

HIERARCHICAL PRODUCTION SYSTEM

Richiro Mizoguchi and Osamu Kakusho
The Institute of Scientific and Industrial Research
Osaka University
Suita, Osaka, JAPAN 565

Abstract

Production systems have been used for representing domain-specific knowledge in a number of successful AI projects. In this short paper, we present a general concept of hierarchical production system(HPS). The HPS is an extended version of the current production system, in which two hierarchical structures of rules are employed. In one hierarchical structure, heuristic rules for conflict resolution are embedded and in another one, domain-specific rules are represented. The introduction of the hierarchical rule structures makes the HPS to be a useful framework for constructing knowledge-based systems.

1. INTRODUCTION

In the field of AI, production systems(PS) have been used for representing domain-specific knowledge in a number of successful projects such as DENDRAL[5] and MYCIN[16]. A PS[1][2][6][10][11][14][15][19][20] consists of three components, data base(working memory or short term memory), a set of condition-action rules, and interpreter. In the simplest case the interpreter operates by scanning the left hand side of each rule until one is found which is successfully matched against the data base. Then, the corresponding right hand side(action) is invoked and the data base is usually changed. In a pure PS all interaction between rules is forced through a very narrow channel, i.e., the data base. Such restriction on interaction gives PS's "modularity", which is one of the most remarkable advantages of PS's. Thanks to the modularity PS's are superior to procedural approaches in flexibility and adaptability.

One draw back of PS's is lack of programmability. Therefore, several PS's rather than pure ones adopt tags, markers, or conflict resolution methods to control the sequence of rule invocation. Consequently, however, they make it hard to understand what is intended.

In this short paper, we present a general concept of hierarchical production system(HPS). The HPS is an extended version of the current PS. It consists of several PS's arranged in a two-directional hierarchy to augment the ability

of the system while retaining its modularity and flexibility.

2. HIERARCHICAL PRODUCTION SYSTEM

As pointed out by Davis[1], PS's have appropriate domains in which they show good performance. By using PS's for knowledge representation in designing problem solving systems, one must decompose the problem domain into almost independent states and embed the domain-specific knowledge into a system regardless of the manner in which it is to be used. So, the competence of the system depends largely on the results of the decomposition.

It is well known that PS's provide a useful framework for embedding domain-specific knowledge into a system with high intelligence. However, a pure PS is probably inefficient because of its excessive modularity. Therefore, certain modifications are made in several dimensions[1][3][7][11][15] in order to augment the competence of the system. In this short paper, we present an additional dimension in which PS's are to be augmented, that is, we propose a hierarchical production system consisting of the current PS's arranged hierarchically.

Decomposition of the problem domains is an important task in building an efficient problem solving system. In speech understanding system, e.g., Hearsay-II [4][8][12], the problem domain is divided into 6 levels such as phrase, word-

sequence, word, syllable, segment, and parameter levels. Kanade[9] has also introduced a multi-level representation of image understanding domain(object, subimage, region, patch, and pixel levels). Nine-level of pattern description is introduced into learning system by Soloway and Riseman[18]. In their systems learning is considered as knowledge-directed interpretation in order to deal with real world domain where a significant amount of knowledge is required. These multi-level representations of the problem domains are natural to the human intuition, and they help the system designer to consider the problem topdown.

In the GPS paradigm, abstraction spaces are employed by Sacerdoti[17]. In his ABSTRIPS system plan making is done as follows: Suppose that the system wants to solve a certain problem. It usually has a collection of actions whose preconditions and effects are completely known. Initially, it makes a rough plan ignoring detailed situations. In this case, the first plan is a sequence of the actions in the highest space. Although it may be inapplicable to the problem domain directly, the plan does not miss the major causal chain. For example, it may contain the action "GO TO A FROM B" regardless of the transportation media or of their subactions required to be performed beforehand. Then, it can refine the plan using the actions on the next lower level. This process is continued until a realizable sequence of actions are obtained. This "Step-wise refinement" of a plan is useful especially when making a large scale plan.

The above observation shows that the hierarchical(multi-level) representation of problem domain and knowledge* plays a crucial role in dealing with complex problems.

Among the issues of PS design, conflict resolution is a major one. Davis[2][3] has discussed the conflict resolution problem by introducing meta-rules. Meta-rules are heuristic ones for guiding the selection of the next rule to be invoked from the conflict set and can be considered as knowledge concerning how to use the lower level knowledge. Furthermore, we can have meta-meta-rules. This hierarchy of rules enables us to embed the strategies into the system while retaining the uniformity of representation. Goldstein and Grimson[7] and Hayes-Roth and Lesser[8] have also considered a similar problem. Although their representations are different, the underlying ideas are same, that is, they want to make their systems efficient by firing appropriate rules in every recognition-act cycle

based on attached heuristic knowledge.

Taking the above two kinds of hierarchical knowledge structures into account, we obtain a general concept of hierarchical production system having the block diagram in Fig. 1.

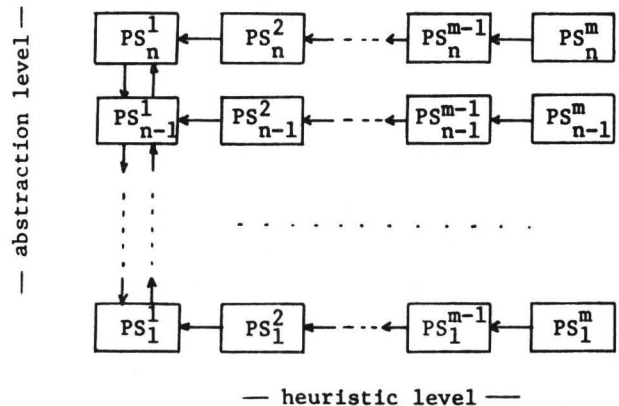


Fig. 1 Hierarchical production system, HPS_n^m . (HPS_1^1 corresponds to the current PS)

The HPS has the following characteristics:

1. The vertical dimension corresponds to the hierarchy of the domain-specific knowledge for step-wise construction of plans or something like that.
2. The horizontal dimension corresponds to the hierarchy of the meta-rules as strategies for conflict resolution.
3. Every component has a usual PS architecture.
4. Interaction between components is restricted to between neighboring ones.
5. Type of the interaction is only the access to the data base.

The HPS is designed to extend the current PS in a natural direction by introducing the hierarchical organization of rules, in which no ad-hoc techniques for conflict resolution are employed. In our system user-defined heuristics can be embedded in the form of rules as well as other knowledge. Furthermore, the hierarchy of the domain-specific knowledge helps the system to invoke appropriate rules, since a PS on a certain level can refer to the plan generated by the PS on the next higher level when determining which rule to fire. Thus, invocation of rules on every level is guided by its next higher level plans.

* The need of hierarchical representation of knowledge is recognized in knowledge representation paradigm[13].

3. CONCLUSIONS

A hierarchical production system has been proposed and its outline has also been described. The HPS enables us to represent the hierarchical domain-specific knowledge and several levels of heuristics.

The HPS has highly parallel computational structure in principle. In the ideal case where decomposition of domain-specific knowledge into a number of independent knowledge is made successfully, every rule and hence every component of the HPS can be activated in parallel. In practice, however, this is not the case. Therefore, we need a monitor or supervisor which arranges the flow of control to work the HPS.

REFERENCES

- [1] Davis, R. and J. King: "An overview of production systems", *Machine Intelligence*, 8, pp. 300-332 (1977).
- [2] Davis, R.: "Generalized procedure calling and content-directed invocation", *Proc. of the Symposium on Artificial Intelligence and Programming Languages*, pp. 45-54 (1977).
- [3] Davis, R. and B. G. Buchanan: "Meta level knowledge: Overview and applications", *Proc. of the 5th IJCAI*, pp. 920-927 (1977).
- [4] Erman, L. D. and V. R. Lesser: "A multi-level organization for problem solving using many, diverse, cooperating sources of knowledge", *Proc. of the 4th IJCAI*, pp. 483-490 (1975).
- [5] Feigenbaum, E. A., B. G. Buchanan, and J. Lederberg: "On generality and problem solving --- A case study involving the DENDRAL program", *Machine Intelligence*, 6, pp. 165-190 (1971).
- [6] Forgy, C. and J. McDermott: "The OPS reference manual", *Carnegie-Mellon Univ. Department of Computer Science* (1978).
- [7] Goldstein, I. P. and E. Grimson: "Annotated production systems -- A model for skill acquisition", *Proc. of the 5th IJCAI*, pp. 311-320 (1977).
- [8] Hayes-Roth, F. and V. R. Lesser: "Focus of attention in the Hearsay-II", *Proc. of the 5th IJCAI*, pp. 27-35 (1977).
- [9] Kanade, T.: "Model representations and control structures in image understanding", *Proc. of the 5th IJCAI*, pp. 1074-1082 (1977).
- [10] Lenat, D. B.: "Automated theory formation in mathematics", *Proc. of the 5th IJCAI*, pp. 833-842 (1977).
- [11] Lenat, D. B. and J. McDermott: "Less than general production system architecture", *Proc. of the 5th IJCAI*, pp. 928-932 (1977).
- [12] Lesser, V. R. and L. D. Erman: "A retrospective view of the Hearsay-II architecture", *Proc. of the 5th IJCAI*, pp. 790-800 (1977).
- [13] Minsky, M.: "A framework for representing knowledge", *The Psychology of Computer Vision*, McGraw-Hill, pp. 211-277 (1975).
- [14] Newell, A. and H. A. Simon: "Human problem solving", *Prentice-Hall* (1972).
- [15] Rychener, M. D.: "Control requirements for the design of production system architectures", *Proc. of the Symposium on Artificial Intelligence and Programming Languages*, pp. 37-44 (1977).
- [16] Shortliffe, E. H.: "MYCIN: A rule-based computer program for advising physicians regarding antimicrobial therapy selection", *Stanford Univ. Computer Science Report*, CS-74-465 (1974).
- [17] Sacerdoti, E. D.: "Planning in a hierarchy of abstraction spaces", *Artificial Intelligence*, 5, pp. 115-135 (1974).
- [18] Soloway, E. A. and E. M. Riseman: "Levels of pattern description in learning", *Proc. of the 5th IJCAI*, pp. 801-811 (1977).
- [19] Waterman, D. A.: "Generalization learning techniques for automating the learning of heuristics", *Artificial Intelligence*, 1, pp. 121-170 (1970).
- [20] Waterman, D. A.: "Adaptive production systems", *Proc. of the 3rd IJCAI*, pp. 296-303 (1973).