ONCOCIN: AN EXPERT SYSTEM FOR ONCOLOGY PROTOCOL MANAGEMENT

Edward H. Shortliffe, A. Carlisle Scott, Miriam B. Bischoff,
A. Bruce Campbell, William Van Melle, Charlotte D. Jacobs

Heuristic Programming Project
Departments of Medicine and Computer Science
Stanford University, Stanford, California 94305

ABSTRACT

We describe an oncology protocol management system, named ONCOCIN, that is designed to assist physicians in the treatment of cancer patients. The system is actually a set of programs, one of which is a rule-based reasoner that encompasses the necessary knowledge of cancer chemotherapy. Representation and control techniques are discussed, and ONCOCIN is contrasted with systems that could be built using EMYCIN. Of particular interest is the need to provide ONCOCIN with an interface that will make the system acceptable to oncologists.

I Introduction

This report describes an oncology protocol management system, named ONCOCIN after its domain of expertise (cancer therapy) and its historical debt to MYCIN [5]. The program consists of a set of interrelated subsystems, the principal ones being:

(1) the Reasoner, a rule-based expert consultant that is the core of the system and the major subject of this report; and

(2) the Interviewer, an interface program that controls a high-speed terminal and the interaction with the physicians using the system.

Work on ONCOCIN began in mid-1979, and the system was installed for prelimianry use in May 1981. This paper describes only the Reasoner, although other system components are mentioned when their interface with the Reasoner is pertinent. We also contrast ONCOCIN with EMYCIN [8], explaining why the EMYCIN formalism was inadequate for our purposes although it did strongly influence the system's rule-based design.

II Overview of the Problem Domain

ONCOCIN is designed to assist clinical oncologists in the treatment of cancer patients. Because the optimal therapy for most cancers is not yet known, clinical oncology research is commonly based on formal experiments that compare the therapeutic benefits and side effects (toxicity) of proposed alternative disease treatments. "Cancer" is a general term for many diseases having different prognoses and natural histories. A treatment that is effective against one tumor may be ineffective against another. Thus a typical cancer research center may conduct many simultaneous experiments, each concerned with a different kind of cancer and its optimal therapy (i.e., the treatment plan with the best chance of cure, remission, or reduction in tumor size, and the least chance of serious side effects).

Each of these experiments is termed a "protocol". Patients with tumors *of* the type being studied, and who are accepted for protocol treatment, are randomly assigned to receive one of two or more possible treatments. The experiment requires close monitoring of each patient's clinical response and treatment toxicity. These data are tallied for all patients treated under the alternative regimens, and in this way the "state-of-the-art" is updated over time.

Each protocol is described in a detailed document, often 40 to 60 pages in length, which specifies the alternative therapies being compared and the data that need to be collected. No single physician is likely to remember the details in even one of these protocol documents, not to mention the 30 to 60 protocols that may be used in a major cancer center. Although an effort is made to have the documents available in the oncology clinics when patients *are* being treated for their tumors, it is often the case that a busy clinic schedule, coupled with a complex protocol description, leads a physician to rely on his memory when deciding drug doses or what laboratory tests to order. Furthermore, solutions for all possible treatment problems cannot be spelled out in protocols. Physicians use their own Judgment in treating these patients, resulting in some variability in treatment from patient to patient. Thus patients being treated on a protocol do not always receive therapy in exactly the manner that the experimental design suggests, and the data needed for formal analysis of treatment results are not always completely and accurately collected.

This work was supported by the National Library of Medicine under Program Project Grant LM-03395 and research career development award LM-00048. The program was developed on the SUMEX Computer Project at Stanford University, a shared national resource supported by the Biotechnology Resources Program of the NIH under grant RR-00785.

The problems we have described reach far beyond the oncology clinic at Stanford Medical Center. There are now several institutions designing protocol management systems to make the details of treatment protocols readily available to oncologists and to insure that complete and accurate data are collected. ONCOCIN is superficially similar to some of the developing systems, but its research objectives are unique in ways we describe in the following section. One overriding point requires emphasis: in order to achieve its goals, ONCOCIN must be used directly by busy clinicians; the implications of this constraint have pervaded all aspects of the system design.

III Design Considerations

The design of the ONCOCIN system has been influenced by the two parallel thrusts of the project's research aims. The first is to perform research into the basic science issues of applied artificial intelligence. Efforts are being made to improve the tools currently available, and to develop new tools for building knowledge-based expert systems for medical consultation. Other areas of AI research include the extension of methods of interaction between rule-based consultation systems and a large database of time-oriented clinical information, decision making based upon trends over time, and the development of techniques for assessing knowledge base completeness and consistency.

The second thrust is to develop a clinically useful oncology consultation tool. One of the primary goals of the project is to demonstrate that a rule-based consultation system with an explanation capability can be usefully applied in a busy clinical environment. While working to achieve this goal, we hope to establish both an effective relationship with a specific group of physicians, and a scientific foundation, that will together facilitate future research and implementation of computer-based tools for clinical decision making.

IV System Overview

The ONCOCIN system will eventually contain knowledge about most of the protocols in use at the Oncology Clinic at Stanford Medical Center. Although protocol knowledge is largely specified in a written document, many questions arise in translating the information into a computer-based format. Knowledge base development has therefore been dependent on the active collaboration of Stanford oncologists. We have started by encoding the knowledge contained in the protocols for treatment of Hodgkins Disease and the non-Hodgkins lymphomas.

In standard use of the system, the physician uses a video display terminal, after examining a patient, to interact with ONCOCIN's data-acquisition program (the Interviewer). The session normally includes: reviewing time-oriented data from the patient's previous visits to the clinic, entering information regarding the current visit, and receiving recommendations generated by the "Reasoner" for appropriate therapy and tests. The Reasoner and Interviewer are linked with one another as shown in Fig. 1. In generating its recommendation, the Reasoner uses initial data about the patient's diagnosis, data about previous treatment, results of current laboratory tests, plus the protocol-specific information in its knowledge base. Before terminating an interaction, the physician can examine the explanation provided with each recommendation. The physician may approve or modify ONCOCIN's recommendation; *any* changes are noted by the system and kept available for future review. ONCOCIN also provides hard-copy backup to complement the on-line interaction and facilitate communication among clinic personnel.

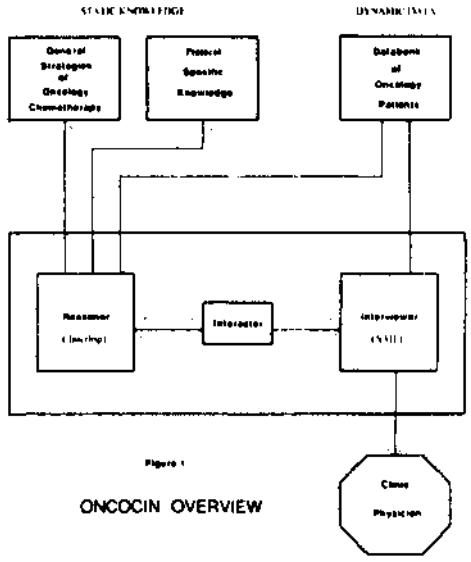
To ensure that busy clinicians will find ONCOCIN fast and easy to use (as well as simple to learn), a special terminal interface has been built. This Interviewer (Fig. 1) includes hardware and software features not often found in current medical systems. A customized keyboard incorporates a keypad for most of the entry and command functions. The Interviewer uses a high-speed video display terminal with multiple windows; simulating the appearance of the flowsheet that currently appears in the clinic chart and which is used to record all the data required for adequate protocol analysis. As the physician enters flowsheet information, relevant patient data are passed to the Reasoner, which is simultaneously considering the case and preparing a recommendation to pass back to the Interviewer for display. The need for a terminal interface that provides rapid response to user type-in, while simultaneously satisfying the computational demands of the reasoning process, has led us to design a complex system architecture with asynchronous processes.

A memo from the MIT Laboratory for Computer Science [7] describes a recent collaboration between MIT and oncologists who have been building a protocol management system at Boston University [3]. They are planning to develop a program for designing new chemotherapy protocols. To our knowledge, this is the only other project that proposes to use AI techniques in a clinical oncology system. However, the stated goals of that effort are different from those of ONCOCIN.

We also implemented the complex protocol for treating oat cell carcinoma of the lung. Because the oat cell protocol is the most complex at Stanford, and it took only a month to encode the relevant rules, we are confident that the representation scheme we have devised will be able to manage, with only minor modifications, the other protocols we plan to encode in the future.

We have chosen a representation that will also eventually allow ONCOCIN to offer a justification for any intermediary conclusions that the system made in deriving the advice.

The Reasoner and the Interviewer each run in a separate fork under the TENEX or TOPS-20 operating systems, therebv approximating a parallel processing system . Separation of the interface program from the reasoning program allows the user to receive prompt attention when entering flowsheet values; the Interlisp program, running in background, is not permitted to slow down the interaction.



ONCOCIN OVERVIEW

V The Reasoner

A Why Not EMYCIN?

ONCOCIN's Heasoner communicates with the Interviewer during a consultation. The need to interact with this specialized interface program is one of several reasons that we chose to build ONCOCIN from scratch rather than to implement it as a new EMYCIN system [8]. Other important differences between ONCOCIN's application and the domains for which EMYCIN systems have been built include the following:

(1) ONCOCIN requires serial consideration of each patient at intervals typically spread over many months. Each clinic visit is a new data point, and EMYCIN's data structures do not easily accommodate multiple measurements of the same attribute over time or inference rules based upon assessment of temporal trends for a given parameter ;

(2) ONCOCIN does not require many of the capabilities provided by EMYCIN.

(3) Because of the nature of the interaction with the Interviewer, ONCOCIN needs to operate in a data-driven mode. Although EMYCIN has limited allowance for forward chaining of rules, it would be inconvenient to force a largely data-driven reasoning process into the EMYCIN format.

Another program, the Interactor, handles interprocess communication. There is also a process that provides background utility operations such as file backup. The Reasoner is written in Interlisp, the Interviewer in SAIL.

B Representation

Knowledge about the oncology domain is represented using four main types of data structure: Contexts, Parameters, Rules, and Control Blocks.

Contexts represent concepts or entities of the domain about which the system needs static knowledge, and aid in organizing the knowledge base. Individual contexts are classified by type (e.g., disease, protocol, *or* chemotherapy) and can be arranged hierarchically. During a consultation, a list of "current" contexts is created as information is gathered. These current contexts together provide a high-level description of the patient in terms of known chemotberapeutic plans. This description serves to focus the system's recommendation process by specifying the situation when a rule can be applied or when data should be requested. Knowledge can be specific to one or more contexts at varying levels of the contextual hierarchy.

Parameters represent the attributes of patients, drugs, tests, etc. that are relevant for the protocol management task (e.g., white blood count, recommended dose, or whether a patient has had prior radiotherapy). Each piece of information accumulated during a consultation is represented as the value of a parameter. There are three steps in determining the value of a parameter. First, the system checks to see if the value can be determined by definition in the current context. If not, the "normal" method of finding the value is used: if the parameter corresponds to a piece of laboratory data that the user is likely to know, it is requested of the user; otherwise, rules for concluding the parameter are tried. Finally, the system may have a (possibly context-dependent) default value that is used in the event that the

This same point led to the development of the VM system [2], a rule-based program that was influenced by EMYCIN but different in its detailed implementation because of the need to follow trends in patients under treatment in an intensive care unit. The development of similar capabilities for ONCOCIN is an active area of research at present. The initial persion of the system, however, encodes only the knowledge from the protocols and does not consider other factors that rely on expert judgment.

normal mechanism fails to produce a value, or the user may be asked to provide the answer as a last resort.

Rules are the familiar productions used in EMYCIN [8] and other rule-based systems; they may be invoked in either data-driven or goal-directed mode. A rule concludes a value for some parameter on the basis of values of other parameters. A rule may be designated as providing a definitional value or a default value as defined above. The rules are categorized by the contexts in which they apply.

As in EMYCIN systems, rules are represented in a stylized format so that they may be translated from Lisp into English for explanation purposes . This representation scheme more generally allows the system to "read" and manipulate the rules. It has also facilitated the development of programs to check for consistency and completeness of the rules in the knowledge base [6].

Below is the English translation of an ONCOCIN rule for determinjng the value for the parameter "attenuated dose"

RULE075

To determine the current attenuated dose for all drugs in MOPP or for all drugs in PAVe:

If: 1) This is the start of the first cycle after a cycle was aborted, and
2) The blood counts do not warrant dose attenuation

Then: Conclude that the current attenuated dose is 75 percent of the previous dose.

Control blocks serve as high-level descriptions of the system's methods for performing tasks. Each contains an ordered set of steps to be used for accomplishing a specific task (e.g., formulating a therapeutic regimen, or calculating the correct dose of a drug). These could be viewed as a script of the events which occur when a patient is being treated on chemotherapy [4]. Note that this data structure allows us to separate control descriptions explicitly from decision rules, a distinction that was often unclear in EMYCIN systems. The design of control blocks was influenced by the prototypes used in the CENTAUR system [1]. The steps in a control block are similar to the prototype's control slot. Because we wish to be able to explain any action that ONCOCIN takes, control blocks can be translated into English using the same translation mechanism that is used to translate rules. For example:

In keeping with the philosophy reflected in other systems we have designed, ONCOCIN is able to produce natural language explanations for its recommendations.

PAVe and MOPP are acronyms for two of the drug combinations used to treat Hodgkins Disease.

ADVISE

To make a recommendation about treating the patient:
1) Formulate a therapeutic regimen.
2) Determine the tests to recommend.
3) Determine suggestions about the patient.
4) Determine the time till the patient's next visit.

To summarize the differences between ONCOCIN's rules and those used in MYCIN and other EMYCIN systems:

(1) Control is separated from domain knowledge, although process information is still codified in a modular format using control blocks;

(2) The contextual information, which defines the setting in which a rule can be applied, is separated from the main body of the rule and used for screening rules when they are invoked (see next section); and

(3) Rules are subclassified to distinguish the major mechanisms by which the values of parameters can be determined (definitional, normal, and default rules).

C Control

When a user specifies the task that ONCOCIN is to perform, the corresponding control block is invoked. This simply causes the steps in the control block to be taken in sequence. These steps may entail:

(1) Fetching data, either by loading previously stored data or by requesting them from the user. This causes parameter values to be set, resulting in data-directed invocation of rules that use those parameters (and that apply in the current context).

(2) Determining the value of a parameter. This causes goal-directed invocation of the rules that conclude the value of the parameter (and apply in the current context). Definitional rules are applied first, then the normal rules, and if no value has been found through these means, the default rules are tried. If a rule that is invoked in a goal-directed fashion uses some parameter whose value is not yet known, that parameter's value is determined so that the rule can be evaluated. However, concluding the value of any parameter, either by the action of rules or when information is entered by the user, may cause data-directed invocation of other rules.

(3) Invoking another control block.

(4) Calling a special purpose function (which may be domain-dependent).

The effects of this control mechanism contrast with the largely backward-chained control used in MYCIN and other EMYCIN systems. In those systems, all rule invocation occurs because the value of a specific parameter is being sought. Rules used to determine the value of that parameter can be referenced in any order, although ordering is preserved for the assessment of the parameters occurring in the conditional statements in each rule's premise. Antecedent (data-driven) rules are used when the user's response to a question, or (less commonly) the conclusion from another rule, triggers one of the system's forward-chained rules. These rules can only be used as antecedent rules and typically have single conditions in their premise.

In ONCOCIN (Fig. 2), on the other hand, initial control is derived from the control block invoked in response to the task selected by the user. Forward- and back-chaining of rules are intermingled , and any rule can be used in either direction.

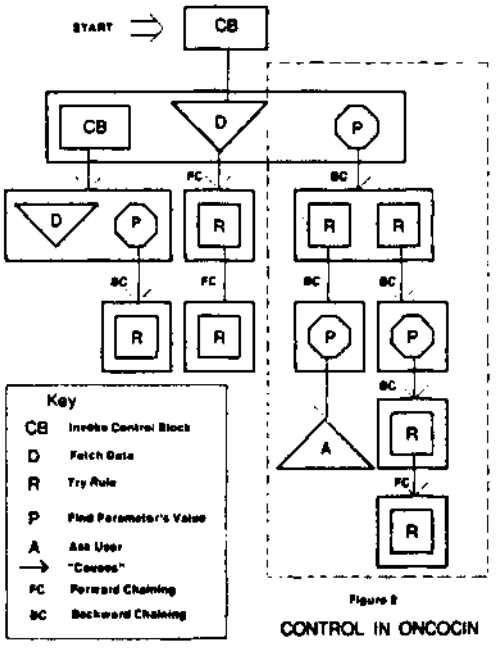


Figure 2

CONTROL IN ONCOCIN

There are trade-offs when choosing between a system which is primarily data-driven and one which is goal-directed. Back-chained systems use rules to generate questions, thus facilitating explanations. The disadvantage, however, is that a goal oriented system may appear slow to get to the main points. On the other hand, data-driven systems are more difficult to implement because they must be able to handle entry of information in arbitrary order. In the case of ONCOCIN. the use of a flowsheet structure for data entry has enabled us to develop a data-driven system. The entries on the flowsheet define all of the important data for therapy decisions . Since the doctors use the Interviewer to enter the same data that they would normally record anyway, the potential frustration of interacting with a goal-directed system is removed.

The broken line in Fig. 2 outlines the portion of the ONCOCIN control structure which is identical to that found in EMYCIN.

VI Why Artificial Intelligence Techniques?

We have learned through the MYCIN experience, and in building other EMYCIN systems as well, that a major part of each development effort has been the encoding of poorly understood knowledge. Enlisting the time and enthusiasm of domain experts has often been difficult, yet progress is usually impossible without active collaboration. Thus there is great appeal to a domain in which much of the needed knowledge is already recorded in thorough, albeit lengthy and complicated, documents (viz., the protocol descriptions that are written for every cancer therapy clinical experiment). Much of the appeal of the ONCOCIN problem domain is the availability of detailed documents that we can study and use for knowledge base development.

As we noted earlier, several other centers have begun to develop protocol management systems, but none has chosen to use techniques drawn from artificial intelligence. Complicated though the chemotherapy protocols may be, they are largely algorithmic, and other groups have been able to encode much of the knowledge using less complex representation techniques. Our reasons for choosing an AI approach for encoding the knowledge of oncology chemotherapy are varied. It should be stressed that all protocols have important loopholes and exceptions; when an aberrant situation arises for a patient being treated, the proper management is typically left unspecified. For example, the lymphoma protocols with which we have been most involved to date include several rules of the form:

IF: there is evidence of disease extension
THEN: refer the patient to lymphoma clinic

or:

IF: there is significant toxicity to vincristine
THEN: consider substituting velban

As shown here, the protocols often defer to the opinions of the attending physicians without providing guidelines on which they might base their decisions. Hence there is no standardization of responses to unusual problems, and the validity of the protocol analysis in these cases is accordingly subject to question. Our long-term goal is to develop approaches to these more complex problems that characterize the management of patients being

The physicians can insert new items into the on-line flowsheet, as they presently do by hand with the paper flowsheet. However, these entries are not used when generating a therapy recommendation.

treated for cancer. It is when these issues are addressed that the need for AI techniques will be most evident. At that point the task domain will begin to look similar in complexity to the decision problems in a system like MYCIN; rules will have uncertainty associated with them (there is currently no need for certainty weights in the rules in ONCOCIN), and close collaboration with experts will once again be required as new rules are written that are not currently recorded in chemotherapy protocols or elsewhere. Even in the short term, however, AI representation and control techniques have permitted us to keep the knowledge base flexible and easily modified. They have also allowed us to develop explanation capabilities and to separate kinds of knowledge explicitly in terms of their semantic categories.

For the present, then, we are trying to build a useful system to which complex decision rules can eventually be added. We know from previous experience that the encoding of knowledge that is not already stated explicitly in protocols will be arduous and will require an enthusiastic community of collaborating physicians. Hence we recognize the importance of one of our research goals noted earlier in this report, viz., to establish an effective relationship with a specific group of physicians so as to facilitate future research and implementation of advanced computer-based clinical tools.

VII Summary

The development of the consultation system addresses basic science questions regarding the optimal representation and utilization of complex medical knowledge. Cancer chemotherapy is a domain that inherently requires consideration of the time course of disease, for example, and we are only beginning to develop techniques for codifying knowledge that relates to temporal analyses of disease or of response to therapy.

The project seeks to identify new techniques for bringing large AI programs to a clinical audience that would be intolerant of systems that are slow or difficult to use. The design of an interface that uses both custom hardware and efficient software should heighten the acceptability of the consultation system. ONCOCIN was implemented for experimental use with lymphoma patients beginning in May 1981. Formal evaluations have been designed that will allow us to determine both the effectiveness and the acceptability of the system's clinical advice.

Finally, ONCOCIN is designed to assist physicians with an important clinical domain in which there is a recognized need for improved data collection and decision making. Furthermore, several other specialty clinics, at Stanford and other centers, use time-oriented treatment protocols, and the techniques developed here are potentially applicable to several clinical domains in addition to oncology chemotherapy planning.

ACKNOWLEDGMENTS

The ONCOCIN System described in this report has relied on the efforts of several physicians and computer scientists. We would like to express our gratitude to the following present and past project members: Paul Chang, Jai Cho, Chuck Clanton, Larry Fagan, Phil Gerring, Patricia Pickett, Brandy Sikic, Motoi Suwa, Randy Teach, and Craig Tovey.

REFERENCES

1. Aikins, J.S. "Prototypes and Production Rules: A Knowledge Representation for Computer Consultations." Doctoral dissertation, Computer Science Department, Stanford University, August 1980.

2. Fagan, L.M. "VM: Representing Time-Dependent Relations in a Medical Setting." Doctoral dissertation, Computer Science Department, Stanford University, June 1980.

3. Horwitz, J. Thompson, H., Concannon, T. et al. "Computer-assisted patient care management in medical oncology." Proceedings of the 4th Symposium on Computer Applications in Medical Care, pp. 771-780, Washington, D.C., November 1980.

4. Schank, R. Scripts, Plans, Goals and Understanding, New York: John Wiley and Sons, 1977.

5. Shortliffe, E.H. Computer-Based Medical Consultations: MYCIN, New York: Elsevier/North Holland Publishing Company, 1976.

6. Suwa, M., Scott, A.C., and Shortliffe, E.H. "An approach to verifying completeness and consistency in a rule-based expert system". Memo HPP-05-81, Stanford Heuristic Programming Project, Stanford University, April, 1981.

7. Szolovits, P. "Artificial intelligence and clinical problem solving". Technical Memo MIT/LCS/TM-140, Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, MA, September 1979.

8. van Melle, W. "A Domain-Independent System that Aids in Constructing Knowledge-Based Consultation Programs." Doctoral dissertation, Computer Science Department, Stanford University, June 1980.