# STRUCTURE BASED CONTROL STRATEGY

C.M.Lou and J.Wang
Institute of Automation
Academia Sinica
Beijing, CHINA

## ABSTRACT

A control strategy is introduced. The tasks in the agenda are organized hierarchically, such that, in general, only tasks in the same level are allowed to compete with each other. KEY tasks with highest priority are also introduced to maintain the flexibility of control. This concept has been applied to a medical diagnosis expert system MDM, and it is implemented on UNIVAC 1100/10. The result is satisfactory.

## INTRODUCTION

This paper introduces a control method based on knowledge structures. The concept of agenda has been successfully applied to the expert systems by several authors (1)(2)(3)(4)(5)• The general agenda method used so far has some drawbacks:

* The efficiency is low as the system is complex.
* The formation of tasks is complicate.
* The formation *of reasons is* difficult.

The concept introduced in this paper is to attempt to overcome some of those drawbacks, especially the efficiency. The essential points of our concept are:

* We treat the reasoning process as a series of selection, evaluation, and generation of tasks. This is the same as in the ordinary agenda methods.
* The tasks in the agenda are organized hierarchically corresponding to the structure *of* knowledge, such that, in general, only tasks in the same level are allowed to compote with each other.
* There are some KEY tasks which have the highest priority, so that when one of them appears in the agenda, this task will be evaluated immediately.

This control strategy is applied to a medical diagnosis expert system MDM with success(6). The system is implemented on UNIVAC 1100/10.

## STRUCTURE OF TASKS

System MDM has a hierarchical structure of knowledge. The form of it is

COURSE . FRAME . SLOT.

The first level knowledge is COURSE. Different COURSES may be defined by peoples with different expertise. The second level knowledge is FRAME, which are units of knowledge of the COURSE correspondingly. The third level knowledge is SLOT, which are the properties of the corresponding FRAME. The set of slots of each FRAME is fixed for simplifying the treatment.

The structure of tasks corresponds to the structure of knowledge. The general form of the task is:

COURSE(x).FRAME(y).SLOT(z).OPERATION(w).,

where x, y, z, w are variables. For tasks of different levels, the corresponding forms are:

COURSE(x).OPERATION(w).
COURSE(x).FRAME(y).OPERATION(w).
COURSE(x).FRAME(y).SLOT(z).OPERATION(w).,

OPERATION consists *of* three elements, they are:

CHECK: to check the content of the task, and see whether it is necessary to evaluate further.
EVALUATION: to evaluate.
DICISION: according to the result obtained, decide the inference direction.

As an example,

COURSE(A).FRAME(B).SLOT(CAUSED-BY).CHECK.

is a task which means to see whether the cause of desease of FRAME B in COURSE A is necessary to be evaluated.

Besides, there are two special tasks which are used frequently. One is

TEST(x),

which asks to conduct some experimental test, and x is the title, e.g., WBC. The other one is

COURSE(x).FRAME(y).COUNT.,

which performs the calculation *of* reliability for the result obtained by the proceeding tasks so far, and the resultant should be returned.

## HEURISTIC RULES

There are two categories of heuristic rules in system MDM:

* The rules that are subordinate to tasks: when a task is being evaluated, rules are selected according to the environment, and its reasoning conclusion is the operation result of the task. E.g.,

  IF: 1) it is necessary to evaluate the cause of the desease, which is represented by an instance of a frame, and
  2) this instance- i.• not confirmative,
  THEN: generate tasks:
  1) COURSE.FRAME.CAUSED-BY.EVAL.,
  2) COURSE.FRAME. CHECK.

  This rule is attached to the task

  COURSE(x) .FRAME(y) . SLOT ( CAUSED-r;Y) .CHECK.

  As another example, a rule may form the judgement on the instance of a frame as reliable, unreliable, or the case in between.

  IF: 1) the score of a frame is below the threshold,
  2) all the contents in slot DG1 are unreliable,
  THEN: this frame is reliable.

* The rules which are subordinate to the system: which are devided into several groups, such as:

  ** initiation rules,
  ** rules for inheritence,
  ** rules that generate reasons,
  ** rules that generate tasks, etc..

  The meaning of these rules will be given in the succeeding sections.

## DISPATCHING TASKS

When using ordinary method, all tasks put in the agenda compete with each other. The one with highest priority is to be selected and evaluated. The heuristic rule for the corresponding control is as:

IF: 1) there are some tasks in the agenda whose priority values are above the threshold, and
2) $X_i$ is the task with highest priority,
THEN: evaluate $X_i$.

The control mechanism of our system is somewhat different. That is, it controls in a hierarchical way. The corresponding rule is:

IF: 1) there is no KEY task in the agenda, and
2) at the moment, there are executable tasks in several levels of agenda, among which, the $N^{th}$ level is with the highest priority,
3) $X_i$ is the task in N level with highest priority,
THEN: evaluate $X_i$. .

The priorities of different levels *of* the agenda are given a priori, such that, level-3 with highest value, and level-1, the lowest.

In general, there is no competition between tasks in difference levels. The advantages of this strategy are obvious. In short,

* the efficiency of search is increased,
* the formation of reasons is simplified,
* the aims of competition between tasks are more clear.

There are some KEY tasks involved in the system, which are generated by some heuristic rules. When the KEY task appears in the agenda, it will be evaluated immediately without exception. One of the purposes of introducing these KEY tasks is to establish the connections of tasks between different agenda levels. At critical moment, the KEY task will change the direction of control of the whole operation.

There are two categories of KEY tasks:

* When a task has been evaluated, the resultant may transfer some tasks to be the KEY tasks, e.g. ,

  IF: a FRAME or a COURSE is confirmed,
  THEN: the DECISION task related to this FRAME or COURSE is a KEY task.

* All tasks in the agenda are with their priorities below the threshold, but there might be some tasks, the evaluation of which will influence the final results of the operation. These tasks are assigned to be the KEY tasks. E.g.,

  IF: 1) in agenda, there is no task whose priority is above the threshold,
  2) there is still a DECISION task in the agenda,
  THEN: this task is a KEY task.

We treat the reasoning process as a series of selection, evaluation, and generation of tasks. Each level of the agenda maintains a high threshold value. The condition for evaluating a task is that its priority is over this threshold. It also maintains a low threshold value. The condition for generating a new task is the score of its initial reason being over this threshold.

## REASONS FOR TASKS

There are two categories of reasons formed in the system:

* reasons formed in the new task as it is generated, e.g.,

  IF: the new task EVALUATION is generated from the task CHECK,
  THEN: add reason to the new task 200

* reasons caused by the environment, e.g.,

IF: 1) in TEST, tasks with risk equal to zero are all evaluated,
2) the instance of FRAME(x) is not affirmative,
3) there is still a CAUSED-BY task in the agenda,
THEN: add reason to the task CAUSED-BY 60.

The summation of the values of different reasons of a task is the total score of that task.

## INHERITANCE OF DEDUCTION INFORMATION

After the evaluation, the task will be cleared out. It is very important to preserve some of the information for further deduction. In MDM, there are some heuristic rules conducting this information inheritance. The related information is stored in a frame-like structure as shown in Fig.1. It can be applied to the succeeding operations by matching the corresponding rules. For example,

IF: 1) task x generates task y,
2) y is a deeper task,
THEN: inherit information PRODUCING, PRODUCED, SCORE.

TASK NAME:
PRODUCING TASK, NAME:
PRODUCED TASK, NAME:
REASON:
SCORE RELATED TO:

Fig.1

## RESULTS

The structure-based control strategy is applied to a medical diagnosis expert system MDM, which is implemented on UNIVAC 1100/10, The result is satisfactory. The following is a part of sequence of tasks generated. It shows the direction of inference controlled by this method.

RUN TASK: 13 PRIORITY: 80
RUN TASK NAME: SHOCK .SHOCK .DECISION
KEY TASK: YES
1. PRODUCE DECISION 80

PRODUCED TASK: 25 RULE: 5
PRODUCED TASK NAME: SHOCK .CARDIOGENIC .CHECK
THE REASON IS:
1. FRAME- -FRAME 160

RUN TASK: 14
RUN TASK NAME: SHOCK .SHOCK .TEST .DECISION
KEY TASK: NO
THE REASON IS:
1. PRODUCE DECISION 80
2. PRODUCE DECISION 80
3. COURSE- -FRAME 210
4. A FRAME BE CERTAINED AMD THIS FRAME IS PRODUCED 60

## CONCLUSION

In this paper we introduce the concept of hierarchical structure to the agenda. The strategy constrains the competition between tasks, simplifies the reason formation mechanism, and preserves the flexibility of the ordinary agenda by introducing the KEY tasks. This strategy is especially important in developing complex systems, such as systems including knowledge acquisition, learning, and problem solving, with higher control efficiency. This is the eventual aim of our work,

## REFERENCES

1. Bruell,P. (1979): "An Agenda Driven Theorem Prover." ATP-53, The Uni. of Texas at Austin, Texas, *U.S.A.*

2. Lenat,D.B. (1982): "AM: Discovery in Mathematics as Heuristic Search." In Knowledge-Based Systems in Artificial Intelligence, by Davis,R., Lenat,D.B., McGraw-Hill, 1-225.

3. Lesser,V.R., Erman,L.D. (1977): "A Retrospective View of the Hearsay-II Architecture." Proc. 5 IJCAI, Cambridge, U.S.A., 790-800.

4. Smith,D.E., Clayton,J.E. (1980): "A Frame-Based Production System Architecture", Proc. 1st AAAI. Stanford, U.S.A., 15-4-156.

5. Tyson,W.M. (1981): "APRVR: A priority-Ordered Agenda Theorem Prover." ATP-66, The Uni. of Texas at Austin, Texas, U.S.A.

6. Wang,J. (1981): "MDM: A Medical Decision Making Model." M.S. Thesis, Institute of Automation, Academia Sinica, Beijing, CHINA.