Stanford Heuristic Programming Project
Memo HPP-77-7

April 1977

Computer Science Department
Report No. STAN-CS-77-612

(12)

ADA042756

RULE-BASED UNDERSTANDING OF SIGNALS

by

H. Penny Nii and Edward A. Feigenbaum

COMPUTER SCIENCE DEPARTMENT
School of Humanities and Sciences
STANFORD UNIVERSITY

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: STAN-CS-77-612, HPP-77-7	2. GOVT ACCESSION NO.
	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): Rule-based Understanding of Signals.	5. TYPE OF REPORT & PERIOD COVERED: Technical Report, April 1977
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): H. Penny Nii, Edward A. Feigenbaum	8. CONTRACT OR GRANT NUMBER(s): ARPA DAHC 15-73-C-0435
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Stanford Computer Science Department, Stanford University, Stanford, CA. 94305	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: ARPA [illegible]
11. CONTROLLING OFFICE NAME AND ADDRESS: Defense Advanced Research Projects Agency, Information Processing Techniques Office, 1400 Wilson Ave., Arlington, VA. 22209	12. REPORT DATE: April 1977
	13. NUMBER OF PAGES: 23
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): Mr. Philip Surra, Resident Representative, Office of Naval Research, Durand 165, Stanford University	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Reproduction in whole or in part is permitted for any purpose of the U.S. Government.

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

KNOWLEDGE-BASED PROGRAM, PRODUCTION RULES, HYPOTHESIS FORMATION, HIERARCHICAL HYPOTHESIS, HIERARCHICAL CONTROL, MODEL-DRIVEN, DATA-DRIVEN, LEVELS OF ABSTRACTION, EVENT-DRIVEN PROGRAM, KNOWLEDGE SOURCES, APPLICATION, EXPECTATION-DRIVEN, HEURISTICS, PROBLEM SOLVING STRATEGY, INCREMENTAL HYPOTHESIS FORMATION, EXPERTS, SIGNAL INTERPRETATION.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

SU/X and SU/P are knowledge-based programs which employ pattern-invoked inference methods. Both tasks are concerned with the interpretation of large quantities of digitized signal data. The task of SU/X is to understand "continuous signals," that is, signals which persist over time. The task of SU/P is to interpret protein x-ray crystallographic data. Some features of the design are: (1) incremental interpretation of data employing many different pattern-invoked sources of knowledge, (2) production rule representation of knowledge, including high level strategy knowledge, (3) "opportunistic" hypothesis formation using

over

DD FORM 1473, 1 JAN 73 — EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) both data-driven and model-driven techniques within a general hypothesize-and-test paradigm; and (4) multilevel representation of the solution hypothesis.

Rule-based Understanding of Signals

STAN-CS-77-612
Heuristic Programming Project Memo 77-7

H. Penny Nii and Edward A. Feigenbaum

ABSTRACT

SU/X and SU/P are knowledge-based programs which employ pattern-invoked inference methods. Both tasks are concerned with the interpretation of large quantities of digitized signal data. The task of SU/X is to understand "continuous signals," that is, signals which persist over time. The task of SU/P is to interpret protein x-ray crystallographic data. Some features of the design are: (1) incremental interpretation of data employing many different pattern-invoked sources of knowledge, (2) production rule representation of knowledge, including high level strategy knowledge, (3) "opportunistic" hypothesis formation using both data-driven and model-driven techniques within a general hypothesize-and-test paradigm; and (4) multilevel representation of the solution hypothesis.

KEYS WORDS

KNOWLEDGE-BASED PROGRAM, PRODUCTION RULES, HYPOTHESIS FORMATION, HIERARCHICAL HYPOTHESIS, HIERARCHICAL CONTROL, MODEL-DRIVEN, DATA-DRIVEN, LEVELS OF ABSTRACTION, EVENT-DRIVEN PROGRAM, KNOWLEDGE SOURCES, APPLICATION, EXPECTATION-DRIVEN, HEURISTICS, PROBLEM SOLVING STRATEGY, INCREMENTAL HYPOTHESIS FORMATION, EXPERTS, SIGNAL INTERPRETATION.

This research was supported by the Defense Advanced Research Projects Agency under ARPA Order No. 2494, Contract No. DAHC 15-73-C-0435.

Rule-based Understanding of Signals

H. Penny Nii
Edward A. Feigenbaum
Heuristic Programming Project
Computer Science Department
Stanford University
Stanford, Ca. 94305

ABSTRACT

Su/X and SU/P are knowledge-based programs which employ pattern-invoked inference methods. Both tasks are concerned with the interpretation of large quantities of digitized signal data. The task of SU/X is to understand "continuous signals", that is, signals which persist over time. The task of SU/P is to interpret protein x-ray crystallographic data. Some features of the design are: (1) incremental interpretation of data employing many different pattern-invoked sources of knowledge, (2) production rule representation of knowledge, including high level strategy knowledge, (3) "opportunistic" hypothesis formation using both data-driven and model-driven techniques within a general hypothesize-and-test paradigm; and (4) multilevel representation of the solution hypothesis.

This work was supported by the Department of Defense, Advanced Research Project Agency, ARPA Contract DAHC15-73-C-0435.

This paper was written for the Workshop on Pattern-Directed Inference Systems, Hawaii, May 1977.

CONTENTS

1 INTRODUCTION AND SUMMARY

This paper describes a design of knowledge-based programs which employ pattern-invoked inference methods. Domain and strategy knowledge are represented as production rules to be invoked when appropriate situations arise in the problem-solving process. The same basic design philosophy is utilized in two task domains, both of which are concerned with the interpretation of large volumes of digitized physical signals. The tasks are (1) the understanding of continuous signals produced by objects and (2) the interpretation of protein x-ray crystallographic data in terms of a three-dimensional model of the molecule. The programs associated with these tasks are called SU/X and SU/P, respectively.

Some of the design concepts in SU/X and SU/P are rooted in the HEARSAY-II program [4, 6-7]. Concepts which have been borrowed are: (a) a global data base, called the blackboard, for the integration of knowledge sources and (b) a multilevel representation of the solution hypotheses. These basic concepts are integrated into a system design that emphasizes: (a) the representation of knowledge in production rules, (b) the representation of the control structure as sources of knowledge related to problem-solving methods and strategies, (c) the capability of the program to explain its reasoning steps, and (d) a level of generality of the basic design concepts leading to application in different tasks or domains.

1.1 Major Themes

The "understanding" of physical signals often requires using information not present in the signal data themselves. Examples of such information are: (a) in the continuous-signal problem, the characteristics of the signal-producing objects, (b) in the protein-modeling problem, the amino acid sequence and the stereochemical and protein chemistry constraints. Each such source of knowledge may at any time provide an inference which serves as a basis for another knowledge source to make yet another inference, and so on, until all relevant information has been used and appropriate inferences have been drawn.

Essential to the operation of the program is its <u>model</u> of the developing hypothesis. The model is a symbol-structure that is built and maintained by the program, contains what is known about the unfolding situation, and thus provides a context for the ongoing analysis. The model is used as a reference for the interpretation of new information, assimilation of new events, and generation of expectations concerning future events. It is the program's "cognitive flywheel".

SU/X and SU/P are "knowledge-based" programs (footnote 1). Their powers are largely derived from the knowledge given to them by "expert" human analysts and/or "expert" algorithms. Major problems in the design of such systems show up vividly in these two programs:

a. Knowledge acquisition. This is a task of systematically ferreting out the informal and semiformal knowledge held by the expert. The breadth and sheer volume of an expert's knowledge is what makes his analysis general and powerful; yet, obtaining that knowledge, which he often does not realize he is using, is a painstaking and inexact process.

b. Knowledge representation. Having acquired the knowledge in its "human" form, we must represent it in a form that is convenient and efficient for machine processing and at the same time reasonably "natural" (bear in mind that the knowledge rarely boils down merely to a set of numbers) -- a difficult and time-consuming task.

c. Integration of multiple, diverse sources of knowledge. Program and information structures must be created by which the various kinds of knowledge can "work together" to form a coherent and accurate hypothesis. When the knowledge exists at many different levels of abstraction and aggregation (say, from alpha-helix substructure all the way down to electron density values in an electron density map), one has a major design problem.

1.2 Major Terms and Concepts

The task of "understanding" the data is accomplished at various levels of analysis. These levels are exhibited in Figure 1.1 for the continuous-signal interpretation problem and in Figure 1.2 for the protein-modeling problem. The most integrated -- the highest -- levels for the two problems involve the description of the signal-producing objects, and the three-dimensional model of the protein. The lowest levels, that is, the levels closest to the data, consist of the line features derived from the signal data, and the atoms and their coordinates in three space.

At each level, the units of analysis are the hypothesis elements. These are symbol-structures that summarize what the available evidence indicates in terms that are meaningful at that particular level.

Bridging between the levels of analysis are _sources of knowledge_ [4,7]. A knowledge source (KS) is capable of putting forth the _inference_ that some hypothesis elements present at its "input" level imply some particular hypothesis elements(s) at its "output" level. A source of knowledge contains not only the knowledge necessary for making its own specialized inferences, but also the knowledge necessary for checking the inferences made by other sources of knowledge. The inferences which draw together hypothesis elements at one level into a hypothesis element at a higher level (or which operate in the other direction) are represented symbolically as links between levels (See figures 1.1 and 1.2). The resulting network, rooted in the input data and integrated at the highest level into a description of the hypothesized problem solution, is called the _current best hypothesis_, or _the hypothesis_ for short. Each source of knowledge holds a considerable body of specialized information that a human expert would generally consider "ordinary". Sometimes this is relatively "hard" knowledge or "textbook" knowledge. Also represented are the _heuristics_, that is, "rules of good guessing" a human expert develops in his area of expertise. These "judgmental" rules are generally accompanied by estimates from human experts concerning the _weight_ that each rule should carry in the analysis.

Each KS is composed of "pieces" of knowledge. By a piece of knowledge we mean a _production rule_, that is, an IF-THEN type of implication formula. The "IF" side, also called the situation side, specifies a set of conditions or patterns for the applicability of the particular rule. The "THEN" side, also called the action side, symbolizes the implications to be drawn (more precisely, various processing events to be caused) if the "IF" conditions are met. (Refer to [2] for an excellent overview of production rules.)

The knowledge of how to perform, that is, how to use the available knowledge sources, is another kind of knowledge that experts possess. This type of knowledge is also represented in the system in the form of _control/strategy_ production rules, which promote flexibility in specifying and modifying strategies of analysis.

Hypothesis formation is an "opportunistic" process. Both _data-driven_ and _model-driven_ hypothesis formation techniques are used within the general _hypothesize-and-test_ paradigm. One of the tasks of the control/strategy knowledge source is to determine the applicability of these methods to different situations. The unit of processing activity is the _event_. Events symbolize such things as "what inferences to make", "what symbol-structures to modify", "what to look for in the data", and so on. The basic control loop for these event-driven programs, is one in which lists of events (events sometimes include new data) and the set of control/strategy rules are periodically scanned to determine the "next thing to do" (footnote 2).

In the following sections we discuss issues related to the representation of the hypothesis, the knowledge sources, and the control structure. Before continuing, however, we will briefly describe the two tasks that have been implemented and list some guidelines for choosing applications in which this type of system organization may be useful.

2 THE TASKS

2.1 Interpretation of Continuous-Signals (SU/X)

The signal-understanding program performs analysis of data derived from a digitized plot of continuous signals, the interpretation of which is to a considerable degree a function of time. Examples of data having this characteristic are electromagnetic and acoustic signals, and signals from hospital patients monitored in an intensive care unit. The "front-end" signal-processing hardware and software detect energy "packets" appearing at various spectral frequencies, and follow these packets in time. The current system is designed to analyze a digitized description of these data. At the end of each time period, say, a few minutes, the user is given an integrated analysis of the interpreted objects within its data purview. [5]

2.2 Interpretation of Three-Dimensional Signal Data: Protein Crystallography (SU/P)

The task of this program is to infer three-dimensional models of protein molecules. The model is derived from an interpretation of the electron density map of the crystallized protein. The density map is, in turn, derived from x-ray diffraction data. These data typically yield a poorly resolved distribution of the electron density within the protein molecule, and the location of individual atoms are generally not identifiable. Traditionally, the protein crystallographer embodies his interpretation of the electron density map in a "ball and stick" molecular model fashioned from metal parts. These parts are strung together to build a model which conforms to the electron density map and is also consistent with protein chemistry and stereochemical constraints. The current system tries to simulate humans who build models incrementally from the most "obvious" regions of the electron density map. The incremental, opportunistic strategies used by our program to form hypotheses closely resemble the problem-solving methods used by human model builders. Refer to [3] for more complete description of the problem.

3 SUITABLE APPLICATION AREAS

Building a signal interpretation system within the program organization summarized above can best be described as "opportunistic" analysis. Bits and pieces of information must be used as opportunity arises to build slowly a coherent picture of the world -- much like putting a jigsaw puzzle together. Some thoughts on the characteristics of problems suited to this approach are listed below:

1. Large amounts of signal data need to be analyzed. Examples include the interpretation of speech and other acoustic signals, X-ray and other spectral data, radar signals, photographic data, etc. (A variation involves understanding a large volume of symbolic data; for example, the maintenance of a global plotboard of air traffic based on messages from various air traffic control centers.)

2. Formal or informal interpretive theories exist. By informal interpretive theory we mean lore or heuristics which human experts bring to bear in order to "understand" the data. These inexact and informal rules are incorporated as KSs in conjunction with more formal knowledge about the domain.

3. Task domain can be decomposed hierarchically in a "natural way" [4]. In many cases the domain can be decomposed into a series of data reduction levels, where various interpretive theories (in the sense described above) exist for transforming data from one level to another.

4. "Opportunistic" strategies must be used. That is, there is no computationally feasible "legal move generator" that defines the space of solutions in which pruning and steering take place. Rather, by reasoning about bits and pieces of available evidence, one can incrementally generate partial hypotheses that will eventualy lead to a more global solution hypothesis.

3.1 Data-Driven vs Model-Driven Hypothesis Formation Methods

We have combined data- and model-driven methods of hypothesis formation in the design of SU/X and SU/P. By "data-driven" we mean "inferred from the input data". By "model-driven" we mean "based on expectation" where the expectation is inferred from knowledge about the domain. For example, a hypothesis generated by a KS which infers an amino acid sidechain from the electron density data is a data-driven hypothesis. On the other hand, a hypothesis about the existence of an amino-acid sidechain that is deduced from topological knowledge of the protein is a model-based hypothesis. In the former case, the data is used as the basis for signal analysis; in the latter case, the primary data is used solely to verify the expectation.

There are no hard-and-fast criteria for determining which of the two hypothesis formation methods is more appropriate for a particular signal-processing task. The choice depends, to a large extent, on the nature of the KSs which are available and on the power of the analysis model available. Our experience points strongly toward the use of a combination of these techniques; some KS's are strongly data dependent while others are strongly model dependent. In the continuous-signal interpretation program, for example, the majority of the inferences are data-driven, with occasional model-driven inferences. The converse is true in the protein model-building wich places more emphasis on model-driven hypothesis generation. The following are guidelines we have used in determining which of the two methods is more appropriate:

1. Signal to Noise Ratio. Problems which have inherently low S/N ratios are better suited to solutions by model-driven programs; the converse is true for problems with high S/N ratios.

2. Availability of a model. A model, sometimes referred to as "the semantics of the task domain", can be available in various forms: (1) input to an abstract level of the hypothesis structure, (2) general knowledge about the task domain, or (3) specific knowledge about the particular task. In the protein crystallography problem, for instance, the amino acid sequence (the topology of the protein) serves as a model for guiding the interpretation of the primary data. However, in the continuous-signal interpretation problem, the model is drawn from general knowledge about the signal sources and from other relevant external sources of information that serve to define the context. If a reliable model is available, the data-interpretation KSs can be used as verifiers rather than generators of inferences; this reduces the computational burden on the signal-processing programs at the "front end".

4 THE NATURE OF THE HYPOTHESIS

In order to integrate a diversity of knowledge about the task domain, the domain is decomposed hierarchically into levels of analysis. We will describe briefly some of the basic ideas on the nature of the hypothesis (footnote 3).

A signal interpretation problem can be viewed as a problem of "transforming" signals representing an object into a symbolic description of the ojbect on a more abstract level. We use the word "transformation" to mean a shift from one representation of an object (digitized signals) to another (symbolic description) using any formal or informal rules.

The data structure hierarchy reflects a plan for the utilization of the various data transformation KSs which contribute to the total data interpretation process. Generally these transformational steps involve data reductions of the primary data in a stepwise fashion from the detailed to the more abstract description of the object. However, we have found that some of the most useful KSs generate inferences spanning several levels. For example, in the protein-modeling problem, a human can "see" in the electron density data, helical substructures without knowing or observing the details of each atom placement. This kind of knowledge is usually very specific to situations; human experts know, and use, many of these specialized, informal bodies of knowledge.

The data structure of the solution hypothesis is a linked network of nodes, where each node (hypothesis element) represents a meaningful aggregation of lower level hypothesis elements. A link between any two hypothesis elements represents a result of some action by a KS and indirectly points to the KS itself. A link has associated with it directional properties. In general, the direction indicates one of the the following: (1) A link which goes from a more abstract to a less abstract level of the hypothesis is referred to as an "expectation-link". The node at the end of an expectation-link is a model-based hypothesis element, and the link represents "support from above" (i.e. the reason for proposing the hypothesis element is to be found at the higher level). (2) A link which goes in the opposite direction, from lower levels of abstraction to higher, is referred to as a "reduction-link". The node at the end of a reduction-link is a data-based hypothesis element, and the link represent "support from below" (i.e. the reason for proposing the hypothesis element is to be found at a lower level). (These directions correspond loosely to "top-down" and "bottom-up" path generation.) Examples of KSs and hypothesis elements generated by the KSs are shown in Figure 2.

The protein-modeling problem posed some difficulties in the design of its hypothesis structure. These can be attributed to several factors. First, the decomposition of the solution space (the three-dimensional model) and the abstractions of the primary data (electron density) do not result in one consistent data hierarchy but result in two hierarchies. Second, the two hierarchies overlap semantically at some levels but are not representationally compatible. Third, very little is known about mapping the object between the two spaces. As indicated in Figure 3, however, the two hierarchies, with a network of links, can be merged into a single representation of the problem space. This representation indicates that hypothesis need not be represented as a strict hierarchy; it can be represented as a more general network of related elements. (Refer to [3] for more detailed description.)

5 THE NATURE OF THE "CONTROL"

A system's performance depends both on the competence of each KS and on the utilization of these KSs within the context of the goals of the task domain.

There are two seperate but equally important issues involved in a design of a knowledge-based performance program: (1) the availability and the quality of the specialist KSs that cooperate in the building of a hypothesis. (These KSs define the hierarchy of abstractions of the hypothesis.) (2) the optimal utilisation of these KSs. If we view the KSs as resources that are available for solving a problem, then the optimal resource allocation strategy is determined by the quality, the size, and the cost of the KSs, and the state of the current hypothesized solution. The control structure must be sensitive to, and be able to adjust to, the numerous possible solution states which arise in the course of solving a problem. Within this viewpoint, then, what is commonly called the "control structure" becomes another totally domain-dependent knowledge source. The notion of a "hierarchic control" is an attempt to come to grips with the issues of resource allocation and "control" strategies.

5.1 Hierarchically Organized Control Structures

In a "hierarchically organized control structure," problem-solving activities themselves form a hierarchy of knowledge necessary for solving the problem. On the lowest level is a set of knowledge sources the tasks of which are to make the primary inferences in the hypothesis network previously described. We refer to this level of knowledge as the "hypothesis-formation" level. At the next level are "meta" KSs that have knowledge about the capabilities of the KSs in the hypothesis-formation level. We refer to this level as the "KS-activation" level; a KS on this level represents a policy on knowledge utilization. At the highest level is the Strategy-KS which analyzes the quality of the current solution to determine what region of the data to anlyze next; it also determines what kind of strategy to use.

Another way to describe this organization is as follows: The KSs are organized hierarchically -- much like the management structure in a corporate environment -- in terms of the scope of their knowledge and the specificity of their functions.

> Example: A KS capable of deciding whether to look for helices or to continue looking for a large amino acid sidechain would possess a higher level of knowledge than a KS whose function is to infer the placement of atoms of some amino acid sidechain. It is a higher level because its area of expertise (choosing the best problem solving strategy for a given situation), is broader in scope and narrower in the knowledge of the processing specifics. It does not have, and it need not have, any knowledge of the details of the execution of the problem-solving strategy it chooses.

This control hierarchy should be clearly distinguished from the hierarchy of hypothesis levels. The hypothesis hierarchy represents an a priori plan for the solution presented by a "natural" decomposition of the analysis problem. The control hierarchy, on the other hand, represents the organization of the problem-solving activities necessary for the formation of the hypothesis. Figure 4 shows a general relationship between the organization of the hypothesis structure and the organization of the control structure. Table 1 summarizes the scope of KSs on each level of control hierarchy.

5.2 Control Structure Implementation

All information needed by the different KSs is contained in a global data structure called the "blackboard". The "blackboard" concept has its origin in HEARSAY [4] and is extended in SU/X and SU/P. The contents of the blackboard in SU/X and SU/P constist of:

1. The current best hypothesis (CBH)

2. The Event-list: A list of changes in the hypothesis which have not yet been processed by any KS. An event also contains the name of the KS and the identifier of the rule which caused the change.

3. The Event: A global variable containing the currently "active event", that is, an event which is currently being processed by some KS. The Event also represents the current focus of attention.

4. The Problems-list: A list of unresolved problems encountered by various KSs. Such problems range from expected data not yet available, to detectable "errors" in the program (e.g. insufficient knowledge).

5. The Event history list: The Event, together with its Predecessor and Successor events form a causal chain of reasoning. In the continuos-signal understanding problem, the Event history list is sometimes used by KS to analyze series of events which occurred over a period of time. More generally, it serves as a data base from which reasoning traces are generated and "how" and "why" questions answered. (See reference [1,8] for some examples of this type of traces.)

5.2.1 Hypothesis Formation Level

At the lowest level of control -- the most data specific level -- are the inference-generating KSs, or the specialist-KSs. Each specialist-KS has the task of creating or modifying hypothesis elements, evaluating inferences generated by other specialist-KSs, and cataloging of missing evidence which are essential for a KS to generate meaningful inferences.

Each specialist-KS has access to the blackboard. Its focus of attention is that portion of the blackboard containing the latest change(s) made to the current hypothesis. Although a KS has access to the entire hypothesis, it normally "understands" only the descriptors contained in two levels, its input level and its output level.

INFERENCE-GENERATION. Inference-generation is the creation or modification of hypothesis elements; it is the "hypothesize" part of the hypothesize-and-test paradigm. An inference-generator may use a data-driven or model-driven hypothesis formation method. As mentioned earlier, a KS is represented as a set of production rules consisting of "situation-action" pairs . The "situation" for the inference-generator is a particular state of those hypothesis elements containing data relevant to the KS. A match between a description in the hypothesis element and the situation-side of a rule indicates that a KS can make some conjectures regarding that hypothesis element. When the appropriate KS is invoked, the "action" part will transform the current hypothesis to a new current hypothesis either by adding new links to the structure, creating new hypothesis elements, or changing the attribute values of a hypothesis element (see Table 1. for a summary).

INFERENCE-EVALUATION. Inference evaluation involves the appraisal of inferences generated by other KSs; it is the "test" part of the hypothesize-and-test paradign. For each inference level there are usually more than one specialist-KS capable of generating inferences on that level. When a KS is invoked because of a particular event, another KS may already have processed the salient event. In such a circumstance, the currently active KS evaluates the inference generated by the other KS. The evaluation can result in the KS agreeing with, disagreeing with, or being indifferent about the particular inference being evaluated. If there is agreement, the confidence in that inference is increased; if there is disagreement, either the confidence value is decreased or an alternative hypothesis is generated. There is no action taken for "I don't know" situations.

PROBLEM-CATALOGING. Problem cataloging involves attempting to identify missing evidence essential for a KS to generate meaningful inferences. If a KS is unable to make new inferences when called upon to do so, it may be due to lack of knowledge about the particular situation or due to lack of necessary information, that is, the current situation does not meet the conditions on the situation sides of the rules. If the specialist-KS is "ignorant" then its knowledge-base need to be augmented in some way. If the cause is due to lack of particular evidence, a KS can request it by placing notice on the Problems-list. This calls the system's attention to a particular situation in which a solution is possible "...if x were true." Since a specialist-KS is not aware of the importance (or the unimportance) of its own immediate needs within the general framework of the solution, the decision to pursue or not to pursue the needs of the specialist-KSs is made by a higher level KS.

5.2.2 KS-Activation Level

At the level immediately above the hypothesis-formation level are the KS-activators whose tasks are to invoke the specialist-KSs as appropriate. The KSs on this level represent various policies and problem-solving strategies related to the utilization of the specialist-KSs. If, for example, events are processed on an earliest-occurences-first policy, we would have a breadth-first strategy; if events are processed on a latest-occurences-first policy, we would have a depth-first strategy.

If there is more than one specialist-KS available to process an event, some policy is needed to guide the order in which these KSs are to be utilized. Different KS-activators can be made to reflect different policies, ranging from fastest-first to most-accurate-first (footnote 4). There are currently two kinds of KS on the KS-activation level, the Event-driver and the Expectation-driver. For each event the Event-driver activates specialist-KSs based on the degree of specialization (and assumed accuracy) of the KSs. The Expectation-driver processes items on the Problems-list on the basis of how critical the needed evidence is to the emerging hypothesis. This evaluation of how-critical for the continuous-signal problem is sharply defined as part of the knowledge of the domain. In the protein-modeling problem, however, the evaluation criteria are much more heuristic, and in fact are just another element of the overall analysis strategy.

The Event-driver. An event type represents an a priori grouping of similar changes to the hypothesis, that is, it represents the abstractions of possible changes to the hypothesis. The changes, together with the identity of the rules which produced the changes, are put on a globally accessible list called the "Event-list". The Event-driver invokes the appropriate Specialist-KSs based on the information contained in the event or group of events.

Expectation-driver. The task of the Expectation-driver is to monitor the items on the Problems-list to see if any events which might satisfy the conditions on the Problems-list have occurred. If the conditions have occurred, it will activate the specialist-KS which had arranged the request. (footnote 5)

5.2.3 Strategy Level

The set of rules at the Strategy-level captures a human expert's knowledge of how to solve a problem. The task of the Strategy-KS -- the highest control level -- is to choose the best problem-solving strategy for the current state of the solution. Its expertise lies, first, in determining how close the current hypothesis is to the actual solution. In neither SU/X nor SU/P are there formal mechanisms to measure the differences between the current best hypothesis and the "right answer". The program detects when the solution hypothesis is "on the right track" by use of heuristic criteria. For example, in the protein modeling problem a large number of connected nodes on the stereo-substructure level may imply that the hypothesis is approaching a solution.

A consistent inability to verify expectation-based hypothesis elements may signal an error in the hypothesis. A more general indication of ineffective hypothesis formation appears as a consistent generation of conjectures whose confidence values are below a threshhold value; and which therefore indicates that the analysis is "bogged down".

A strategy-KS must also decide on a course of action once a difference between the hypothesis and the "right answer" is found. Note that these two functions of the Strategy-KS -- noticing weak parts of the hypothesized solution and choosing the appropriate corrective actions -- correspond to the situation and the action parts of production rules. Currently, the Strategy-KS can take one of three possible actions:

1. invoke the Expectation-driver to see if the local needs/goals are satisfiable by recent event(s);

2. invoke the Event-driver to process the latest changes in the hypothesis;

3. decide what region of the data space to work on next, i.e. determine the region of minimal ambiguity in the data.

6 GOAL-DIRECTED ACTIVITY: SOME SPECULATIONS

Our experience indicates that although the data-driven and model-driven hypothesis formation methods in combination are powerful, some situations are best handled with a goal-driven method, i.e. utilizing a goal structure and goal-seeking search processes. In the programs described, the occasional lack of certain evidence can halt the whole problem-solving process. However, the need for missing evidence may already be known and catalogued on the Problems-list. Under such a circumstance the obvious solution is to set a goal for "seeking" that evidence. Within the context of the current implementation, a goal-directed search through the solution space can be accomplished by: (1) adding a Goal-driver on the KS-activation control level, (2) implementing a backward-chaining mechanism for the rules as in the MYCIN system [1], and (3) adding rules to the Strategy-KS to choose between data-driven, model-driven and goal-driven methods of hypothesis formation as appropriate.

7 SUMMARY AND CONCLUDING REMARKS

SU/X and SU/P are two application programs that have been written to reason toward an understanding of digitized physical signals. The essential features of the programs' design are: (1) data- and model-driven, opportunistic modes of hypothesis formation in which the "control" is organized hierarchically, and (2) a globally accessible hypothesis structure augmented by focus-of-attention and historical information which serve to integrate diverse sources of knowledge. The basic design is similar in many ways to the HEARSAY-II Speech Understanding System design. It is applicable to many different types of problems, especially to those problems that do not have computationally feasible "legal move generators" and must therefore resort to opportunistic generation of alternate hypotheses.

The use of production rules to represent control/strategy knowledge offers the advantages of uniformity of representation and accessibility of knowledge for purposes of augmentation and modification of the knowledge base. Because the line-of-reasoning is often a complex compounding of the elemental steps indicated by the rules, a dynamic explanation capability is needed. We did not discuss this important feature of the programs. Nor did we discuss the facility which allows assignment of an expert's degree of uncertainty for each rule entered. The use of this facility is not well developed currently in the programs discussed. (See References 8 and 9 for similar but better developed capabilities in the MYCIN program.) We believe that facilities for explanation and for inexact inference must be integrated into the program design at the initial stages.

Footnotes

1. SU/X was implemented in the context of a military signal-understanding application. It is a large INTERLISP program that performed well on a variety of complex signal-interpetation tasks within the domain. SU/P, also written in INTERLISP, is under development.

2. The events are stored in three lists, each of which requires its own special treatment; knowledge-based events i.e. events specifically related to changes in the hypothesis; time-based events, i.e., those events specifically related to expectations of "what will happen when"; and problems, i.e. expectations fr_m the programs' "model of the situation" for which the clinching confirmatory or disconfirmatory evidence has not yet been found.

3. As mentioned earlier, the design of the hypothesis structure in SU/X and SU/P is based on the concepts found in HEARSAY-II. We refer you to [4,7] for a more detailed description.

4. The issues of focus of attention and resource allocation policies, as described by Hayes-Roth and Lesser [6], are important ones. A subsequent paper will describe the implementation of these policies within the SU/X and SU/P framework.

5. The problems which are "need-for-evidence" can be viewed as "subgoals-to-be-achieved". The systems are currently biased toward an opportunistic mode of hypothesis formation, and the implicit strategy for such subgoals is "wait and see".

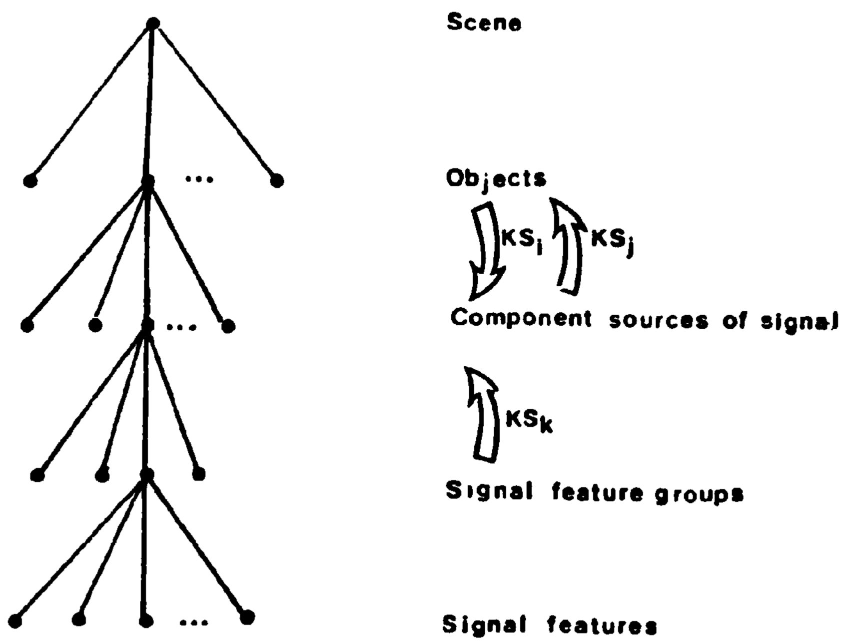


Figure 1.1 Hypothesis Structuré (SU/X)*

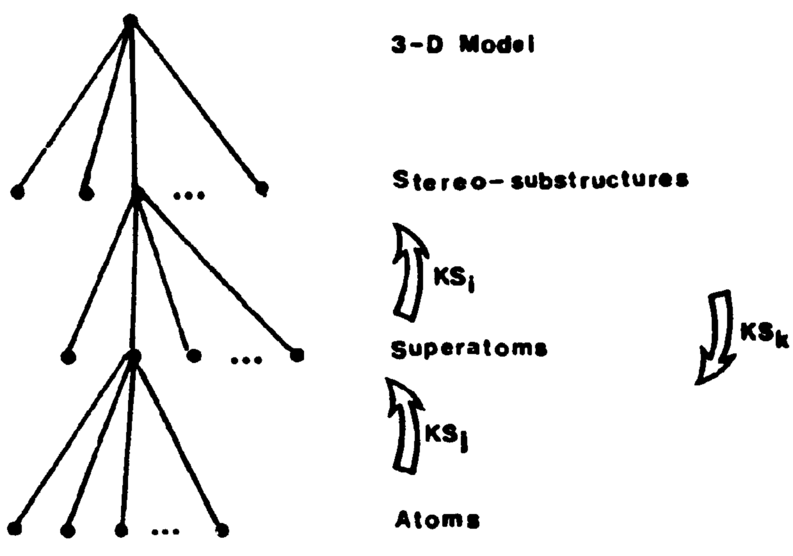
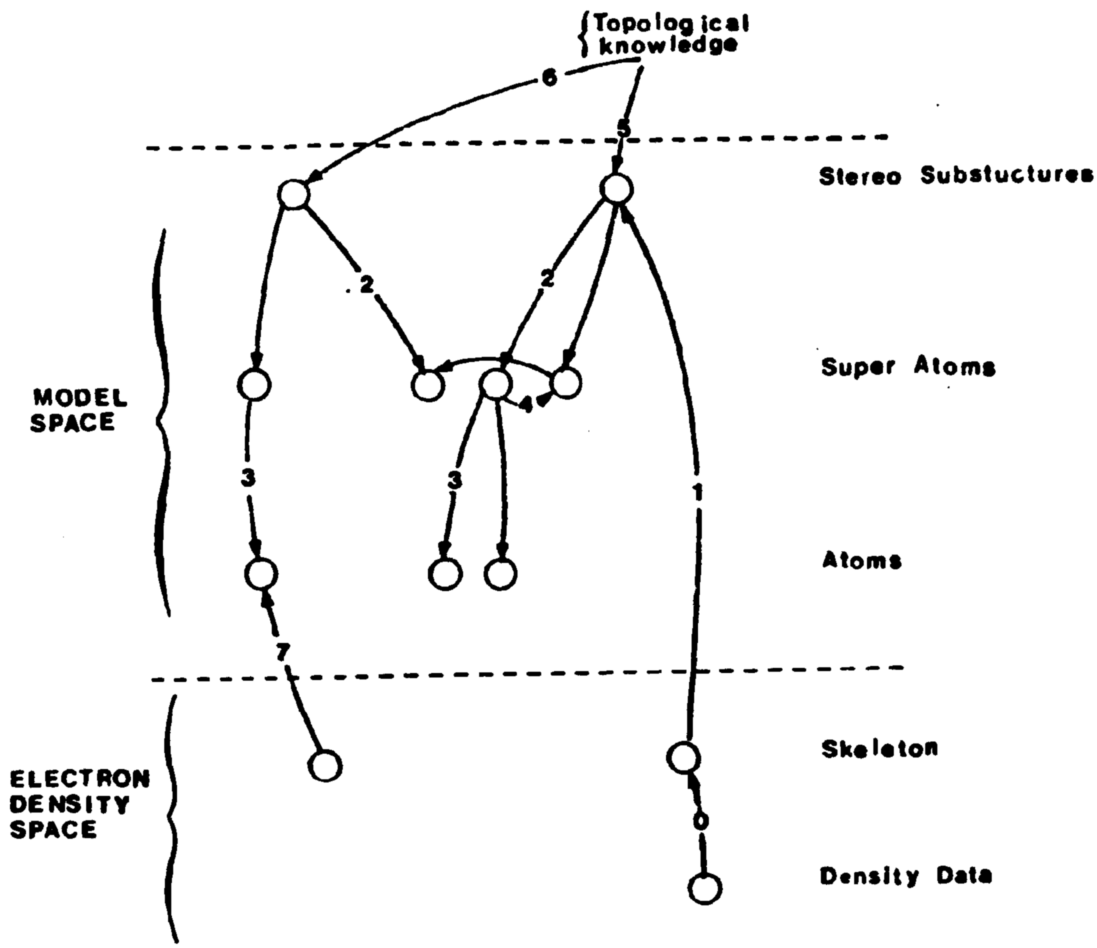


Figure 1.2 Hypothesis Stucture (SU/P)*

* The nodes represent hypothesis elements.
The arrows represent KSs which infer hypothesis element(s) on one level from hypothesis elements on another level.

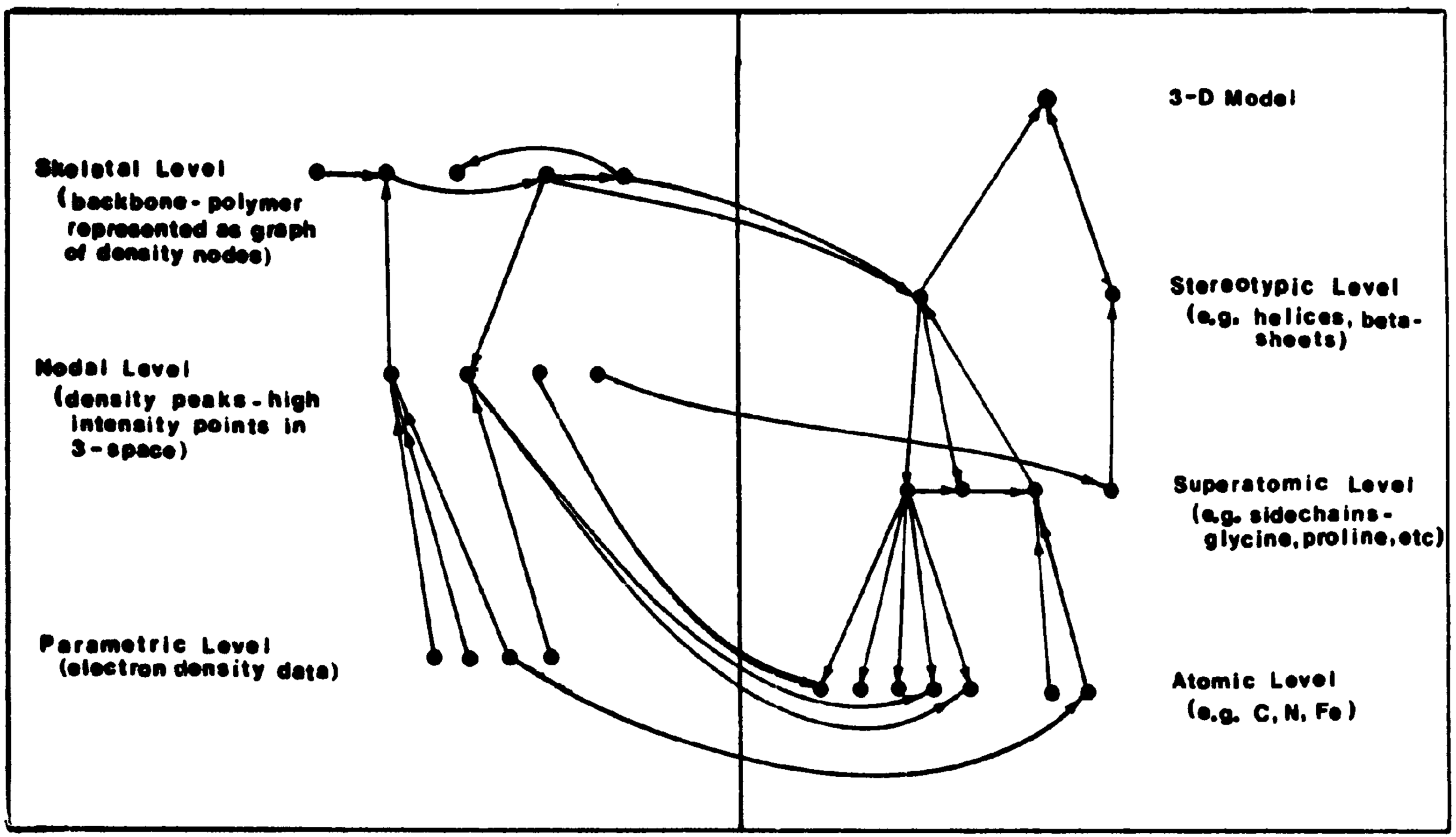


KNOWLEDGE SOURCES

0. Skeletonization
1. Helix Identification (Skeletal)
2. Sidechain Identification
3. Bond Rotation
4. Sequence Identification
5. Helix Identification (Topological)
6. Cofactor Identification
7. "Heavy Atoms" Identification

Knowledge Source Utilization in Hypothesis Formation

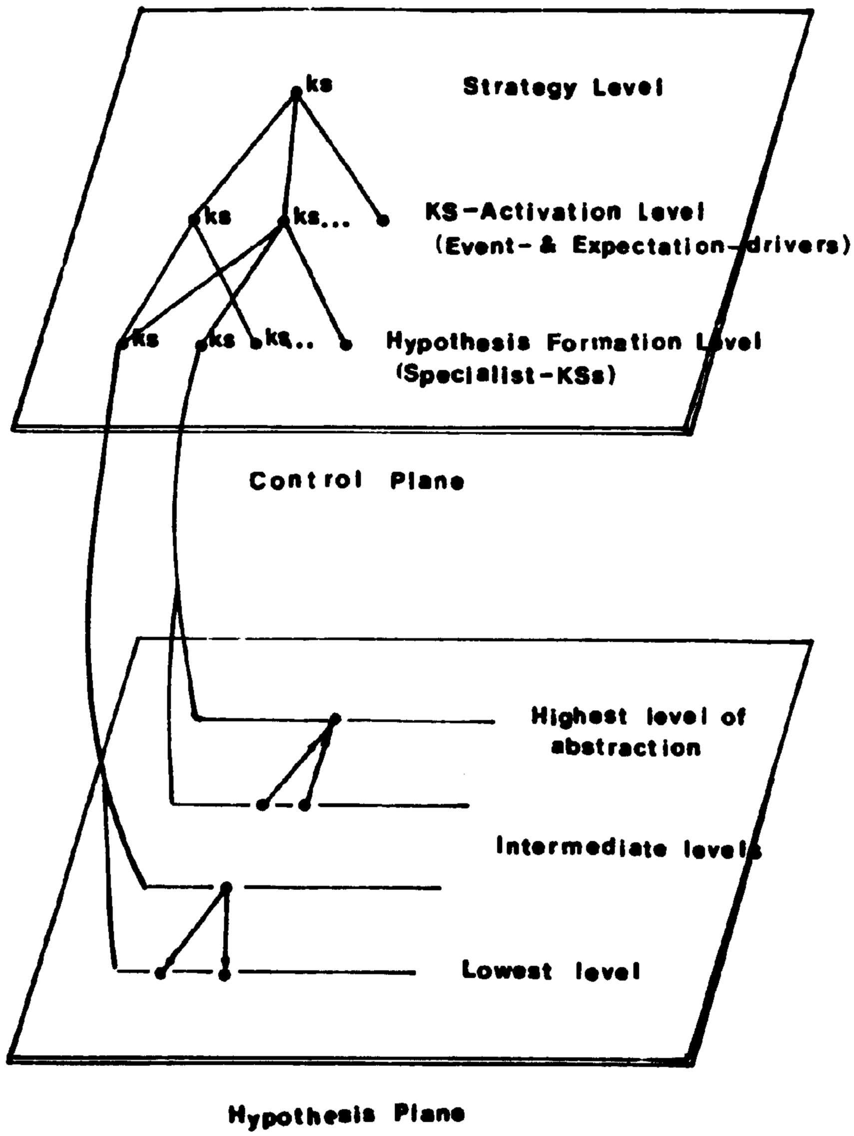
Figure 2.



Electron Density Space

Protein Model Space

Figure 3. Hypothesis Construction in the Protein Modeling Problem



Relationship between Hypothesis Hierarchy and Control Hierarchy

Figure 4.

Specialist-KS (on Hypothesis-formation Level)

Has access to:

1. primary data,
2. hypothesis elements,
3. facts, and
4. events in the Event history list.

May act to:

1. change the values of attributes of hypothesis elements or
2. change the links (relationships) in the hypothesis structure, and
3. inform the system of its actions by:
    a. putting on the Eventlist the type of changes that were made, or
    b. putting unresolved prolblems on the Problems-list, or
    c. ask to be recalled at a later time (generate time-based event).

Event- and Expectation-Drivers (on Knowledge-Source-Activation Level)

Has access to:

1. events on the Eventlist,
2. items on the Problems-list, and
3. time-based events.

May act to: invoke appropriate Specialist-KSs in an appropriate sequence to reflect its resource allocation policy.

Strategy-KS (on Strategy Level)

Has access to:

1. Eventlist,
2. Problems-list,
3. time-based events,
4. Current-Best-Hypothesis (or a summary of CBH if available), and
5. Event- and Expectation-Drivers.

May act to:

1. choose the appropriate KSs on the KS-Activation level, and/or
2. change the focus of attention (i.e. choose and event, a problem, a dormant region of the hypothesis, or a different region of the data to process next).

Summary of KS Activities on Different Control Levels

Table 1.

Bibliography

1. Davis, R., Buchanan, B.G. and Shortliffe, E.H. Production rules as representation for a knowledge-based consultation program. Stanford Computer Science Report STAN-CS-75-519, Stanford University, Oct., 1975. Journal of Artificial Intelligence, Vol. 8, No. 1 (February 1977), 15-45.

2. Davis, R. and King, J. An overview of production systems. Machine Intelligence 8: Machine Representation of Knowledge, Elcock, E.W. and Michie, D. (eds.), John Wiley, 1977. Also, Stanford Computer Science Report STAN-CS-75-524, Stanford University, 1975.

3. Engelmore, R.S., and Nii, H. P. A knowledge-based system for the interpretation of protein x-ray crystallographic data. Heuristic Programming Project Report HPP-77-2 (also STAN-CS-77-589), January, 1977.

4. Erman, L.D., and Lesser, V.R. A multi-level organization for problem solving using many, diverse, cooperating sources of knowledge. Proc. of the 4th IJCAI, 1975., 483-490.

5. Feigenbaum, E.A., Nii, H.P., et.al. HASP Final Report, vols. 1-4, Technical Report under ARPA Contract N66314-74-C-1235, Systems Control, Inc., Palo Alto, Ca., 1975. (Classified document)

6. Hayes-Roth, F., and Lesser, V.R., Focus of Attention in a distributed logic speech understanding system. Computer Science Department Technical Report, Carnegie Mellon University, January 12, 1976.

7. Lesser, V.R., Fennel, R.D. , Erman, L.D., and Reddy, R. Organization of the HEARSAYII speech understanding program. IEEE Transaction on Acoustic, Speech, and Signal Processing, 1975, ASSP-23, 11-23.

8. Shortliffe, E. H., et.al. Computer-based consultations in clinical therapeutics: explanation and rule acquisition capabilities of the MYCIN system. Computers and Biomedical Research, August, 1975.

9. Shortliffe, E. H., and Buchanan, B. G. A model of inexact reasoning in medicine. Mathematical Bioscience, 23, 351-379, 1975.