjective estimation begins and perform sensitivity analyses to cover her tracks. Utilities can be assessed from either physicians or patients, and formal assessments avoid certain logical limitations of informal reasoning about relative worth. Technical innovations have developed: micro-computer and mini-computer based systems for performing sensitivity analyses; bedside techniques and summary graphs of generic or common problems; new graphical techniques for expressing the results of analysis.

Our group has perhaps the broadest experience in using these techniques in individual patient settings. We strongly advocate their use and train students and physicians to apply them to medicine. However, in reflecting about our efficacy in helping the technique diffuse, we are struck by a basic problem. Formal decision theory is far more than a mechanical engine for inference. Its application is complex. It takes many months of hard work before a fledgling fellow becomes competent in the technique and before his results can be trusted. Even then fellows generally work within a relatively restricted paradigm of decision analysis (i.e. using the most convenient models, following the traditional practices of the Division that the current software supports). Why? Because decision theory is, in a way, not an inference engine but merely a technique for exploring models that conform to certain structure. Although far more general, its mathematical place is medicine may be no greater than that of specific mathematical models—calculating the pharmacokinetics of specific drugs in patients with renal insufficiency, locating the patient on an acid-base nomogram, calculating the valve area in a patient subjected to cardiac catheterization. We believe that the argument for using all such models is that they allow the physician to explore the implications of certain observations, to prognosticate about the effects of interventions or the effects of future data, or to cogitate about the implications of alternative underlying data—performing sensitivity analyses, if you will. How do models facilitate these activities? They make an analogy between the real-world and a well-understood formal system. Powerful techniques from mathematics are employed to manipulate this formal system; the results may then be transferred by the inverse analogy back to the real-world case, hopefully without too much distortion.

The real limitation in the application of decision theory to medicine is that the technique does not encompass the categorical side of medical inference and offers little except formal advice about

how useful models can be manipulated. In fact, it is a trivial task to teach a student to "fold back" or evaluate a decision tree, to calculate a Bayesian probability revision, to perform a sensitivity analysis, or even to use a seemingly complex computer program to perform these tasks. The true impediments to using this technique lie not in the formal skills we can teach but in the informal skills of model construction, revision and interpretation that we rarely teach explicitly. In many ways, we have fallen prey to the same behavior patterns for which we have so loudly criticized our mentors—rather than teaching how to structure and interpret decision models, we offer our colleagues and students an apprenticeship in which they can observe and be criticized by experienced analysts.

Reflecting on our experience in teaching these formal techniques, we are drawn to consider four basic activities in which categorical reasoning—perhaps even in which expertise and experience—are reflected: 1) the structuring of a decision problem, 2) the de-bugging or validation of a decision model, 3) the interpretation of the results of an analysis, and 4) the application of clinical judgment or common sense to ascertain that the conclusions are reasonable. In this subproject, we propose to bring the techniques of artificial intelligence that encompass categorical reasoning to bear on these activities. We believe that this system will contain components of both frame-based and rule-based reasoning and that a knowledge base will be required on two levels. At level one, basic knowledge of the medical domain will be required. This knowledge will be represented in and drawn from the same uniform knowledge base proposed earlier in this application. Level two consists of meta-level knowledge about modeling: what represents coherent tree structure and consistent modeling techniques (emphasizing such basic principles as symmetry in tree structure, the use of subtrees and the linkage of related quantitative expressions to underlying shared processes, implications of node orderings on the required probabilities and on conditional dependencies, and assumptions about parameters (probabilistic) implied by different structures), for example.

In the meta-level domain, we also need to encode knowledge of the expected behavior of certain kinds of models and knowledge of how certain observed behaviors of a model subjected to sensitivity analyses might indicate specific errors in the underlying problem structure. The first step required to represent this high level expert knowledge will be the development of a flexible representation scheme from which inferences about model structure and behavior might be drawn.

Our existing computer programs are 1) a Pascal compiled program that interprets trees (DECISION MAKER), 2) an alternative tree folding program (written in IQLISP) that produces BASIC code which is then passed to a standard BASIC compiler to generate a compiled form of the tree model, and 3) a relatively new implementation in compiled Pascal that again interprets tree structure but which includes more sophisticated graphics and modeling of Markov processes. Each of these program represents the tree model as a set of isolated nodes with various attributes specifying conditions for connecting nodes. There is no meta-level knowledge contained within the programs about how the physician should build a decision tree to represent a specific problem. Each program is a computational engine to explore the tree model and present results in graphical or tabular form.

A prototype expert system has been developed to construct decision trees on an IBM PC. The program employs a frame-based knowledge representation of disease processes, diagnostic tests, and therapeutic interventions coupled to a control structure containing heuristics for node expansion. The system constructs trees by considering all tests, treatments, and outcomes in its knowledge base, which is currently limited to a small subset of diseases. A knowledge acquisition module allows the rapid creation and archiving of any additional frames. The combinatorial explosion of tree structure is limited by rules contained in the knowledge frames and by the automatic recognition of subtrees. The program is capable of generating both classical decision trees and Markov processes. Although the system cannot assess actual numerical probabilities and utilities, it can describe the required conditional probabilities and generate multiattribute utility vectors for each outcome state. The user supplies the baseline values of these quantities and can then use the

program to evaluate the tree either numerically or symbolically. The program can then suggest to the user which sensitivity analyses might be most fruitfully examined.

5.7.1 Structuring the Model

We propose to develop a knowledge base for tree construction on two levels. The medical domain-specific knowledge (e.g., what tests and treatments are available, what are the constraints imposed on data gathering, how test or prognostic results can be interpreted, what prognoses might be expected under alternative treatment plans) will be maintained in the uniform knowledge base as frames, rules and associations. The surface representation of this data will again be in a limited natural language. During the tree creation phase, the evolving decision tree and its parameters will be displayed in graphic form. The physician will have the option of describing options (such as the duplication of sections of the evolving model as subtrees) in the same simplified language but will also be able to point to regions of the tree with a mouse.

The decision tree itself will also be represented in the knowledge base. The simplest way to represent such a tree is in terms of concepts that represent each choice, chance and outcome node, and other concepts that represent the links between them. This provides a natural network representation, so that subtrees can be simply handled. In addition, because links are also represented as concepts, additional information about the links is stated simply as additional roles of their corresponding concepts. In addition, a representation of binding contexts will be needed, to complete the representation of the computational abilities of DECISION MAKER. This does not, however, capture the useful relationships between different decision trees. What we need to develop are taxonomic organizing principles for classifying decision trees. As a result, each tree would be "close to" trees with similar properties in some meaningful sense. It is likely that there will be several organizing principles capturing the important criteria for comparison of trees and thus several hierarchies in which to locate the trees. These organizing principles might be the nature of the primary decision (fixed, time of choice, cyclic, etc.), the type of the model, and so forth. The important characteristic of the tree representation is that it simplify the problem of finding other trees needed by the Decision Analysis system for comparison, debugging, and explanation. This requirement arises even within the context of a single large tree, where it is important to find homologous subtrees.

For example, a simple interaction between a particular represented tree and other knowledge in the overall knowledge base might be:

> In a patient with angina [specifying a link into the knowledge base], consider the option of whether or not [decision node] to perform coronary arteriography [pointer to test concept]. The angina is disabling [pointing to a potential utility model in the knowledge base] and persists despite maximum medical therapy. The patient has already undergone exercise stress testing [pointing to a Bayesian and discriminant model in the knowledge base which will estimate the probability of various severities of coronary disease] and developed angina and 2mm ST depression after 2 minutes of stage II work [test result which produces a revised probability].

Note that we are not, at least initially, proposing to parse text of this nature. Rather, we want to be sure that each of these descriptors is represented in the knowledge base. This patient problem description will produce a decision tree based on a region of the knowledge base. Presumably the program will interrogate the user about relevant areas on the surface of the patient-problem description, that is, places in which there is additional potential clinical information (represented in the knowledge base) or concepts that the physician might or might not wish to consider. For

example, the program would examine the data elements that estimate the probability and severity of coronary artery disease and discover that the patient's smoking history, blood pressure patterns and serum lipid levels are all relevant—indicating that the program should perhaps generate questions about these data elements. As the program generates tree structure, the model and generated data (e.g., that the probability of coronary artery disease is 0.9) would be revealed to the physician.

The program would enforce proper habits in problem representation. For example, although it might allow the physician to model prognosis based on coronary anatomy after catheterization and based on overall data (without reference to revealed anatomy) in the "no catheterization" part of the model, it would warn the physician that this violation of symmetry would produce improper results if the clinician attempts to perform a sensitivity analysis on the probability of left main coronary artery obstruction. [To understand the rationale underlying this problem, realize that prognosis is poorer with or without surgery if anatomic lesions are more severe. Thus, increasing the probability of left main disease should worsen prognosis in both surgically and medically treated cohorts. However, if the underlying coronary anatomy is not modeled in the "no catheterization" branch, then there would be no way for the increased probability of left main disease to affect the prognosis of patients treated conservatively.] We would accomplish this by explicitly representing that the overall result in the "no cath" branch depends on anatomy, even though the precise form of the dependency has not been described. Thus, we can realize that performing a sensitivity analysis on probability of left main obstruction will violate this assumption. In general, most of our critiquing power will come from explicitly recording dependency on assumptions such as this. If the physician requested such an analysis later, the program would again provide warning and might refuse to provide the analytic results until proper tree structure was provided. Thus, as the tree is being structured and as analyses are being requested from the computational engine we envision the program critiquing [Miller83] the proposed model.

As Miller has shown, the requirements for an artificial intelligence system to provide useful criticism of a management plan is a far simpler task than *de novo* plan generation. The knowledge base necessary for such activities is more limited and can even apply a set of heuristics for evaluations because one may be relieved of the vast burden of providing a usable "legal move generator." We therefore expect to be able to provide useful criticism of decision trees suggested by human analysts or even perhaps our present rather simple tree-building program.

We presume that evolving tree structure will be represented in an internal form that will allow the program to hypothesize that certain areas of a large decision tree are actually common subtrees and bring that hypothesis to the attention of the analyst. Such identification will be a far more complex task in an evolving decision tree than the simple common subexpression analysis technique used by our present tree compiler. The knowledge base surrounding a given decision region will include not only the information needed to construct a decision tree de novo, but will also contain links to related trees constructed for prior patients and perhaps even template decision trees for common problems.

Additional issues of interest in structuring the decision space include choosing which sensitivity analyses to perform. The nature of the decision model may limit the ability to perform certain analyses and can render the results of other analyses invalid. And even in a moderately-sized model, there are too many possible parameter combinations to perform every conceivable sensitivity analysis. With a little expertise (meta-knowledge once again, or medical knowledge of what the parameters are and how confident we are about them), the program may provide advice about such selections.

Another feature of model construction is the choice of a utility structure which provides reasonable information in a balanced way. The inappropriate choice of a utility scale can bias an analysis or render important differences tiny. A knowledge-base of utility models may indicate which scales

or attributes are most feasible. It would be a simple matter to decide when to use quality adjustments, how to model life expectancy (e.g. DEALE), or when to use simpler scales. We would also plan to develop a limited knowledge base of published health status indexes from which an analyst could make reasonable selections.

5.7.2 Debugging the Model

Having introduced the concept of providing advice and critiquing a decision model as it is being built, let us now pass on to the next phase of clinical decision analysis in which categorical reasoning and expertise appear both important and poorly understood by naive analysts—the importance of debugging a decision model. First, we believe that the purpose of a decision model is not to provide an simple answer to the question "What strategy of management is best?" Rather, we believe that the purpose of a decision model is to allow the physician to explore hypotheses about what would be best under alternative assumptions, i.e., to allow the analyst to perform sensitivity analysis. However, a problem arises: 1) few physician have the experience to understand the expected behavior of a specific model under extreme circumstances or when near a threshold point; and 2) most decision models initially contain errors, i.e., they have bugs, just as most computer programs do.

Thus, an interesting problem arises. When a model is explored—pushed to its limit—and when anomalous or poorly understood behavior is observed, is that a bug in the model or an insight into the medical domain? Given our current state of understanding about clinical decision analysis, the distinction between these two possibilities is an art form and is often made based on the experience of having seen similar behavior or having constructed a similar model before. We plan to use the nature of the model and knowledge about the problem domain available in the knowledge base by looking at the nodes that are generalizations of the clinical problem and generalizations of the model. These nodes in turn will provide access to the similar models that will provide expectations about behavior.

To allow the program to reason about the behavior of a decision model we shall need to develop a structure for representing a model's performance as its inputs are varied. In a way, this representation problem should be analogous to representing the temporal evolution of a pathophysiologic process, not as a series of snapshots or a stack of plates, but as a stream of states where one reasons about the nature of the stream (i.e., slowly developing congestive heart failure versus the sudden development of "flash" pulmonary edema, where the former suggests fluid and sodium overload or a chronic process and where the latter suggest pulmonary emboli, ischemia, papillary muscle dysfunction, chordal rupture and the like). Thus, we shall need to develop a representation for both univariate and multivariate sensitivity analyses. At this point, one recognizes behavior patterns by the shape of various graphical summaries of such analyses. We shall develop a representation for describing the output of models so that performance can be classified and linked into a relevant region of the knowledge base. Perhaps qualitative sensitivity analysis descriptions might take a form such as:

1. Increases in parameter p bias towards strategy A
2. Increases in parameter p result in increases in parameter q
3. If p increases and q does not, then r will increase (multivariate)
4. Increases in p increase the difference in expected utility between A and B, but at a decreasing marginal rate

5. Changes in p, q, and r will never result in B being dominant, unless s is much higher than the baseline.

At this point it would be well to pause for a moment and reflect on the knowledge base to which we continually refer and in which we plan to represent a wide variety of diverse information. If we attempted to build such a uniform knowledge base for broad domains in medicine (for example, for the whole field of internal medicine addressed by INTERNIST or CADUCEUS), the task would be unmanageable. That is not our plan. We shall be addressing several narrow but somewhat related domains—specific aspects of cardiology and nephrology, not a global view of the entire specialty. In a way, this is a kind of depth-first representation scheme. The domains will be quite narrow and the knowledge will be represented with multiple levels of detail but there will also be extraordinary breadth in the kind of knowledge applicable to each narrow domain.

In additional to the qualitative reasoning about model performance described above, we often reason about subtle changes in quantitative performance. For example, certain parameters are often linked, modeling a dependency on a common factor or a modulated behavior ascribed to therapy, e.g., the concept of drug efficacy. We can recall many models in which a subtle bug was identified by either observing or failing to observe an expected relation, sometimes with the "funny" behavior occurring in the third significant figure and being passed off, by an inexperienced fellow, as "round off error." Sometimes such minute errors are signposts to failures of the model. For example, we constructed a model to explore prognosis in patients with a patient with cancer using the declining exponential approximation of life expectancy and found inconsistencies because the cohort that develops cancer was not receiving "credit" for its survival prior to recurrence.

5.7.3 Interpreting the Results

Model interpretation is the next large domain of clinical decision analysis in which categorical reasoning should have impact. It is not uncommon to see an inexperienced analyst describe a model purely in terms of its surface behavior, e.g., "empiric therapy is better than performing a diagnostic test and the threshold sensitivity for the test is 0.98, a value higher than can be reasonable expected." The real issue might be the slope of the line describing the relation, e.g., the strength of the relation between sensitivity and outcome or, perhaps, the fact that over the entire range of reasonable values for test sensitivity, the difference between the expected utility of empiric therapy and that of testing never exceeds 3 days—the decision is a "close call." Of course, the definition of a close call depends on the setting and perhaps on patient preference—3 days may be more salient in some situations than in others. We anticipate that our program should be able to properly identify such situations and provide enhanced understanding to the physician.

Of course, one of the greatest difficulties with using a formal model is that its output is basically quantitative and the physician basically reasons qualitatively. Thus, the analyst must be able to "understand" the performance of the model to trust its result. What is really meant by understand is to find a qualitative scenario which is compatible with the model's performance. In this regard, the problem is characteristic of expert reasoning in many other domains—the mechanisms used for antegrade inference often are sharply at variance with the concepts used for retrospective explanation, but both must be present for the expert advice to be properly understood and used. In the same way, we shall develop the ability for the expert program to support its basically quantitative conclusions with explanations that utilize basic pathophysiologic principles and associational data that will be available in the knowledge base.

Furthermore, we anticipate the availability of multi-level explanations, much like the analogy of peeling an onion. We would hope that explanations of surface behavior should be expressible in terms of hypotheses about the categorical reasons for that behavior based on well established

data or basic pathophysiology. This multi-level explanation should be in the spirit of backtracking explanation as done by Davis in the MYCIN project and by Swartout in the explainable digitalis therapy advisor. It also falls squarely in the traditions of multi-level models and inference nets begun by CASNET and extended in the ABEL program. Besides multiple levels, we will also provide explanations tailored to different aspects of the model. For example, why are certain things included or not included (medical relevance)? Why were particular structures chosen? What exactly does a certain parameter encompass? What simplifying assumptions were made along the way? Note that all of these are questions about the model construction, not the results. The explanation of results will draw on the qualitative description of sensitivity analyses described above, as well as medical interpretation of the events in the model. Although the task will be complex, we would like the program to recognize what the crucial factors in the model were, and present them to the physician with some medical interpretation.

5.7.4 Applying Common Sense Clinical Judgment

One of the major problems with most probabilistic reasoning schemes—and decision analysis is a prototypical example—is that they can produce conclusions that totally miss the mark. Such gross errors can only be detected if one steps back and looks at the analysis in the context of the total patient. Most programs and certainly almost all probabilistic programs are incapable of taking such an overview. In fact, when we teach students and physicians the applications of clinical decision analysis as a special instance of medical informatics, we take great care to assure them that the physician will always have a role and must always step back and ask "Is that reasonable?"

We propose to use our integrated knowledge base and layered program to step back a level from its decision analysis and use categorical reasoning to balance the interpretation of the model's output against an overview of the problem domain. Clearly, we shall not be able to offer a general solution to the problem of clinical judgment and common sense as it applies to providing and expert overview of an analysis, but we believe that we shall be able to encode sufficient knowledge in several specific problem domains to allow this process to proceed. Thus, we shall be focusing this work on some of the specific repetitive template analyses offered below.

5.7.5 Specific Models

In this part of our research plan we shall broaden our horizons slightly from coronary artery disease and fluid and electrolyte therapy to encompass some interesting and recurrent clinical problems in which a well developed analytic model might be tested. Obviously the management of patients with both stable and unstable coronary artery disease with respect to pharmacologic therapy and possible surgery or angioplasty is such a medical domain. We also believe that the proper timing of valve replacement surgery in patients with valvular heart disease is another cardiologic problem of potentially great interest because of the deep physiologic reasoning available both for model building and for the generation of coherent explanations.

One of the most common problems referred for clinical decision analysis is the management of patients with actual thrombo-embolic disease or at high risk for thrombo-emboli, particularly in the setting of an increased risk for standard anti-coagulant therapy, e.g., in an elderly patient or a patient with a hemorrhagic diathesis. Because such problems are both common and can addressed by a relatively uniform set of models, we hope to develop a fairly rich knowledge base for modeling in that area. Furthermore, the underlying physiology of coagulation is sufficiently well-understood to allow deep reasoning.

We shall also be developing a set of prototypical analyses for various aspects of nephrology, often

with the common thread of whether to subject a patient to a biopsy or to begin empiric therapy that carries some real risk itself. We also plan to develop a knowledge base, including template trees, for the selection of therapy for chronic renal failure (i.e., transplantation [cadaveric versus living related donor] and dialysis [center and home dialysis, peritoneal dialysis]) and the use of nephrotoxic drugs in patients with a renal allograft (i.e., whether or not to continue cyclosporine in to prevent rejection in a patient developing nephrotoxicity on that drug). We shall also be developing a model for the treatment of patients with rapidly progressive glomerulonephritis, a severe type of renal disease that most often leads to end stage renal disease, despite vigorous therapy with toxic drugs, such as immunosuppressive agents. The issue here is whether to risk that therapy, with its low likelihood of success, or whether to allow end stage renal renal disease to (safely) and inexorably develop, assuming that new therapies render that state tolerable. We shall also consider specific analyses in nephrology involving the decision of whether or not to obtain a renal arteriogram, e.g., in an elderly patient with a small kidney and moderate renal insufficiency who might have renal artery stenosis.

Finally, in the later years of this investigation once regions of the decision analytic knowledge base are well-developed, we shall attempt to develop a limited ability for a program that can use a parser and the existing knowledge base to examine a limited area of full text articles retrieved from MEDLINE by using a hand-tuned search strategy to update the knowledge base recurrently in a relatively automatic fashion. Until an adequate parser is available, such parsing might be performed by a student with little knowledge of the problem domain. The purpose of this subproject is to gain experience in the maintenance of a detailed knowledge base and the recognition and management of inconsistencies between new and existing knowledge. In general, knowledge bases developed for AIM projects have continued to expand (e.g., the INTERNIST project) but relatively little attention has been devoted to knowledge substitution as motivated by new, often quantitative, facts, e.g., the results of a clinical trial.

5.8 A Categorical Approach to Reasoning About Utilities

Perhaps the loudest objections to the application of decision analysis to medicine arise around the problem of utility assessment or patient values. Among the basic steps of clinical decision analysis, the quantification of attitudes toward alternative outcomes is the least familiar to and most challenging of physicians. The mechanics of decision theory are quantitative; decision making is approached by separating probabilities from values, combining the two, and then manipulating the result. The underlying model of reasoning is quantitative and each aspect is addressed by mathematical techniques. One might argue, however, that utilities are the embodiment of human values and that those values are basically qualitative; therein lies the conflict.

In this aspect of our research we propose to examine the necessity of and alternatives to the quantitative specification of values. We shall be relying on the integrated modular philosophy behind our evolving medical reasoning system. In particular we shall be focusing on decision analytic models using two different utility formulations: 1) the classic utility function based on a fixed mathematical model parameterized by responses to traditional lottery questions, and 2) a partially specified model based on qualitative preference assertions and orderings on hypothetical outcomes. The approach we shall develop will be equally applicable to unidimensional and multiattribute domains. We hypothesize that the qualitative utility model will have different performance characteristics and may imply either different decisions or provide different insights to the decision maker than will the usual numerical approach. We plan to apply both utility models to an array of fairly standardized decision situations (usually represented as a template decision tree) and to examine both system performance and user comfort with the result. In particular, we believe that the categorical utility models may be better able to provide explanations to physician and patient.

The optimal setting for utility assessment involves fairly immediate feedback to the utility source (e.g., the patient or the physician) based on an examination of the assessee's responses for inconsistency and based on the implications of the resulting utility structure (in our case, either a set of constraints or a parameterized model) in the relevant medical problem space. Because we would like to explore our alternatives in such an optimal setting, we shall often constrain this aspect of our research to a well-formulated decision problem that has been instantiated as a working (and debugged) decision tree and which corresponds to a well-delineated neighborhood of our knowledge base so that adequate explanations can be generated.

Prior work on developing computer programs for utility assessment rely on the assessee to specify a functional form for the utility function (e.g., multiplicative or additive decompositions (multiattribute), linear or exponential unidimensional forms) and then lead the user through a fixed pattern of questions designed to determine the relevant parameters of the utility function [Schl71, Keen76]. These programs calculate the parameters of the utility function based on users' responses to the lottery questions. They typically provide a fixed set of functional forms, with associated algorithms for generating questions to elicit the parameters. Some programs (e.g. Keeney and Sicherman's MUFCAP [Keen76] for multiattribute assessment) also provide for some structuring by the user and verification procedures to validate the underlying assumptions. However, these programs are fundamentally inflexible, in that they cannot accommodate data that are not directly part of the elicitation algorithm. For example, they usually ask for lotteries bounded by extreme values, since this reduces the number of questions necessary to completely specify the utility function. These programs also rely on certainly equivalents, rather than arbitrary inequalities between lotteries.

Although such programs have proved useful to some decision sciences consultants, placing the burden of model selection on the user is particularly serious for those who are not expert in utility theory (e.g. physicians or patients). It is unlikely that such programs could be used routinely

in medicine, where neither physician nor patient would have the background to choose a relevant utility function nor to understand inconsistencies that might arise. Most medical applications have taken a far more simplistic view of utility assessment—using either simple lotteries, categorical scales, or time tradeoffs. There has been little work addressed to providing feedback during utility assessment.

We have developed a fairly standard utility assessment instrument that employs a set of flip cards to guide a trained interviewer or physician through an assessment task. A limited number of gross inconsistencies can be identified but there is no mechanism for iteratively refining the specified utility function. We have successfully applied this technique to two cohorts of patients in a randomized prospective drug trial and have been able to identify changes in "quality of life" with greater sensitivity than certain general health status measures. We have also applied modifications of this instrument (to appropriately adapt the underlying scenarios) to patients with angina who were candidates for coronary bypass surgery, to patients with chronic renal failure who were candidates for transplantation, and most recently to patients with breast masses who may be candidates for alternative treatments for breast cancer. Those studies show the feasibility of assessing utilities and have identified consistent discrepancies among different assessment techniques, but have not been integrated into a decision support system.

The assessment of utilities depends on the presentation of a set of oversimplified choices from which preferences curves can be derived. It might be more reasonable, however, to ask the user a set of general questions (e.g., is more always better [i.e., is the function monotonic]) to help select an array of potential functions. The same set of tradeoff questions should then be used, to whatever extent is possible, to constrain the parameters of these alternative models of preference. The implications of these models will then be provided to the user both in terms of hypothetical oversimplified choices and in terms of choices in the decision problem being addressed. We shall perform some of these evaluations with patients using either a microcomputer or a lisp machine with appropriate graphics to provide relevant feedback. It will be difficult to impossible to incorporate all these ideas below in a microcomputer implementation. It makes more sense to use a lisp machine to implement the overall framework, transferring a subset of the results to the microcomputer environment. For this reason we may well need two lisp machines at the NEMC, but this hospital has promised to help fund one of them from contributions to a research development campaign.

5.8.1 AI Techniques For Reasoning About Utility Functions

We shall develop a utility reasoning package that will provide for more flexible utility assessment. The following section outlines some of our approaches, and their implications for a utility assessment tool. Note that while the "assessment tool" will be designed for a multiattribute framework, its methods apply directly to single-attribute utilities.

The central object of our assessment procedure is a preference model. In its most general form, a preference model is an arbitrary collection of assertions about an individual's preferences. These assertions can be qualitative preference properties (e.g., independence or limited dependence of preference for different attributes, qualitative behavior of single-attribute functions—monotonic, risk averse), or hypothetical preference choices. Preference choices (certainty equivalents of lotteries, for example) are usually the only form of input to utility assessment tools, and current tools typically can accept only restricted sequences of choice data. The ultimate goal for our program is to be able to make maximal use of any collection of such data but to allow other less constrained forms of preference assertion. Consistent with developments in other parts of this application, the qualitative utility assertions (e.g., "I'd rather not be dead;" "Life the way I am now isn't worth living.") and quantitative comparisons (e.g., "Six years of life with angina are scarcely better than three years the

way I used to be") will be expressed in a stilted language that will be parsed (initially by hand) into our qualitative preference representation. For example, notions about preference for life years over various health states will be represented by assertions about conditional utility functions. We might say in this case that the utility for life years under good health is monotonically increasing, while it is constant or even decreasing in certain morbid states. Note that our preference representation forms just one part of the overall patient specific model described elsewhere in this proposal.

5.8.2 Incompletely Specified Utility Functions

The reason we are willing to view preference models as arbitrary collections of assertions is that we will not require utility functions to be completely specified. Removing this restriction adds a great degree of flexibility to the assessment system. If we are able to derive sufficient constraint (perhaps to determine a decision for a particular case) before specification is complete, then we have escaped with significantly less assessment effort. We may feel more confident in the result, because it depends on fewer and weaker assumptions. These weaker assumptions may represent the use of mathematically complex models for which a complete assessment would be intractable.

Even under the most simplifying of assumptions, the task of assessing a utility function in complete detail can be tedious, painful, and subject to serious cognitive biases. Noting that completeness is generally not necessary for decisions, Winkler suggests that the development of utility assessment aids that use less than complete information is one of the most promising research topics [Wink82] in utility analysis.

Given a probabilistic model of the decision (represented as a decision tree), we can transform the incomplete preference model into a symbolic expression of expected utility for each strategy. The model's parameters may be described by an arbitrary constraint network. If instead of a typical probability tree we employ a categorical representation of the decision problem (described elsewhere in the proposal), our expected utilities are derived from a combination of the two constraint representations.

Detecting cases where incomplete assessment is sufficient is an interesting and difficult task in itself. Symbolic and qualitative reasoning techniques developed by AI researchers fit in well with our framework, and provide a wealth of mechanisms for establishing dominance in such partially specified domains. In particular, we expect to make substantial use of the qualitative mathematics package being developed by Sacks [Sack84] for analysis of symbolic representations of expected utility. We have already been using this system for analysis of qualitative properties (direction, risk aversion, higher order risk properties) of single-attribute functional forms.

5.8.3 Automatic Maintenance of Assumptions and Structural Sensitivity Analysis

All assumptions underlying the structure of the utility model will be explicitly represented in the assessment tool. A truth maintenance system [McAl82] will handle modular assertion and retraction of these assumptions, constructing new mathematical structures as the premises are changed. Since these assumptions are fairly fine-grained, the range of combinations of assumptions represent a wide continuum of possible mathematical structures, and should vary smoothly in complexity.

Hypothetical preference choices will also be treated as assertions. This contrasts with traditional assessment tools, where the input data is immediately interpreted with respect to the particular mathematical model employed for the utility function. Since our system will change model structures dynamically, we need to represent the data in a model-independent fashion.

Employing these mechanisms, we can expect our assessment tool to be capable of structural sensitivity analysis, whereby we can test the robustness of the result to changes in the model's form.

This is often a far more satisfying test than traditional parameter-tweaking sensitivity analysis. Farquhar [Farq83] stresses the value of structural sensitivity analysis, and laments the inadequacies of current facilities.

5.8.4 Formalization of Utility-Theoretic Knowledge

A knowledge-based approach to utility assessment has the additional benefits accruing from explicit encoding of knowledge of utility theory. The multiattribute decomposition knowledge base already compiled encompasses a broad range of the important results developed by utility theorists over the years, and the existing theorem language should support incorporation of many others. Collection of these results into a uniform (executable) representation should provide an interesting opportunity to compare and combine different decomposition approaches in a wide variety of problems.

Yet another aspect of utility theory that will be a candidate for formalization is stochastic dominance [Whit78, Bawa82]. Researchers (particularly in finance) have developed a set of algorithms that test for the dominance of alternatives under progressively more restrictive assumptions about the utility function. For example, first-order dominance is appropriate for any monotonic utility function; a much more powerful algorithm may be used when decreasing absolute risk aversion may be assumed. Given our flexible framework for representing these sorts of assumptions about utility structure, a knowledge base associating the algorithms with the qualitative properties would be quite useful.

5.8.5 Using and Explaining Categorical Utility Models

Having developed the alternative representations for expressed preferences, we shall develop mechanisms for evaluating the "expected utility" of such models. Because any formal tree structure can be reduced to normal form by moving the embedded decision nodes "up front," we can translate each strategy into a probability distribution over the outcomes on the leaf nodes. Combined with our incompletely specified utility function, we get a representation for expected utility consisting of a probability weighted set of constraint models. A straightforward generalization of this would be to allow incompletely specified probabilistic models, resulting in the same form of expected utility representation. Comparing expected utilities described by constraints on symbolic expressions will require symbolic and qualitative reasoning techniques, including those provided by Sacks' QMR [Sack84].

Recall that two of the major motivations for developing a decision model are to allow exploration (i.e., a set of "what if" questions) and to provide insight into the problem structure. Because coherent explanations are more likely to be qualitative, it may well be that a constraint based utility model may have more explanatory power. The qualitative representation maintains a greater amount of problem structure, since it does not reduce all preferences to a flat quantitative scale.

Decision-analytic models which do not maintain ties between quantitative constructs and their qualitative justifications are notoriously opaque. Since our preference models will be based on qualitative assertions and we explicitly represent the utility-theoretic knowledge that determines the mathematical structures, these ties are a fundamental part of our system. Furthermore, partial specification may allow us to eliminate some of the dependence on numerical input that contributes to difficulties with explanation. Our utility model will be a part of a uniformly evolving knowledge base so that explanations will be couched in terms of the specific problem domain under consideration.

5.8.6 Interpretation Based on Descriptive Models of Preference Choice

Psychologists and economists have identified numerous biases in preference choice which may lead to violations of utility theory. This has disturbing consequences for the validity of utility assessment and has been sufficient argument for many to abandon formal decision analysis altogether. The framework we are proposing, however, may provide an opportunity to exploit these psychological results in an assessment tool.

Since we will be using partially specified functions and model-independent representations of preference choices, we should be able to de-couple our interpretation of an individual's stated choices and our prescriptive model of decision making. In other words, our system may employ any psychological theory of how the subject makes preference choices in assessment, while still using the data to generate a classical expected utility function.

As a special case, we could implement the usual assumption that the assessor is maximizing expected utility. The other cases are more interesting, however. For example, suppose we assume that the assessor is making choices according to prospect theory [Kahn79]. In that case we would adjust the lottery according to the editing procedures described by Kahneman and Tversky, and fit the result to their model (in this case a formula with value functions and an additional parameter to transform the probabilities). Since prospect theory has more parameters than utility theory, the conclusions we can draw are strictly weaker (although they do assume risk properties which may give us additional constraint). Still, these results may still go a long way in constraining the utility function in assessment.

Naturally, we can only use descriptive theories that we can relate to utility in some way. Though this criterion rules out some possibilities, there are still quite a few candidates in the literature, such as Bell's regret theory [Bell82]. An assessment tool that could switch between alternate descriptive theories of preference choice would be a valuable tool for psychological resource, and may provide a novel form of sensitivity analysis for decision making.

5.9 Dissemination of Modeling Techniques into Practice and Education

In prior sections of this application we proposed to develop an integrated artificial intelligence management system which includes the application of decision-analytic and other probabilistic techniques and to bring categorical reasoning techniques from artificial intelligence to bear on yet unsolved problems in clinical decision analysis. In this final section we propose to extend existing and implement new personal computer programs for building and analyzing decision models and for communicating the results of such analyses to experienced clinicians. Our group has spent the past several years developing and extending a decision analysis program (DECISION MAKER) which took its first form in Fortran on a PDP11/03. Our natural requirements for recursive procedures and easy access to a variety of personal computers led to an implementation in Pascal which has been available on the IBM, Apple, and Heath/Zenith computer line. Because of the initially small address space on these machines (64K segments), we used an interpreted Pascal (UCSD P-code). Several attempts to generate an overlayed machine code version, for increased speed, failed in the small memory environment. The standard IBM and Microsoft Pascals available in IBM's 8088 environment did not allow the user access to the full 640K of address space in any convenient manner. This problem was alleviated with the appearance of Turbo Pascal, which has now become almost an accepted standard. We were able to re-implement DECISION MAKER in that environment with approximately a ten-fold gain in speed.

Over the past year our group has developed a second very similar tree folding program (SML-TREE) which relies heavily on the screen characteristics of the IBM PC's color-graphics card and which provides a more standardized tree representation for input and display. In that program, we have also used windowing techniques to allow the analyst to capture a snapshot of the decision tree folded back to various levels.

Beside the technical provision of convenient tree evaluation, our research in this area has produced certain basic extensions to the decision-analytic formalism which allow it to model more complex processes. We can discuss these developments under four main topics: subtrees, Boolean nodes, Markov cycle trees and dual utility representations.

A subtree has the same role in representing decision models as does the subroutine in representing computational processes. It is a region of common tree structure which can be linked to (i.e., can be called by a variety of parent nodes). During tree evaluations after a subtree is called, it returns a value which is then passed to the calling node. For a subtree to return a call-specific value, there must be some mechanism for specifying its arguments. In our system, a subtree does not have arguments explicitly (that is, all its variables are global), but it is possible to specialize the global environment to produce a local one. Such local environments are created by adding arguments to a LIFO binding stack of variables and values. Thus, the specification of local bindings prior to linking to the subtree is quite analogous to the use of local variables to create an environment in a subroutine. The use of subtree notation has three main advantages: 1) it emphasizes areas of symmetry among regions of the tree, thus minimizing the chance of leaving out considerations in one instance while including them in another; 2) it allows fairly complex trees to be represented in a smaller number of nodes, roughly decreasing tree size by a log order of magnitude; and 3) it emphasizes links among parameter values so that the models are more physiologically reasonable.

A Boolean node is a control structure which allows testing to occur during tree evaluation. Essentially, it permits the analyst to create dynamically changing tree structure. Trees that use such structure can be used to model recursive decisions (i.e., try one therapy first and if it fails move on to choosing another) in very compact form. It also allows the analyst to create a unified model that can be evaluated in different levels of detail or with different structural assumptions.

Markov cycle trees are used to model Markov processes within a classic decision tree. A cycle

tree may be viewed as a special kind of terminal node—one in which the utility is calculated by a process rather than by an equation or a variable look up. The Markov node contains arbitrary internal structure which is in essence a probability tree leading from each potential state defined in the process to all its possible next states after single transitions. In such a probability tree, the terminal position (or state node) simply indicates where members of the cohort following a particular path in a particular cycle should be collected. The process runs until a termination criterion is met—either an arbitrary number of cycles, a minimum cohort size is reached or until the accumulating utility fails to change. Because the probabilities on the chance nodes of the probability tree and the incremental utilities assigned to each state are specified by mathematical expression, they can vary with the environment in which the Markov process is evaluated and with the clock cycle of the Markov process. Special expressions can be used to describe the tail effects of the processes. Because these tail effects can refer to other structures in an arbitrary tree, Markov processes can be patched into arbitrary places within a decision tree. It is even possible to structure one Markov process to run within another Markov process. Although such models impose a heavy computational burden, they are occasionally necessary. For example, if one were using a Markov process to model prognosis in a cohort of patients who received a prosthetic device, if the failure rates of that valve depend on how long it has been implanted, and if secondary and tertiary replacements are possible, then embedded Markov processes may be needed to accurately model prognosis.

One of the major problems in decision analysis is the commensuration of basically incommensurable quantities. In policy analyses, this recognition has led to cost-effectiveness analysis, in which dollars and health outcomes are kept separately. Of course such separation of attributes need not be limited to dollars. We have structured a number of problems where health effects are best kept separate. A recent one that comes to mind is the case of a pregnant woman who presents early in her third trimester with leaking amniotic fluid. Management strategies that tend to delay delivery will increase the probability of a live born child but will also increase the chance of maternal morbidity secondary to infection. We found it useful to express those results in terms of mothers lost per additional healthy baby gained. In an analysis of the management of patients with possible temporal arteritis, Chang and Fineberg found it useful to express their analytic results as steroid side effects encountered per case of blindness averted. Certain problems arise, however, in analyzing complex decision trees with separate utility structures. If trees have embedded decision points or logical control points (Boolean nodes) that are evaluated on the fly, the analyst must be certain that all the switches or choices point the same way during both evaluations. Furthermore, the calculation of all probabilities and control values is duplicated. We have begun to develop a different approach to this problem, initially with dual utility structures, but eventually with n-dimensional structures. This model proposes that a utility is no longer a value that is multiplied by a set of path probabilities during tree evaluation, but rather that it is a vector of values with each attribute being separately maintained.

We hope to use this approach to help physicians begin to address the issue of constrained resources in their decision making processes. The technology of applying such multiattributed analyses (of which cost-effectiveness analysis is but one) to the individual patient have only been considered in the broadest terms, but it is clear that physicians will need to fold such considerations into their clinical logic. At the NEMC we have the advantage of a powerful management information system (to which members of the Division of Clinical Decision Making have access) which can provide very good estimates of at least the near-term resource costs of alternative therapeutic plans. We hope to begin to use that database to develop an approach to patient management in settings of constrained resources.

Unfortunately, some of these enhancements have evolved in separate implementations of the

program, partially because increasing complexity has limited program size. We believe, however, that larger memory sizes are becoming sufficiently commonplace in personal computing that the time has come to combine many of these features into a single, user-friendly program that provides good graphics and can use the newer mouse-technologies, when appropriate, for tree input and editing.

Although its use would be limited to physicians with experience in model building or with access to a library of template analyses (which we shall be developing), we view this revision of our existing software as the development of a productivity tool. We hope to develop a format of communication that will allow the analysts to move back and forth between the tree analysis program, a word-processor, a standard database (in which the physician might keep references to and facts summarized from the literature), a spreadsheet program and possibly a graphics package. Obviously, we are not proposing to develop an integrated package such as Symphony, Framework or Encore. We merely hope to develop a decision analytic module that might interact successfully with one or more of those packages, or which will take the philosophy of Sidekick as an overlay that could be called for calculation while the other software was active.

We expect that the artificial intelligence based systems developed in other aspects of this proposal will not find immediate application in practice because it will be several years before machines with excellent Lisp environments or even Lisp machines themselves will be widely available in medical environments. Thus, we believe that there will be an ongoing need to identify pieces of the integrated system that might be trimmed into stand-alone programs that could function in the present microcomputer environment, be it the 8088/8086 line (IBM and its clones) or be it the 68000 line (Apple MacIntosh/Lisa) because these lines will soon become the de facto standard in medical education.

Furthermore, we believe that there will be a growing and almost immediate need for good educational software for medical students in the very near future. We believe that such educational uses will in the short run be based on micro-computer/personal workstation technology and that there will soon develop a community of medical schools with that capacity. Even if the schools themselves do not force their students in that direction, we believe that medical students will arise from college environments where owning a person computer is as commonplace and as necessary as owning a typewriter. These medical students and schools will be looking for relevant software.

As aspects of the knowledge base evolve, we believe that it will be possible to spin off a variety of programs for such a microcomputer environment. Those programs might help communicate the facts in the knowledge base, might teach the student certain processes or approaches to clinical problems, or might even be an interesting array of case material with expert commentary about reasoning. The process of spinning off the relevant sections of the primary knowledge base will be similar to the process of generating derived facts for knowledge base consistency. Therefore, the same kinds of knowledge base tools should make this process relatively painless [see section D.4].

The Tufts University School of Medicine will be moving into the area of medical informatics and education about information sciences for all our students over the next several years. The investigators involved in this application will be playing a major role in that educational program. Thus, we shall have the opportunity to place the educational scientists who will be developing that curriculum in contact with the computer scientists and clinical researchers developing this knowledge base. We believe that the timing of that effort will dovetail nicely with the completion of the improved integrated version of our microcomputer system and that the availability of an avenue for such collaboration and communication will help ensure the diffusion of our research into medical education.

6 Literature Cited

[Asbe84] Asbell, I. J. "A Constraint Representation and Explanation Facility for Renal Physiology." *MIT Lab for Computer Science, Technical Report 318*, 1984.

[Bawa82] Bawa, V. S. "Stochastic dominance: A research bibliography." *Management Science* 28:698-712, 1982.

[Bell82] Bell, D. E. "Regret in decision making under uncertainty." *Operations Research* 30:961-981, 1982.

[Beta71] Betaque, N. E., Gorry, G. A. "Automating Judgmental Decision Making for a Serious Medical Problem." *Management Science* 17:B-421, 1971.

[Blei72] Bleich, H. L. "Computer-Based Consultation: Electrolyte and Acid-Base Disorders." *AJM* 53:285, 1972.

[Blois80] Blois, M. S. "Clinical Judgment and Computers." NEJM 303:4 192-197, 1980.

[Brom83] Bromley, H., Patil, R. S. and Widman, L. E. "An Approach to Therapy Formulation for Acid-Base and Electrolyte Disorders." *A paper submitted to AAAI-83 in the topic area of Expert Systems* 1983.

[Chap85] Chapman, D. "Nonlinear planning: A rigorous reconstruction." submitted to IJCAI-85, 1985.

[Clan83] Clancey, W. J. "The epistemology of a rule-based expert system: A framework for explanation." *Artificial Intelligence* 20:215-251, 1983.

[Elst78] Elstein, A. S., Shulman, L. A., and Sprafka, S. A. "Medical Problem Solving: An Analysis of Clinical Reasoning." *Harvard University Press* Cambridge, Mass., 1978.

[Eric80] Ericsson, K. A. and Simon, H. A. "Verbal reports as data." *Psychological Review* 87:215, 1980.

[Farq83] Farquhar, P. H. "Research directions in multiattribute utility analysis." in Hansen, ed., *Essays and Surveys on Multiple Criteria Decision Making*, 1983.

[Feig77] Feigenbaum, E. A., "The art of artificial intelligence: Themes and case studies of knowledge engineering." IJCAI 5 1014-1029, 1977.

[Fike71] Fikes, R. E. and Nilsson, N. J. "STRIPS: A new approach to the application of theorem proving to problem solving." *Artificial Intelligence* 2:189-208, 1971.

[Gorr67] Gorry, G. A. "A System for Computer-Aided Diagnosis." *Project MAC, Massachusetts Institute of Technology* Technical Report TR-44, 1967.

[Gorr73] 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." *AJM* 55:473, 1973 .

[Gorr78] Gorry, G. A., Silverman, H., and Pauker, S. G. "Capturing Clinical Expertise: A Computer Program that Considers Clinical Responses to Digitalis." *AJM* 64:452, 1978.

[Gran82] Granville, R. A. "An Introduction to NLP." 1982.

[**Hayes79**] Hayes, P. J. "The naive physics manifesto." in Expert Systems in the Micro-Electronics Age, (ed) D. Michie, Edinburgh University press, May 1979.

[**Kahn79**] Kahneman, D., and Tversky, A. "Prospect theory: An analysis of decision under risk." *Econometrica* 47:263-291, 1979.

[**Kass78**] Kassirer, J. P., and Gorry, G. A. "Clinical Problem Solving: A Behavioral Analysis." *Annals of Int Med* 89:245, 1978.

[**Kass82**] Kassirer, J. P., Kuipers, B. J., and Gorry, G. A. "Toward a Theory of Clinical Expertise." *AJM* 73:251, 1982.

[**Keen76**] Keeney. R. L. and Sicherman, A. "Assessing and analyzing preferences concerning multiple objectives: An interactive computer program." *Behavioral Science* 21:173-182, 1976.

[**Kuip84a**] Kuipers, B. J. and Kassirer, J. P. "Causal reasoning in medicine: analysis of a protocol." *Cognitive Science* 8:363, 1984.

[**Kuip84b**] Kuipers, B. J. "Common-sense reasoning about causality: deriving behavior from structure." *Artificial Intelligence* 24:169-204, 1984.

[**Kuip85**] Kuipers, B. J., Kassirer, J. P., and Moskowitz, A. "Protocol extract: Critical decision node." Medical expertise project memo 15, 1985.

[**Long80**] Long, W. "Criteria for Computer Generated Therapy Advice in a Clinical Domain." *IEEE Proceedings of Computers in Cardiology Conference* Williamsburg, Virginia, 1980.

[**Long82**] Long, W. J., Naimi, S. and Criscitiello, M. G. *A Knowledge Representation for Reasoning about the Management of Heart Failure* in Proceedings of the Computers in Cardiology Conference, Seattle, WA, October 12-15, 1982.

[**Long83**] Long, W. J., Russ, T. A., Locke, W. B. "Reasoning from Multiple Information Sources in Arrhythmia Management." IEEE Frontiers of Engineering and Computers in Health Care 1983.

[**Long83a**] Long, W. J., and Russ, T. A. "A Control Structure for Time Dependent Reasoning," International Joint Conference on Artificial Intelligence 1983, August 1983.

[**Long83b**] Long, W. J. "Reasoning About State from Causation and Time in a Medical Domain," American Association for Artificial Intelligence 1983 Conference, August 1983.

[**Long84**] Long, W. J., Naimi, S., Criscitiello M. G., Pauker, S. G., and Szolovits P. "An Aid to Physiological Reasoning in the Management of Cardiovascular Disease." 1984 Computers in Cardiology Conference, 1984.

[**McAl82**] McAllester, D. A. "The reasoning utility package user's manual." MIT AI Lab memo 667, 1982.

[**Miller83**] Miller, P. L. "Medical Plan-Analysis by Computer." MEDINFO-83 593-596, 1983.

[**Mose84**] Moser, M. G. "An overview of NIKL, the new implementation of KL-ONE." Bolt Beranek and Newman Inc. Technical Report No. 5421, 1984.

[**Nils80**] Nilsson, N. J. "Principles of artificial intelligence." Tioga Press, 1980.

[**Pati81**] Patil, R. S. "Causal Representation of Patient Illness for Electrolyte and Acid-Base Diagnosis." *MIT-LCS Technical Report 267* 1981.

[**Pati82**] Patil, R. S., Szolovits, P., and Schwartz, W. B. "Information Acquisition in Diagnosis." *Proceedings of AAAI-82* pp 345–348, 1982.

[**Pauk76**] Pauker, S. G., Gorry, G. A., Kassirer, J. P., and Schwartz, W. B. "Toward the Simulation of Clinical Cognition: Taking a Present Illness by Computer." *AJM* 60:981, 1976.

[**Pauk77**] Pauker, S. G. and Szolovits, P. "Analyzing and Simulating Taking the History of the Present Illness: Context Formation." in *Computational Linguistics in Medicine* (Schneider/Sagvall Hein; eds.), North-Holland Publishing Company, pp 109-118, 1977.

[**Popl82**] Pople, H. E., Jr. "Heuristic Methods for Imposing Structure on Ill-Structured Problems: The Structuring of Medical Diagnostics," in P. Szolovits, (Ed.) *Artificial Intelligence in Medicine* Westview Press, 1981.

[**Rieg77**] Rieger, C., and Grinberg, M. "The declarative representation and procedural simulation of causality in physical mechanisms." IJCAI 5, 1977.

[**Rubi75**] Rubin, A. D. "Hypothesis Formation and Evaluation in Medical Diagnosis." *MIT-AI Technical Report AI-TR-316* 1975.

[**Ruth81**] Rutherford, C. J., Davies, B., Barnett, A. I. and Desforges, J. F. "A computer system for decision analysis in Hodgkins Disease." *MIT-LCS Technical Report LCS/TR-271* 1981.

[**Sace74**] Sacerdoti, E. D. "Planning in a hierarchy of abstraction spaces." *Artificial Intelligence* 5:115-135, 1974.

[**Sack84**] Sacks, E. "Qualitative Mathematical Reasoning." master's thesis, MIT, 1984.

[**Safr76**] Safran, C., Desforges, J. F., and Tsichlis, P. N. "Diagnostic planning and cancer management." *MIT-LCS Technical Report LCS-TR-169* 1976.

[**Schl71**] Schlaifer, R. "Computer Programs for Elementary Decision Analysis." Division of Research, Graduate School of Business Administration, Harvard University, 1971.

[**Schw70**] Schwartz, W. B. "Medicine and the Computer: The Promise and Problems of Change." *NEJM* 283:1257, 1970 .

[**Schw73**] Schwartz, W. B., Gorry, G. A., Kassirer, J. P, and Essig, A. "Decision Analysis and Clinical Judgment." *AJM* 55:459, 1973.

[**Sher81**] Sherman, H. "A Comparative Study of Computer-Aided Clinical Diagnosis of Birth Defects." *MS Thesis, Electrical Engineering and Computer Science, MIT* Cambridge, Ma., 1981.

[**Smith78**] Smith, B. C. "A Proposal for a Computational Model of Anatomical and Physiological Reasoning." Massachusetts Institute of Technology, Artificial Intelligence Laboratory Report No. AI-Memo 493 1978.

[**Stef81a**] Stefik, M. "Planning with constraints (MOLGEN: Part 1)." *Artificial Intelligence* 16:111-140, 1981.

[**Stef81b**] Stefik, M. "Planning and meta-planning (MOLGEN: Part 2)." *Artificial Intelligence* 16:141-170, 1981.

[**Swar77**] Swartout, W. R. "A Digitalis Therapy Advisor with Explanations." *MIT-LCS Technical Report TR-176*, 1977.

[**Swar83**] Swartout, W. R. "A system for creating and explaining expert consulting programs." *Artificial Intelligence* 21:285, 1983.

[**Szol76**] Szolovits, P., and Pauker, S. G. "Research on a Medical Consultation System for Taking the Present Illness." in *Proceedings of the Third Illinois Conference on Medical Information Systems* University of Illinois at Chicago Circle, 1976.

[**Szol77**] Szolovits, P., Hawkinson, L., Martin, W. A. "An Overview of OWL, a Language for Knowledge Representation." in Rahmstorf, G., and Ferguson, M., (Eds.), *Proceedings of the Workshop on Natural Language Interaction with Databases*, International Institute for Applied Systems Analysis, Schloss Laxenburg, Austria, 1977.

[**Szol78**] Szolovits, P., Pauker, S. G. "Categorical and Probabilistic Reasoning in Medical Diagnosis." *Artificial Intelligence* 11:115, 1978.

[**Szol82**] Szolovits, P. "Artificial intelligence and medicine." in *Artificial Intelligence in Medicine* (Szolovits, ed.) AAAS Selected Symposium 51, Westview, 1982.

[**Teac81**] Teach, R. L., and Shortliffe, E. H. "An analysis of physician attitudes regarding computer-based clinical consultation systems." *Computers and Biomedical Research* 14:542, 1981.

[**Tver74**] Tversky, A. and Kahneman, D. "Judgment under uncertainty: Heuristics and biases." *Science* 185:1124-1131, 1974.

[**Warn61**] Warner, H. R., Toronto, A. F., Veasey, L. G., and Stephenson, R. "A mathematical approach to medical diagnosis: Application to congenital heart disease." *JAMA* 177:3 177-183 1961.

[**Weed71**] Weed, L. L., "Medical Records, Medical Education, and Medical Care," The Press of Case Western University, 1971.

[**Well85a**] Wellman, M. P. "Reasoning about preference models." master's thesis, MIT, in preparation.

[**Well85b**] Wellman, M. P. "Reasoning about assumptions underlying mathematical models." submitted to the Workshop on Coupling Symbolic and Numerical Computing in Expert Systems, 1985.

[**Whit78**] Whitmore, G. A. and Findlay, M. C. "Stochastic Dominance." Lexington Books, 1978.

[**Wile80**] Wilensky, R. "Meta-planning." in Proceedings of AAAI-80 334-336 1980.

[**Wilk84**] Wilkins, D. E. "Domain-independent planning: Representation and plan generation." *Artificial Intelligence* 22:269-301, 1984.

[**Wink82**] Winkler, R. L. "Research directions in decision making under uncertainty." *Decision Sciences* 13:517-533, 1982.

[Wins84] Winston, P. H. "Artificial Intelligence." Addison-Wesley, 1984.

[Yeh83] Yeh, Alexander Sen. "PLY: A system of plausibility inference with a probabilistic basis." master's thesis, MIT, 1983.

[Yeh85] Yeh, A. "Flexible Data Fusion (& Fission)," submitted to IJCAI-85, 1985.

Appendix: Case Material for Protocol Analysis CASE 1

The patient is a 45 year old woman, mother of two and wife of a malpractice attorney, who presented to a community hospital emergency ward with fever and malaise.

Her work-up included a CBC, chest xray and urine analysis, which were all reported to be normal. The patient was sent home with the tentative diagnosis of viral illness and she was instructed to stay in bed, drink fluids and take aspirin as needed.

Her symptoms of fever and malaise progressed over the next week and she sought medical attention at the same emergency ward.

The patient was admitted to the hospital and started on tetracycline and erythromycin. Several days later she began to develop symptoms of lethargy, paresis and dysesthesia in her right upper extremity.

A CT scan of the head was interpreted as being consistent with encephalitis.

At this point she was transferred to a tertiary care hospital. On admission there she was found to have a temperature of 38C with other vital signs noted to be stable. SKIN - was without rashes, LUNGS - were clear, HEART - revealed a systolic ejection murmur , no clicks, gallops or rubs, ABDOMEN - soft, nontender and without organomegaly, EXTREMITY - no edema. NEURO - was within normal limits. Laboratory Data: HCT 35, WBC 13,000 with normal differential cell count.

A neurologist was consulted and an arteriogram was performed.

The carotid arteriogram showed no evidence of intracranial vasculitis. A repeat CT scan of her head revealed evidence of four small cerebral emboli (compared to the CT scan obtained 5 days previously) over the cerebellum, left and right cerebral hemispheres.

An echocardiogram was obtained which revealed a 5 mm vegetation on the anterior leaflet of her mitral valve.

A cardiologist noted a murmur of mitral regurgitation, with no gallops and clear lungs on physical exam. He also noted a palpable spleen tip and several splinter hemorrhages.

Note: *At this point the dilemma is whether to collect more diagnostic information (i.e., stop antibiotics and obtain blood cultures) or to proceed directly to mitral valve replacement. While it would be optimal to culture the patient off antibiotics and complete a course of*

antibiotics for which the offending organism is sensitive, the prior course of antibiotics limits the likelihood of obtaining a positive blood culture and the patient will remain at risk for another cerebral embolus. Performing a mitral valve replacement now will remove the source of emboli, but without completing a course of antibiotics, the prosthetic valve may become infected with the same organism, which is very likely to be still present.

CASE 2

The patient is a 63 year old woman, retired social worker, with a 5 year history of coronary artery disease, who was seen by her family doctor because episodes of chest pain occurring with increased frequency after minimal exertion.

Her doctor admitted her to the cardiac care unit of a community hospital. On physical exam he noted: Bp 130/85, NECK- left carotid bruit, LUNGS - clear , HEART - Regular rhythm, prominent S4, no S3, no murmurs, ABDOMEN - soft nontender, no organomegaly, EXTREMITIES - without edema. EKG: Sinus Rhythm, normal conduction times, normal axis, minimal st segment depression in the inferior leads, but not significantly changed from her prior tracings.

She was continued on propranolol (40 mg Qid), isosorbide dinitrate (20 mg qid) and was started on diltiazem (60mg tid).

The patient had one episode of chest pain in the hospital which was associated with 1.5 mm of ST segment depression in V1 to V4.

The patient responded to sublingual nitroglycerine, and her ST segments returned to baseline. Her dose of diltiazem was increased to 80 mg tid. Two sets of cardiac enzymes at that point (36 hours) were negative for infarction. Plans were made to send the patient to a university hospital for further evaluation.

After transfer, the patient was pain-free for 36 hours, without EKG or cardiac enzyme evidence of infarction.

Cardiac catheterization was performed which revealed a 50 percent stenosis of the left main coronary artery, a 70 percent occlusion of the proximal LAD, a 60 percent occlusion of the obtuse marginal branch of the circumflex, and an 80 percent stenosis of the right coronary artery. There was good distal run off in the LAD and right coronary arteries.

A cardiothoracic surgeon was concerned about the left carotid bruit because of a history of a stroke 12 years before

Noninvasive carotid artery studies suggested tight carotid stenosis of the internal carotid arteries bilaterally.

A four vessel aortic arch arteriogram revealed a 70 percent occlusion of the left common carotid artery, with an ulcerating plaque, and the right internal carotid was totally (100 percent) occluded.

Note: *The dilemma at this point is which procedure(s) should be done, and in what order. Among the many options are:*

- *Left carotid endarterectomy first followed within a few weeks by coronary artery bypass graft surgery,*

- *Coronary artery bypass surgery followed in a few weeks by left carotid endarterectomy.*
- *Simultaneously perform both procedures.*

Each of these options has its risks and benefits.