UNIFYING DATA-DIRECTED AND GOAL-DIRECTED CONTROL:

An Example And Experiments

Daniel D. Corkill, Victor R. Lesser, and Eva Hudlicka

Computer and Information Science Department
University of Massachusetts
Amherst, Massachusetts, 01003

ABSTRACT

Effective control in a multi-level cooperating knowledge source problem solver (such as Hearsay-II) requires the system to reason about the relationships among competing and cooperating knowledge source (KS) instantiations (both past and potential) that are working on different aspects and levels of the problem. Such reasoning is needed to assess the current state of problem solving and to develop plans for using the system's limited processing resources to the best advantage. The relationships among KS instantiations can be naturally represented when KS activity is viewed simultaneously from a data-directed and a goal-directed perspective. In this paper we show how data- and goal-directed control can be integrated into a single, uniform framework, and we present an example and experimental results of sophisticated focusing using this framework.

I INTRODUCTION

The multi-level cooperating knowledge source model of problem solving, as posited by the Hearsay-II architecture, poses interesting control problems. Effective control using such a problem solving approach requires the control component to reason about the relationships among competing and cooperating knowledge source (KS) activities (both past and potential) and among KS activities working on different aspects and levels of the problem. Such reasoning is required in order to assess the current state of problem solving and to determine how the system should use its limited processing resources to the best advantage.

For example, the control component needs to develop and reason about sequences of KS activities

relating to a particular approach to one aspect of the problem. This allows these activities to be scheduled as a coherent unit and to be eliminated as a unit if the approach proves unproductive. A second example is the implementation of an opportunistic scheduling strategy where the partial solution of a high-level problem is used to focus the system on low-level activities required to solve the remainder of the problem (focus of attention through subgoaling). Other examples are selecting a specialized KS to resolve the system's confusion over competing partial solutions and instantiating activities to produce input data necessary for performing an important activity (precondition-action backchaining). All of these examples rely on the control component's ability to evaluate the the potential effects of KS activities from a non-local context.

The data-directed and instantaneous scheduling mechanisms developed for the Hearsay-II speech understanding system could reason about KS relationships in only a rudimentary way [5]. That level of reasoning was sufficient for the KSs used in the final configuration of that speech system [3]. However, the limitations of this rudimentary control have become increasingly apparent to us and others as the multi-level cooperating KS model has been applied to different task domains [1].

Nii and Feigenbaum with SU/X [8], Engelmore and Nii with SU/P [2], and Erman, et al., with Hearsay-III [4] recognized these limitations and consequently have developed systems with enhanced control capabilites. These enhancements permit more sophisticated control over scheduling by allowing the KS scheduling queues to be manipulated under program control. However, these modifications do not explicitly formalize the relationship among KS activities. Such relationships are left to the user to build. We feel these relationships need to be explicitly formalized if domain-independent control strategies are to be developed. The premise of this paper is that these relationships become apparent in a control framework in which KS activity can be viewed simultaneously from a data-directed and a goal-directed perspective.

In this paper, we first review the data-directed scheduling mechanisms of Hearsay-II. Next, we indicate how data-directed and goal-directed control can be integrated into a single, uniform framework through the generation of goals from data-directed events and we show the structural relationships among KS activities that

this framework creates. We show an example of this framework performing sophisticated focusing of KS activity and present experimental results that show the advantages of the unified approach over a purely data-directed approach to control.

II DATA-DIRECTED HEARSAY-II SCHEDULING

Figure 1 presents a high-level schematic for data-directed control in Hearsay-II. KSs are invoked in response to particular kinds of changes on the blackboard, called blackboard events. The blackboard monitor knows which events at which levels interest each KS. The occurence of a blackboard event does not guarantee that there is, in fact, sufficient information on the blackboard for a KS to be executed. The blackboard monitor executes a precondition procedure for each interested KS to make a more detailed examination and, if sufficient information is found, a KS instantiation (KSI) is created and placed onto the scheduling queue. The scheduler calculates a priority rating for each KSI on the scheduling queue, selecting for execution the one with the highest rating. Execution of the KSI causes changes to the blackboard which trigger additional blackboard events and the process continues.

Although this data-directed Hearsay-II architecture has many advantages, it is severly limited in its ability to plan its interpretation activities. Scheduling is instantaneous — only the immediate effects on the state of problem solving are considered. There is no inference process used to determine the effects of executing

a KS beyond its immediate effects on the system state. Another limitation of this scheduling approach occurs when the precondition procedure cannot find sufficient information for the KS to be instantiated. The scheduler does not record which information is missing and has no way of re-evaluating the priorities of a pending KS that can generate the missing information or instantiating the KS if it is not already present. In the data-directed architecture it is assumed that if the information is really important it will eventually be generated based on normal scheduling considerations.

To remedy these control limitations within the basic Hearsay-II architecture, we next present an augmented version of the architecture that integrates data- and goal-directed control of KS activity via the generation of goals from blackboard events. Within this augmented architecture, a wide range of scheduling paradigms can be implemented efficiently: from those based on an instantaneous, statistical and data-directed approach to those based on complex planning of goal-directed activity. In this way, the system developer can tailor the control to the specifics of the task domain and KS configuration.

III GOAL-DIRECTED HEARSAY-II SCHEDULING

Figure 2 presents a high-level schemantic of Hearsay-II as augmented to accomodate goal-directed scheduling. A second blackboard, the goal blackboard, is added that mirrors the original (data) blackboard in dimensionality. The goal blackboard
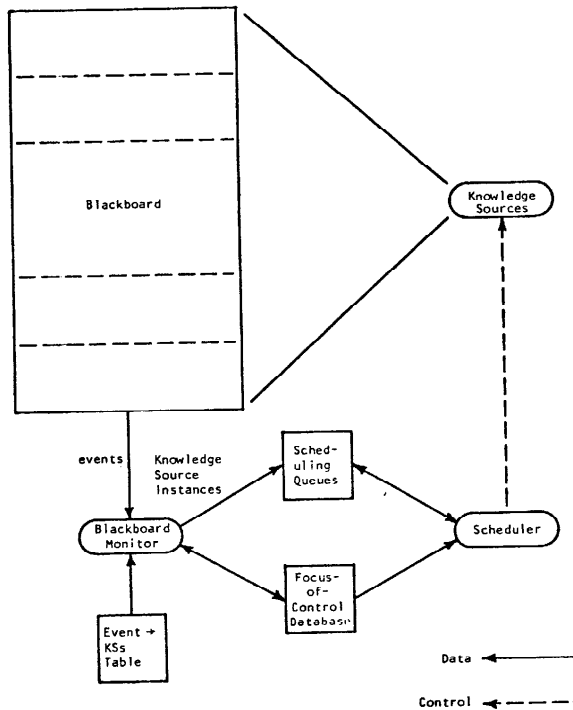
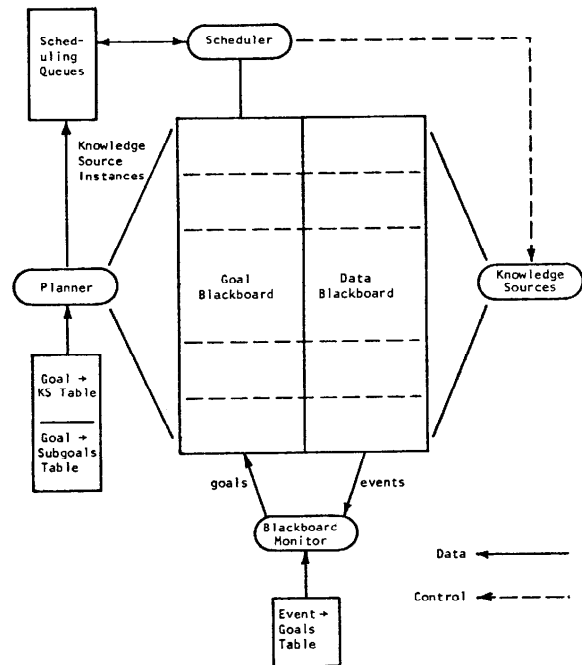FIGURE 1: Data-Directed Hearsay-II Architecture

FIGURE 2: Goal-Directed Hearsay-II Architecture

contains goals, each representing a request to create a particular state of hypotheses on the data blackboard in the (corresponding) area covered by the goal. For example, a simple goal would be a request for the creation of a hypothesis with specific attributes above a given belief in a particular area of the data blackboard.

The integration of data-directed and goal-directed control into a single, uniform framework is based on the following observation:

> The stimulation of a precondition process in the data-directed architecture not only indicates that it may be possible to execute the knowledge source, but that it may be desirable to do so in order to achieve the goal implicit in the output of the KS.

In order to make these implicit goals explicit, the event-to-KS mapping contained in the blackboard event table is split into two steps: event-to-goals and goals-to-KSs. The blackboard monitor watches for the occurrence of a data blackboard event, but instead of placing KSIs on the scheduling queue, it uses the event-to-goals mapping to determine the appropriate goals to generate from the event and inserts them onto the goal blackboard. Goals may also be placed on the goal blackboard from external sources. Placing a high-level goal onto the goal blackboard can effectively bias the system towards developing a solution in a particular way.

A new control component, the planner, is also added to the architecture. The planner responds to the creation of goals on the goal blackboard by developing plans for their achievement. In their simplest form these plans consist of goal/KSI relationships which specify one or more KSs which can potentially satisfy the created goals. The planner uses the goal-to-KS mapping to create these KSIs. More sophisticated planning activities consist of building goal/subgoal, precondition goal/KSI, and overlapping goal relationships [1]. The scheduler uses the relationships between the KSIs and the goals on the goal blackboard as a basis for its scheduling decisions.

We have implemented a version of the goal-directed Hearsay-II architecture in a distributed interpretation system which produces a dynamic map of vehicles moving through a geographical area [7]. Figure 3 shows how goal-directed focusing can be used in this application to increase the priority rating of low-level KSI based on the creation of a high-level hypothesis. The processing levels in order of increasing abstraction are: signal location (SL), group location (GL), vehicle location (VL), and vehicle track (VT).*

The creation of SL hypothesis H:SL:01 on the data blackboard causes the planner to create GL goal G:GL:01 on the goal blackboard. This goal indicates that the system should attempt to form a GL hypothesis using H:SL:01. The planner next

---

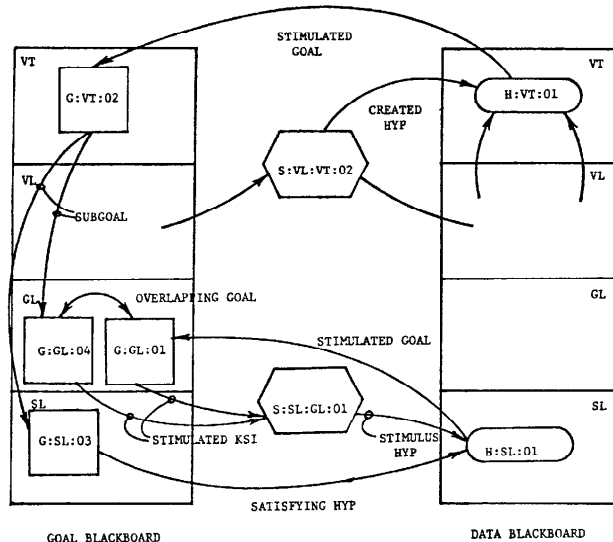*. Additional processing levels used in the system are omitted here.

FIGURE 3: An Example

instantiates KSI S:SL:GL:01 to try to achieve this goal. The rating of a KSI is a function of the belief of its stimulus hypotheses and the priority rating of its stimulus goals (if any). The priority of a goal is a function of the belief of its stimulus hypotheses, its level on the blackboard, and its relationships with other goals. Assume that H:SL:01 is weakly believed and consequently S:SL:GL:01 is given a low execution rating. Processing continues with other SL hypotheses and eventually creates a VT hypothesis H:VT:01 with a moderately high belief. The creation of this hypothesis causes a number of goals to be created, including the goal shown in the figure, G:VT:02. This goal indicates that the system should attempt to extend H:VT:01.

The planner uses domain knowledge in the form of a goal-to-subgoal mapping for decomposing this high-level goal into a SL level subgoal, G:SL:03. This subgoal indicates in what area is necessary to have SL hypotheses in order to eventually extend the VT hypothesis. Subgoal G:SL:03 is given the same priority rating as its parent goal G:VT:02. The planner finds that H:SL:01 has already been created in this area and can satisfy G:SL:03. The planner then creates subgoal G:GL:04 and finds that goal G:GL:01 overlaps with it. The planner adds G:GL:04 as a second stimulus goal of the low-rated KSI S:SL:GL:01. The addition of the higher priority goal causes the rating of the KSI to be increased based on its potential contribution to the track extension goal G:VT:02.

Subgoaling can reduce the combinatorics often associated with the top-down elaboration of hypotheses. Top-down elaboration is generally used for two different activities: the generation of the lower-level structure of a hypothesis (to discover details) and the determination of which existing low-level hypotheses should be driven-up to create or verify a high level hypothesis based on

expectations (for focusing). Top-down elaboration of hypotheses is best suited only to the first activity -- subgoaling on the goal blackboard is a more effective way to perform expectation-based focusing. When hypothesis elaboration is used as a focusing technique, the elaboration process has to be conservative in order to reduce the number of hypotheses generated and to reduce the possibility of generated low-level hypotheses being used as "real data" by knowledge sources in other contexts. Because subgoals are distinct from hypotheses, they can be liberally abstracted (such as supplying a range of values for an attribute) and under-specified (such as supplying a "don't care" attribute). Therefore, subgoaling the high-level goal of generating the expectation-based hypothesis (including the use of "level-hopping") avoids the combinatorial and context confusion problems associated with the use of top-down hypothesis elaboration for focusing.

Planning operations, such as subgoaling and precondition goal/KSI chaining, permits sophisticated opportunistic focusing to be performed by the planner and scheduler. Highly rated low-level hypotheses can be driven up in a data-directed fashion while high-level goals generated from strong expectations can be subgoaled downward to control low-level synthesis activities (as in the above example). Similarly, processing in low rated areas can be stimulated if a highly rated knowledge source requires the creation of a precondition goal in that area.

IV FOCUS OF ATTENTION EXPERIMENTS USING SUBGOALING

We are beginning to experiment with the use of subgoaling as a focus of attention mechanism. Our goal is a set of rigorous experiments that quantify those situations in which subgoaling outperforms a simpler, purely data-directed approach. The characteristics we are varying include the confusability of the input data and the power of the KSs to resolve this confusability and to make effective predictions. To vary the power of KSs we are using a semi-formal model for simulating KSs of different power through the use of an oracle [6,7]. We also plan to vary the weighting factor used by the scheduler for evaluating KSIs. This weighting determines, in part, the balance between data-directed and goal-directed focusing by adjusting the relative contributions of the priority of goals that are potentially satisfied by a KSI and the predicted quality of the hypotheses produced by the KSI.

Figure 4 illustrates a simple scenario in which subgoaling high-level expectations effectively reduces the amount of processing required to generate the correct answer. In this figure, there are two tracks: one representing the signals from an actual vehicle and the other a false ghost vehicle. The actual track data consists of a sequence of high belief SL hypotheses surrounding an area of low belief SL hypotheses. The ghost track consists of a uniform sequence of medium belief SL hypotheses. The two tracks are

LEGEND

———————— True Signal
— — — — — — Ghost Signal
·—·—·—·—·— False Signal

▬▬▬▬▬ Low Belief
▭▭▭▭▭ Medium Belief
▭▭▭▭▭ High Belief

TRACK HYPOTHESES CREATED WITHOUT SUBGOALING

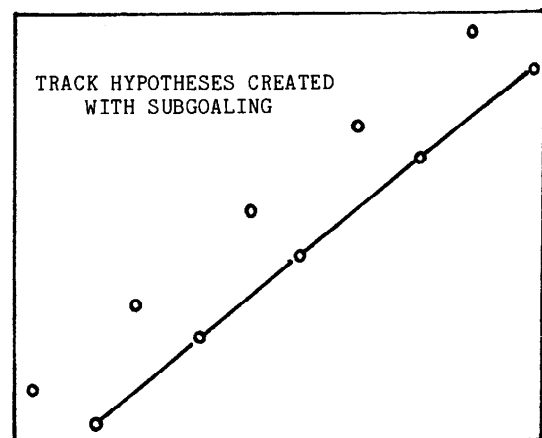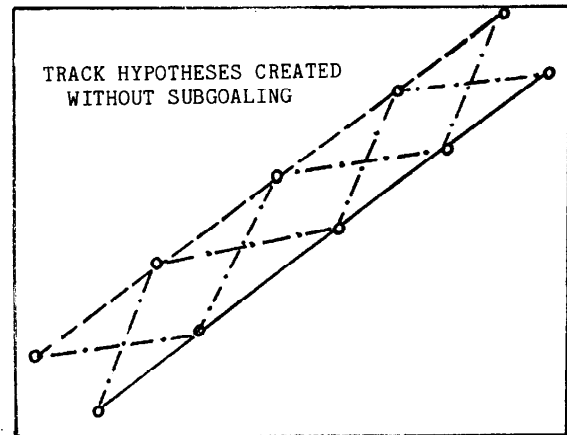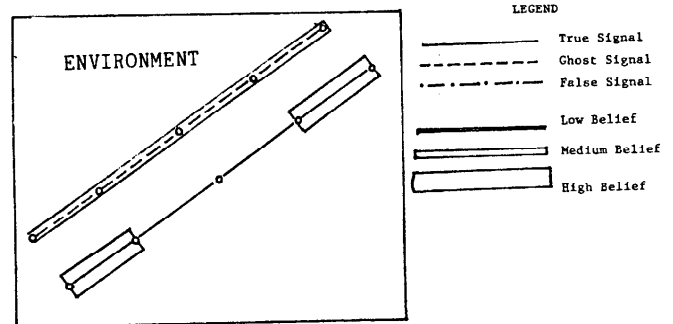TRACK HYPOTHESES CREATED WITH SUBGOALING

FIGURE 4: The Experiment

sufficiently close that the system can produce track hypotheses composed of locations from both the actual and the ghost track. For simplicity, each vehicle is assumed to emit a single signal frequency.

Without focusing through the creation of subgoals, the system executes 54 KSIs to completely generate the correct track. With subgoal focusing

based on subgoaling at the VT level the system requires 28 KSIs. This significant speedup comes from the system avoiding considerable work in attempting to develop track hypotheses that integrate high belief data from the actual track with medium belief false data and by interconnecting medium belief false data before extending high belief actual data with low belief actual data. In purely data-directed scheduling, these activities seem reasonable from the scheduler's local view of the effects of KS actvity. However, by affecting the decisions of the scheduler with subgoals which represent predictions, much of the system work involved in processing data that partially correlates with the actual track or that is of medium belief can be avoided.

V CONCLUSION

We have shown how data- and goal-directed control can be naturally integrated into a single uniform control framework, permitting the development of a wide range of different scheduling and planning strategies for controlling knowledge source (KS) activity. This framework increases the number of task domains in which the multi-level, cooperative KS model of problem-solving (used in the Hearsay-II architecture) is an effective approach.

We have also presented an example and an experiment indicating the potential advantages of a sophisticated focusing strategy based on subgoaling over a purely data-directed strategy which does not construct complex relationships among KS activities.

REFERENCES

1. Daniel D. Corkill and Victor R. Lesser.
   A goal-directed Hearsay-II architecture: Unifying data-directed and goal-directed control.
   Technical Report 81-15, Computer and Information Science Department, University of Massachusetts, Amherst, Massachusetts, 01003, June 1981.

2. Robert S. Engelmore and H. Penny Nii.
   A knowledge-based system for the interpretation of protein x-ray crystallographic data.
   Technical Report Stan-CS-77-589, Computer Science Department, Stanford University, Stanford, California, February 1977.

3. Lee D. Erman, Frederick Hayes-Roth, Victor R. Lesser, and D. Raj Reddy.
   The Hearsay-II speech understanding system: Integrating knowledge to resolve uncertainty.
   Computing Surveys 12(2):213-253, June 1980.

4. Lee D. Erman, Philip E. London, and Stephen F. Fickas.
   The design and an example use of Hearsay-III.
   Proceedings Seventh International Joint Conference on Artificial Intelligence, pages 409-415, August 1981.

5. Frederick Hayes-Roth and Victor R. Lesser.
   Focus of attention in the Hearsay-II system.
   Proceedings Fifth International Joint Conference on Artificial Intelligence, pages 27-35, August 1977.

6. Victor R. Lesser, Jasmina Pavlin, and Scott Reed.
   Quantifying and simulating the behavior of knowledge-based interpretation systems.
   In Proceedings of the First Annual National Conference on Artificial Intelligence, pages 111-115, August 1980.

7. Victor Lesser, Daniel Corkill, Jasmina Pavlin, Larry Lefkowitz, Eva Hudlicka, Richard Brooks, and Scott Reed.
   A high-level simulation testbed for cooperative distributed problem solving.
   Technical Report 81-16, Department of Computer and Information Science, University of Massachusetts, Amherst, Massachusetts, 01003, June 1981.

8. H. Penny Nii and Edward A. Feigenbaum.
   Rule based understanding of signals.
   In D. A. Waterman and Frederick Hayes-Roth, editors, Pattern-Directed Inference Systems, pages 483-501, Academic Press, 1978.