Report 85-17
Stanford -- KSL

Rationale for Knowledge Base Redesign in a Medical Advice System.
Mark Musen, Curtis Langlotz, et al.,
1985

card 1 of 1

Knowledge Systems Laboratory
Report No. KSL 85-17

RATIONALE FOR KNOWLEDGE BASE REDESIGN IN A MEDICAL ADVICE SYSTEM

Mark Musen, Curtis Langlotz, Lawrence Fagan, and Edward Shortliffe

Medical Computer Science Group
Knowledge Systems Laboratory
Stanford University
Stanford, California 94305

ONCOCIN is a rule-based expert system to advise on cancer chemotherapy. Although shown to provide excellent advice, the program could not be easily adapted to *critique* a physician's treatment plan without incorporation of additional knowledge of the structure of experimental protocols. A separate effort to automate the encoding of new oncology protocols was impeded by the lack of structural organization in the knowledge base. In both cases, problems arose because ONCOCIN's knowledge representation scheme did not reflect the hierarchy of control knowledge inherent in oncology protocols. The limitations of current knowledge representation techniques in ONCOCIN are discussed.

Introduction

ONCOCIN is a medical expert system that assists physicians in the treatment of cancer patients enrolled in chemotherapy protocols. Like its predecessor program MYCIN [1], ONCOCIN has been judged in blinded evaluations to give advice of equal quality to that of actual clinical experts [2]. ONCOCIN is used in the daily care of patients at Stanford's oncology clinic.

The high acceptability and performance of the system reinforce a number of important early design decisions that have been previously described [3]. However, it has only been through the effort of building the initial prototype, studying its acceptance in the oncology clinic, and attempting to improve on its scope and performance that we have recently been able to identify major limitations in our initial architecture for storing oncology knowledge in the computer.

Absence of an overt representation of the hierarchical relationship of elements in the knowledge base has forced programmers to develop new knowledge structures to drive several extensions to the basic consultation system. In this paper, we discuss how failure of ONCOCIN to encode the underlying structure of oncology protocols explicitly in its initial knowledge base has limited the system's explanation abilities and complicated expansion of the system for new protocols. Having identified the problems, our group is now developing a new version of ONCOCIN designed for single-user workstations that simultaneously addresses these past limitations in knowledge representation.

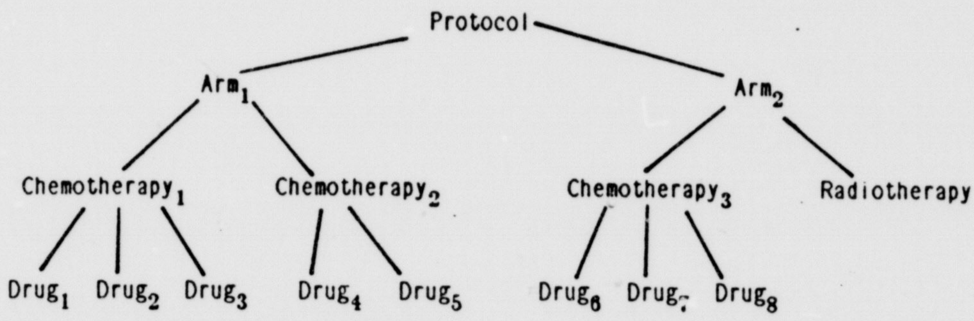
The Domain of Oncology Protocols

Outpatient therapy for cancer patients in academic centers is routinely based on special *protocols* that compare two or more treatment regimens. These regimens generally consist of sequences of multiple drugs repeated at intervals, where each sequence is referred to as a cycle of chemotherapy. Radiation therapy or surgery may also be administered. Each protocol *arm* specifies the chemotherapies and other treatments to which individual patients may be randomly assigned. Protocols also indicate alterations in the standard treatment that are made in accordance with patients' responses to the drugs and radiation, with resultant lowering of drug dosages or complete delay of scheduled treatment as necessary.

The components of an oncology protocol conform to an explicit structural hierarchy. The

#113

hierarchy for a typical protocol might be diagrammed as follows:



The Original ONCOCIN Knowledge Base

A detailed discussion of knowledge representation in ONCOCIN has appeared previously [3]. ONCOCIN's static knowledge of oncology protocols is encoded using three types of data structures: Control Blocks, Frames, and Rules.

Control Blocks serve as high-level descriptions of the system's methods for performing tasks. Each contains an ordered set of steps to be taken in accomplishing a specific action (e.g. formulating a therapeutic regimen or determining when a patient's next clinic visit should be scheduled). The control blocks capture the procedural knowledge needed to care for cancer patients in general, independent of the particular protocol that actually specifies the therapy.

Frames are data structures used to store static protocol knowledge that is declarative in nature. Each slot in a frame contains either a particular piece of information or a pointer to another frame where the actual data (or yet another pointer) may be found. The sequences of chemotherapies in each protocol, the standard dosages of drugs, and the duration of each chemotherapy cycle are examples of knowledge represented by frames. The data in frames are never changed during a consultation nor used for inference except by rules.

Rules are productions similar to those used in MYCIN that may be used for either data-driven or goal-directed inference. ONCOCIN has over 400 protocol-dependent rules that are used to determine (1) when to delay treatment or abort a cycle of chemotherapy, (2) how to adjust drug dosages because of treatment-induced low blood counts or other adverse reactions, and (3) when to order required laboratory tests and how to interpret their results. As in other rule-based systems [1], each of ONCOCIN's rules is indexed in the knowledge base by both the premise statements in its left-hand-side and by the conclusion that the rule can reach. Although most rules pertain to more than one protocol, ONCOCIN does not index the rules by protocol nor any other higher organizational framework.

ONCOCIN does denote as part of each rule the list of *contexts* in which the rule is applicable. Contexts may indicate a type of cancer, a chemotherapy, a drug, or a combination of these concepts to encode a specific position in the hierarchy of protocol knowledge--for example, the use of "*mustard* in *MOPP* chemotherapy for *lymphoma*". During a consultation, ONCOCIN keeps track of the "current" context, a reflection of the location in the knowledge hierarchy where reasoning happens to be focused. Because the knowledge base lacks true hierarchical organization, however, each time the current context changes, ONCOCIN must successively re-examine its rules to see which apply in the new context. This inefficiency slows the consultation and motivates much of our effort to redesign the knowledge base.

New Research Directions

Once the initial ONCOCIN prototype was installed in our oncology clinic in May 1981, feedback from physicians using the system helped focus our attention on two new areas of research.

One complaint regarded the fashion in which ONCOCIN offers its advice. When physicians use

BLANK PAGE

ONCOCIN, they enter information regarding a patient's clinical condition into the computer. The system then produces a treatment recommendation. If the physician prefers to alter the program's suggestion in some minor way, he or she must then enter the different treatment and indicate why ONCOCIN's decision is being overridden. The physician always remains the final decision-maker regarding a patient's therapy; if he or she must countermand ONCOCIN frequently, overriding the system can become annoying.

Another criticism raised by the system's new users pointed to the limited number of protocols that were encoded for the system. Initially, ONCOCIN could be used only for patients with lymphoma. Although protocols for breast cancer were later added to the knowledge base, entering additional protocols proved to be a tremendously time consuming process for both programming staff and collaborating oncologists. Attention has therefore turned to developing a means to expedite the incorporation of new protocols for ONCOCIN.

Developing the Critiquing Model

To deal with the first complaint, we recognized that if the system were modified to monitor and critique the therapy plans proposed by users, it could conduct a consultation in a much less disruptive manner. In addition, doctors would routinely suggest treatment first and thereby remain more actively involved in the consultation process. The educational role of the system would also be heightened because a critiquing system could not only point out the differences between its own recommendation and the physician's, but it could also help the physician make an informed choice between the two by explaining the differences.

Early experience suggests that this mode of interaction will enhance a program's acceptability for some applications [4, 5]. Little would be gained by the critiquing approach if the consultation program were to enter into a lengthy analysis of *every* plan entered by the user. Only when a significant disagreement has occurred should a critiquing expert system interrupt to explain the problem it has noted.

In the first step of the critiquing process, the physician enters the patient data that ONCOCIN uses to formulate its recommendation. Instead of showing the recommendation to the user, however, advice is withheld until after the physician has entered his or her proposed treatment. The critiquing module then uses a process called ***hierarchical plan analysis*** to compare systematically the physician's plan with the one formulated by ONCOCIN. Explanations of clinically significant differences are generated if the physician requests them. Once satisfied, the physician can (1) accept ONCOCIN's recommendation, (2) modify ONCOCIN's recommendation, or (3) choose to follow his or her original recommendation.

It is the plan analysis phase of critiquing which led to the need for an explicit hierarchy to augment the ONCOCIN knowledge base. Hierarchical plan analysis requires that domain specific knowledge about the components of a recommendation and their inter-dependencies be represented distinctly in the knowledge base. For example, decisions about radiation therapy are dependent on the initial protocol choice; decisions about the dosing adjustments for a certain drug are dependent on whether that agent is to be included in the therapy in the first place. It makes no sense to critique the dose of a particular drug when the primary difference between the the physician's plan and ONCOCIN's recommendation involves the choice of what drugs to give.

Hierarchical plan analysis is thus an evaluation process to find the most general set of differences which completely account for the significant disagreements between two therapy plans. Analysis begins with the top component in a hierarchy of treatment options. At progressively deeper levels, the critiquing module determines whether analogous components of the computer and the physician recommendations differ in clinically significant ways. Evaluation procedures specifically designed for each component of the hierarchy are used.

Development of the critiquing module required significant augmentation of the ONCOCIN knowledge base. In particular, the process of analyzing the physician's plan required the representation of hierarchical relationships that had not previously been made explicit. Revising ONCOCIN for critiquing provided the first suggestion that the initial knowledge representation scheme was not nearly as complete or as flexible as we had assumed.

Automating Knowledge Acquisition

A second focus of research that emerged from our experience with the ONCOCIN prototype involves expediting the encoding of the many new protocols needed for the system. This process of eliciting and entering new domain knowledge from experts is termed *knowledge acquisition* and remains a chief bottleneck in expert systems development [6].

Our experience with ONCOCIN and our review of dozens of oncology protocols indicates that these protocols are particularly well suited for attempts at automated knowledge acquisition. First, the core knowledge of each protocol has already been formalized in a protocol document. More important, protocols are highly structured hierarchically and the protocol knowledge is also both *predictable* and *constrained*. For each concept that appears in oncology protocols, we can anticipate the nature of most of its possible values. For example, we can predict that all chemotherapies will be comprised of drugs (selectable from a finite list) and that all drugs will have some numeric dose. Knowledge of the domain allows us to determine *a priori* what range of choices are appropriate for most concepts.

We have consequently begun work on a knowledge editing tool called *OPAL* [7]. OPAL allows an oncologist to work independently at a display terminal, entering information about new protocols using a "fill in the blanks" approach. The program solicits new knowledge by asking the user to fill out tabular forms produced on a large bit-mapped display. The formatting is easily understood by physicians and is often identical to the tables that appear in protocol documents. The physician need never be concerned with how this protocol knowledge might ultimately be represented in the computer for use by ONCOCIN.

OPAL captures protocol knowledge at a very high level. When a physician uses OPAL to describe a new oncology protocol, this knowledge must then be mapped into a format that can be used by the ONCOCIN reasoning program. There is an isomorphism between knowledge represented graphically by OPAL and the control blocks, frames, and rules that form the ONCOCIN knowledge base. The challenge, then, is to translate between the two representations.

OPAL and the ONCOCIN Knowledge Base

As in our efforts to adapt ONCOCIN for a critiquing mode, the attempt to transfer information between OPAL and the ONCOCIN knowledge base has been impeded by the relatively unstructured nature of knowledge representation in ONCOCIN. Although prototype software has been developed to effect part of this transfer, the program is awkward and inefficient.

Exchange of knowledge between OPAL and ONCOCIN is hampered because ONCOCIN's production rules occur linearly in the knowledge base with no underlying organization. The knowledge in oncology protocols has an intrinsic hierarchical structure; OPAL capitalizes on this structure to organize and control a physician's interaction with the system. But once OPAL has captured a protocol's knowledge with all of these detailed hierarchical relationships preserved, the generation of rules for ONCOCIN's "flat" knowledge base requires that the hierarchical state that defines the applicability of each rule be converted to the form of a "context specification".

The existing contexts in ONCOCIN unfortunately do not tell us everything we would like to know about the use of a rule in the knowledge hierarchy. For example, "protocol" is not recognized by the current system as a specific context. The concept of "protocol arm" cannot even be represented in the present knowledge base. Although we could always add different contexts to the knowledge base to represent these supplementary concepts, the context system itself is inflexible and awkward to extend once a knowledge base is partially complete. Worse, increasing the number of contexts would only compound the processing inefficiencies already discussed that occur each time ONCOCIN changes the current context.

OPAL is designed not only for entry of new protocol data but also for graphical review of protocols already encoded in the ONCOCIN knowledge base. When information concerning a specific protocol is to be transferred from ONCOCIN back to OPAL, the current representation requires that the entire knowledge base be examined sequentially to search for rules with appropriate contexts. A similar exhaustive search of frame-based data must be performed as well. Inefficiency again results from ONCOCIN's inability to represent explicitly the intrinsic hierarchical relationship among the elements in its knowledge base.

We eventually want physicians to be able to use ONCOCIN for all of the cancer protocols at Stanford. Yet as more and more protocols are added via OPAL, the knowledge base runs the risk of exceeding allocatable memory in the computer if all protocols are loaded in at the same time. Unfortunately, the present representation scheme offers no direct way to partition the knowledge base when necessary because the knowledge for the various protocols is so intertwined.

Revising Knowledge Representation in ONCOCIN

Despite the excellent performance of the program, our experiments with both the critiquing model and with automating knowledge acquisition have pointed out limitations in the format of ONCOCIN's knowledge base. Efforts are now being directed at reimplementing ONCOCIN using a hierarchically organized representation that is based on object-oriented programming.

Each protocol, chemotherapy, and drug in the knowledge base is being treated as a discrete "object". Relationships between objects are clearly defined and each object is linked to the particular rules that conclude the object's properties. The hierarchical structure of the knowledge explicitly identifies the rules that are relevant at any stage of the consultation. The additional structure also allows partitioning of the knowledge base as necessary and permits direct flow of information between ONCOCIN and OPAL.

Conclusions

The design of ONCOCIN departed from the purely rule-based approach of MYCIN by moving certain procedural knowledge out of rules into special *control blocks*. Making control knowledge explicit and separate from the decision rules helped add modularity to the knowledge base and led to more efficient processing during consultations. It was not initially obvious, however, that in each step of control block invocation, only a small portion of the decision knowledge would be applicable--due to the highly structured nature of oncology protocols. In the present system, ONCOCIN must consider and then discard large numbers of irrelevant rules at each step. Furthermore, the inefficiency compounds as more protocols (and hence more rules) are added to the system. The current reimplementation of the system on workstations is designed in part to overcome these deficiencies in our original approach.

Recently, rule-based systems have been criticized for an inability to reason directly about their own structural knowledge [8]. In the case of ONCOCIN, tasks that depend on the hierarchical relationships among elements in the knowledge base are clearly hindered by the present representation scheme. Both the critiquing module and OPAL must create additional, hierarchical representations of domain knowledge in order to interface with the existing knowledge base. Although not discussed here, an editing tool that displays graphically the relationships among elements in the ONCOCIN knowledge base is also forced to generate similar additional knowledge structures [9].

Pople [10] has discussed limitations of the INTERNIST knowledge base that emerged during a decade of research with that important program. As with ONCOCIN, a failure to represent hierarchical relationships in the knowledge is cited as the major failure of INTERNIST--a problem to be addressed by the successor program CADUCEUS.

While our group has not abandoned the rule-based approach in designing ONCOCIN's new knowledge representation scheme, decision rules now fit into a hierarchy that relates directly to the control knowledge. The resulting representation should yield more efficient consultations and facilitate partitioning the knowledge base as the number of protocols in the system increases. By building the intrinsic structure of oncology protocols directly and explicitly into the knowledge base, we can enhance the performance of all tasks that must access and manipulate that knowledge.

References

1. Buchanan, B. G. and Shortliffe, E.H. *Rule Based Expert Systems: The MYCIN Experiments of the Stanford Heurisitic Programming Project*. Addison-Wesley, 1984.

2. Hickam, D. H., Shortliffe, E. H., Bischoff, M. B., Scott, A. C., Jacobs, C. D. A Computer Based Treatment Consultant for Clinical Oncology: The Quality of Computer-Generated Advice. Heuristic Programming Project Memo HPP-84-9, Stanford University, 1984. (submitted for publication)

3. Shortliffe, E. H., Scott, A. C., Bischoff, M. B., Campbell A. B., van Melle, W., Jacobs, C. D. ONCOCIN: An Expert System for Oncology Protocol Management. Proceedings of the Seventh International Joint Conference on Artificial Intelligence, Vancouver, British Columbia, Aug., 1981, pp. 876-881.

4. Miller, P. L. "Critiquing Anesthetic Management: The 'ATTENDING' Computer System." *Anesthesiology 59* (1983), 362-369.

5. Langlotz C. P. and Shortliffe, E. H. "Adapting a Consultation System to Critique User Plans." *International Journal of Man-Machine Studies 19*, 5 (1983), 479-496.

6. Duda, R. O., and Shortliffe, E. H. "Expert Systems Research." *Science 220* (1983), 261-268.

7. Differding, J. C., Musen, M., Lane, C. D., Fagan, L., Shortliffe, E. H. Knowledge Acquisition for Well Structured Application Areas. (in preparation)

8. Clancey, W. J. "The Epistemology of a Rule-Based Expert System: A Framework for Explanation." *Artificial Intelligence 20*, 3 (1983), 215-251.

9. Tsuji, S. and Shortliffe, E. H. Graphical Access to the Knowledge Base of a Medical Consultation System. Proceedings of AAMSI Congress 83, American Association for Medical Systems and Informatics, May, 1983, pp. 551-555.

10. Pople, H. Heuristic Methods for Imposing Structure on Ill-structured Problems: The Structuring of Medical Diagnosis. In *Artificial Intelligence in Medicine*, P. Szolovits, Ed.,Westview Press, Boulder, CO, 1982.

Acknowledgements

This work was supported by NIH grant RR-01631 and ONR contract NR 049-479. Computing facilities were provided by the SUMEX-AIM resource under NIH grant RR-00785. Dr. Musen (Fellow in General Internal Medicine) and Dr. Shortliffe (Faculty Scholar) have also received support from the Henry J. Kaiser Family Foundation. We thank Jay Ferguson, Christopher Lane, and Joan Differding for many helpful discussions.

FILMED FROM BEST AVAILABLE COPY