Explicit Representation and Retrieval of Contextual Knowledge for Real-World Agents*

Roy M. Turner
Department of Computer Science
University of Maine
Orono, ME 04469–5752
rmt@umcs.maine.edu

Abstract

For several years, our laboratory has worked on representing and using contextual knowledge to control real-world agents, for example, autonomous underwater vehicles (AUVs). Our approach, called *context-mediated behavior* (CMB), represents contexts as first-class objects called contextual schemas (c-schemas). Each represents a class of situation in which the agent may find itself, and it contains knowledge about how the agent should behave in that context. C-schemas are also instrumental in memory organization, forming the indexing structures of a content-addressable memory. Retrieval is augmented with a diagnostic process to find the appropriate set of c-schemas for a situation, which are then merged to create a coherent view of the context. Currently, we are extending our approach to handle more complex kinds of planning and acting and for use in multiagent systems (e.g., to provide context-appropriate ways to organize and reorganize the agents). This workshop paper gives an overview of the work, including the current status and future plans.

It is generally agreed that context plays an important part in controlling a natural agent's behavior, and that it should play a similar role for artificial agents. Indeed, it is difficult to imagine appropriate behavior without some notion of the context in which the behavior occurs.

Natural agents seem to take context into account automatically when selecting behavior. For example, a person responds differently to the sound of a car horn depending on whether he or she is driving, crossing a street on foot, or sitting at a sidewalk café. A person who is an avid biker still does not consider riding his or her bike on a train, even if the bike is aboard. A person entering a library or church automatically lowers his or her voice without conscious thought.

Somewhat paradoxically, although much context-sensitive behavior is automatic and unconscious, humans are able to reason about the context if need be. For example, a person can enumerate things that are appropriate to do in the library and things that are not, and he or she can list attributes of such contexts.

Although one should not blindly mimic nature in artifacts (jets are better flying machines than ornithopters, for example), often nature provides a good starting point. In this case, automatic recognition of context without conscious attention seems to be a good idea, just as any other sort of "effortless" reasoning would be a good idea. Once the context is identified, by whatever manner, it frees the agent from making decisions about context at each choice point in behavior until the context changes. And the ability to reason about the context, which implies explicit context representation, is also a good idea, since it allows learning about the context, considering hypothetical situations ("If I were in the context of riding a train, what kinds of things would I be able to do to pass the time?"), and possibly merging knowledge about multiple contexts to determine how to behave in novel situations.

There have been many approaches to imbuing intelligent agents with context-awareness. Context was taken into account, for example, by even the earliest planning programs and rule-based expert systems; in those systems, context was encoded as filter conditions and preconditions of operators and as part of rule antecedents. These systems, however, although dealing with explicit knowledge about context, dealt with context itself implicitly. Except in rare systems, there was no attempt to represent the context as an object in its own right, and when there was such an attempt, the representation was meager at best.

In the late 1980s and early 1990s, there began to be interest in representing context explicitly, including work by the author (Turner 1989), Brézillon (Brézillon *et al.* 1997), McCarthy (McCarthy 1995), Giunchiglia (Giunchiglia 1993) and colleagues, and Guha (**?**). Research in this area increased rapidly and began to be reported in the major AI conferences, numerous workshops, and in the CONTEXT (International and Interdisciplinary Conference on Modeling and Using Context) conference series.

This paper presents an overview of the work of our laboratory[1] on context-sensitive reasoning, which we call *context-mediated behavior* (CMB) (Turner 1998). This approach

*This work was funded in part by grants N0001-14-96-1-5009 and N0001-14-98-1-0648 from the Office of Naval Research and BCS–9211914 from the National Science Foundation. The content of this paper does not necessarily reflect the position or the policy of the U.S. government, and no official endorsement should be inferred.

[1]The Maine Software Agents and Artificial Intelligence Laboratory, MaineSAIL; MaineSAIL.umcs.maine.edu.

relies on explicit context representation coupled with automatic context recognition to provide an agent with information about how to control all facets of its behavior. We first discuss our view of what context is, then our representation mechanism, contextual schemas (c-schemas). We then describe one of the mechanisms we have used to organize and retrieve c-schemas as needed, the process of selecting the appropriate one(s) to use for a given situation, and how our agent, Orca (Turner 1995), an intelligent controller for autonomous underwater vehicles, makes use of the contextual knowledge. We then look briefly at some of our ongoing work on context-sensitive reasoning for advanced planning and multiagent systems.

Context

Context is an amazingly elusive thing to define, much less represent, as has been evident from papers in and discussions at the CONTEXT conferences. Some circumscribe context very tightly, while others assert that a context is essentially infinite in scope. Consequently, authors are obliged to define what they mean by context early in their papers.

We are no exception. In our approach to context-sensitive reasoning, we use the term *situation* to mean the complete state of the world and agent(s). We use the term *context* to mean a class of situations, each with the same general implications for an agent's behavior. The process of situation assessment, then, is that of assigning a situation to a particular known context.

This is a very imprecise definition, of course, but an intuitive one. For example, the situation might be that an AUV is between Boston and Cambridge, it is July, and it has been given the goal of collecting CTD (conductivity, temperature, and depth) measurements. This situation would very likely be an instance of several known contexts, including: operating in the Charles River; operating in summer; operating in the Charles River in summer; taking CTD data; taking CTD data in the Charles River; and so on. Which contexts would be known about would depend on the AUV's knowledge base, which might in turn depend on its own history. We can imagine the AUV selecting the most specific, say performing data collection in the Charles River in summer, and using that context to guide its behavior. This context might suggest being very careful when surfacing or operating near the surface, given the likely presence of shells, sailboats, etc.

Explicit Representation of Context

Our work is an outgrowth of early work on case-based reasoning, so it should be no surprise that our representation scheme is similar. In fact, our context representation is much like a generalized case: instead of representing a record of a particular situation, it represents a generalization of a class of cases (situations). We call these generalized cases *contextual schemas*, or *c-schemas* for short.

The exact form and content of c-schemas is domain dependent. Broadly, however, a c-schema contains two types of information, descriptive and prescriptive. Descriptive information describes the context and entities expected to appear in it. In the AUV domain, for example, descriptive information in a c-schema would include information about the physical setting; properties of the AUV specific to the situation, including typical goals the agent has in the context; the presence of other agents, if any; and any context-specific meanings of concepts. A c-schema representing the context of "being in a harbor," C-HARBOR, might contain information about:

- the setting: the water column is shallow, there is clutter on the bottom, there is surface traffic
- concepts: e.g., different membership functions to define the fuzzy linguistic values associated with "depth" ("shallow", "too deep", etc.) (Turner 1997).

A c-schema needs to contain enough descriptive information to allow the agent to identify the context by matching the c-schema to the situation and to realize when the situation no longer is an instance of the context represented.

The prescriptive information contained in a c-schema specifies how the agent should behave while in the context. There are several kinds of such information. *Event-handling knowledge* is needed to allow the agent to determine when an event has occurred (recognition), what the event means (diagnosis), how important it is (assessment), and what to do about it (response). For example, in a harbor, a sonar contact is likely to be with the shore, a vessel traveling on the surface, or one that is docked or moored. In the context of operating in a hostile harbor, however, a sonar contact takes on a different, and possibly more sinister, meaning: it could be a mine. How this is diagnosed and what to do about it depends on the context. In this context, avoiding the mine would likely be the best course, while in the context of a demining mission, the response would be different (report it or disable it).

Attention-focusing knowledge is needed to allow the agent to appropriately and efficiently select what to work on at any given time. This includes knowledge about which goals are reasonable to pursue or not in the current context, as well as the relative importance of particular goals. As noted below, we are extending this kind information in current work to include information about kinds of resources that might be available, resource and time constraints that might be present, etc.

The way goals should be achieved varies by context, and so *action-selection knowledge* is needed to link goals to appropriate methods for achieving them (called *procedural schemas*, or p-schemas, in our approach). For example, in most missions, localization goals and self-preservation goals are important. However, in the context of rescuing a diver, such goals are not as important: it wouldn't make sense to abandon the diver to take a GPS fix, or even to respond to a leak, unless the leak would incapacitate the AUV before it could finish the rescue attempt.

Some aspects of behavior having little to do with goals are also context-dependent. The example of lowering one's voice when entering a library or church falls into this category. We call such behavioral parameter settings *standing orders* and represent them as part of the c-schema. For example, an AUV entering a harbor should automatically tighten its depth envelope (to avoid surface traffic and bottom clutter); when docking, it should disable obstacle avoidance (or risk a very frustrating docking experience); and when in the possible presence of mines, it should slow down.

Retrieving Contextual Schemas: Context Assessment

The process of assessing the current context should be separate from the agent's other reasoning processes. This not only models automatic context recognition, in that it does not interfere with or need be taken into account by the other reasoning, but it also allows the context assessment process to be used with a variety of reasoners.

In our approach, context assessment is a diagnostic process analogous to medical differential diagnosis (Arritt & Turner 2003b). Features of the agent and world take the place of signs and symptoms, and c-schemas (the contexts) take the place of diseases. This is reasonable, as the process of medical diagnosis has as its result the recognition that the patient is in the context of having a particular disease. In medical diagnosis, recognizing the disease allows the doctor to treat it; for an intelligent artificial agent, recognizing the context allows the agent to know how to behave appropriately.

We borrow from the work in AI in medicine for context assessment. In particular, we use an abductive differential diagnosis model based on that used in INTERNIST-I/CADUCEUS (Miller, Pople, & Myers 1982).

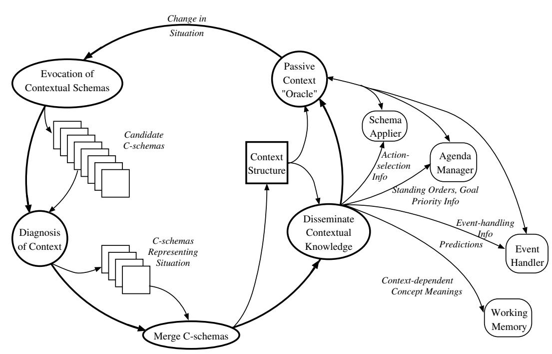

Figure 1: Context management in Orca. (From [Turner, 1998].)

Figure 1 shows the process of context assessment. The process starts when features of the current situation *evoke* particular hypotheses about what the context might be. Although the evocation can be done in any manner, we use a conceptual memory based loosely on Kolodner's CYRUS program (Kolodner 1984; Lawton, Turner, & Turner 1999). Such a memory is content-addressable: features of the situation are presented to the memory as a probe, and the memories indexing structures are traversed based on those features and second-order elaborated features to find the set of the most closely-matching (most specific) c-schemas. The c-schemas returned are marked with how strongly they are evoked by the situation.

Once a candidate set of c-schemas is found, then they are partitioned into logical competitor sets (Feltovich *et al.* 1984). Loosely, each LCS contains those hypotheses that explain the same set of features in the environment. The hypotheses are ranked, based on the evoking strengths and on successful and violated predictions about the situation. The topmost LCS is then examined more closely to see if there is any additional information that will allow it to be reduced to a single hypothesis. Additional reasoning may be needed at this point, and, in the worst case, some actions may need to be taken, requiring the assessment process to interact with the agent's reasoning mechanism. Once the topmost LCS is "solved," features explained by it are marked as such, new LCSs are created based on any new information found, and the process continues.

The process terminates with a set of c-schemas that each fit the situation. At this point, the c-schemas are merged to form a coherent picture of the current context. In this way, existing c-schemas can be combined to represent novel contexts. For example, if an AUV is on a sampling mission near Georges Bank, there are whales nearby, there is surface traffic (from whale-watching boats), and power is low, then the AUV may in fact be in a novel context. However, there may be c-schemas that fit the situation: operating in shallow water, surface traffic present, low power, on a sampling mission, and possibly even operating around whales. These can then be combined to form a representation of the current context from which behavioral information can be obtained.

Using Contextual Knowledge

Although representation and retrieval of contextual knowledge are very important aspects of context-sensitive reasoning, the context management process must still be integrated into the agent's overall reasoning process. We take the approach of having a separate context management module, CONMAN, that can interact with the agent's other reasoning modules.[2]

Figure 1 shows how CONMAN is integrated into the Orca AUV control agent. Other agent modules can register with the context manager to be apprised of new contextual knowledge when the context changes, or they can query CONMAN to obtain the knowledge as needed.

In addition, recent work has focused on augmenting CONMAN to allow it to function as a working memory for those agents, such as Orca, that need that functionality. In this way, the agent's view of its current situation is automatically colored by the context, for example, by having highly-predicted features of the world show up in working memory, marked with the appropriate degree of belief, without the agent having observed or otherwise inferred them.

CONMAN can be used with other kinds of systems that do not have the ability to query for contextual knowledge nor need a working memory. For example, we have used our context-management approach to automatically provide context-appropriate weights and network structure for a neural net (Arritt & Turner 2003a). One could imagine using

[2]The older, published name was ECHO, for "embedded context handling object," but we found the new name irresistible.

CONMAN to provide context-dependent search heuristics for a problem-solving or game-playing agent, constraint-based heuristics for constraint-based reasoners, operators for planners, or rule sets for rule-based systems.[3]

Status and Future Work

Currently, most of the context-mediated behavior approach outlined in this paper has been implemented in the Orca AUV control agent. CONMAN is currently undergoing redesign to make it a more complete working memory for Orca.

In addition, current work is focused on merging c-schemas, which is a very difficult problem. If knowledge from two (or more) c-schemas does not conflict about something, say how to handle an event, then there is no problem. However, the information could conflict, either completely (e.g., surface versus land on the bottom) or partially (e.g., change the sonar rate to twice per second versus twice per minute). Indeed, even detecting a conflict may be difficult. Merging contextual knowledge is the topic of PhD dissertation work in our lab.

Context also comes into play in another project in our lab focused on selecting the appropriate level of commitment to actions and goals during continuous (on-line) planning (Albert, Turner, & Turner). In this project, resources, location, and other important features of the situation organize goals and actions so that the agent can know when the schedule new goals and actions in the evolving template for its future actions. We envision context coming into play here, both in providing appropriate heuristics for organizing and in helping to specify the organizing features that are appropriate in the situation.

A major focus of our work is on multiagent systems, in particular organizing and reorganizing heterogeneous, open MASs such as autonomous oceanographic sampling networks (AOSNs) (Curtin *et al.* 1993; Turner & Turner 2001). We are currently exploring the use of our context-mediated behavior approach to help select appropriate organizational structures.

The basic idea is the same: contexts encode classes of situations, but in this case, some encode situations involving a MAS. These contain as part of their prescriptive information knowledge about which organizational structure is appropriate for the situation. This allows features of the situation to be used to automatically find starting points for organization design. For example, if the situation is one in which there is little an agent can expect to know about other agents, then the corresponding c-schema may suggest a kind of market-based approach, such as some variant of the contract net protocol (Smith 1980). Alternatively, if the situation is one in which communication is limited, yet global coherence is required, then the corresponding c-schema may suggest some sort of hierarchical organization. When multiple c-schemas fit the situation—i.e., the situation is an instance of more than one known context or, equivalently, it is an instance of a novel context—then the c-schema merger process will have to merge organizational structures. This might result in a novel structure, or it might result in a hybrid structure, for example, an overall hierarchy with contracting done in some "departments."

There are several open questions in this research. For example, who should be responsible for assessing the context in a MAS, a single agent, a subset of the MAS, or the MAS as whole? Also, if several agents each assess the context independently, then how can they negotiate to agree on what the context is?

[3] This would be similar in some respects to early AI in medicine work that partitioned rule sets (Chandrasekaran *et al.* 1979).

Conclusion

Context-mediated behavior is one approach to context-sensitive reasoning. It relies on explicitly representing not only contextual knowledge, but also contexts themselves as first-class objects that can be reasoned about. This allows retrieval of all knowledge about a context by matching the current situation to one or more known context representations (c-schemas), which contain the contextual knowledge. The contextual knowledge can then be used to make predictions about the current situation and to prescribe how the agent should behave while in the context. By separating the context-assessment process from the agent's other reasoning, we achieve three things: the automatic (from the perspective of the other reasoning) assessment of context, with the concomitant automatic retrieval of knowledge about how to behave; the ability focus research on the context assessment process itself, irrespective of the agent, including how to have the process learn from experience; and the ability easily to add contextual reasoning to agents that would not ordinarily be able to do explicit context assessment (e.g., neural networks).

The work has been and is being implemented in Orca, an intelligent controller for AUVs and other autonomous intelligent agents. However, we believe that the approach is generic and will be appropriate for a wide range of single-agent and multiagent systems.

References

Albert, E.; Turner, E. H.; and Turner, R. M. Appropriate commitment planning. Submitted to the International Workshop on Planning Under Uncertainty and Execution Control for Autonomous Systems, held in conjunction with the International Conference on Automated Planning and Scheduling (ICAPS'06).

Arritt, R. P., and Turner, R. M. 2003a. Context-specific weights for a neural network. In *Proceedings of the Fourth International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT'03)*, 29–39. Stanford, CA: Springer, New York.

Arritt, R. P., and Turner, R. M. 2003b. Situation assessment for autonomous underwater vehicles using a priori contextual knowledge. In *Proceedings of the Thirteenth International Symposium on Umanned Untethered Submersible Technology (UUST)*. Durham, NH: The Autonomous Undersea Systems Institute, Lee, NH.

Brézillon, P.; Gentile, C.; Saker, I.; and Secron, M. 1997. SART: A system for supporting operators with contextual knowledge. In *Proceedings of the 1997 International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT–97)*, 209–222.

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

Curtin, T.; Bellingham, J.; Catipovic, J.; and Webb, D. 1993. Autonomous oceanographic sampling networks. *Oceanography* 6(3).

Feltovich, P. J.; Johnson, P. E.; Moller, J. A.; and Swanson, D. B. 1984. LCS: The role and development of medical knowledge and diagnostic expertise. In Clancey, W. J., and Shortliffe, E. H., eds., *Readings in Medical Artificial Intelligence*. Reading, Massachusetts: Addison–Wesley Publishing Company. 275–319.

Giunchiglia, F. 1993. Contextual reasoning. *Epistemologia*. Special issue on I Linguagi e le Macchine XVI.

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

Lawton, J. H.; Turner, R. M.; and Turner, E. H. 1999. A unified long-term memory system. In *Proceedings of the International Conference on Case-Based Reasoning (ICCBR'99)*.

McCarthy, J. 1995. A logical ai approach to context.

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

Smith, R. 1980. The contract net protocol: High-level communication and control in a distributed problem solver. *IEEE Transactions on Computers* C–29(12):1104–1113.

Turner, R. M., and Turner, E. H. 2001. A two-level, protocol-based approach to controlling autonomous oceanographic sampling networks. *IEEE Journal of Oceanic Engineering* 26(4).

Turner, R. M. 1989. 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*, 940–947.

Turner, R. M. 1995. Intelligent control of autonomous underwater vehicles: The Orca project. In *Proceedings of the 1995 IEEE International Conference on Systems, Man, and Cybernetics*. Vancouver, Canada.

Turner, R. M. 1997. Determining the context-dependent meaning of fuzzy subsets. In *Proceedings of the 1997 International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT-97)*.

Turner, R. M. 1998. Context-mediated behavior for intelligent agents. *International Journal of Human–Computer Studies* 48(3):307–330.