A Frame-based Production System Architecture

David E. Smith and Jan E. Clayton
Heuristic Programming Project*
Department of Computer Science
Stanford University

ABSTRACT

We propose a flexible frame-structured representation and agenda-based control mechanism for the construction of production-type systems. Advantages of this architecture include uniformity, control freedom, and extensibility. We also describe an experimental system, named WHEEZE, that uses this formalism.

Introduction

The success of MYCIN-like production systems [4] [7] [9] has demonstrated that a variety of types of expertise can be successfully captured in rules. In some cases, however, rules alone are inadequate necessitating the use of auxiliary representations (e.g. property lists for parameters in MYCIN). Other limitations result from the use of goal-directed control.

In this paper we outline a flexible schemata for constructing high performance production-like systems. The architecture consists of two components:

1. An extensible representation (utilizing a frame-structured language) which captures production rule knowledge.
2. An agenda-based control mechanism allowing considerable freedom in tailoring control flow.

We have used this architecture in the development of a system named WHEEZE, which performs medical pulmonary function diagnosis based on clinical test results. This system is based on two earlier efforts, PUFF [7], an EMYCIN-based production rule system [11], and CENTAUR [1] [2], a system constructed of both rules and prototypes.

An Alternative Representation for Productions

Figure 1 shows how a typical PUFF rule would be transformed into our representation. Each assertion is represented as a frame in the knowledge base, with antecedent sub-assertions appearing in its *Manifestation* slot. The number associated with each manifestation is its corresponding importance. Similarly, the certainty factor and findings from the rule are given separate slots in the assertion. Assertions appearing in the *SuggestiveOf* and *ComplementaryTo* slots are those worth investigating if the original assertion is comfirmed or denied respectively (numbers following these assertions are *suggestivities*).

Implicit in the production rule representation is a function which indicates how to compute the "belief" of the consequent assertions given belief in the antecedent assertion. Unfortunately, evaluation of the antecedent assertion involves modal logic (since greater shading is required than simple binary values for belief and disbelief). Therefore, a "HowToDetermineBelief" slot is associated with each assertion indicating how its belief is to be computed.

If: 1) The severity of Obstructive Airways Disease of the patient is less than or greater to mild, and

2) The number of pack-years smoked is greater than 0, and

3) The number of years ago that the patient quit smoking is 0

Then: It is definite (1000) that the following is one of the conclusion statements about this interpretation: Discontinuation of smoking should help relieve the symptoms.

OADwithSmoking:

Manifestation	((OAD-Present 10) (PatientHasSmoked 10) (PatientSillSmoking 10))
SuggestiveOf	((SmokingExacerbatedOAD 5) (SmokingInducedOAD 5))
ComplementaryTo	((OADwithSmoking-None 5))
Certainty	1000
Findings	"Discontinuation of smoking should help relieve the symptoms. "
HowToDetermineBelief	*function for computing the minimum of the beliefs of the manifestations*

Figure 1. English translation of PUFF rule (top) and Corresponding WHEEZE Frame for OADwithSmoking (bottom). Numbers appearing in the Manifestation, SuggestiveOf and ComplementaryTo slots are importance and suggestivity weightings.

----

*This research was supported in part by ONR contract N00014-79C-0302 and by NIH Biotechnology Resource Grant RR-00785.

The declarative nature of this representation facilitates modification and extension. For example, the addition of related knowledge, such as justifications, explanations, and instructional material, can be accomplished by the addition of slots to already existing assertions. The single uniform structure alleviates the need for any auxilliary means of representation.

Considerable efficiency has been gained by the use of rule compilation on production systems [10] [11]. We feel that a technique similar to this could also be used effectively on our representation but have not yet fully investigated this possibility.

AN AGENDA BASED CONTROL MECHANISM

Depth-first, goal-directed search is often used in production systems because questions asked by the system are focused on specific topics. Thus, the system appears to follow a coherent line of reasoning, more closely mimicking that of human diagnosticians. There are, however, many widely recognized limitations. No mechanism is provided for dynamically selecting or ordering the initial set of goals. Consequently, the system may explore many "red herrings" and ask irrelevant questions before encountering a good hypothesis. In addition, a startling piece of evidence (strongly suggesting a different hypothesis) *cannot* cause suspension of the current investigation and pursuit of the alternative.

Expert diagnosticians use more than simple goal-directed reasoning. They seem to work by alternately constructing and verifying hypotheses, corresponding to a mix of data- and goal-directed search. Furthermore, they expect these systems to reason in an analogous manner. It is desirable, therefore, that the system builder have control over the dynamic reasoning behavior of the system.

To provide this control, we employ a simple relaxation of goal- and data-directed mechanisms. This is facilitated by the use of an agenda to keep track of the set of goals to be examined, and their relative priorities. The control strategy is:

1. Examine the top assertion on the agenda.
2. If its sub-assertions (manifestations) are known, the relative belief of the assertion is determined. If confirmed, any assertions that it is *suggestive* of are placed on the agenda according to a specified measure of suggestivity. If denied, complementary assertions are placed on the agenda according to a measure of suggestivity.
3. If it cannot be immediately verified or rejected then its unknown sub-assertions are placed on the agenda according to a measure of importance, and according to the agenda level of the original assertion.

By varying the importance factors, *SuggestiveOf* values, and the initial items placed on the agenda, numerous strategies are possible. For example, if high-level goals are initially placed on the agenda and subgoals are always placed at the top of the agenda, depth-first goal-directed behavior will result. Alternatively, if low-level data are placed on the agenda initially, and assertions suggested by these data assertions are always placed below them on the agenda, breadth-first data driven behavior will result.

More commonly, what is desired is a mixture of the two, in which assertions suggest others as being likely, and goal directed verification is employed to investigate the likely assertions. The example below illustrates how this can be done.

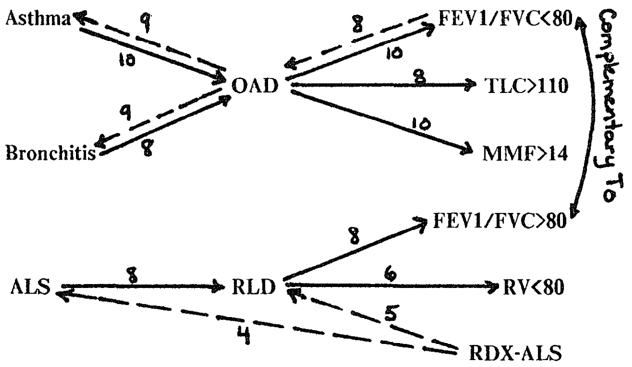

Figure 2. A simplified portion of the WHEEZE knowledge base. The solid lines indicate Manifesatation links (e.g. OAD is a manifestation of Asthma), the dashed lines represent SuggestiveOf links. The numbers associated with the links are the corresponding "importances" and "suggestivities" of the connections.

In the knowledge base of figure 2, suppose that RDX-ALS is confirmed, suggesting RLD to the agenda at level 6 and ALS at level 4. RLD is then examined, and since its manifestations are unknown, they are placed at the specified level on the agenda. The agenda now contains FEV1/FVC$\geq$80 at level 8, RV<80 and RLD at level 6, and ALS at level 4. FEV1/FVC$\geq$80 is therefore selected, and suppose that it is found to be false. Its complementary assertion (FEV1/FVC<80) is placed at level 8 on the agenda. and is immediately investigated. It is, of course, true, causing OAD to be placed at level 8 on the agenda. The diagnosis proceeds by investigating the manifestations of OAD; and, if OAD is confirmed, Asthma and Bronchitis are investigated.

While many subtleties have been glossed over in this example, it is important to note that:

1. The manipulation of *SuggestiveOf* and *importance* values can change the order in which assertions are examined, therefore changing the order in which questions are asked and results printed out. (In the example, FEV1/FVC was asked for before RV.)

2. Surprise data (data contrary to the hypothesis currently being investigated) may suggest goals to the agenda high enough to cause suspension of the current investigation. (The surprise FEV1/FVC value caused suspension of the RLD investigation in favor of the OAD investigation. If the suggestivity of the link from FEV1/FVC<80 to OAD were not as high, this would not have occurred.)

3. Low-level data assertions cause the suggestion of high-level goals, thus selecting and ordering goals to avoid irrelevant questions. (In the example, RLD and ALS were suggested and ordered by the low-level assertion RDX-ALS.)

Thus, extreme control flexibility is provided by this mechanism.

Besides the mechanism proposed above, there have been several other attempts to augment simple goal directed search. Meta-rules [5] can be used to encode strategic information, such as how to order or prune the hypothesis space. They could also be used, in principle, to suspend a current investigation when strong alternatives were discovered. In practice, however, meta-rules for accomplishing this task could be quite clumsy. In the CENTAUR system [1] [2], procedural attachment mechanisms (in disease prototypes) are used to capture the control information explicitly, and "triggering" rules serve to order the initial hypothesis space.

Our solution differs from these earlier attempts by proposing a single uniform control mechanism. It is sufficiently straightforward that tailoring of the control flow could potentially be turned over to the domain expert.

RESULTS

Not suprisingly, WHEEZE exhibits the same diagnostic behavior as its predecessors, PUFF and CENTAUR, on a standard set of 10 patient test cases. In refining the knowledge base, suggestivities and importance factors were used to great advantage to change the order in which questions were asked and conclusions printed out. This eliminated the need to carefully order sets of antecedent assertions.

The representation described has proven adequate for capturing the domain knowledge. In some cases, several rules were collapsed into a single assertion. In addition, the combination of representation and control structure eliminated the need for many awkward interdependent rules (e.g. rules with screening clauses).

Representation of both the rule and non-rule knowledge of the PUFF and CENTAUR systems has been facilitated by the flexibility of the architecture described. This flexibility is the direct result of the uniform representation and control mechanism. Further exploitations of this architecture appear possible, providing directions for future research.

ACKNOWLEDGEMENTS

We would like to thank Jan Aikins, Avron Barr, James Bennett, Bruce Buchanan, Mike Genesereth, Russ Greiner, and Doug Lenat for their help and comments.

REFERENCES

[1] Jan. S. Aikins. Prototypes and Production Rules: an Approach to Knowledge Representation for Hypothesis Formation. *Proc 6th IJCAI*, 1979. Pp. 1-3.

[2] Jan S. Aikins. *Prototypes and Production Rules: A Knowledge Representation for Computer Consultations.* Doctoral dissertation, Dept. of Computer Science, Stanford University.

[3] Jan S. Aikins. Representation of Control Knowledge in Expert Systems. *Proc 1st AAAI*, 1980.

[4] James Bennett, Lewis Creary, Robert Englemore and Robert Melosh. *SACON: A Knowledge Based Consultant for Structural Analysis.* Computer Science Report CS-78-699, Dept. of Computer Science, Stanford University, September 1978.

[5] Randall Davis and Bruce G. Buchanan. Meta-Level Knowledge: Overview and Applications. *Proc 5th IJCAI*, 1977. Pp. 920-927.

[6] Russell Greiner, Douglas Lenat. A Representation Language Language. *Proc 1st AAAI*, 1980.

[7] J. C. Kunz, R. J. Fallet, et. al.. *A Physiological Rule Based System for Interpreting Pulmonary Function Test Results.* HPP-78-19 (Working Paper), Heuristic Programming Project, Dept. of Computer Science, Stanford University, December 1978.

[8] Stephen G. Pauker and Peter Szolovits. Analyzing and Simulating Taking the History of the Present Illness: Context Formation. In Schneider and Sagvall Hein (Eds.), *Computational Linguistics in Medicine.* North-Holland, 1977. Pp. 109-118.

[9] E. H. Shortliffe. Computer-based Medical Consultations: MYCIN. New York: American Elsevier, 1976.

[10] William van Melle. A Domain-independent Production-rule System for Consultation Programs. *Proc 6th IJCAI*, 1979. Pp. 923-925.

[11] William van Melle. *A Domain-independent System that Aids in Constructing Knowledge-based Consultation Programs.* Doctoral dissertation, Dept. of Computer Science, Stanford University.