Report 82-36
Stanford -- KSL

Explanatory Power for Medical Expert Systems: Studies in the Representation of Causal Relationships for Clinical Consultations. J.W. Wallis, Edward H. Shortliffe, 1982

card 1 of 1

Heuristic Programming Project
Report No. HPP 82-36

1982

Explanatory Power for Medical Expert Systems: Studies in the Representation of Causal Relationships for Clinical Consultations

J. W. Wallis, and E. H. Shortliffe

Published in Methods of Information in Medicine 21 (3/1982),
F.K. Schattauer Verlag, Stuttgart - New York.

Reprinted with permission.

Meth. Inform. Med. *21* (1982) 127—136

Explanatory Power for Medical Expert Systems: Studies in the Representation of Causal Relationships for Clinical Consultations[*]

(From the Heuristic Programming Project, Departments of Medicine and Computer Science, Stanford University, Stanford, California)

J. W. Wallis, E. H. Shortliffe

This paper reports on experiments designed to identify and implement mechanisms for enhancing the explanation capabilities of reasoning programs for medical consultation. The goals of an explanation system are discussed, as is the additional knowledge needed to meet these goals in a medical domain. We have focussed on the generation of explanations that are appropriate for different types of system users. This task requires a knowledge of what is *complex* and what is *important*; it is further strengthened by a classification of the associations or causal mechanisms inherent in the inference rules. A causal representation can also be used to aid in refining a comprehensive knowledge base so that the reasoning and explanations are more adequate. We describe a prototype system which reasons from causal inference rules and generates explanations that are appropriate for the user.

Key-Words: Medical Decision Making, Consultation Systems, Explanation, Artificial Intelligence, Expert Systems

AUSSAGEKRAFT MEDIZINISCHER BERATUNGSSYSTEME: UNTERSUCHUNGEN ÜBER DIE DARSTELLUNG URSÄCHLICHER BEZIEHUNGEN FÜR KLINISCHE KONSULTATIONEN

Diese Arbeit berichtet über Experimente, die dazu dienten, Methoden zur Verbesserung der Aussagekraft von Grundlagenprogrammen für medizinische Beratungen zu erkennen und anzuwenden. Diese Programme benutzen eine Codierung der medizinischen Information in einem Spezialbereich, um mittels logischer Folgerungen die möglichen Ursachen eingetragener Symptome auszugeben. Die Ziele des Erklärungssystems werden im Zusammenhang mit dem zusätzlichen Wissen, das zur Lösung dieser Aufgabe im medizinischen Bereich benötigt wird, diskutiert. Wir haben uns darauf konzentriert, solche Erklärungen zu erzeugen, die für die unterschiedlichen Systembenutzer geeignet sind. Diese Aufgabe benötigt eine Kenntnis davon, was komplex und was wichtig ist in den Beratungen, und wird von einer Klassifizierung der ursächlichen Verbindungen, welche den Regeln für die Folgerungen unterliegen, unterstützt. Das Folgerungsnetz kann auch benutzt werden, um die Entwicklung eines Systems für die Erkennung von Zusammenhängen zu unterstützen, kann aber nicht genug Informationen enthalten, um neue Folgerungen aus dem bestehenden Wissen abzuleiten. Wir beschreiben den Prototyp eines solchen Systems, das aus dem Folgerungsnetz Schlüsse ableitet und dem Benutzer geeignete Erklärungen darbietet.

Schlüssel-Wörter: Medizinische Entscheidungsfindung, Konsultationssystem, Erklärung, künstliche Intelligenz, Experten-System

Introduction

Computer science research devoted to the development of consultation programs has become known as »expert systems research« or »knowledge engineering« [7]. Much of the work is relevant to the design of clinical decision making programs [19]. For example, researchers in the development of expert systems have increasingly recognized the importance of explanation capabilities in encouraging the acceptance of their programs, an area that is also critical in medical consultation system development [9, 22].

Good explanations serve four functions in a consultation system: (1) they provide a method for examining the program's reasoning if errors arise when the system is being built; (2) they assure users that the reasoning is logical, thereby increasing user acceptance of the system; (3) they may persuade users that unexpected advice is appropriate; and (4) they can educate users in areas where their knowledge may be weak. These diverse roles impose several requirements upon the system. For example, the explanations must adequately represent the reasoning processes of the program, and they should allow the user to examine the reasoning history or underlying knowledge at various levels of detail. In addition, although the program's approach to a problem need not be identical to an expert's approach, the program's overall strategy and reasoning steps must be understandable and seem logical, regardless of the user's level of expertise. This means that the system must have the capability to tailor its explanations to the varying needs and characteristics of its users.

In this paper we describe experiments in the design and implementation of a prototype explanation program. After briefly describing previous work in the development of explanation capabilities for consultation programs, we introduce the representation techniques used in our experimental system. The program's explanation capabilities are then described. Subsequent sections of the paper discuss the nature of causal reasoning in expert systems and its relation to explanation. We also suggest a useful scheme for classifying commonly used inference rules.

[*] This work was supported by Contract NR 049—479 from the Office of Naval Research and by a grant from the Henry J. Kaiser Family Foundation. Dr. Shortliffe is recipient of research career development award LM00048 from the National Library of Medicine. The computing was undertaken on the SUMEX-AIM machine, a shared national resource funded by the Biotechnology Resources Program of the NIH under grant RR-00785.

Previous Work

Our past work in explanation for consultation systems has dealt primarily with the ability to cite the production rule [4] involved in a particular decision. One example of this approach is the explanation system for MYCIN, our rule-based program to assist in the selection of antimicrobial therapy for patients with bacteremia or meningitis [17, 21]. This program is able to answer questions about *how* it has reached a particular conclusion (i.e., what rules led to the pertinent inference) and about *why* it has asked a particular question (i.e., which rules can use the requested information). The capability can be used for a specific run of the program or for general querying of the knowledge base.

MYCIN's explanation capability is illustrated in Fig. 1. Although the program's responses provide an accurate description of a portion of its reasoning, to understand the overall reasoning scheme a user needs to request a display of *all* the rules that are used. Additionally, rules such as those mentioned in Fig. 1 are largely designed for efficiency and therefore frequently omit underlying causal meachanisms that are known to experts but may be necessary for a novice to understand a decision. The rule guiding the choice of carbenicillin with an aminoglycoside, for example, does not mention the synergism of the two drugs when combined in the treatment of serious *pseudomonas aeruginosa* infections. Finally, while MYCIN does have a limited sense of discourse (viz., an ability to modify responses based on the topic under discussion), its explanations are customized to neither the questioner's objectives nor characteristics.

MYCIN's explanation capabilities were expanded by Clancey in his work on the tutorial system named GUIDON [2]. In order to use MYCIN's knowledge base and patient cases for tutorial purposes, Clancey found it necessary to incorporate knowledge about teaching. This knowledge, expressed as »tutorial rules«, and a four-tiered measure of the baseline knowledge of the student (»beginner«, »advanced«, »practitioner«, or »expert«), have enhanced the abilitiy of a student to learn efficiently from MYCIN's knowledge base. Clancey has also noted problems arising from the frequent lack of underlying »support« knowledge which is needed to explain the relevance and utility of a domain rule [3].

More recently, Swartout has developed a system that generates explanations from a record of the development decisions made during the writing of a consultation program to advise on digitalis dosing [23]. The domain expert provides information to a »Writer« subprogram, which in turn constructs the advising system. The traces left by the writer, a set of domain principles, and a domain model are utilized to produce explanations. Thus both the knowledge acquisition process and automated programming techniques are intrinsic to the explanations generated by Swartout's system. Responses to questions are customized for different kinds of users by keeping track of what class is likely to be interested in a given piece of code.

Whereas MYCIN generates explanations that are usually based on a single rule*), Weiner has described a system named Blah [25] that can summarize an entire reasoning chain in a single explanatory statement. The approach developed for Blah was based on a series of psycholinguistic studies [11, 12, 26] that analyzed the ways in which human beings explain decisions, choices, and plans to one another. For example, Blah structures an explanation so that the differences between alternatives are given before the similarities (a practice that was noted during the analysis of human explanations).

```
**HOW DID YOU DECIDE TO TREAT FOR
  PSEUDOMONAS?
ITEM—2 is Pseudomonas aeruginosa associated with
  meningitis. I decided to treat ITEM—2 on the basis of
  clinical evidence only.

The following rules concluded about:
      the likelihood that Pseudomonas aeruginosa is one
  of the organisms which might be causing the infection

Rule        Cumulative        last question asked
            Certainty     before conclusion was made

RULE 545      (.29)                 46
RULE 557      (.64)                 48
RULE 578      (.82)                 50

**PRINT 545

If:     1) The infection which requires therapy is mening-
           itis,
        2) A: A smear of the culture was not examined, or
           B: Organisms were not seen on the stain of the
              culture,
        3) The type of the infection is bacterial,
        4) The patient has not had neurosurgery,
        5) The age of the patient is greater than 10 days,
           and
        6) The infection was acquired while the patient was
           hospitalized
Then:      There is evidence that the organisms (other than
           those seen on cultures or smears) which might be
           causing the infection are e. coli (.75), staphy-
           lococcus coag-pos (.3), pseudomonas aeruginosa
           (.3) or klebsiella pneumoniae (.5)

**DO YOU TREAT PSEUDOMONAS AERUGIN-
  OSA BACTEREMIA WITH CARBENICILLIN?
Yes, for treatment of PSEUDOMONAS AERUGINOSA
  in BACTEREMIA, the drugs of choice (ranked by rela-
  tive efficacy, assuming sensitivity) are:

  1st  choice:
       CARBENICILLIN-AND-TOBRAMYCIN
       CARBENICILLIN-AND-GENTAMICIN
       AMIKACIN-AND-CARBENICILLIN
  2nd  choice:
       TOBRAMYCIN
       GENTAMICIN
       AMIKACIN
```

Fig. 1: An example of an interaction with MYCIN's explanation program. User input is in capital letters and follows a double asterisk. Note that the rule for selecting a drug to cover *pseudomonas* bacteremia is adequate for allowing MYCIN to reach the correct conclusion but that the underlying reason for combining two drugs is unclear.

The tasks of interpreting questions and generating explanations are confounded by the problems inherent in natural language understanding and text generation. A consultation program must be able to distinguish general questions from case-specific ones, and questions relating to

*) Although MYCIN's »WHY« command has a limited ability to integrate several rules into a single explanation [20], the user wishing a high level summary must specifically augment the »WHY« with a number that indicates the level of detail desired. We have found that the feature is therefore seldom used. It would, of course, be preferable if the system »knew« on its own when such a summary were appropriate.

specific reasoning steps from those involving the overall reasoning strategy. As previously mentioned, it is also important to tailor the explanation to the user, giving appropriate supporting causal and empiric relationships. It is to this last task that the research presented in this paper is aimed. We have avoided problems of natural language understanding for the present, concentrating instead on representation and control mechanisms that permit the generation of explanations customized to the knowledge and experience of either physician or student users.

Design Considerations: The User Model

For a system to produce customized explanations, it must be able to model the user's knowledge and motivation for using the system. At the simplest level, such a model can be represented by a single measure of what the user knows in this domain, and how much he wants to knew (i.e., to what level of detail he wishes to have things explained). One approach is to record a single rating of a user's *expertise*, similar to the four categories mentioned above for GUIDON. The model could be extended to permit the program to distinguish subareas of a user's expertise in different portions of the knowledge base. For example, the measures could be dynamically updated as the program responds to questions and explains segments of its knowledge. If the user demonstrates familiarity with one portion of the knowledge base, then he probably also knows about related portions (e.g., if a physician is familiar with the detailed biochemistry of one part of the endocrine system, it is likely he knows the biochemistry of other parts of the endocrine system as well). This information can be represented in a manner similar to GOLDSTEIN's rule pointers, which link analogous rules, rule specializations, and rule refinements [8]. In addition, the model should ideally incorporate a sense of dialogue to facilitate user interactions. Finally it must be self-correcting (e.g., if the user unexpectedly requests information on a topic that the program had assumed he knew, it should correct its model prior to giving the explanation). In our recent experiments we have concentrated on the ability to give an explanation appropriate to the user's level of knowledge and have deemphasized dialogue or model correction.

Knowledge Representation

Form of a Conceptual Network

We have found it useful to describe the knowledge representation for our prototype system in terms of a semantic network (Fig. 2)*. It is similar to other network representations

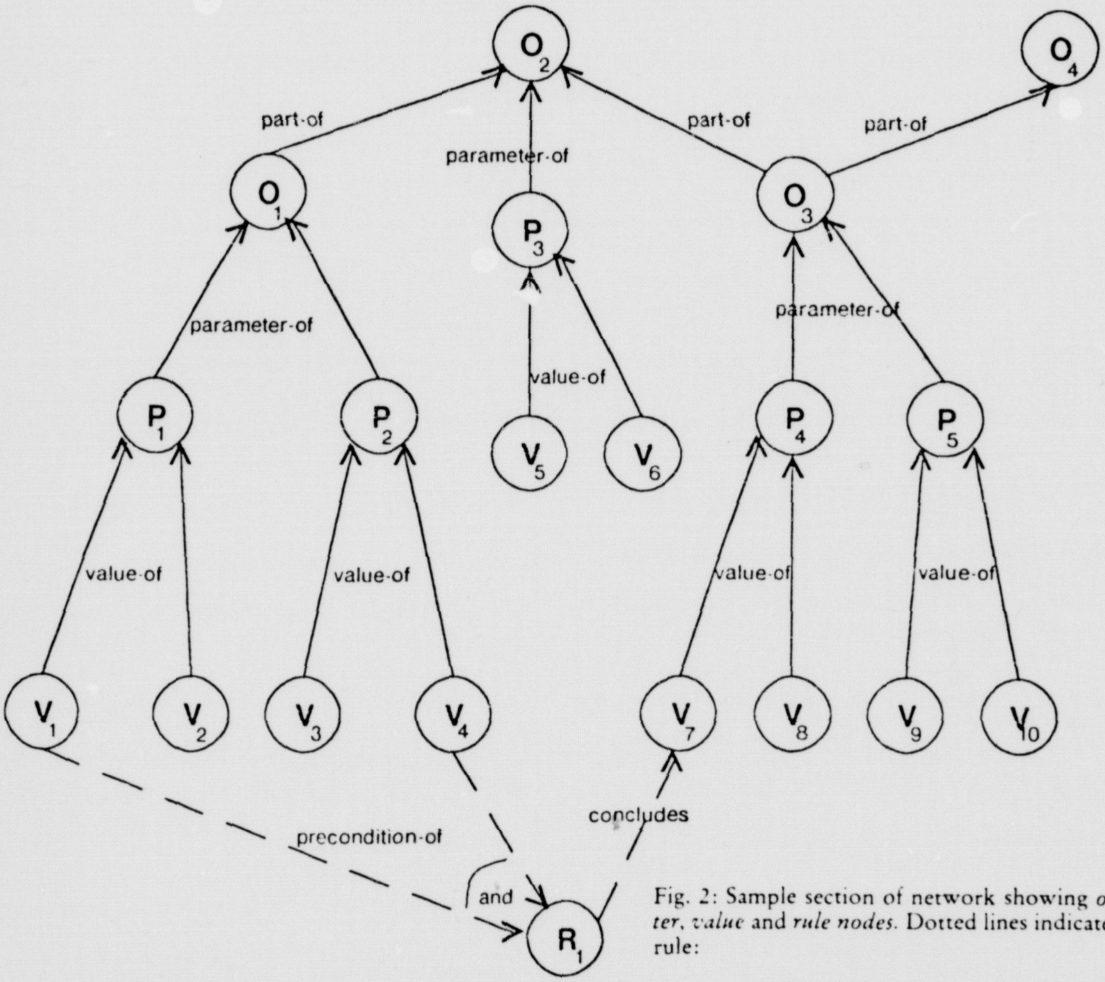

Fig. 2: Sample section of network showing *object, parameter, value* and *rule nodes*. Dotted lines indicate the following rule:

```
IF:     PARAMETER—1 of OBJECT—1
            is VALUE—1, and
        PARAMETER—2 of OBJECT—1 is VALUE—4
THEN:   Conclude that PARAMETER—4 of OBJECT—3
            is VALUE—7
```

*) The descriptive power of a semantic network provides clarity when describing this work. However, other representation techniques used in artificial intelligence research could also have captured the attributes of our prototype system.

used in the development of expert systems [6, 27] and has been influenced by RIEGER's work on the representation and use of causal relationships [16]. A network provides a particularly rich structure for entering detailed relationships and descriptors in the domain model. *Object nodes* are arranged hierarchically, with links to the possible attributes *(parameters)* associated with that object. The *parameter nodes*, in turn, are linked to the possible *value nodes*, and *rules* are themselves represented as nodes with links that connect *value nodes*. These relationships are summarized in Table 1.

Table 1: Relationships among object nodes, parameter nodes, value nodes and rule nodes.

Type of Node	Static Information (Associated with Node)	Dynamic Information (Consultation-Specific)
Object node	part of link (hierarchic) parameter list	
Parameter node	object link value-node list default value text definition	
Value node	parameter-node link precondition-rule list conclusion-rule list importance complexity ask first/last	contexts for which this value is true certainty factor explanation data ask state
Rule node	precondition list (Boolean) conclusion certainty factor rule type complexity text justification	explanation data

The *certainty factor* (CF) associated with value and rule nodes (Table 1) refers to the belief model developed for the MYCIN system [18]. A CF of +1 associated with a value indicates that it is known to be true in a given context (e.g., for a specific patient in a given consultation); similiarly −1 designates a value known to be false. There is a continuous range of intermediate values, with CF = 0 indicating the indifferent state. Measures of certainty are propagated from premises to conclusions using a combining function [18] which considers both the belief in the value of the relevant parameters and the CF for the inference rule (a static measure of the rule's inference strength on the same −1 to +1 scale).

Ask first/last (Table 1) is a property that controls whether the value of a parameter is to be requested from the user before an attempt is made to compute it using inference rules from the knowledge base. The *text justification* of a rule is provided when the system builder has decided not to break the reasoning step into further component parts but wishes to provide a brief summary of the knowledge underlying that rule. *Complexity, importance*, and *rule type* are described in more detail below.

Rules and Their Use

In the network (Fig. 2), rules connect value nodes with other value nodes. This contrasts with the MYCIN system in which rules are functionally associated with an object-parameter pair and succeed or fail only after completion of an exhaustive search for *all* possible values associated with this pair. To make this clear, consider a rule of the form:

If: DISEASE-STATE of the LIVER is ALCOHOLIC CIRRHOSIS
Then: It is likely (.7) that the SIZE of ESOPHAGEAL VEINS is INCREASED.

When evaluating the premise (if-condition) of this rule to decide whether it applies in a specific case, a MYCIN-like system would attempt to determine the certainty of *all* possible values of the DISEASE-STATE of the LIVER, producing a list of values and their associated certainty factors. Our experimental system, on the other hand, would only investigate rules that could contribute information specifically about ALCOHOLIC CIRRHOSIS. In either case, however, rules are chained together through a mechanism that is goal-oriented and known as »backward chaining«.

Because our prototype system reasons backwards from single values rather than from parameters, it saves time in reasoning in most cases. However, there are occasions when this approach is not sufficient. For example, if a value is concluded with absolute certainty (CF = 1) for a parameter with a mutually exclusive set of values, this necessarily forces the other values to be false (CF = −1). Lines of reasoning that result in conclusions of absolute certainty (i.e., reasoning chains in which all rules make conclusions with CF = 1) have been termed »unity paths« [20]. In cases with mutually exclusive values of parameters, complete investigation of one value requires consideration of any other value that could be reached by a unity path. Thus the representation must allow quick access to such paths.

When reasoning by elimination, similiar problems arise if a system focuses on a single value. One needs the ability to conclude a value by ruling out all other possible values for that parameter; this entails a slight modification of the organizational and reasoning scheme. One strategy is to use this elimination method in cases of mutually exclusive options only after the normal backward chaining process fails (provided that the possibilities represented in the knowledge base are known to span *all* potential values).

Complexity and Importance

The design considerations for adequate explanations require additions to the representation scheme described above. To provide customized explanations, appropriate for different levels of expertise, we have found it useful to associate a measure of *complexity*, both with the inference rules and with the concepts about which they are concluding. Because some concepts are key ideas in a reasoning chain and should be mentioned regardless of their *complexity*, a measure of *importance* associated with concepts is useful as well. Both measures are presently specified at the time knowledge is added to the system, but a dynamic modification of these initial values would improve the flexibility of the approach.

Although *complexity* and *importance* are related, one cannot necessarily be predicted from the other. For example, biochemical details of the endocrine system are *complex*, but are not *important* to an understanding of endocrine abnormalities, yet the same *complexity* of biochemical detail is *important* for understanding the glycogen storage diseases. A measure of a fact's *importance* was also used by CARBONELL in the form of »relevancy tags«, supplemented by »distance« in a semantic network [1], but he did not distinguish between the two concepts discussed here.

Explanation Capabilities

Tailored Explanations

The measurements of *complexity* and *importance* described above facilitate the generation of tailored explanations. Consider a linear causal chain representing a simplified causal mechanism for the existence of kidney stones (Fig. 3). A sample explanation dialogue based on this reasoning chain might be as follows*:

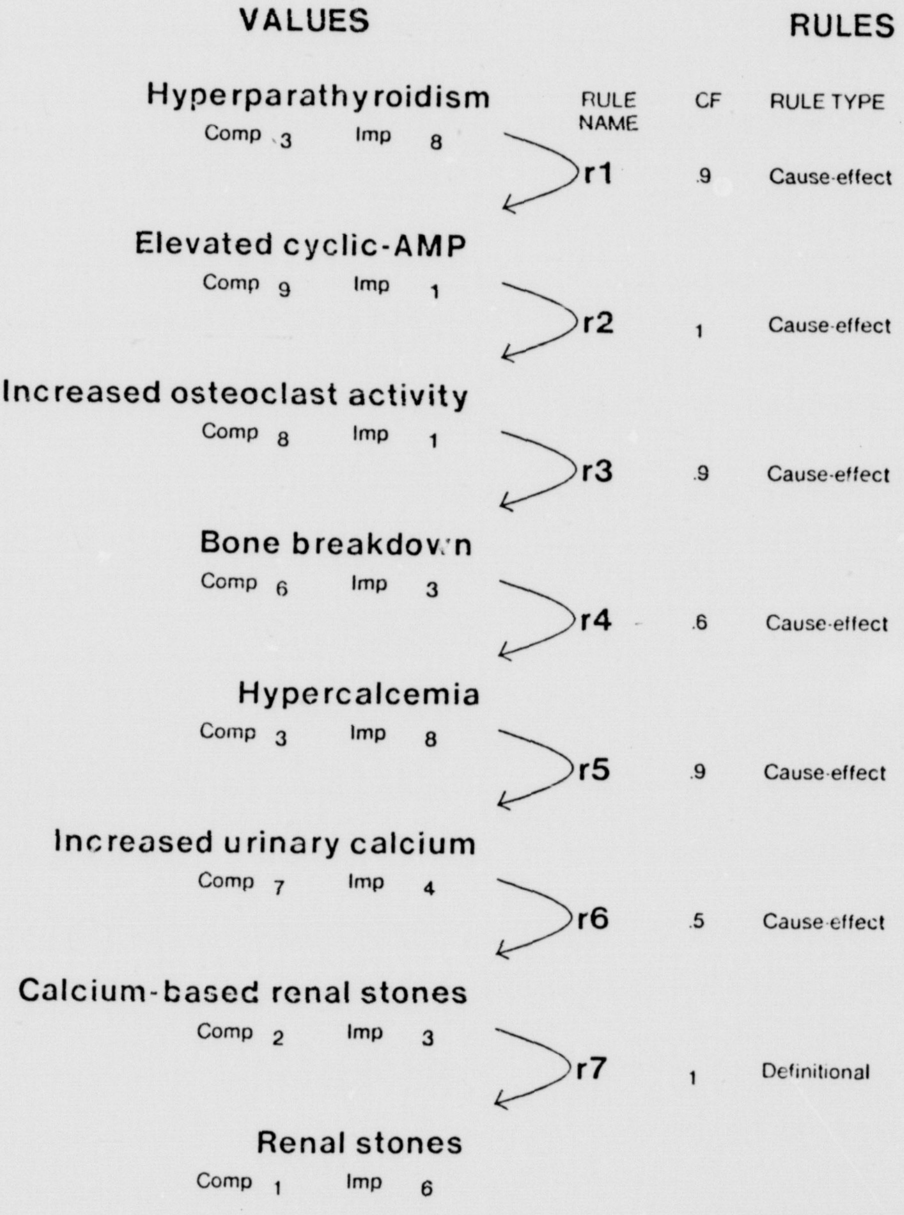

Fig. 3: An example of a small section of a causal knowledge base, with the *complexity* (Comp) and *importance* (Imp) illustrated for the value nodes (concepts). This is a highly simplified causal chain provided for illustrative purposes only. For example, the effect of parathormone on the kidney (promoting retention of calcium) is not mentioned but would have an opposite causal impact on urinary calcium. This reasoning chain is linear (each value has only one cause) and contains only *cause-effect* and *definitional* rules. *Sample Interactions 1* and *2* (see text) are based on this reasoning chain.

*) Our program functions as shown except that the user input requires a constrained format rather than free text. We have simplified that interaction here for illustrative purposes. The program actually has no English interface.

Sample Interaction 1:

**EXPERTISE 3 [the user indicates his level of expertise on a scale of 1 to 10 where 10 = expert]

**HOW DOES HYPERPARATHYROIDISM CAUSE RENAL STONES?
Renal stones are caused by hypercalcemia
Hypercalcemia is caused by hyperparathyroidism

**MORE DETAIL
Renal stones are caused by increased urinary calcium
Increased urinary calcium is caused by hypercalcemia
Hypercalcemia is caused by increased bone breakdown
Increased bone breakdown is caused by hyperparathyroidism

**EXPERTISE 7 [the user indicates that he is more expert than above]

**HOW DOES HYPERPARATHYROIDISM LEAD TO INCREASED BONE BREAKDOWN?
Bone breakdown is caused by increased osteoclast activity
Increased osteoclast activity is caused by increased cyclic-AMP
Increased cyclic-AMP is caused by hyperparathyroidism

This sample dialogue demonstrates: (1) the user's ability to specify his expertise, (2) the program's ability to employ the user's expertise to adjust the amount of detail it offers, and (3) the user's option to request more detailed information about the topic under discussion.

Two user-specific variables are used to guide the generation of explanations*):

> *Expertise:* A number representing the user's current level of knowledge. As is discussed below, reasoning chains that involve simpler concepts as intermediates are collapsed to avoid the display of information that might be obvious to the user.
> *Detail:* A number representing the level of detail desired by the user when receiving explanations (by default a fixed increment added to the *expertise* measure). A series of steps that is excessively detailed can be collapsed into a single step to avoid flooding the user with information. However, if the user wants more detailed information, he can request it.

As was shown in Fig. 3, a measure of *complexity* is associated with each value node. Whenever an explanation is produced, the concepts in the reasoning chain are selected for exposition on the basis of their *complexity*; those concepts with *complexity* lying between the user's *expertise* level and the calculated *detail* level are used**). Consider, for example, the five-rule reasoning chain linking six concepts as shown in Fig. 4. When intermediate concepts lie outside the desired range (concepts B and E in this case), broader inference statements are generated to bridge the nodes that are appropriate for the discussion (e.g., the statement that A leads to C would be generated in Fig. 4). Terminal concepts in a chain are always mentioned, even if their *complexity* lies outside the desired range (as is true for concept F in the example). This approach preserves the logical flow of the explanation without introducing concepts of inappropriate complexity.

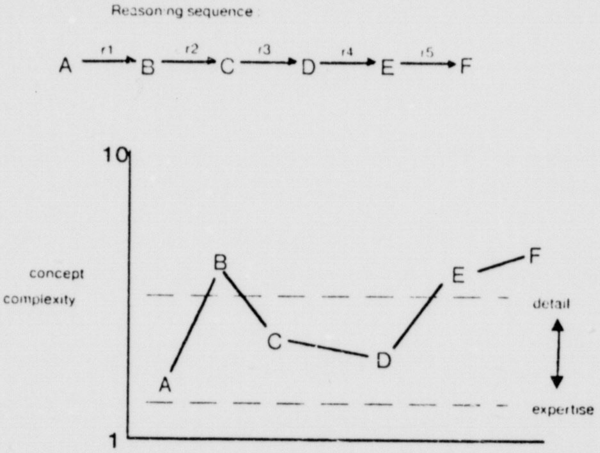

Fig. 4: Diagram showing the determination of which concepts (parameter values) to explain to a user with a given *expertise* and *detail* setting. The letters A through F represent the concepts (values of parameters) that are linked by the inference rules *r1* through *r5*. Only those concepts whose *complexity* falls in the range between the dashed lines (including the lines themselves) will be mentioned in an explanation dialogue. Explanatory rules to bridge the intermediate concepts lying outside this range are generated by the system.

We have also found it useful to associate a *complexity* measure with each inference rule to handle circumstances in which simple concepts (low *complexity*) are linked by a complicated rule (high *complexity*)*). This situation typically occurs when a detailed mechanism, one that explains the association between the premise and conclusion of a rule, consists of several intermediate concepts that the system builder has chosen not to encode explicitly**). When building a knowledge base, it is always necessary to limit the detail at which mechanisms are outlined, either because the precise mechanisms are unknown or because minute details of mechanism are not particularly useful for problem solving or explanation. Thus it is useful to add to the knowledge base a brief *text justification* (Table 1) of the mechanism underlying a rule.

Consider, for example, the case in Fig. 5 which corresponds to the same reasoning chain represented in Fig. 4. Although rule *r3* links two concepts (C and D) that are within the *complexity-detail* range for the user, the relation-

*) Another variable we have discussed but not implemented is a focusing parameter which would put a ceiling on the number of steps in the chain to trace when formulating an explanation. A highly focussed explanation would result in a discussion of only a small part of the reasoning tree. In such cases, it would be appropriate to increase the *detail* level as well.

**) The default value for *detail* in our system is the *expertise* measure incremented by 2. When the user requests more detail, the *detail* measure is incremented by 2 once again. Thus, for the three interchanges in *Sample Interaction 1*, the *expertise-detail* ranges are 3—5, 3—7, and 7—9 respectively. *Sample Interaction 2* (below) demonstrates how this scheme is modified by the *importance* measure for a concept.

*) The opposite situation does not occur; rules of low *complexity* do not link concepts of higher *complexity*.

**) PATIL has dealt with this problem by explicitly representing causal relationships about acid-base disorders at a variety of different levels of detail [13].

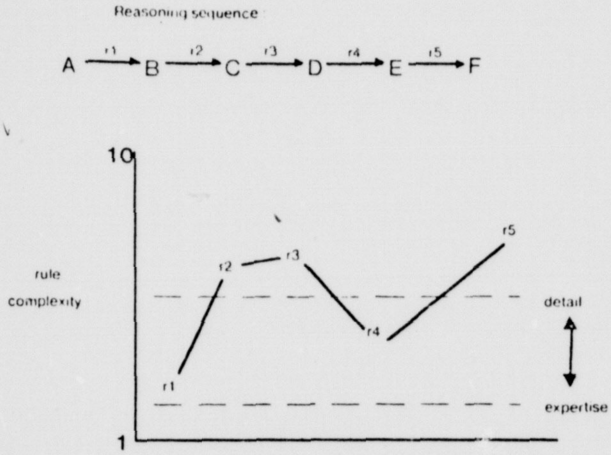

Fig. 5: Diagram showing the determination of which rules to explain further for a user with a given *expertise* and *detail* setting. When a rule is mentioned because of the associated concepts, but the rule itself is too *complex*, further text associated with the rule is displayed.

ship mentioned in rule *r3* is itself considered to be outside this range. When generating the explanation for this reasoning chain, the program mentions concepts C and D, and therefore mentions rule *r3* despite its *complexity* measure. Since the rule is considered too *complex* for the user, however, the additional explanatory text associated with the rule is needed in this case. If the rule had fallen within the *complexity-detail* range of the user, on the other hand, the *text justification* for the rule would not have been required*).

Further modulation of rule and concept selection is accomplished using the *importance* measure associated with parameters. A high *importance* forces the inclusion of a reasoning step in an explanation, thereby overriding the *complexity* considerations that were shown in Figs. 4 and 5. When the *importance* level of a concept is two or more points above the *expertise* of the user, the item is included in the explanation. Consider, e.g., the following dialogue which demonstrates the way in which the *importance* measure is used:

Sample Interaction 2:

```
**EXPERTISE 6
**HOW DOES HYPERPARATHYROIDISM CAUSE
  RENAL STONES?
  Renal stones are caused by increased urinary calcium
    Increased urinary calcium is caused by hypercalcemia
      Hypercalcemia is caused by increased bone break-
      down
        Bone breakdown is caused by increased osteoclast
        activity
          Increased osteoclast activity is caused by hyper-
          parathyroidism
```

Note that this example shows a response to the same question asked in *Sample Interaction 1*. This time, however, the *expertise* level is six rather than three. Hypercalcemia is therefore mentioned only because its *importance* level of eight (see Fig. 3) is two points higher than the *expertise* of the user; the *complexity* level of hypercalcemia does not fall within the *expertise-detail* range of the user and thus would not have been included in the explanation if it were not for its high *importance*. The other items mentioned are either terminal concepts in the chain (renal stones and hyperparathyroidism) or have a *complexity* measure lying within the user's *expertise-detail* range of six to eight.

Many reasoning chains are not as simple as those shown in Figs. 3—5. When explaining a branched reasoning chain, for example, the explanation system can set aside the branches of the chain and mention them only when it is appropriate to the level of detail required by the user. This feature provides users with an overview of the reasoning process to help them decide whether it is necessary to examine the more detailed steps. The capability is illustrated in the following dialogue which involves a patient with hypercalcemia, a possible malignancy, and prolonged bed rest:

Sample Interaction 3:

```
*WHY DOES THE PATIENT HAVE INCREASED
 SERUM CALCIUM?
 Increased serum calcium is suggested by immobilization
 and malignancy
**MORE DETAIL
  Increased serum calcium is implied by increased bone
  breakdown
    Increased bone breakdown is suggested by 2 paths of
    reasoning:
      Increased bone breakdown is implied by increased
      osteoclast activity
        Increased osteoclast activity is implied by pro-
        longed immobilization
      Increased bone breakdown is also implied by malig-
      nant bone invasion
```

Types of Rules

Our refinement of the rule types presented by Clancey [3] yields five types of rules*) that are relevant to explanation strategies:

definitional: the conclusion is a restatement of the precondition in different terms;
cause-effect: the conclusion follows from the precondition by some mechanism, the details of which may not be known;
associational: the conclusion and the precondition are related, but the causal direction (if any) is not known;
effect-cause: the presence of certain effects is used to conclude about a cause with some degree of certainty;
self-referencing: the current state of knowledge about a value is used to update that value further**).

The importance of distinquishing between *cause-effect* and *effect-cause* rules is shown in Fig. 6, which considers a simplified network concerning possible fetal Rh incompatibility in a pregnant patient. Reasoning backwards from the goal question »Is there a fetal problem?«, one traverses three steps that lead to the question of whether the parents are Rh incompatible; these three steps use *cause-effect* and *definitional* links only. However, in order to use the laboratory data concerning the amniotic fluid to form a conclusion about the presence of fetal hemolysis, *effect-cause* links must be used.

*) An example of this approach is included in *Sample Interaction 4* in the next section.

*) Rules considered here deal with domain knowledge, to be distinguished from strategic or meta-level rules [5].

**) In many cases these rules can be replaced by strategy rules (e.g., »if you have tried to conclude a value for this parameter and have failed to do so, then use the default value for the parameter«).

The sample interactions in the previous section employed only *cause-effect* and *definitional* rules. An explanation for an *effect-cause* rule, on the other hand, requires a discussion of the inverse *cause-effect* rule (or chain of rules), and a brief mention of alternate possibilities to explain the certainty measure associated with the rule. As discussed above, the *expertise* of a user may also require that the program display a text justification for the causal relationships cited in a *cause-effect* rule. Consider, for example, an interaction in which an explanation of the *effect-cause* rule in Fig. 6 is produced:

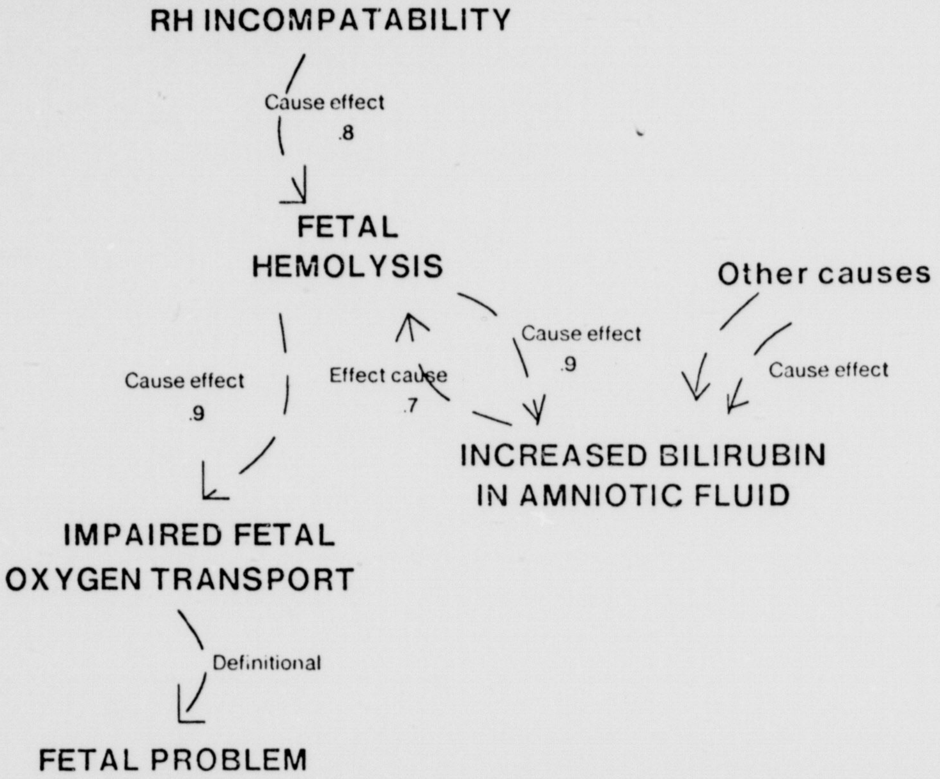

Fig. 6: A simple causal network showing the difference in reasoning between *effect-cause* and *cause-effect* rules in the medical setting. The number beside a link indicates the *certainty factor* (CF) associated with the rule. Note than an actual rule network for this domain would be more complex, with representation of intermediate steps, associated medical concepts, default values, and definitions.

Sample Interaction 4:

**WHY DO INCREASED BILIRUBIN COMPOUNDS IN THE AMNIOTIC FLUID IMPLY FETAL HEMOLYSIS?

Fetal hemolysis leads to bilirubin compounds in the fetal circulation; equilibration then takes place between the fetal plasma and the amniotic fluid, leading to increased bilirubin compounds in the amniotic fluid

While the relationship in this direction is nearly certain, the inverse relationship is less certain because of the following other possible causes of increased bilirubin compounds in the amniotic fluid:

Maternal blood in the amniotic fluid from trauma
Maternal blood in the amniotic fluid from prior amniocentesis

The response regarding the equilibration of fetal plasma and amniotic fluid is the stored text justification of the *cause-effect* rule that leads from »fetal hemolysis« to »increased bilirubin in amniotic fluid«. The individual steps could themselves have been represented in causal rules if the system builder had preferred to enter rule-based knowledge about the nature of hemolysis and bilirubin release into the circulation. The second component of the response, on the other hand, is generated from the other *cause-effect* rules that can lead to »increased bilirubin in amniotic fluid«.

The other types of rules require minor modifications of the explanation strategy. *Definitional* rules are usually omitted for the expert user on the basis of their low *complexity* values. An explanation of an *associational* rule indicates the lack of known causal information, and describes the degree of association. *Self-referencing* rules frequently have underlying reasons that are not adequately represented by a causal network; separate support knowledge associated with the rule [3], similar to the text justification shown in *Sample Interaction 4*, may need to be displayed for the user when explaining them.

Causal Links and Statistical Reasoning

We have focussed this discussion on the utility of representing causal knowledge in an expert system. In addition to facilitating the generation of tailored explanations, the use of causal relationships strengthens the reasoning power of a consultation program and can facilitate the acquisition of new knowledge from experts. However, an attempt to reason from causal information faces many of the same problems that have been encountered by those who have used statistical approaches for modeling diagnostic reasoning. It is possible to generate an *effect-cause* rule and to suggest its corresponding probability or certainty only if the information given in the corresponding *cause-effect* rule is accompanied by additional statistical information. For example, Bayes' Rule may be used to determine the probability of the i:th of k possible »causes« (e.g., diseases), given a specific observation (»effect«):

$$P(\text{cause}_i \mid \text{effect}) = \frac{P(\text{effect} \mid \text{cause}_i)\ P(\text{cause}_i)}{\sum\limits_{j=1} P(\text{cause}_j)\ P(\text{effect} \mid \text{cause}_j)}$$

This computation of the probability that the i:th possible »cause« is present given that the specific »effect« is observed, $P(\text{cause}_i \mid \text{effect})$, requires knowledge of the *a priori* frequencies $P(\text{cause}_i)$ for each of the possible »causes« (cause_1, cause_2, …, cause_k) of the »effect«. These data are not usually available for medical problems, and are dependent upon locale and prescreening of the patient population [19, 24]. The formula also requires the value of $P(\text{effect} \mid \text{cause}_j)$ for all *cause-effect* rules leading to the »effect«, not just the one for the rule leading from cause_i to the »effect«. In Fig. 6, for example, the *effect-cause* rule leading from »increased bilirubin in amniotic fluid« to »fetal hemolysis« could be derived from the *cause-effect* rule leading in the opposite direction only if all additional *cause-effect* rules leading to »increased bilirubin in amniotic fluid« were known (the »other causes« indicated in the figure) and if the relative frequencies of the various possible causes of »increased bilirubin in amniotic fluid« were also available. A more realistic approach is to obtain the inference weighting for the *effect-cause* rule directly from the expert who is building the knowledge base. Although such subjective estimates are fraught with danger in a purely Bayesian model [10], they appear to be adequate when the numerical weights are supported by a rich semantic structure [19, 28].

Similarly, problems are encountered in attempting to produce the inverse of rules that have Boolean preconditions. For example, consider the rule

IF: (A and (B or C))
THEN: Conclude D

Here D is known to imply A (with a certainty dependent on the other possible causes of D and their relative frequencies) only if B or C is present. While the inverse rule could be generated using Bayes' Rule given the *a priori* probabilities, one would not know the certainty to ascribe to cases where both B *and* C are present. This problem of conditional independence tends to force assumptions or simplifications when applying Bayes' Theorem. Dependency information can be obtained from databanks or from an expert, but cannot be derived directly from the causal network.

It is instructive to note how the Present Illness Program (PIP) and CADUCEUS, two recent medical reasoning programs, deal with the task of representing both *cause-effect* and *effect-cause* information. CADUCEUS [15] has two numbers for each manifestation of disease, an »evoking strength« (the likelihood that an observed manifestation is caused by the disease) and a »frequency« (the likelihood that a patient with a disease will display a given manifestation). These are analogous to the inference weightings on *effect-cause* rules and *cause-effect* rules respectively. However, the first version of the CADUCEUS program (INTERNIST—1) does not allow for combinations of manifestations that give higher (or lower) weighting than the sum of the separate manifestations*), nor does it provide a way to explain the inference paths involved.

PIP [14, 24] handles the implication of diseases by manifestations by using »triggers« for particular disease frames. No weighting is assigned at the time of frame invocation; instead PIP uses a scoring criterion that does not distinguish between *cause-effect* and *effect-cause* relationships in assigning a numerical value for a disease frame. While the information needed to explain the program's reasoning is present, the underlying causal information is not**).

In our experimental system, the inclusion of both *cause-effect* rules and *effect-cause* rules with explicit certainties, and the ability to group manifestations into rules, allow flexibility in constructing the network. Although causal information taken alone is insufficient for the construction of a comprehensive knowledge base, the causal knowledge can be used to propose *effect-cause* relationships for modification by the system builder. It can similarly be used to help generate explanations for such relationships when *effect-cause* rules are entered.

Conclusion

We have argued that a need exists for better explanations in medical consultation systems, and that this need can be partially met by incorporating a user model and an augmented causal representation of the domain knowledge. The causal network can function as an integral part of the reasoning system and may be used to guide the generation of tailored explanations and the acquisition of new domain knowledge. Causal information is useful but not sufficient for problem solving in most medical domains. However, when it is linked with information regarding the *complexity* and *importance* of the concepts and causal links, a powerful tool for explanation emerges.

Our prototype system has been a useful vehicle for studying the techniques we have discussed. Topics for future research include: 1) the development of methods for dynamically determining *complexity* and *importance* (based on the semantics of the network rather than on numbers provided by the system builder); 2) the discovery of improved techniques for using the context of a dialogue to guide the formation of an explanation; 3) the use of linguistic or psychologic methods for determining the *reason* a user has asked a question so that a customized response can be generated; and 4) the development of techniques for managing the various levels of *complexity* and *detail* inherent in the mechanistic relationships underlying physiological processes. The recent work of Patil, Szolovits, and Schwartz [13], who have separated such relationships into multiple levels of detail, has provided a promising approach to the solution of the last of these problems.

Acknowledgments

We are indebted to the following individuals for their advice and their assistance in reviewing earlier drafts of this paper: Bruce Buchanan, Bill Clancey, Greg Cooper, Mike Genesereth, Bill Swartout, Randy Teach, and Gio Wiederhold.

References

[1] Carbonell, J. R.: AI in CAI: An Artificial-Intelligence Approach to Computer-Assisted Instruction. IEEE Transactions on Man-Machine Systems. MMS *11* (1970) 190—202.

[2] Clancey, W. J.: Tutoring Rules for Guiding. A Case Method Dialogue. Int. J. Man-Machine Studs. *11* (1979) 25—49.

[3] Clancey, W. J.: The Epistemology of a Rule-Based Expert System. To appear in Artificial Intelligence (1983).

*) This problem is one of the reasons for the move from INTERNIST—1 to the new approaches used in CADUCEUS [15].

**) Recently the ABEL program, a descendent of PIP, has focussed on detailed modeling of causal relationships [13].

[4] Davis, R., King, J.: An Overview of Production Systems. In E. W. Elcock and D. Michie (Eds): Machine Representation of Knowledge. (New York: Wiley 1976).

[5] Davis, R.: Application of Meta-Level Knowledge to the Construction, Maintenance, and Use of Large Knowledge Bases. Doctoral Dissertation. Memo STAN-CS-76-552, HPP-76-7, Stanford University, July 1976.

[6] Duda, R. O., Hart, P. E., Barrett, P., Gaschnig, J., Konolige, K., Reboh, R., Slocum, J.: Development of the PROSPECTOR System for Mineral Exploration. Final Report, SRO Projects 5821 and 6415, SRI International, Menlo Park, Calif. (October 1978).

[7] Feigenbaum, E. A.: The Art of Artificial Intelligence: Themes and Case Studies of Knowledge Engineering. In AFIPS Conference Proceedings of the National Computer Conference Vol. 47, p. 227ff. (Montvale, N. J.: AFIPS Press 1978).

[8] Goldstein, I.: Developing a Computational Representation of Problem Solving Skills. AI Memo 495, Artificial Intelligence Center, Massachusetts Institute of Technology, October 1978.

[9] Gorry, G. A.: Computer-Assisted Clinical Decision Making. Meth. Inform. Med. *12* (1973) 45—51.

[10] Leaper, D. J., Horrocks, J. C., Staniland, J. R., deDombal, F. T.: Computer-Assisted Diagnosis of Abdominal Pain Using Estimates Provided by Clinicians. Brit. med. J. 1972, IV, 350—354.

[11] Linde, C.: The Organization of Discourse. In T. Shopen and J. M. Williams (Eds): Style and Variables in English. (Cambridge, MA.: Winthrop Press 1978).

[12] Linde, C., Goguen, J. A.: Structure of Planning Discourse. J. Soc. Biol. Struct. *1* (1978) 219—251.

[13] Patil, R. S., Szolovits, P., Schwartz, W. B.: Causal Understanding of Patient Illness in Medical Diagnosis. Proceedings of the 7th International Joint Conference on Artificial Intelligence, pp. 893—899. Vancouver, British Columbia, August 1981.

[14] Pauker, S. G., Gorry, G. A., Kassirer, J. P., Schwartz, W. B.: Toward the Simulation of Clinical Cognition: Taking a Present Illness by Computer. Amer. J. Med. *60* (1976) 981—995.

[15] Pople, H.: Heuristic Methods for Imposing Structure on Ill-Structured Problems: The Structuring of Medical Diagnostics. In P. Szolovits (Eds): Artificial Intelligence in Medicine. AAAS Symposium Series, Westview Press (forthcoming 1982).

[16] Rieger, C.: An Organization of Knowledge for Problem Solving and Language Comprehension. Artif. Intell. *7* (1976) 89—127.

[17] Scott, A. C., Clancey, W. J., Davis, R., Shortliffe, E. H.: Explanation Capabilities of Production-Based Consultation Systems. Amer. J. Comput. Linguist. Microfiche 62, 1977.

[18] Shortliffe, E. H., Buchanan, B. G.: A Model of Inexact Reasoning in Medicine. Math. Biosci. *23* (1975) 351—379.

[19] Shortliffe, E. H., Buchanan, B. G., Feigenbaum, E. A.: Knowledge Engineering for Medical Decision Making: A Review of Computer-Based Clinical Decision Aids. Proceed. IEEE *67* (1979) 1207—1224.

[20] Shortliffe, E. H., Davis, R., Axline, S. G., Buchanan, B. G., Green, C. C., Cohen, S. N.: Computer-Based Consultations in Clinical Therapeutics: Explanation and Rule Acquisition Capabilities of the MYCIN System. Comput. biomed. Res. *8* (1975) 303—320.

[21] Shortliffe, E. H.: Computer-Based Medical Consultations: MYCIN. (New York: Elsevier/North-Holland 1976).

[22] Shortliffe, E. H.: Consultation Systems for Physicians: The Role of Artificial Intelligence Techniques. In Proceedings of the Third National Conference of the Canadian Society for Computational Studies of Intelligence, Victoria, British Columbia, May 1980. Also in B. Webber and N. Nilsson (Eds): Readings in Artificial Intelligence, (Menlo Park, Ca.: Tioga Press 1981).

[23] Swartout, W. R.: Explaining and Justifying in Expert Consulting Programs. Proceedings of the 7th International Joint Conference on Artificial Intelligence, Vancouver, British Columbia, August 1981.

[24] Szolovits, P., Pauker, S. G.: Categorical and Probabilistic Reasoning in Medical Diagnosis. Artif. Intell. *11* (1978) 115—144.

[25] Weiner, J. L.: BLAH: A System which Explains its Reasoning. Artif. Intell. *15* (1980) 19—48.

[26] Weiner, J. L.: The Structure of Natural Explanation: Theory and Application. Systems Development Corp. SP-4305 (1979).

[27] Weiss, S. M., Kulikowski, C. A., Amarel, S., Safir, A.: A Model-Based Method for Computer-Aided Medical Decision-Making. Artif. Intell. *11* (1978) 145—172.

[28] Yu, V. L., Fagan, L. M., Wraith, S. M., Clancey, W. J., Scott, A. C., Hannigan, J., Blum, R. L., Buchanan, B. G., Cohen, S. N.: Antimicrobial Selection by a Computer: A Blinded Evaluation by Infectious Disease Experts. J. Amer. Med. Ass. *242* (1979) 1279—1282.

Addresses of the authors: Edward H. Shortliffe, M.D., Ph.D., Room TC-117, Division of General Internal Medicine, Stanford University School of Medicine, Stanford, CA 94305; Jerold W. Wallis, M.D., current address: Dept. of Medicine, University of Michigan Medical Center, Ann Arbor, Michigan, USA.