Planning for the Control of an Interpretation System*

Norman Carver and Victor Lesser

COINS Technical Report 89-39
April 1989

Abstract

Interpretation is a complex and uncertain process which requires sophisticated evidential reasoning and control schemes. We have developed a framework which models interpretation as a process of gathering evidence to manage uncertainty. The key components of the approach are a specialized evidential representation system and a control planner with heuristic focusing. The evidential representation scheme includes explicit, symbolic encodings of the sources of uncertainty in the evidence for the hypotheses. This knowledge is used by the control planner to identify and develop strategies for resolving the uncertainty in the interpretations. Since multiple, alternative strategies may be able to satisfy goals, the control process can be seen to involve a search. Heuristic focusing is applied in parallel with the planning process in order to select the strategies to pursue and control the search. The control plan framework allows the use of a flexible focusing scheme which can switch back and forth between strategies depending on the nature of the developing plans and changes in the domain.

*This work was supported by the Air Force Systems Command, Rome Air Development Center, Griffiss Air Force Base, New York 13441-5700, and the Air Force Office of Scientific Research, Bolling AFB, DC 20332 under Contract No. F30602-85-C-0008. This contract supports the Northeast Artificial Intelligence Consortium (NAIC).

1 Introduction

Interpretation is the process of determining a high-level, abstract view of a set of data based on a hierarchical specification of possible viewpoints. Hypotheses representing interpretations of some subset of the data are developed using the hierarchical support relations to identify the legal "evidence" for the hypotheses. Complex interpretation tasks require sophisticated evidential reasoning and control schemes which can deal with the uncertainty inherent in the interpretation process. For example, combinatorial considerations preclude complete construction and evaluation of all potential interpretations: control must be exercised over the creation and refinement of hypotheses. This means that the system must compare alternative hypotheses based on partial knowledge without being sure whether it has even created all of the correct interpretations. In many domains the uncertainty is compounded by the volume of data being too large to be completely considered and/or by the data being uncertain and possibly incorrect.

We have developed an interpretation framework based on a model of interpretation as a process of gathering evidence to manage uncertainty. The key components of the approach are a specialized evidential representation system and a control planner with heuristic focusing. The evidential representation scheme includes explicit, symbolic encodings of the *sources of uncertainty* in the evidence for the hypotheses. That is, evidence provides uncertain support for a hypothesis because there are conditions under which the evidence may fail to support the hypothesis: the sources of uncertainty in the evidence. For example, while some piece of sensor data in a vehicle monitoring system can be used to support a particular vehicle hypothesis, the resulting evidence is uncertain because the data may actually be due to a sensor malfunction or may support a competing, alternative vehicle hypothesis. Our model of interpretation associates a set of sources of uncertainty with the evidence for the hypotheses: partial evidence, uncertain evidence inference, uncertain evidence premise, alternative evidence interpretation (representing the relations between alternative hypotheses), conflicting evidence, etc. This knowledge is used by the control process in elucidating strategies for meeting the problem-solving goals since the *purpose* of interpretation actions is to resolve the uncertainty in the hypotheses. The evidential representation scheme is discussed in section 2.1.

Control decisions are made through an incremental planning process which identifies, selects, and implements problem-solving strategies. The available strategies are defined as a hierarchy of *control plans*. Control planning involves determining the subgoals of the current plan which must be satisfied next and then matching each subgoal to the possible control plans to determine how it might be satisfied. Primitive control plans represent actions and have corresponding functions for executing the actions. Planning and execution are interleaved—that is, the plans are only elaborated to the point of selecting the next action—because the outcome of actions is uncertain. In general, there will be many partial control plan instances which could be further elaborated at any point in the processing—i.e., many possible strategies for pursuing the system goals. The alternative control plan instances represent the choices of which hypotheses to resolve uncertainty in, what sources of uncertainty to eliminate, and how to do it. Thus, one of the major issues for interpretation systems is the development of an effective focus-of-attention scheme. In our system focusing is accomplished as part of the multi-stage process of refining and elaborating the control plans. Maintaining a framework of control plan instances allows the system a great deal of control flexibility since the focus-of-attention can move back and forth between strategies by refocusing in the plan instance hierarchy. Refocusing decisions are based on data-directed factors such as the outcome of control plans, the characteristics of the developing interpretation hypotheses, and data availability. Control planning and focusing are discussed in section 2.2.

The work presented here grew out of our experience developing a focusing scheme for plan recognition [1]. Plan recognition systems have tended to ignore the practical aspects of focusing the interpretation process and of comparing alternative hypotheses to determine just what it is that the system believes. The scheme we developed used an explicit record of the application of focusing heuristics to guide the system and allow it to revise its interpretations. However, the granularity of the control was too coarse for many domains, the focusing assumptions information was of limited value in controlling backtracking, and hypothesis uncertainty was confused with the control decisions (see [2]). Of interest to us also has been work on planning for control by Clancey [4], Hayes-Roth [7], and Durfee and Lesser [6]. However, none of these systems provide a completely suitable framework for interpretation. Clancey's tasks and meta-rules are really control plans and their substeps. The framework is limited by the fact that meta-rules directly invoke subtasks so there is no ability to search for and consider alternative strategies; focusing is implicit in the meta-rule preconditions. Of course, this may be fine for classification problems [3] which can be more exhaustive than can interpretation systems. The Hayes-Roth work on blackboard systems for control is more general than our work (which concentrates on interpretation). However, this generality means that little guidance is provided in how to structure control knowledge—e.g., the need for uncertainty knowledge as a basis for elaborating control plans. Another drawback of the control blackboard approach is its reliance on an agenda mechanism which must consider all possible actions on each loop. By contrast, focusing in parallel with the control plan hierarchy as we do provides, in effect, a partitioned agenda: only those actions immediately applicable to the in-focus plan or goal are considered. The incremental planning approach of Durfee and Lesser builds abstract models of the interpretation data and uses these models to guide further processing. This idea can be handled as one type of problem-solving strategy in our system with the addition of

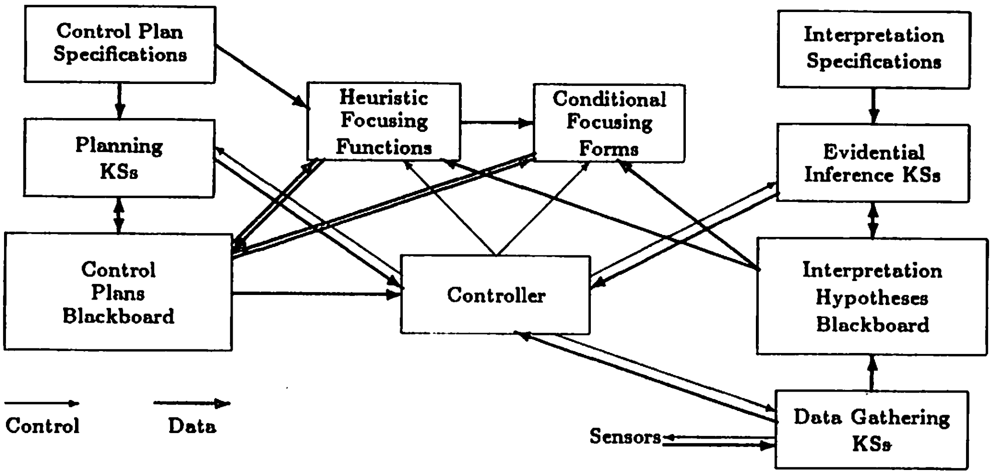

Figure 1: Architecture of System

appropriate abstract operators and control plans representing the strategy. Most work on evidential representation systems relies on having a fixed set of alternatives among which to partition belief (as in classification problems) and having atomic hypotheses. As we discuss in section 2.1, this is not the case for interpretation so such work is not directly applicable. Our use of symbolic representations of uncertainty was inspired, in part, by Cohen's work on symbolic representations of evidence called *endorsements* [5]. These symbolic representation of uncertainty make it possible to understand the relations between the hypotheses and the methods applicable to resolving the uncertainty. This is important because numeric sumaries of evidential uncertainty do not provide the kind of information needed to understand how to go about resolving the uncertainty in the hypotheses.

2 Planning to Resolve Uncertainty

Figure 1 illustrates the major components of our system. The controller is responsible for executing the basic control loop (see Figure 4). Control plan instances created by the planning process are maintained on the Control Plans Blackboard. The expansion of the control plans is accomplished using Planning KSs which are created from the Control Plan Specifications. Interpretation hypotheses are developed by executing interpretation actions using the corresponding Evidential Inference KSs and the Data Gathering KSs. The hypotheses are maintained on the Interpretation Hypotheses Blackboard along with information about the evidence supporting the hypotheses, the uncertainty in that evidence, and the relations between the hypotheses. The Heuristic Focusing Functions are defined as part of the Control Plan Specifications and are applied to the control plans under the direction of the controller. The Conditional Focusing Forms are essentially demons which can be defined by the focusing heuristics in order to modify the flow of control and refocus within the control plan hierarchy. The satisfaction condition of each of these functions is checked following the execution of an inference or data gathering KS.

2.1 Evidence and Sources of Uncertainty

For each class of interpretation hypothesis, H, the interpretation hierarchy specification defines both a set of sets of lower-level, support hypotheses, $\{\{S_i\}_j\}$, and a set of higher-level, explanation hypotheses, $\{E_i\}$. Hypotheses may be supported by multiple sets of supporting hypotheses to model those domains in which there can be multiple *sources of evidence*. For example, in vehicle monitoring, vehicles might be supported by data from a variety of sensor types so that $\{S_i\}_1$ might represent the support from acoustic sensors and $\{S_i\}_2$ that from radar. The lowest-level units in the hierarchy correspond to the data to be interpreted. The highest-level units represent the most abstract interpretation of the data and require no explanation for their occurrence.

The model of evidence that we use is based on the requirements for control—i.e., that the *sources of uncertainty* for each hypothesis can be used to drive the control process. At any point during the interpretation process, each hypothesis instance is based on a *set* of evidential inferences of the form: $S_{k_l} \Rightarrow H$ (supporting evidence) or $E \Rightarrow H$ (explanation evidence) where $S_{k_l} \in \{S_i\}_l$, $\{S_i\}_l \in \{\{S_i\}_j\}$, and $E \in \{E_i\}$. There are then the following potential classes of sources of uncertainty for the correctness of a hypothesis:

- Incomplete Evidence:
  - The set of supporting evidential inferences $\{S_j\}_l$ (where $S_{k_l} \in \{S_j\}_l$) may be incomplete—i.e., $\{S_j\}_l \subset \{S_i\}_l$.
  - There may not yet be any explanation evidence.
- Uncertain Evidence:
  - The premise hypothesis of an evidential inference may be uncertain—i.e., some S_{k_l} or E_k is uncertain.
  - There may be alternative interpretations for the evidence—i.e., for some $S_{k_l} in \{S_i\}_l$ the correct inference may be $S_{k_l} \Rightarrow H'$.
  - There may be alternative explanations for the hypothesis.
  - It may be uncertain whether an inference satisfies the hierarchy constraints—- i.e., it is uncertain whether $\{S_j\}_l \subset \{S_i\}_l$ or whether $E \in \{E_i\}$ for H.
- Conflicting Evidence:
  - Support evidence doesn't exist—i.e., some S_{k_l} doesn't exist for H.
  - Explanation evidence doesn't exist—i.e., no $E \in \{E_i\}$ matches H.

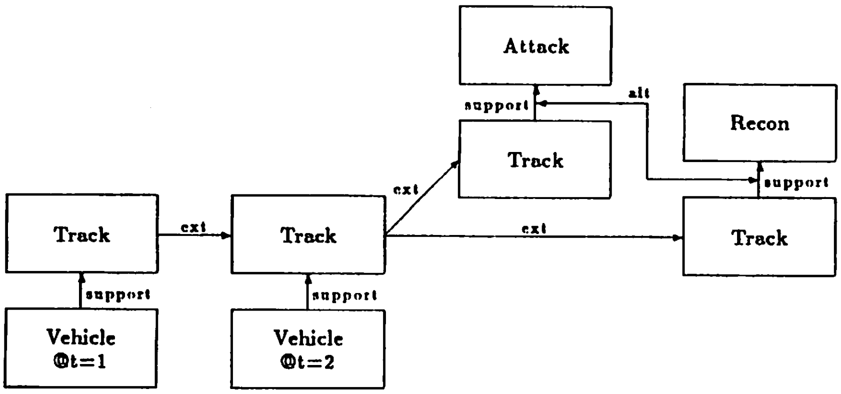

Figure 2: Hypothesis Representations

As evidential inferences are made by the interpretation process, symbolic expressions are created which represent the specific instances of these classes of uncertainty which exist in the evidence. These symbolic expressions represent the sources of uncertainty in the evidence and are associated with the hypothesis the evidence supports. For example, the Attack hypothesis in Figure 2 includes the expression, (Alt-Interp "track" "recon"), to represent the uncertainty in its supporting track evidence due to the existence of an alternative interpretation for the track (the Recon hypothesis). Further examples of the symbolic sources of uncertainty may be seen in Figure 5 as the *result* of the plan Get-Sources-of-Uncertainty (which determines the sources of uncertainty which exist in the specified hypothesis). We have also used this framework to extend the knowledge typically found in an interpretation system by including sources of uncertainty for the evidential relations which do not result from alternative interpretations due to interpretation hypotheses. For example, acoustic sensor data in a vehicle monitoring system provides uncertain support for a vehicle because the data may be the result of a sensor malfunction or a weather disturbance. We can specify such factors as sources of uncertainty for particular evidence even though these factors are not explicitly included as interpretation hypotheses (there is no sensor malfunction hypothesis in the interpretation specifications). Because we represent these uncertainty factors, knowledge can be applied to confirm or disconfirm the uncertainty. However, when the methods for discounting these uncertainties require evidential reasoning and interpretation, then the factors must be included as additional interpretation hypotheses–e.g., ghost tracks.

Interpretation hypotheses are compound structures because they may include parameters. The values of a hypothesis' parameters are defined in terms of the parameters of its support and explanation hypotheses. The inclusion of parameters (with continuous values) is one of the key characteristics of interpretation problems which distinguishes them from classification problems. Clancey [3] contrasts classification problem solving in which a solution is selected from a fixed set of alternatives with what he calls "constructive" problem solving in which a solution must be "formulated." He also points out that most medical "diagnosis" expert systems are really doing classification and are only useable for routine diagnosis in which the possibility of multiple, interacting diseases is ignored. Interpretation is clearly a form of constructive problem solving. We don't simply gather evidence to decide among a set of predetermined alternatives, but must gather evidence just to determine what the set of alternatives consists of. For example, in vehicle monitoring there is no way of knowing a priori how many vehicles the data might support nor where the vehicles might be. Sensor evidence for vehicle hypotheses not only supports the belief in a vehicle, it also defines the vehicle tracks by constraining the vehicle type and position.

Clancey mentions that the difference between classification and constructive problem solving has important consequences for choosing a knowledge representation. Gathering evidence for an interpretation hypothesis not only *justifies* the interpretation hypothesis, it also *refines* it by constraining its parameter values. Thus, every time evidence is added to a hypothesis it may result in a change in the parameter values of the hypothesis. However, since the evidence is uncertain, there may be alternative evidence which must also be pursued. Consequently, multiple versions of each hypothesis, called *extensions*, must be used to represent the alternative hypothesis refinements supported by the (uncertain) evidence. As part of our representation of the uncertainty in the hypotheses, we have developed a scheme for representing the alternative extensions of hypotheses and the interrelations between these extensions and between competing hypotheses. The advantage of this framework is that we can simultaneously reason about the uncertainty in a hypothesis due to alternative possible extensions of the hypothesis and due to competing hypotheses. Figure 2 shows a simple example of such an extension framework. In this example, there are four extensions of the Track hypothesis, each of which is caused by the addition of evidence which further constrains the Track. Of particular interest are the alternative extensions created by the competing interpretations of Track as support for an Attack mission or a Recon mission. These alternative extensions are used to recognize the *alternatives* relation between the support evidence for the Attack hypothesis and for the Recon hypothesis. The representation framework also allows us to represent the relations between evidence at different levels in the evidential hierarchy. For instance, additional evidence gathered for one of the vehicle hypotheses may constrain that hypothesis in such a way that it is only consistent with either the Attack hypothesis or the Recon hypothesis—but not both. Since this evidence is uncertain, though, it cannot be assumed that it is correct. The representation creates alternative extensions of the relevant hypotheses in order to represent the relations created by the addition of this evidence.

The power of this representation comes from its usefulness for differential diagnosis. The basic strategy for resolving interpretation uncertainty is to gather support and explanation for the hypothesis (or else explain why the evidence cannot be found) and then do

Name	Eliminate-Sources-of-Uncertainty
Description	Eliminates the sources of uncertainty from the hypothesis ?hyp until the belief in ?hyp is greater than ?belief.
Goal Form	(Have-Eliminated-SOUs ?hyp ?belief)
In Variables	(?hyp ?belief)
Out Variables	()
Temp Variables	(?sou)
Subgoals	((Have-Source-of-Uncertainty . (Have-SOU ?hyp ?sou)) (Have-Eliminated-SOU . (Have-Eliminated ?hyp ?sou)))
Sequence	(ITERATION (GREATER (belief ?hyp) ?belief) (SEQUENCE Have-Source-of-Uncertainty Have-Eliminated-SOU))
Constraints	()

Figure 3: Control Plan Specification

differential diagnosis to discount alternative uses of the data. Having complete support and explanation evidence for a hypothesis provides *necessary* evidence for the hypothesis, but it does not confirm the hypothesis because there still may be alternative explanations for all of the evidence. Thus the only way to gather *sufficient* evidence to confirm the hypothesis is to do differential diagnosis on the evidence. The explicit sources of uncertainty information in conjunction with the representation of hypothesis extensions provides the basis for doing this. In the example in Figure 2 the knowledge of the alternatives relation between the evidence for the Attack and Recon hypotheses allows us to understand why each is uncertain and what must be done to resolve the uncertainty. It also allows us to understand that each of these hypotheses negatively affects the belief in the other. Thus instead of trying to directly support a hypothesis we might also resolve its uncertainty indirectly by disproving its alternative.

Uncertainty in a plan hypothesis is typically resolved by gathering evidence to directly eliminate the sources of uncertainty for the hypothesis. However, in some domains it may also be possible to resolve uncertainty by gathering *independent* evidence for a hypothesis—that is, using alternative sources of evidence. It should be noted, though, that evidence is not necessarily independent just because it is based on a separate source; independent evidence must include independent sources of uncertainty. For example, if evidence from radar and from radio emissions detection could both be affected by the same kind of weather disturbances then one source of evidence could not be used to resolve this source of uncertainty in evidence from the other source.

2.2 Control Plans and Heuristic Focusing

Control plans are defined using specifications like the one in Figure 3 for the plan Eliminate-Sources-of-Uncertainty. The goal of the plan is specified by the Goal Form, (Have-Eliminated-SOUs ?hyp ?belief). All of the variables in the goal form must be listed in either the Input Variables clause or the Output Variables clause of the specification. Input Variables must be supplied to the plan when it is instantiated and Output Variables are bound upon completion of the plan to return results. Many actions are capable of

repeat: Pursue-Focus on each element of Current-Focus-Set until nil.

Pursue-Focus(focus)
case on type(focus):

plan	Focus on variable bindings to select plan instances for Current-Focus-Set. Expand plan instances to next subgoals. Focus on subgoals to select subgoals for Current-Focus-Set.
subgoal	Match goal to applicable plans. Focus on plans to select new focus elements for Current-Focus-Set.
primitive	Execute primitive plan action to get result. Update plan states and select new focus element for Current-Focus-Set. Check refocus and subgoal conditions.

Figure 4: Control Planning Loop

returning multiple bindings for variables—e.g., determining an available sensor unit. To deal with this fact, variables are allowed to take on *multiple valued* values which represent a set or range of bindings. Focusing then determines the value(s) to use in subsequent plan expansion. This example has the input variables ?hyp and ?belief in the goal form and no output or result variables. In addition to the input and output variables, the plan specification also includes a Temp Variables clause which lists variables used to hold any subgoal results that are not part of the plan goal form. Here, the temporary variable ?sou is used to hold the source of uncertainty being worked on by the plan. Each plan is realized by a sequence of subgoals. The subgoals of the plan are defined in the Subgoals clause in terms of the goal forms for the subgoals. These goal forms are used to identify control plans applicable to satisfying the subgoals by unifying the subgoal goal forms with the control plan goal forms of all possible control plans. In the example, there are two subgoals, Have-Source-of-Uncertainty and Have-Eliminated-SOU, whose goal forms are specified. The subgoal sequence is specified in the Sequence clause. This clause uses a shuffle grammar to express strict sequences, concurrency, alternatives, optional subsequences, and iterated subsequences. Eliminate-Sources-of-Uncertainty iterates a sequence of the two subgoals until the required level of belief in ?hyp is reached. The two subgoals represent the actions of identifying the current sources of uncertainty in ?hyp and eliminating a source of uncertainty in ?hyp. The plan thus proceeds by identifying the sources of uncertainty, selecting (through focusing) a source of uncertainty to be eliminated, eliminating the source of uncertainty, and repeating this sequence as necessary. The sequencing constraints of the Sequence clause can be augmented with additional ordering constraints and constraints on the subgoal variable values by placing additional conditions in the Constraints clause.

The basic control planning loop is detailed in Figure 4. An example control plan instance is represented in Figure 5 as an AND/OR tree of plan and subgoal nodes. The situation represented is such that the top-level plan to solve the interpretation problem has created a subgoal of resolving the uncertainty in the Attack hypothesis of Figure 2. In order to pursue this subgoal, the subgoal form is unified with the goal forms of the defined control plans to determine which plans are applicable to satisfying the subgoal.

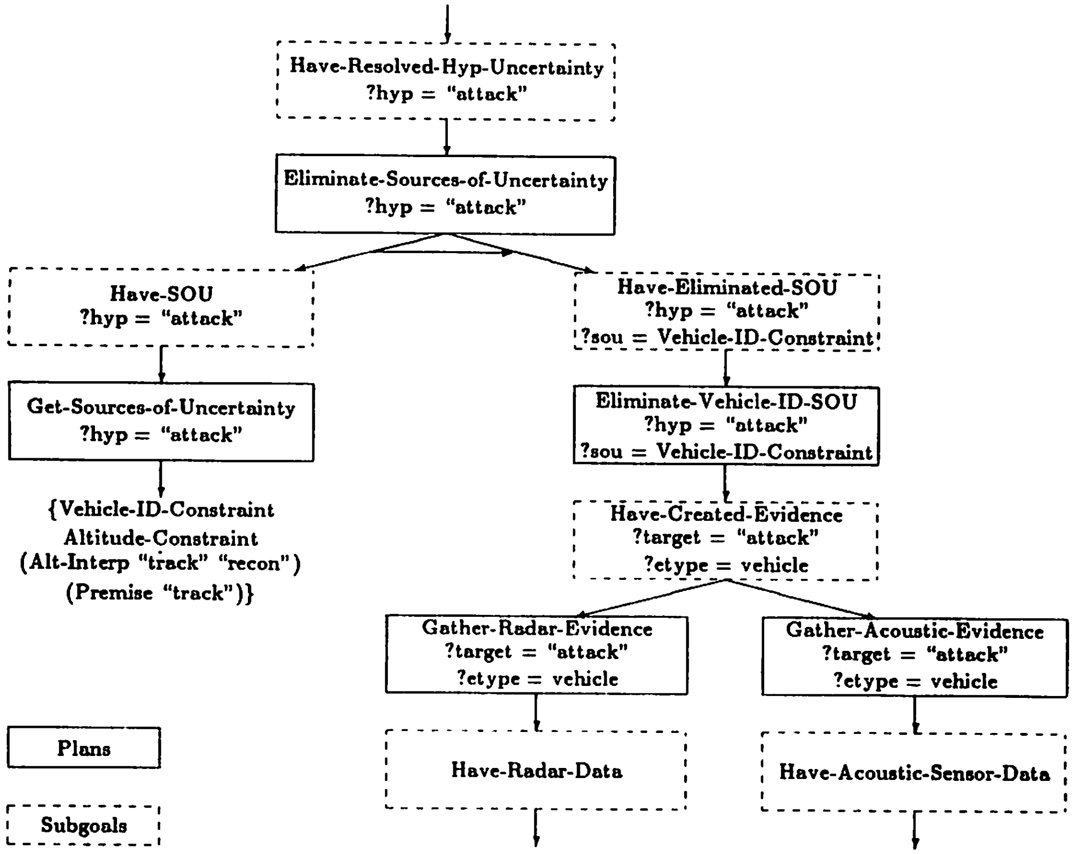

Figure 5: Control Plan Instance

In this case, only a single plan is relevant to satisfying this goal, Eliminate-Sources-of-Uncertainty, the plan whose specification was shown in Figure 3. In general, there would be multiple matches corresponding to multiple possible strategies for satisfying the goal and focusing knowledge would be applied to select the plan(s) to focus on. Pursuing an in-focus control plan means expanding the control plan to create subgoals representing the substeps of the plan which need to be satisfied next. The first two subgoals of the Eliminate-Sources-of-Uncertainty plan are shown in Figure 5 even though they are sequential steps (signified by the horizontal arrow between their arcs). The first subgoal, Have-SOU, can be satisfied by the primitive plan, Get-Sources-of-Uncertainty. Primitive plans represent actions which may be carried out with corresponding Knowledge Sources. Primitives may generate information for the planning process—e.g., determining an available sensor to generate new data—or may generate interpretation evidence—e.g., to create an evidential inference. Actions may fail or may succeed and return results. In the case of Get-Sources-of-Uncertainty here, the action succeeds and returns a multiple-valued value consisting of the symbolic representations of the current sources of uncertainty in the attack hypothesis. This result is bound to the variable ?sou of the primitive and the status of the plan is set to complete. The outcome of the action (the change in status and the variable bindings) is propagated to the subgoal the primitive satisfies and then in turn to the plan containing the subgoal. The change in the state of the subgoal may mean that this plan has failed or has succeeded which would cause additional propagation. In the example, it simply changes the state of the plan so that it is expecting the next subgoal (Have-Eliminated-SOU) and binds the temp plan variable ?sou (which will be used to bind the subgoal variable ?sou). Multiple-valued values such as that just bound to ?sou are used to represent a set of alternative bindings for a variable—i.e., uncertainty over the correct value. Because ?sou is used as an input to the next subgoal and because it has a multiple-valued value, it is necessary to apply focusing knowledge at this point to select the value(s) to be used for further expansion. The single-value version(s) of the subgoal are then used to match and select applicable control plans. In the example, the heuristic focusing knowledge in the Eliminate-Sources-of-Uncertainty control plan which is associated with the variable ?sou, has selected the source of uncertainty Vehicle-ID-Constraint as the (sole) in-focus binding for ?sou.

Focusing heuristics represent meta-level knowledge relative to the knowledge in the control plans. Whereas control plans embody problem-solving strategies for interpretation, focusing heuristics embody strategies for selecting the appropriate problem-solving strategies. In our framework, focusing heuristics are associated with particular control plans. There are several points at which focusing decisions must be made so we partition the focusing knowledge into four different classes: variable, subgoal, matching, and updating. *Variable* focusing knowledge is associated with each of the variables of the control plan environment and is used to select among competing bindings for a variable. This occurs when actions return *multiple-valued* values (as discussed above). *Subgoal* focusing knowledge is used to select among multiple active subgoals for a plan instance. Control plans can specify that certain subgoal sequences are able to be carried out in parallel, however, it still may be preferable to sequence the subgoals due to uncertainty over their satisfaction and results. *Matching* focusing knowledge is used to select among the multiple plans which are applicable to satisfying a subgoal when there are multiple plans goal forms which match a subgoal form. *Updating* focusing knowledge is associated with each subgoal of a control plan and is used to decide how to proceed when a plan for satisfying the subgoal completes (succeeds or fails). In general, plan expansion is not exhaustive so completion of a plan will leave alternative expansions of the plan and alternative matching plans which might be pursued to try to satisfy a subgoal. Updating heuristics decide whether to accept a result and propagate it or to pursue existing alternatives instead. Thus this knowledge is partially responsible for controlling "backtracking" of the system.

A common problem with meta-level focusing knowledge is conflicts. That is, we may have heuristic knowledge that says to "Prefer A" and other knowledge that says to "Prefer

B" instead. The cause of such a conflict is the generality of the heuristics which fails to provide information about the proper context in which to apply the knowledge. For example, general heuristics for selecting the "best" data to use to create evidence might say to prefer data that is: well sensed, in time slices with a small number of data clusters, in tight clusters, etc. These heuristics may very well conflict by preferring different data. However, if we understand the purpose for which the evidence is to be used, we can avoid this problem because we understand that certain data characteristics are most important in particular contexts. Thus, the number of time clusters is less critical for extending a track than for creating evidence of a new track since the existing track constrains the data selection. Hierarchical control plans provide context with which to disambiguate heuristics since the control plan structure represents the purpose of the plans. Thus, by associating heuristic focusing knowledge with particular control plans and allowing the heuristics to examine the control plan hierarchy we provide the context to disambiguate the heuristics.

The basic control process described above is highly top-down and depth-first. However, the uncertainty of the interpretation process requires a strong bottom-up control component as well. We accomplish this with several extensions to the basic focusing scenario which make it possible for the system to shift its focus between competing strategies in response to the characteristics of the developing plans and factors such as data availability. Focusing is extended by allowing variable and matching focus decisions to be: absolute, postponed, or preliminary. *Absolute* focusing heuristics simply select a single path to be pursued—subject, of course, to potential plan failure (which is handled by the updating process). However, focusing heuristics may not always be able to select a single "best" path to pursue. Instead, they may need to partially expand each of several competing strategies to gather more specific information about the situation before being able to select the best alternative.

We handle this nondeterminism by allowing multiple paths to be expanded with *postponed* focusing. In order to postpone focusing and pursue multiple paths, we must specify the conditions under which the alternatives should be reevaluated and how to reevaluate them. A postponed focusing decision creates a *refocus form* which specifies the paths to be pursued, the conditions for refocusing, and a refocus handler. Refocus conditions are evaluated following the execution of any action (only actions generate new knowledge). When they are satisfied, the refocus handler is invoked and reevaluates the choices within the new context in order to eliminate the multiple foci. An example of a postponed focusing decision occurs late in the refinement of the control plan in Figure 5. There are two control plans applicable to satisfying the subgoal Have-Created-Evidence: Gather-Radar-Evidence, which uses radar data to create the desired evidence, and Gather-Acoustic-Evidence, which uses acoustic sensor data. The system is uncertain about how to proceed because it cannot be sure which source of evidence will provide the best track evidence without knowing more about the actual data which is available. To handle this situation, the focus decision is postponed until the first subgoal of each alternative plan is satisfied—that is, until the potential data is determined. The refocus handler is then used to evaluate the focusing alternatives in light of the additional information by evaluating the relative quality of the available data for each alternative plan.

Preliminary focus decisions are similar to postponed decisions except that refocusing involves a reexamination of all of the original alternatives as opposed to just those that were initially focused on. Preliminary focus decisions are used when one alternative is likely to be the best—subject to certain reservations about its progress or under a particular assumption about the situation. The refocus conditions can then monitor the progress of the choice or the validity of the assumptions. For example, they may be used to limit the amount of effort expended on one alternative by including refocus conditions which set a limit on the amount of time to be expended or the level of completion to be reached. This is important in plans like Eliminate-Sources-of-Uncertainty shown in Figures 3 and 5. Resolving the uncertainty in one particular hypothesis to the exclusion of all other alternatives could result in the system missing important domain activities or losing the opportunity to gather useful data. Thus such a plan would be selected with a refocus form which limits the amount of time spent on the plan or causes the choice of the plan to be reconsidered following each plan iteration.

Preliminary focus decisions may be combined with postponed decisions in order to pursue multiple options, refocus among them, and still limit the entire choice. They make it possible to define opportunistic methods for refocusing. Preliminary and postponed focus decisions also control the system's backtracking since they effectively define the backtrack points and the conditions under which the system backtracks. This provides the system with a form of nonchronological backtracking for a domain where dependency-directed backtracking is ineffective [2]. Toward this end, the refocus conditions may also refer to plan failure.

3 Conclusion and Status

This work is a further example of the utility of making control decisions through planning. It expands on existing research with respect to planning for the control of an interpretation system in three significant ways: the control task is viewed as being driven by the need to resolve uncertainty, the uncertainty of the interpretation hypotheses is represented explicitly and symbolically, and the process of finding the correct control plan may be seen to involve a search process which requires focusing. The combination of control plans with parallel focusing and a symbolic representation of the interpretation uncertainty provides a flexible framework which can be used to implement sophisticated control strategies.

The use of control plans changes the nature of interpretation control reasoning. Typically, control decisions have involved first rating all of the interpretation hypotheses and the data to select the "best" item to pursue and *then* determining how to pursue it. Such decisions are extremely complicated since they have to simultaneously consider a variety of factors and they obscure the strategies being used since the strategies are implicit in the ratings functions. The fundamental problem with focusing on hypotheses is that it's really impossible to decide which hypotheses to pursue—let alone *how* to pursue them—without understanding *why* you are pursuing them—i.e., without knowing what purpose they serve with respect to the overall system goal. Rather than trying to include such knowledge (along with heuristic focusing knowledge) within a complex rating function, the control planning process described here expresses the problem-solving strategies as explicit control plans. The selection of hypotheses to pursue and the methods to pursue them then flows naturally from the selection of strategies to meet the current problem-solving goals and the instantiation of these strategies. In addition, hierarchical focusing in conjunction with plan refinement makes it practical to express the focusing heuristics explicitly as well. This is because only a limited number of alternatives are considered at any point and because the control plan structure provides detailed context information.

We presently have a prototype implementation of the interpretation framework presented in this paper which simulates a system for monitoring aircraft. A variety of data sources such as acoustic sensors, radar, and emissions detectors are included. Additional sources of evidence such as terrain, air defense positions, and weather information are also available. Active control over evidence gathering can be effected through control of the operations of some of the sensors. Interpretation hypotheses cover a variety of missions including those involving coordination of multiple aircraft. One of the areas of current research in this project is the issue of languages for expressing focusing knowledge. In particular, we are looking into the factors needed for focusing in real-time applications. This includes such things as estimated processing and elapsed times and estimates of the quality of the evidence from alternative strategies.

References

[1] Carver, Norman, Victor Lesser, and Daniel McCue, "Focusing in Plan Recognition," *Proceedings of AAAI-84*, 1984, 42–48.

[2] Carver, Norman, *Evidence-Based Plan Recognition*, TR 88-13, Computer and Information Science Department, University of Massachusetts, 1988.

[3] Clancey, William, "Classification Problem Solving," *Proceedings of AAAI-84*, 1984, 49–55.

[4] Clancey, William, "From GUIDON to NEOMYCIN and HERACLES in Twenty Short Lessons," *AI Magazine*, 7 (3) 1986, 40–60.

[5] Cohen, Paul, *Heuristic Reasoning About Uncertainty: An Artificial Intelligence Approach*, Pitman, 1985.

[6] Durfee, Edmund, and Victor Lesser, "Incremental Planning to Control a Time-Constrained, Blackboard-Based Problem Solver," *IEEE Transactions on Aerospace and Electronic Systems*, September, 1988.

[7] Hayes-Roth, Barbara, "A Blackboard Architecture for Control," *Artificial Intelligence*, 26, 1985, 251–321.