35

An Expert System for Oncology Protocol Management

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

This chapter describes an oncology protocol management system, named ONCOCIN after its domain of expertise (cancer therapy) and its historical debt to MYCIN. The program is actually a set of interrelated subsystems, the primary ones being:[1]

1. the *Reasoner*, a rule-based expert consultant that is the core of the system; and
2. the *Interviewer*, an interface program that controls a high-speed terminal and the interaction with the physicians using the system.

The Interviewer is described in some detail in Chapter 32. This chapter describes the problem domain and the representation and control techniques used by the Reasoner. We also contrast ONCOCIN with EMYCIN

[1]Each program runs in a separate fork under the TENEX or TOPS-20 operating systems, thereby approximating a parallel processing system architecture. Another program, the *Interactor*, handles interprocess communication. There is also a process that provides background utility operations such as file backup. This chapter does not describe these aspects of the system design or their implementation. Details are available elsewhere (Gerring et al., 1982).

(Chapter 15), explaining why the EMYCIN formalism was inadequate for our purposes, even though it did strongly influence the system's rule-based design.

35.1   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 complex 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 a given tumor must meet certain eligibility criteria before they are accepted for treatment on the protocol; ineligible patients are treated in accordance with the best state-of-the-art therapy and are therefore not part of a formal clinical experiment.[2] Patients accepted for protocol treatment, on the other hand, 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 alternate 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 alternate therapies being compared and the data that need to be collected. Therapies may require as many as eight to ten drugs, given simultaneously or in sequence, continuously or intermittently. In addition, pharmacologic therapy may be combined with appropriate surgery or radiation therapy. 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 (any one of which may be used to guide treatment of the patients under the care of a single physician). Although an effort is made to have the documents available in the oncology clinics when patients are being treated for their

---

[2]Unfortunately, for many tumors the best state-of-the-art therapy may cause intolerable toxicity or be only partially effective. That is why there is a constant search for improved therapeutic plans.

tumors, it is often the case that a busy clinic schedule, coupled with a complex protocol description, leads a physician to rely on memory when deciding on drug doses and laboratory tests. 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. In some cases, patients suffer undue toxicity or are undertreated simply because protocol details cannot be remembered, located, or are not explicitly defined.

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.[3] ONCOCIN is superficially similar to some of the developing systems, but both its short- and long-term goals are unique in ways we describe below. 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.

35.2   Research Objectives

The overall goals of the ONCOCIN project are

1. to demonstrate that a rule-based consultation system with explanation capabilities can be usefully applied and can gain acceptance in a busy clinical environment;
2. to improve the tools currently available, and to develop new tools, for building knowledge-based expert systems for medical consultation; and
3. to establish both an effective relationship with a specific group of physicians and a scientific foundation, which will together facilitate future research and implementation of computer-based tools for clinical decision making.

---

[3]A memo from the M.I.T. Laboratory for Computer Science (Szolovits, 1979) describes a collaboration between M.I.T. and oncologists who have been building a protocol management system at Boston University (Horwitz et al., 1980). 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 differ from those of ONCOCIN.

Hence ONCOCIN's research aims have two parallel thrusts: to perform research into the basic scientific issues of applied artificial intelligence, and to develop a clinically useful oncology consultation tool. The AI component of the work emphasizes the following:

1. the implementation and evaluation of recently developed techniques designed to make computer technology more natural and acceptable to physicians;
2. extension of the methods of rule-based consultation systems so that they can interact with a large data base of time-oriented clinical information;
3. the design of a generalized control structure, separate from the domain knowledge, with the hope that the general system can be usefully applied in other problem areas with similar tasks;
4. continuation of basic research into mechanisms for making decisions based on data trends over time;
5. the design of a rapid, congenial interface that can bring a high-performance AI system to a group of users who are not experienced with AI or with computers in general; and
6. the development of techniques for assessing knowledge base completeness and consistency (see Chapter 8).

35.3   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 Hodgkin's disease and the non-Hodgkin's lymphomas.[4] In generating its recommendation, the system uses initial data about the patient's diagnosis, results of current laboratory tests, plus the protocol-specific information in its knowledge base. As information is acquired, it is stored on-line in files associated with the patient.

After examining a patient, the physician uses a video display terminal to interact with ONCOCIN's data-acquisition program (the Interviewer;

---

[4]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 hopeful 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.

STATIC KNOWLEDGE   DYNAMIC DATA

```
┌──────────────┐   ┌──────────────┐        ┌──────────────┐
│   General    │   │  Protocol-   │        │  Data Base   │
│  Strategies  │   │              │        │     of       │
│     of       │   │  Specific    │        │  Oncology    │
│  Oncology    │   │  Knowledge   │        │  Patients    │
│ Chemotherapy │   │              │        │              │
└──────────────┘   └──────────────┘        └──────────────┘
```

```
┌─────────────────────────────────────────────────────────────┐
│  ┌──────────────┐   ┌────────────┐     ┌──────────────┐      │
│  │   Reasoner   │   │ Interactor │     │  Interviewer │      │
│  │              │◄─►│            │◄───►│              │      │
│  │  (Interlisp) │   │            │     │   (SAIL)     │      │
│  └──────────────┘   └────────────┘     └──────────────┘      │
└─────────────────────────────────────────────────────────────┘
```

```
                                          ┌──────────────┐
                                          │    Clinic    │
                                          │  Physician   │
                                          └──────────────┘
```
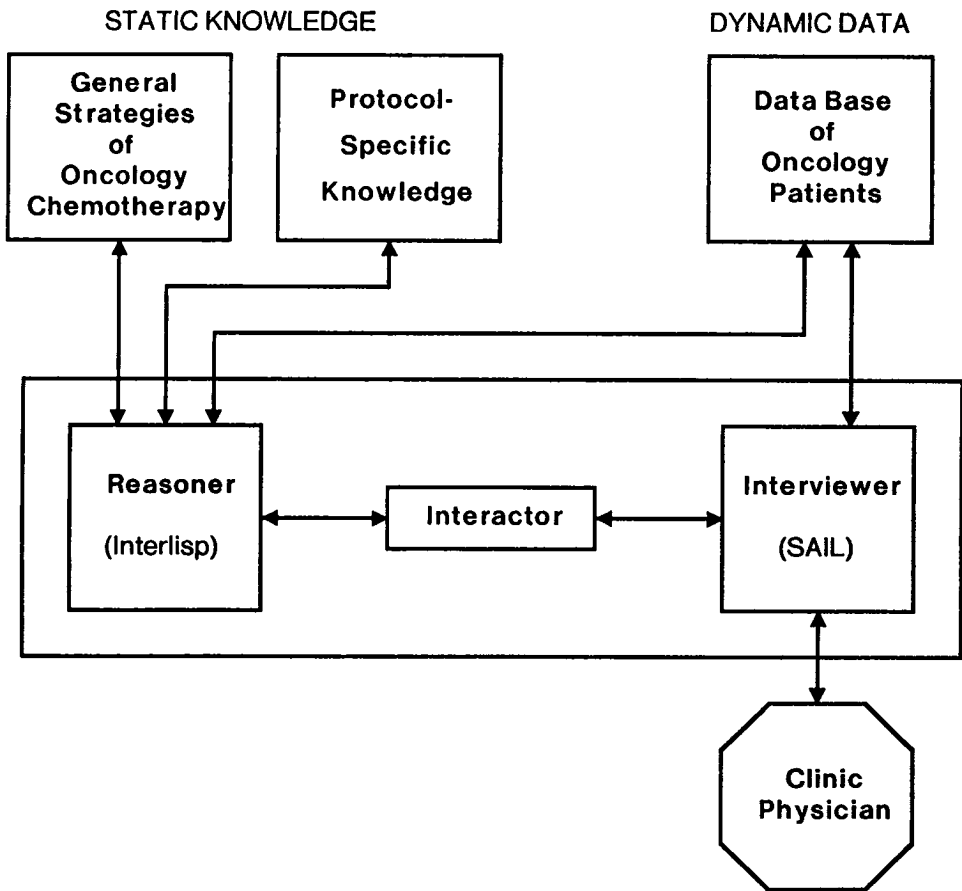
FIGURE 35-1   Overview of ONCOCIN.

see Chapter 32), 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, of appropriate therapy and tests. The Reasoner and Interviewer are linked with one another as shown in Figure 35-1. Each is able to use a data base of prior patient data. In addition, the Reasoner has access to information regarding the execution of chemotherapy protocols (control blocks) and specific information (rules) about the chemotherapy being used to treat the patient. Before terminating an interaction, the physician can examine the explanation provided with each recommendation.[5] The physician may approve

---

[5]We have chosen a representation that had also facilitated early work to allow ONCOCIN to offer a justification for any intermediary conclusions that the system made in deriving the advice (Langlotz and Shortliffe, 1983).

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.

35.4   The Reasoner

35.4.1   Why Not EMYCIN?

ONCOCIN's Reasoner communicates with the Interviewer during a consultation. Although EMYCIN's interactive routines provided a means for us to develop a prototype system quickly, the need to interact eventually with a specialized interface program is one of several reasons that we chose to build most of ONCOCIN from scratch rather than to implement it as a new EMYCIN system (Chapter 15). 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 patients at intervals typically spread over many months. Each clinic visit is a new data point, and conventional EMYCIN context trees and case data tables do not easily accommodate multiple measurements of the same attribute over time.

2. Expert-level advice from ONCOCIN also requires inference rules based on assessment of temporal trends for a given parameter.[6] Because EMYCIN assumes that a consultation is to be given at a single point in time, it does not provide a mechanism for assessing trends or interacting with a data bank of past information on a case.

3. ONCOCIN does not require many of the capabilities provided by EMYCIN. For example, the simplified interaction mediated through the Interviewer allows questions to be answered directly without dealing with the complexities of natural language understanding.

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

---

[6]This same point led to the development of Fagan's VM system (Chapter 22), a rule-based program that was influenced by EMYCIN but differed 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.

35.4.2   Representation

Knowledge about the oncology domain is represented using five main data structures: contexts, parameters, data blocks, rules, and control blocks.[7] In addition, we use a high-level description of each of these structures to serve as a template for guiding knowledge acquisition during the definition of individual instances.[8]

Contexts represent concepts or entities of the domain about which the system needs static knowledge. 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 chemotherapeutic plans. This description serves to focus the system's recommendation process.

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 from 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 normal mechanism fails to produce a value, or the user may be asked to provide the answer as a last resort.[9]

Data blocks define logical groupings of related parameters (e.g., initial patient data or laboratory test results). A data block directs the system to treat related parameters as a unit when requesting their values from the Interviewer, storing the values on a patient's file, or retrieving previously stored values.

Rules are the familiar productions used in MYCIN 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

---

[7]There are a few additional data structures designed to coordinate the interaction between the Reasoner and the Interviewer.

[8]The knowledge base editor is based on the similar programs designed and implemented for EMYCIN. A graphics editor has also been developed for use on the LISP machine workstations to which we intend to transfer ONCOCIN (Tsuji and Shortliffe, 1983).

[9]This "pure" description of ONCOCIN's technique for assigning values to parameters is actually further complicated by the free-form data entry allowed in the Interviewer. The details of how this is handled, and the corresponding relationship to control blocks, will not be described here.

value or a default value as defined above. The rules are categorized by the context in which they apply.

As in EMYCIN systems, rules are represented in a stylized format so that they may be translated from Interlisp into English for explanation purposes.[10] 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 (Chapter 8).

Below are the English translations of two ONCOCIN rules. Note that Rule 78 provides a default value for the parameter "attenuated dose."[11]

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 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.

RULE078

After trying all other methods to determine the current attenuated dose for all drugs:

IF: The blood counts do warrant dose attenuation
THEN: Conclude that the current attenuated dose is the previous dose attenuated by the minimum
        of the dose attenuation due to low WBC and the dose attenuation due to low platelets.

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). Note that this data structure allows us to separate control descriptions explicitly from decision rules, a distinction that was often unclear in EMYCIN systems. 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:

> *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.
>
> *DOSE*
> To calculate the correct dosage of the drug:
> > 1) Determine the current attenuated dose.
> > 2) Determine the units in which the drug should be measured.
> > 3) Determine the maximum allowable dose of the drug.
> > 4) Determine the route of administration.
> > 5) Determine the number of days for which the drug should be given.
> > 6) Compute the dose based upon body surface area.

---

[10]In keeping with the philosophy reflected in other systems we have designed, ONCOCIN is able to produce natural language explanations for its recommendations. See also the critiquing work of Langlotz and Shortliffe (1983).

[11]PAVe and MOPP are acronyms for two of the drug combinations used to treat Hodgkins' disease.

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

35.4.3   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 the following:

1. *Fetching a data block,* 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 by 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. In addition, 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. Figure 35-2 shows the goal-oriented procedure used in EMYCIN. All invocation of rules results 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 maintained 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
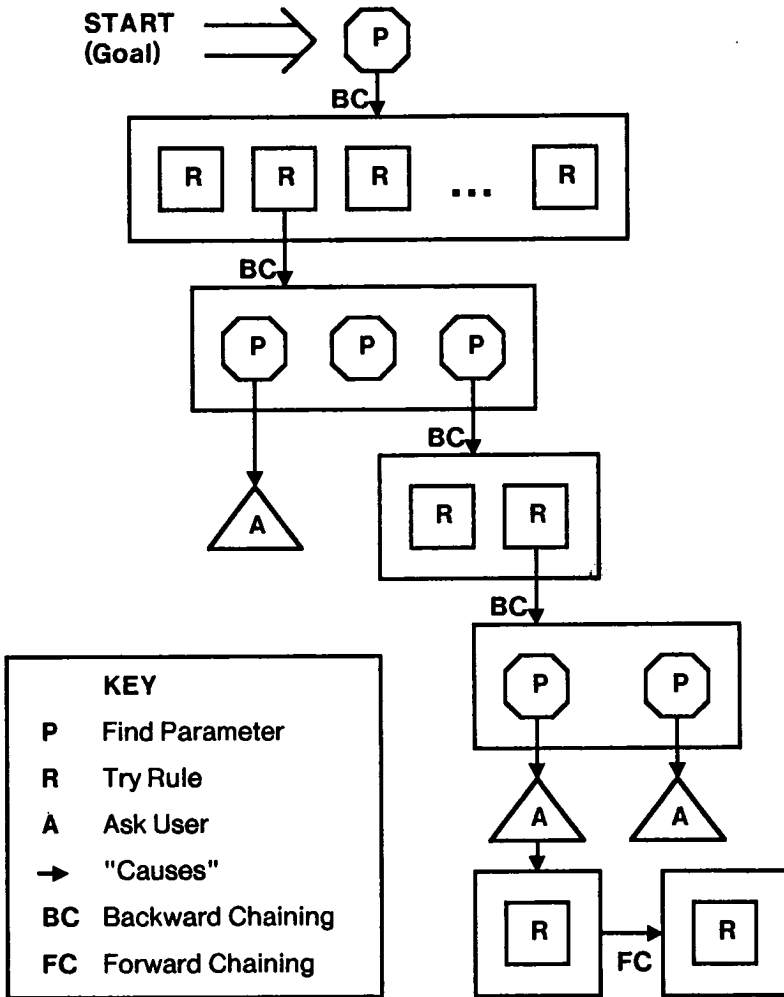
FIGURE 35-2   Control in EMYCIN.

one of the system's forward-chained rules. These rules can only be used as antecedent rules, they typically have single conditions in their premises, and repeated forward chaining is permitted only if one rule concludes with certainty that the premise of another is true.

In ONCOCIN (Figure 35-3), on the other hand, initial control is derived from the control block invoked in response to the task selected by the user. Forward chaining and backward chaining of rules are intermingled,[12] and any rule can be used in either direction.

---

[12]The broken line in Figure 35-3 outlines the portion of the ONCOCIN control structure that is identical to that found in EMYCIN (Figure 35-2).

START ⟹ **CB**
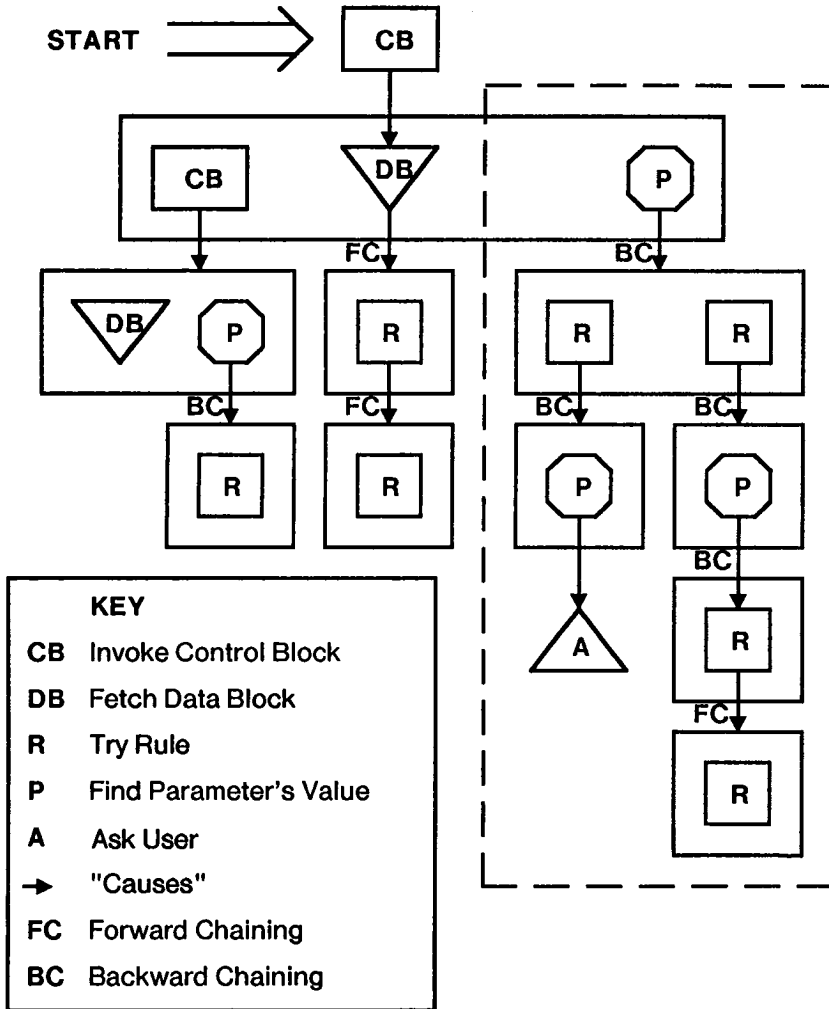
KEY

CB Invoke Control Block

DB Fetch Data Block

R   Try Rule

P   Find Parameter's Value

A   Ask User

➡   "Causes"

FC Forward Chaining

BC Backward Chaining

FIGURE 35-3   Control in ONCOCIN.

35.5   Why Artificial Intelligence Techniques?

We have learned from the MYCIN experience, and in building other EMY-CIN 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.[13] 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 following form:

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

> 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. One goal is to develop approaches to these more complex problems that characterize the management of patients being treated for cancer. It is when these issues are addressed that the need for AI techniques is most evident and the task domain begins to look similar in complexity to the decision problems in a system like MYCIN. Rules will eventually have uncertainty associated with them (we have thus far avoided the need for certainty weights in the rules in ONCOCIN), and close collaboration with experts has been required in writing new rules that are *not* currently recorded in chemotherapy protocols or elsewhere. In addition, however, AI representation and control techniques have already allowed 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 (Langlotz and Shortliffe, 1983; Tsuji and Shortliffe, 1983).

---

[13]Because we need a high-speed interface to ensure the system's acceptance by physicians, we have been forced to design a complex system architecture with asynchronous processes. We have also wanted to allow each process to run in whatever computer language seems most appropriate for its task. ONCOCIN subprocesses are currently written in Interlisp, SAIL, and assembler (Gerring et al., 1982). We have not described the total system or our reasons for making these design decisions, but we believe the structure is necessary to achieve acceptance of the system in a clinical setting.

35.6 Conclusion

In summary, 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 a novel interface that uses both custom hardware and efficient software has heightened the acceptability of ONCOCIN. Formal evaluations are underway to allow us to determine both the effectiveness and the acceptability of the system's clinical advice.

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