

A View of Diagnostic Reasoning as a Memory-directed Task

Roy M. Turner

Department of Computer Science
University of New Hampshire
Durham, NH 03824
rmt@cs.unh.edu

Abstract

Diagnostic reasoning underlies many intelligent activities, including (but not limited to) situation assessment/context recognition, natural language understanding, scene recognition, interpretation of scientific observations, and, of course, medical diagnosis and other forms of fault-finding. In this paper, we present a memory-directed, schema-based approach to diagnostic reasoning. Features of the problem are used to "evoke" one or more possible diagnoses, stored as schemas. Schemas contain information about their "manifestations" that be used to confirm or deny the diagnosis and, in some applications, information that can be used to take action based on the diagnosis. Potential advantages of the approach include cognitive plausibility, rule exception handling via (generalized) case-based reasoning, applicability to multiple domains, extensibility from experience, and a natural way to organize knowledge about what to do after a diagnosis is made.

Introduction

Diagnosis is central to much of intelligence. We can view all of the following as instances of the process of diagnosing features present in a reasoner's input: assessing its current situation (context recognition); understanding natural language; interpreting scientific observations by determining which theories they fit; and, of course, diagnosing medical problems and other kinds of faults.

We are interested in developing a computer-based model of diagnostic reasoning that supports our work in situation assessment and context-sensitive reactive reasoning in the autonomous underwater vehicle (AUV) domain, as part of the ORCA project (Turner & Stevenson, 1991). Such an approach would take features of the current problem-solving environment as input (derived ultimately from sensors or from information from other agents) and produce as output an assessment of the current problem-solving situation. This assessment would then provide the reasoner with the information necessary to assure that its behavior is appropriate for the context.

To be useful in a variety of real-world applications, a computer-based approach to diagnostic reasoning should have several properties. It should be general, that is, not tailored to one particular domain such as medical diagnosis. It should facilitate knowledge acquisition, both initially and throughout the problem solver's lifetime. This includes facilitating learning from the reasoner's own experience. Since complete knowledge is impossible to obtain in almost any domain, the reasoner should be able not only to handle usual situations, but also exceptions that follow no known rules. Finally, we should keep in mind that diagnostic reasoning is often only a means to an end. Consequently, a diagnostic reasoner should facilitate additional reasoning based on its diagnoses, for example, by allowing treatment information (for medical systems) or context-specific actions (for, e.g., reactive planners) to be associated with its representation of the diagnosis (a disease or context).

We are developing an approach to diagnostic reasoning that is based on the view of diagnosis as a memory-directed task; consequently, our approach is called MD, for Memory-based Diagnosis. This does *not* mean that memory is the sole mechanism underlying diagnosis in our approach; indeed, we have argued elsewhere (Turner, 1989a; 1989b) for a view of diagnosis as a planning task. Instead, we view the overall process as being *guided* by the organization of knowledge in the reasoner's memory.

It should be stressed that this is not completed work; rather, we build on some early work done in the domain of medical diagnosis during the MEDIC project (Turner, 1989a; 1989b; 1989c), and current work is ongoing in the area of situation assessment and context-specific reactive reasoning for AUVs. We believe that this approach shows promise not only for our domain but for diagnostic reasoning in general.

Diagnostic Reasoning

Diagnosis is fundamentally an abductive reasoning task (Reggia *et al.*, 1985): from a set of features of a problem or situation ("symptoms"), hypothesize a cause ("diagnosis") based on a *priori* knowledge

linking causes to effects and effects to likely causes. The kind of reasoning done by diagnosticians (e.g., physicians) to implement this abductive inference is *hypothetico-deductive reasoning*: hypothesize a diagnosis, then test predictions made should that hypothesis be true.

Though most researchers agree on this much, differences of opinion arise with respect to a reasonable computational model of diagnosis. The prevalent model is backward chaining rule-based reasoning (Buchanan & Shortliffe, 1984). Chandrasekaran *et al.* (1979) describe diagnosis as *classification*: the process of assigning a problem to a particular location in a taxonomy of problems. Reggia and colleagues (1985), in perhaps the most formal treatment, view diagnosis as abduction, in particular parsimonious set covering. As mentioned above, the author has elsewhere argued for a view of the act of diagnosis as a planning task, view that attempts to tease out *how* diagnosis is done rather than what it is *per se*. The mechanism described in this paper is one part of the latter approach.

Most successful approaches to diagnosis have been rule-based, for example MYCIN (Buchanan & Shortliffe, 1984) and its descendants. Such systems, however, seem to miss an essential flavor of the hypothetico-deductive reasoning style as seen in humans, in which the reasoner is essentially "reminded" of possible diagnoses based on features in the problem.¹ Humans seem to be able to relate symptoms to diagnoses with relatively little cognitive effort, and with an accuracy that increases with the human's experience. This characteristic would be highly desirable in a computer-based diagnostic reasoner as well. To capture this aspect of diagnostic reasoning in a computer model would seem to require a memory of diagnoses, linked together (organized) to facilitate retrieval of diagnoses at appropriate times.

One system that begins to address this is INTERNIST-1/CADUCEUS (Miller *et al.*, 1982; Pople, Jr., 1982) (hereafter referred to as INTERNIST), a diagnostic reasoner in the domain of internal medicine. This system directly approaches diagnosis as an abductive task. Symptoms are stored with information about what diseases they *evoke*—that is, that they should bring to mind for an expert diagnostician. Unfortunately, INTERNIST's knowledge is relatively unstructured and the program has no facility to restructure its knowledge based on experience, nor does it provide a means of easily associating additional knowledge (e.g., treatments) with diagnoses. Another system, MDX (Chandrasekaran *et al.*, 1979) organizes diagnoses in a hierarchy that is traversed (by using rules) to find more specific diseases. How-

¹This is not to say that all reminders are based on surface features; other, deeper, features as they are discovered may also form the basis for hypotheses.

ever, the links between diseases in MDX's memory are static, and contain little or no information to help focus the reasoner's attention on one disease over another. Two other systems, CASEY (Koton, 1988) and MEDIC (Turner, 1989a; 1989b; 1989c), respectively a case-based reasoner (e.g., (Kolodner *et al.*, 1985)) and a schema-based reasoner (a generalization of a case-based reasoner), are more similar to the approach presented here. The kind of memory used by these programs allows a degree of interconnectivity not present in MDX, while at the same time providing mechanisms to allow updating from experience. However, though both of these systems use a content-addressable, flexible memory, they both also rely solely on mechanisms external to the memory to decide the appropriateness of evoked diagnoses.

What is desirable is an approach that can use the flexible, content-addressable memory style of CASEY or MEDIC, augmented with information such as INTERNIST has to provide a measure of likelihood of the diagnoses the reasoner is reminded of by the memory. Such a scheme would provide the reasoner with a metric, potentially updatable from experience, to allow it to effectively direct the diagnostic process.

Memory-based Diagnosis

Memory-based Diagnosis (MD) is based on using a content-addressable memory of schemas representing diagnoses to perform the hypothesis generation phase of hypothetico-deductive reasoning. The memory structures are organized to facilitate retrieval based on cues in the diagnostic problem, and they contain information useful in the context of the diagnoses they represent.

Before proceeding to describe the MD approach, a brief word on terminology is in order. Our terminology is a melange of terms borrowed from INTERNIST, the memory and case-based reasoning literature, and psychological research on the process of diagnosis. "Symptom" and "manifestation" will be used synonymously to mean a feature of a given situation. A "diagnosis" is a conclusion of a fault or disease causing the problem or a characterization of the current situation. A symptom's "evoking strength" (Miller *et al.*, 1982) for a diagnosis is a measure of how strongly its presence indicates the presence of that diagnosis. A diagnosis' "manifestation strength" for a symptom is a measure of how strongly the symptom is expected, given the diagnosis. The two terms "diagnostic task" (Miller *et al.*, 1982) and "logical competitor set" (LCS) (Feltovich *et al.*, 1984) are used synonymously to mean a set of diseases that are in competition to explain the same subset of manifestations; a reasoner seeks to "solve" an LCS by finding the best diagnosis from among its members.

In the MD approach, information about known diagnoses are represented as schematic memory struc-

tures. Elsewhere, we have referred to similar memory structures as *contextual schemas* (c-schemas) (Turner, 1989a; 1989b); we will continue to use that terminology here, though the reader should be aware that we are extending our prior definition of c-schema slightly to accommodate diagnoses such as diseases. Figure 1 shows two example c-schemas.

Note that c-schemas contain additional information apart from the manifestation list (list of those manifestations that are expected). The actual contents vary depending on domain, but, in general, c-schemas can contain any information useful in the context of the diagnosis. In the case of a disease, this might be treatment information. In the case of situation assessment for an autonomous vehicle, this might include context-specific ways to handle unanticipated events, to order goals, to set parameters, or to select action, as discussed elsewhere (Turner, 1989c; 1990). The point is that by making the diagnosis, the information needed to deal with the diagnosis is immediately at hand.

The overall MD process can be described as follows:

1. Find candidate diagnoses.
2. Evaluate their "fit" to the current situation.
3. Form logical competitor sets (LCSs): sets of diagnoses in which each diagnosis competes to explain the same subset of the features in the problem.²
4. Solve the LCSs, obtaining one or more non-competing diagnoses to explain the problem.
5. Merge the diagnoses into an overall picture of the situation.

Finding candidate diagnoses. C-schemas are organized in a memory to facilitate their retrieval as well as their modification from future experience. The memory used in MEDIC is modeled after the CYRUS memory program (Kolodner, 1984), which is essentially a set of highly-interconnected discrimination nets. Such memories have been used with good results in case-based reasoners (CBR) and a schema-based reasoner (MEDIC) to find needed cases and schemas based on features of the current problem-solving environment. They are often referred to as "MOP memories", since the primary memory structures are memory organization packets (MOPs). MOPs are schemas that encode information about a class of episodes (in CYRUS); their primary function, as their name suggests, is to organize more specialized MOPs, which organize more specialized ones, etc., until the level of individual episodes ("cases", in the CBR literature) is reached. C-schemas are a kind of MOP; consequently, they

²This may, in turn, require looping to earlier steps as new information is uncovered; see, e.g., (Miller *et al.*, 1982).

serve as organizing points in the memory, with more general c-schemas indexing ones that are more specialized. We use the term "case" for the individual episodes in memory: cases of diseases that have been seen, situations that have been encountered, and so forth.

We should point out that our approach does not exclude case-based reasoning—far from it. MD, which is a kind of schema-based reasoning, can be validly viewed as generalized CBR: that is, CBR that uses generalized cases instead of or in addition to "real" cases of problem solving. We have argued elsewhere (Turner, 1989a) for the utility of this view.

C-schemas are linked together by *indices*, each of which is a feature type/value pair. The feature types present in a c-schema's indices correspond to types of features expected to be present in the diagnosis represented by the c-schema. For example, a medical reasoner's c-schemas might contain such feature types as: symptoms, patient characteristics, medical history, etc. The "value" half of an index is an indication of how the indexed c-schema differs from its parent. For example, a c-schema representing "lung disease" will index one representing "sarcoid", a common lung problem in inner-city hospitals in some regions of the country. Some feature type/value pairs we might expect as indices from "lung disease" to "sarcoid" would be: "patient home/Atlanta", "patient race/black", etc. Values are chosen that are different from values of the corresponding feature type in the parent c-schema; these are considered *predictive* differences by Kolodner; we can also think of them as *evocative* differences, since their presence *evokes* the indexed c-schema—that is, it "brings to mind" the diagnosis it contains. Figure 2 shows a portion of a hypothetical c-schema memory.

Traversal of a MOP memory entails first comparing features of the current problem to the features predicted by some subset of its top-level memory structures, then selecting potential indices based on salient (evocative) differences.³ These potential indices are then compared to the c-schema's actual indices to produce a set of one or more c-schemas. Traversal continues in this manner using these structures until no further c-schemas are found. The c-schemas (and/or possibly episodes) found are, in our approach, the set of most specific possible diagnoses for the problem.

One difference in the MD approach from typical MOP memory applications is that we are interested in how strongly the current situation evokes a given diagnosis: this provides one measure of how well the diagnosis fits the problem, in the same manner as in INTERNIST. To provide for this, we augment

³This process is actually quite a bit more complex than this, as described fully by Kolodner (1984).

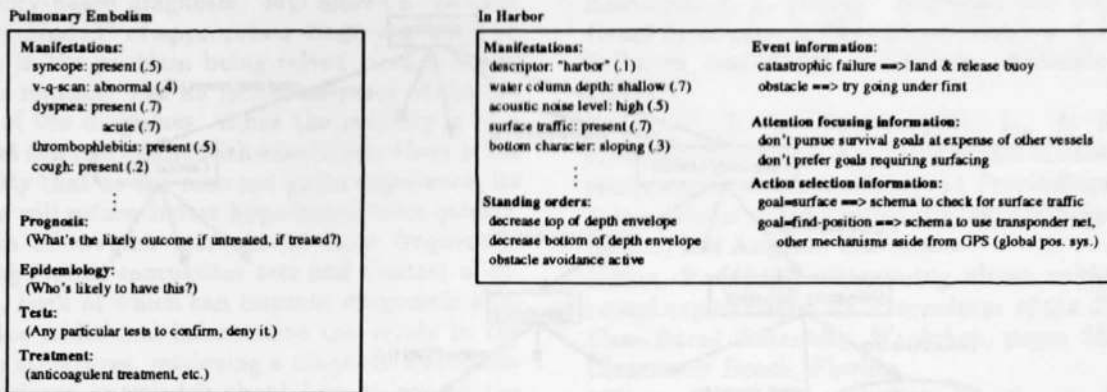


Figure 1: Contextual schemas from two different domains: medical diagnosis (left) and AUV control (right). Numbers refer to manifestation strengths.

the indices of c-schemas with an *evoking strength*, a measure of how strongly the reasoner should be reminded of a c-schema, given its parent c-schema and the current situation. An important property of MOP memories is that they can be updated from experience, and, as new episodes are added, the memory reorganizes its indices appropriately as well as updates the generalized information in its MOPs. In our scheme, the evoking strengths would also be updated when new information is added. This would theoretically allow the reasoner's "reminders" of possible diagnoses to become more accurate as it gains experience.

Of course, how strongly a diagnosis is evoked by a particular set of features in the problem description is not solely a function of the indices from the c-schema's parent. The strength with which the parent is evoked is also important. In addition, a c-schema can be reached through possibly many paths in a MOP memory, including multiple indices from the same parent; these different paths should contribute to the overall assessment of evoking strength. While we have not yet worked out the details of the combination function for evoking strengths, we can offer some initial observations. First, a c-schema should pass evoking strength to its children in proportion to how strongly it is evoked, yet we would not want simple multiplication of probability-like numbers, since this would result in a progressive decrease in evoking strength the deeper memory is traversed—the counter-intuitive situation of c-schemas that are better and better matches being less and less evoked! Second, when a c-schema is indexed more than once by a c-schema, or by more than one c-schema, its evoking strength should be a function of the evoking strengths of all the ways it is indexed, since these represent potentially differ-

ent ways the c-schema matches the current situation. However, care will have to be taken to ensure that, as can happen in a MOP memory, the c-schema is not reached by two or more paths that are essentially the same, just permutations on the order in which features were examined.

The traversal process just described can determine a set of c-schemas comprising the potential diagnoses in one of at least two ways. As described above, memory traversal will create a set of all c-schemas encountered that the traversal process was not able to find children of, given the current problem features. The obvious drawback of this scheme is that the size of the set may grow quite large, and it may contain c-schemas that have very little to do with the actual problem (i.e., they were found solely due to slight resemblances). This means that the reasoner will have more hypotheses to select from, but will have to do more work to determine which hypothesis or small set of hypotheses is actually the correct diagnosis. A second approach would be to use information during memory traversal to narrow the search for hypotheses. The evoking strengths could be used during memory search to prune the search, in the manner of either beam search or a best-first search method. In this case, when considering memory traversal from some c-schema in memory, only the best n children might be considered next, where "best" is determined by evoking strength. With appropriate care, the memory could return either a single likely diagnosis or a set of likely diagnoses for the reasoner to consider.

These methods of finding a set of potential diagnoses are not mutually exclusive. It may be that different techniques are appropriate in different situations. For example, when in a hurry, the latter technique may be used, potentially sacrificing com-

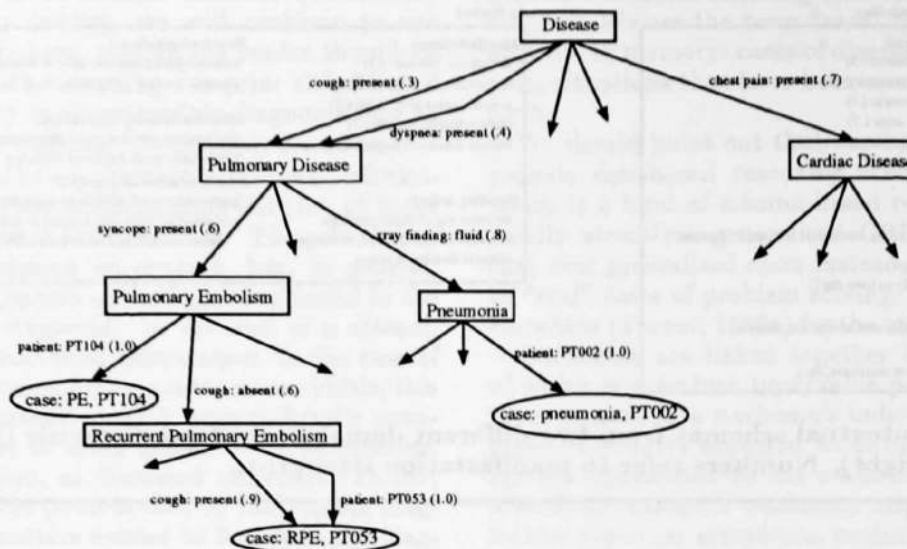


Figure 2: A portion of a c-schema memory. Numbers are evoking strengths.

pleteness for speed. In addition, it may be that if a reasoner is able to learn from its experiences, it can change the mechanism it uses. When a novice, a reasoner may painstakingly elucidate all possible diagnoses and carefully examine each one. With more experience, the reasoner's memory may become well-tuned, both in terms of its organization and the accuracy of its evoking strengths. In this case, the reasoner may use the latter strategy and immediately "home in" on the best diagnosis.

Evaluate fit of diagnoses. For this portion of the process, we turn to other work on diagnostic reasoning. We can, for example, use a mechanism patterned after INTERNIST's and formulate an overall score for each potential diagnosis from summing a positive component, based on the evoking strength, and a negative component, based on both manifestations predicted but not present and on unexplained manifestations. A better version of this scheme may be to consider manifestations that are explained as part of the positive component.

Logical competitor set formation and solution. The mechanism for these two steps can also be borrowed from INTERNIST. LCSs are useful for a diagnostician because, unlike rule-based reasoning, LCSs guide the strategies selected for information gathering. This allows the reasoner to play one disease off against others. For a more complete discussion of this, see (Miller *et al.*, 1982), (Feltovich *et al.*, 1984), or, in the context of a memory-based reasoner, (Turner, 1989a).

Merging diagnoses. For medical diagnosis and related tasks, when several complementary diagnoses are found, they can simply be output as a diagnosis list.⁴ For other diagnostic tasks, however, it is not so simple. For example, in situation assessment for an autonomous underwater vehicle controller, the "diagnoses" are classes of situations; at any one time, the vehicle may be in several: operating with low power, under ice, within a harbor, etc. In domains such as this, the reasoner must merge the diagnoses to form an overall picture of its situation. This is an intended area for future research.

One exciting extension of the idea of representing diagnoses as c-schemas is the possibility of using a c-schema to represent a group of diagnoses that often occur together or that are often considered as a group during diagnosis. The former case should drastically improve the reasoner's ability to quickly diagnose a situation in which several different diagnoses are present, especially since the presence of one diagnosis often masks the presence of another (see, e.g., Sticklen (1987)). The latter case essentially means storing logical competitor sets as c-schemas; since LCS formation can be difficult, this, too, can improve the reasoner's performance, as suggested by Feltovich *et al.* (1984).

Conclusion

In this paper, we have briefly sketched memory-based diagnosis, an approach for using a dynamic conceptual memory to aid the process of diagnosis.

⁴See Sticklen (1987), however, for a different view.

tic reasoning. There are several potential benefits of memory-based diagnosis. MD allows a reasoner to be "reminded" of appropriate diagnoses based on features in the problem being solved in such a way that the reasoner has an initial estimate of the accuracy of the diagnoses. Since the memory is of a type that can reorganize with experience, there is the possibility that as the reasoner gains experience, its memory will return better hypotheses more quickly. MD also allows the reasoner to store frequently-occurring logical competitor sets and clusters of diagnoses, both of which can improve diagnostic ability. Since additional information can reside in the memory structures, retrieving a diagnosis automatically retrieves knowledge about how to act on the diagnosis (e.g., treatment for a disease, appropriate behavior for a kind of situation, etc.). As the memory organizes diagnoses hierarchically and allows case-based reasoning, situations which do not follow usual rules can be stored and retrieved easily. And, finally, to the extent that the underlying memory is cognitively plausible (e.g., (Kolodner, 1984)), so is MD; indeed, there is some psychological evidence supporting the schematic representation of context (Chi *et al.*, 1982) and LCSs (Feltovich *et al.*, 1984) in expert problem solvers.

As we have said, this work is just beginning. Earlier work in the MEDIC project was encouraging, and we are currently investigating the approach in the domain of AUV control as part of the ORCA project. We believe that memory-based diagnosis is a promising approach to diagnostic reasoning, a kind of reasoning that underlies much behavior we consider intelligent.

References

Buchanan, B. G. & Shortliffe, E. H. (1984). *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley Publishing Company, Reading, Massachusetts.

Chandrasekaran, B., Gomez, F., Mittal, S., & Smith, J. (1979). An approach to medical diagnosis based on conceptual structures. In *Proceedings of the Sixth International Joint Conference on Artificial Intelligence*, Stanford, California.

Chi, M. T. H., Glaser, R., & Rees, E. (1982). Expertise in problem solving. In Sternberg, R. J., editor, *Advances in the Psychology of Human Intelligence*, volume 1, pages 7-75. Erlbaum, Hillsdale, NJ.

Feltovich, P. J., Johnson, P. E., Moller, J. A., & Swanson, D. B. (1984). LCS: The role and development of medical knowledge and diagnostic expertise. In Clancey, W. J. & Shortliffe, E. H., editors, *Readings in Medical Artificial Intelligence*, pages 275-319. Addison-Wesley Publishing Com-

pany, Reading, Massachusetts.

Kolodner, J. L. (1984). *Retrieval and Organizational Strategies in Conceptual Memory*. Lawrence Erlbaum Associates, Publishers, Hillsdale, New Jersey.

Kolodner, J. L., Simpson, R. L., & Sycara-Cyranski, K. (1985). A process model of case-based reasoning in problem-solving. In *Proceedings of the International Joint Conference on Artificial Intelligence*, Los Angeles, California.

Koton, P. (1988). Reasoning about evidence in causal explanations. In *Proceedings of the DARPA Case-Based Reasoning Workshop*, pages 260-270, Clearwater Beach, Florida.

Miller, R. A., Pople, Jr., H. E., & Myers, J. D. (1982). INTERNIST-1, an experimental computer-based diagnostic consultant for general internal medicine. *New England Journal of Medicine*, 307:468-476.

Pople, Jr., H. E. (1982). Heuristic methods for imposing structure on ill-structured problems: The structuring of medical diagnostics. In Szolovits, P., editor, *Artificial Intelligence in Medicine*, pages 119-189. Westview Press, Boulder, Colorado.

Reggia, J. A., Nau, D. S., & Peng, Y. (1985). A formal model of diagnostic inference. I. Problem formulation and decomposition. *Information Sciences*, 37:227-256.

Sticklen, J. (1987). *MDX2: An Integrated Medical Diagnostic System*. PhD thesis, Department of Computer and Information Science, The Ohio State University.

Turner, R. M. (1989a). *A Schema-based Model of Adaptive Problem Solving*. PhD thesis, School of Information and Computer Science, Georgia Institute of Technology. Technical report GIT-ICS-89/42.

Turner, R. M. (1989b). Using schemas for diagnosis. *Computer Methods and Programs in Biomedicine*, 30(2/3):199-208.

Turner, R. M. (1989c). When reactive planning is not enough: Using contextual schemas to react appropriately to environmental change. In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*, pages 940-947, Detroit, MI.

Turner, R. M. (1990). A mechanism for context-sensitive reasoning. Technical Report 90-68, Department of Computer Science, University of New Hampshire, Durham, NH 03824.

Turner, R. M. & Stevenson, R. A. G. (1991). ORCA: An adaptive, context-sensitive reasoner for controlling AUVs. In *Proceedings of the 7th International Symposium on Unmanned Untethered Underwater Submersible Technology (AUV '91)*.