Appropriate Commitment Reactive Planning

Erik Albert and **Elise H. Turner** and **Roy M. Turner**
Computer Science Department
University of Maine
Orono, ME 04469-5752
{albert17, eht, rmt}@umcs.maine.edu

Abstract

We propose a method for reactive mission planning in dynamic, real-world environments that allows for *appropriate* commitments to future actions and goals. We consider appropriate commitments to be those that are less likely to become invalidated, or that are important enough to justify the risk. By augmenting a reactive mission planner to organize its actions around these commitments, we hope to create a planner that is able to react quickly to unanticipated events, easily incorporate new goals, coordinate its actions with other agents, and perform missions more efficiently and effectively.

Introduction

Fully autonomous mission planning for unmanned vehicles operating in dynamic and uncertain domains, such as the underwater vehicle domain, is a very challenging problem. Due to the ever-evolving nature of the environment, an agent needs to be able to respond quickly to unanticipated events such as an effector failure or a new obstacle. In addition, these agents need to be able to prepare for future actions and events such as a rendezvous with a collection vessel.

Agents operating in dynamic environments operate with a scarcity of knowledge about their situation. The exact locations of obstacles such as ships, and hazards such as fishing nets and currents, cannot be fully known ahead of time. In many missions of interest, an agent also operates without full knowledge of the goals of the mission itself. In a collaborative, long-term scientific mission, an agent may know little (if any) of the goals of the mission before it begins.

The single biggest challenge to mission planning in these complex environments is that commitments to specific actions and situations often become invalidated. There are trade-offs involving the level of commitment in any planning system. While these commitments prove to be useful, and sometimes essential, for organizing and coordinating actions, there is a chance of each commitment not being realized. When this occurs in a planning system, the plan (or a portion of the plan) is invalidated and the planner needs to replan or to correct the invalidated portion. If the agent is operating using a fixed plan, the invalidation of the plan leads to mission failure.

Even plans created by a decision-theoretic (Blythe 1999) or other stochastic approach have a high likelihood of failure in a highly dynamic environment, and the plans they create only contain contingencies for failures that can be predicted. A scarcity of knowledge about the domain or the mission hampers the ability of these approaches to create robust plans and the computational expense limits the ability to use these systems onboard a fully autonomous vehicle.

Reactive planners (Agre and Chapman 1987; Firby 1987; Georgeff and Lansky 1987) approach this problem by remaining fully, or at least mostly, reactive to the current situation rather than relying on a fixed plan. In the extreme, behavior-based variants of reactive planning (e.g., Bonasso and Barratt 1993; Komerska et al. 1999; Turner et al. 1993; Smith et al. 1996; Bellingham et al. 1990; Bellingham et al. 1994), no commitments to future actions are made and an agent continually chooses one action to perform based on the current situation.

By allowing for a reactive plan (at some level of detail), moderate reactive planners (e.g., Gat 1991; Bonasso et al. 1997; Jensen and Veloso 1998; Choi et al. 2004), including the Orca planner (Turner 1995), improve upon behavioral reactive planners by allowing for some commitment to future actions. Typically, however, there is only minimal commitment to future specific actions, even those that can be predicted with a high degree of confidence will be needed. These commitments are important because they impose some structure to a mission plan that can be exploited to organize the remainder of the goals and actions in the mission.

What we propose in this paper is a method for explicitly identifying important goal interactions in a dynamic reactive plan, allowing a reactive planner to make some justified commitments to future actions or situations. These predictions in turn will allow the agent to rationally interleave and organize actions from multiple goals without the need to use expensive deliberate methods. When new goals arise, the agent will be able to insert them into its plan where it makes sense, or to reorganize its plan appropriately.

The AUV Domain

Our research has focused primarily on mission planning for the autonomous underwater vehicle (AUV) domain. However, the planning mechanisms introduced can be used for planning in any dynamic environment by replacing the domain-dependent knowledge base. The AUV domain has several properties that make it ideal for this research. The first is the need for full autonomy. In many scenarios, it is impossible to keep in constant communication with an AUV to allow for human-assisted planning techniques. Deep-sea missions, missions under surface cover, and stealth missions are all examples of times when communication with the AUV is not available. The second interesting property of the AUV domain is its truly dynamic nature. The composition of the environment, such as currents and traffic, can be in constant flux.

Regardless of the complexity of planning in the AUV domain, there exist many important applications for the military, industry, and academia, such as surveillance, mine detection and clearing, salvage and rescue, underwater inspection, underwater construction, and autonomous oceanographic sampling networks (Curtin et al. 1993).

The Orca Reactive Mission Planner

Orca is a reactive mission-level planner for the AUV domain (Turner 1994; 1995). Rather than controlling an AUV (or other type of unmanned vehicle) directly, Orca is designed as the mission-level reasoner for a hierarchical control architecture such as the MSEL software architecture (Blidberg and Chappell 1986). Figure 1 shows Orca situated in the mission planning role in the MSEL architecture. This hierarchical approach presents Orca with an abstract "virtual AUV" allowing it to be more broadly applicable to other autonomous vehicles and operate as a "soft" real-time system.

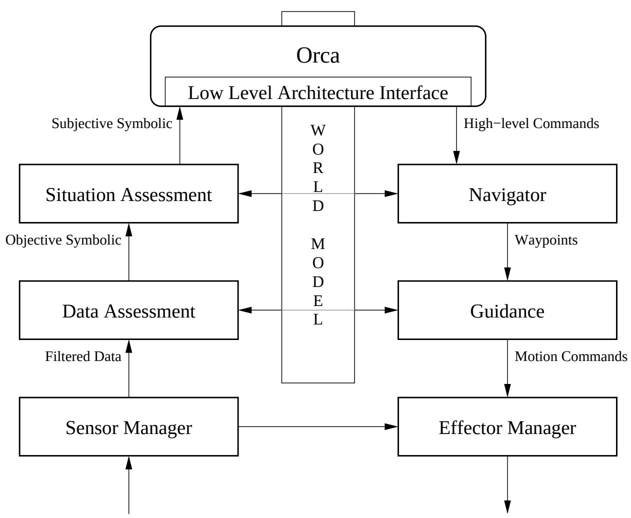

Figure 1: Orca situated in the MSEL software architecture

Orca uses a knowledge base of schemas for all of its reasoning. A schema is a representation of patterns that exist in the real world and in problem solving. Procedural schemas (p-schemas) are similar to hierarchical plans and specify the steps that must be taken to achieve goals. These steps can be primitive actions, other p-schemas, or subgoals. Procedural schemas form a generalization/specialization hierarchy and multiple p-schemas can exist in the knowledge base to achieve a single goal.

Orca uses an agenda to list the goals that comprise its intentions. At any time, Orca focuses on the one best goal from the agenda for the current situation. A p-schema is found to achieve the goal and Orca expands the partial plan, executing primitive actions as they are found. If the situation changes, Orca will look for a more-appropriate specialization of the existing p-schema, or will begin work on a different goal from the agenda.

There are several shortcomings of this approach to focusing attention. The Orca planner currently has no method for giving an overarching view of the mission as a whole. Orca cannot summarize or predict its intended actions over the course of a mission, other than just listing the goals (and what work has been accomplished towards solving these goals) in its agenda. This can lead to difficulty when coordinating with other agents over the course of a mission, as well as a reduced utility in being able to rationally organize actions from different goals.

Orca can also be narrow-minded in its approach to solving the goals in its agenda. Because Orca only ever focuses attention on one goal in its agenda, it cannot serendipitously or opportunistically work on other goals in parallel. This may cause a vehicle to have to, for example, double-back and transit to a location to perform an action for a second goal that would have best been performed while the vehicle was in the area originally.

Appropriate Commitment Reactive Planning

Our work intends to augment the Orca reactive planner with the ability to make *appropriate* commitments to future actions and situations. We consider appropriate commitments to be those which are based on predictable features of the environment, mission, and plan which are important enough to justify the risk (and cost) of those predictions not being realized. For mobile agents in real-world environments, these features include location and high-cost or shared resources.

In order to effectively support these commitments, we are addressing two important issues. The first is the necessity for an explicit representation of the intentions and commitments of the planner that is flexible enough to support dynamic reorganizations and goal additions. This representation should make explicit important interactions between goals so that reasoning about these interactions can be automatic and inexpensive. Secondly, we need a technique for focusing attention (deciding what to do next) that can utilize all of the information encoded in the plan representation.

In moderate reactive systems, there is some representation (called a reactive plan) of the current intentions and priorities of the planner, its active behaviors, and possibly a record of what work has already been performed by the agent. To facilitate the identification of – and organization around – appropriate commitments, we will explicitly store information about the current plan in a *reactive plan network*. This planning structure will be used to represent all of the goals of the planner, any commitments to future actions, and all of the schemas (at various levels of detail) currently being used to achieve goals. This new planning structure will support a reactive agent's ability to dynamically order and reorder actions and goals without breaking convention, group actions that can be executed together, temporally separate actions that compete for a shared resource, and consider alternative courses of action.

The reactive plan network will also contain *organizational nodes*, which are used to explicitly identify interactions and relationships between different goals and actions. Each of these nodes represents an *organizational feature* of the plan or environment; these features are costly to acquire, expensive or limited, and can be predicted with a high confidence. We have identified location, temporally constrained actions, and resources as being important organizational features in real-world domains.

Figure 2 shows a simple example of a reactive plan network. In this diagram, plan components are drawn using circles (goals), triangles (p-schemas), and squares (executable actions). There is also an organizational node, drawn as a double-circle, connected to several actions in the plan. Plan components that are part of the same subplan to solve a goal are connected by specialization links. These links connect a goal to the p-schema that achieves the goal, and each p-schema is connected (after being expanded) to the actions, subschemas, and subgoals that are its steps. Constraint links are used to enforce constraints between plan components such as ordering of plan steps.

Superficially, the reactive plan network appears very similar to a hierarchical task network (e.g., Ghallab, Nau, and Traverso 2004; Horling et al. 1999), however, rather than just being used to create a plan, the reactive plan network is also used to maintain different pieces of the plan at various levels of detail, keep a record of the actions that have already been completed, monitor plan execution, and identify a broad class of interactions between various plan components. Different areas of the reactive plan network may be explored at different times, and the representation does not implicitly give any indication about the order that various actions will be executed.

The organizational nodes of the reactive planning network explicitly identify a broad class of both positive and negative goal interactions, which can be used by the planner to easily exploit positive interactions and resolve negative ones.

Positive goal interactions include common subgoals (Thangarajah, Padgham, and Winikoff 2003), actions from disjoint goals that can be achieved at the same time or at the same location, and commitments to future actions that allow the agent to choose from a set of alternatives a less-expensive, or more optimal, course of action that would not have been available otherwise.

Negative goal interactions, on the other hand, arise from resource limitations as well as limitations of the agent. For example, a physical agent cannot perform actions in two separate locations at the same time. Negative interactions occur frequently when actions require the use of limited and/or shared resources.

Focus of attention in an appropriate commitment reactive planner, for both execution and plan refinement, is driven by an activation metaphor. Sources of activation include *intention* and *organization*. Intentional activation is derived from the importance (priority) of the goals in the plan and causes the planner to focus on completing the primary tasks.

Each organizational node in a reactive plan network can also lend organizational activation to connected plan components based on heuristics governed by the type of organizational feature. This type of activation can be used by the planner to take advantage of positive interactions while alleviating negative ones. There are several basic strategies for resolving negative goal interactions (Schneider and Detweiler 1988; Freed 1998) including:

- **circumventing**: choosing non-conflicting alternatives;
- **delaying/interrupting**: temporally separating conflicts;
- **shedding**: removing low-priority or infeasible tasks; and
- **time sharing**: alternating the control of a resource.

All of these strategies can be implemented using heuristics that take into consideration the “advice” of all pertinent organizational nodes.

Choosing the next action to execute, deciding when and how the plan should be refined, and choosing between alternative methods to achieve a goal are all forms of commitment and are all controlled by the same activation-based mechanism. The planning process can be thought of as finding a consensus between the advice of the organizational strategies and the intentions of the planner.

Supporting Adaptability

The appropriate commitment reactive planning approach maintains high-level reactivity by maintaining a flexible and adaptable planning structure. The planner does not immediately make commitments to how and when goals and tasks will be carried out when they are added to the plan. Instead, this process is controlled through the activation metaphor. Changes to the activation of a plan component by its connected organizational nodes can cause the planner to either add detail, make a commitment, or conversely, to delay action or expansion, backtrack, or commit to an alternative course of action.

When a new goal is added to the reactive plan network, it is connected to the organizational nodes that represent the organizational features identified as being influential to the schema(s) that could be used to solve the goal. If these organizational nodes share a high level of activation, the planner may immediately begin working on solving this goal. On the other hand, if activation is low, the planner may ignore the goal until a more appropriate situation arises.

Figure 3 shows the major knowledge components used by the Orca planner. In addition to the reactive plan network that is used to represent the current intentions and approach of the planner, there is also an episodic memory which stores all of the procedural schemas and a working memory which stores the general knowledge, domain knowledge, and situational knowledge of the planner. This working memory is available to all components of the planning system.

As updates about the situation arrive from the low level architecture, they are posted to the working memory by a module called the event manager. This module can also monitor for anticipated events and notify different planning components when an event occurs. For example, the event manager will notify the planner/executor component when

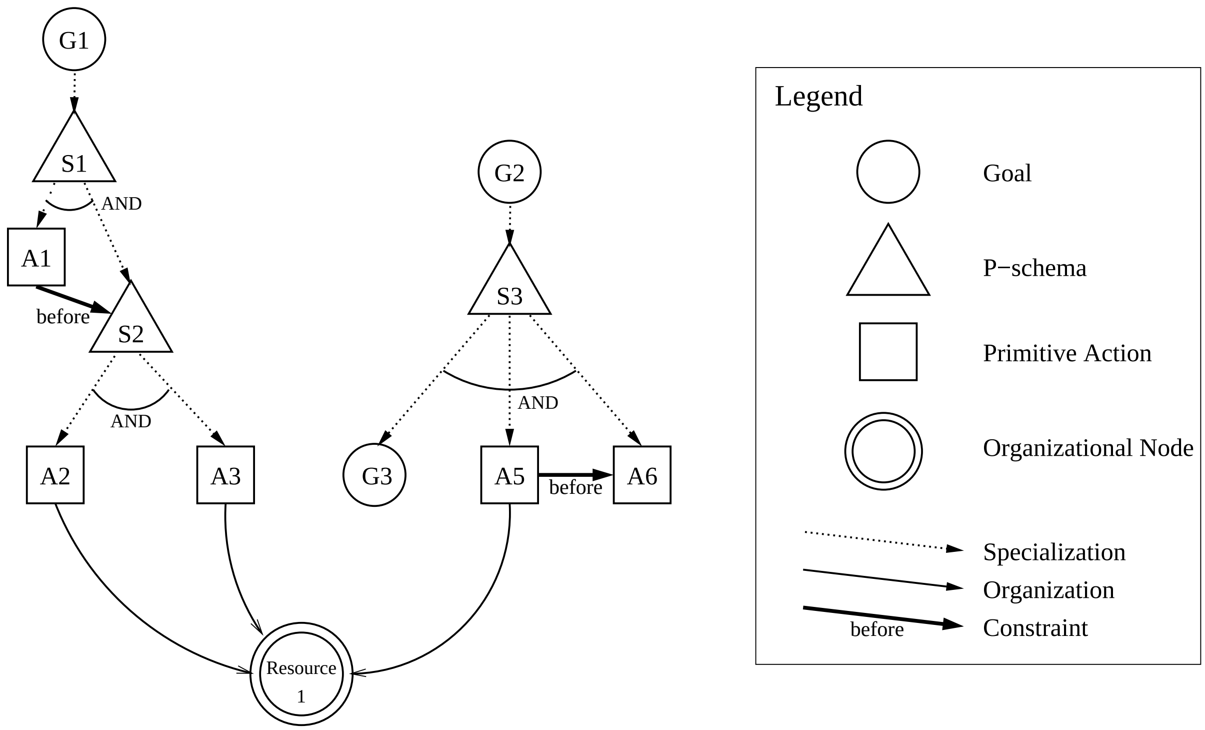

Figure 2: A Simple Reactive Plan Network

an effector event (e.g., waypoint achieved or effector failure) or an organizational feature event (e.g., resource exhausted or acquired) occurs. These organizational events allow the planner to update each related organizational node as the situation changes. Such events may cause the activation of new goals (e.g., to handle an incipient problem), the deletion of a goal, a change to how a goal is satisfied, or a re-evaluation of how to order the actions for execution.

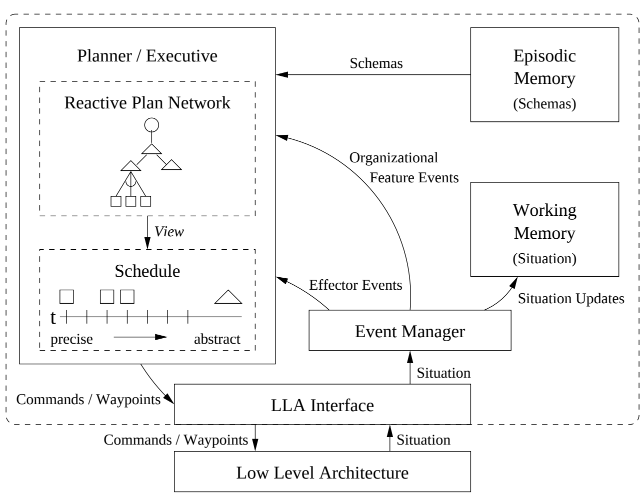

Figure 3: Orca's Knowledge Components

Conclusions and Future Work

In this paper we propose a reactive mission-level planning system for use in a hierarchical reactive control system that allows for appropriate commitments to future actions. The planner will use its commitments to organize the goals and actions in its reactive plan and incorporate new goals. The goal of this work is to create a controller for fully-autonomous vehicles operating in real-world domains that has the following abilities:

- **The ability to remain responsive to changes in the environment and mission** and be able to quickly adapt and incorporate new goals without the need to replan from scratch;
- **The ability to dynamically order and reorder actions and goals**, from multiple goals, and to perform non-mutually exclusive actions from disjoint goals in parallel. By organizing around organizational features, the planner should be able to accomplish tasks efficiently and to rectify negative goal interactions; and
- **The ability to maintain conventions** to remain predictable and able to cooperate with other agents that follow conventions and protocols.

In order to achieve these properties in a reactive mission planner, we are developing an organizational structure called a reactive plan network that allows us to explicitly represent and reason about individual plan components and their interactions. Components of the reactive plan network receive activation from both intentional and organizational sources, and this activation is used by the planner to identify when to add detail to the plan, when to choose between alternative courses of action, and when to commit to primitive actions. Intentional activation causes the planner to give preference to high-priority goals and to follow convention, while organizational activation allows the planner to group actions that should be executed together and to separate or order actions that have a negative interaction, such as competition for a shared resource.

By reasoning about plans at various levels of detail, but only committing to appropriate actions, the plan remains flexible and the agent is able to respond quickly to unanticipated events. This flexible nature of the plan also allows for new goals to be added at any time, with the organizational nodes of the reactive plan network grouping new goals with any related goals and actions already in the plan.

We are currently implementing a new, agent-based version of the Orca planner as a test-bed for our appropriate commitment approach to reactive mission planning. Future areas of work for this project include

- Determining the proper level of detail for plan expansion. This involves determining when to find p-schemas that could be used to achieve a goal, when to select between alternatives, and when to expand sub-goals and sub-schemas. Activation heuristics need to be developed to control how the focus of attention mechanism handles plan expansion. The planner will need to expand components to various levels of detail in order to predict and exploit organizations of actions based on the strategies defined in this paper without over-expanding (an unnecessary expense which can also lead to over-commitment).
- Finalizing the activation model. Ideally, the activation model would be a simple numerical system, with little or no propagation, in order to maximize its efficiency. However, it may be that the different types of "advice" given by organizational nodes cannot be abstracted using a single number.

  For example, consider a goal that has been given a low activation due to the fact that it is temporally constrained and must be achieved at a time in the future. If the planner only sees the low activation level, it may continue to attempt to solve the goal using different alternatives rather than simply ignoring the goal for the time being.

  The activation model may need to be extended into an *advice* model that encodes the reason for the activation assigned by an organizational node. It might also be satisfactory for the organizational nodes to be allowed to add constraint links to the reactive plan network to resolve these issues.
- As the planner interleaves actions and subschemas from various procedural schemas, a problem of maintaining consistency arises. Procedural schemas may not contain a great deal of justification about the order, or information about the relationships, between their steps, and the planner needs to be careful when reordering them not to do so in such a way that violates any preconditions. We need to determine the best approach for maintaining consistency in a reactive plan network, which may involve posting standing orders, adding protection intervals (Tate 1975) or causal links (McAllester and Rosenblitt 1991), and/or posting contingencies to follow if a precondition is violated.
- Development of a method or set of criteria used for identifying new organizational features of the environment or reactive plans.

Acknowledgments

The authors would like to thank the United States Office of Naval Research for its support of this work under grants N00014-01-1-0818 and N00014-00-1-0614. The content does not necessarily reflect the position or the policy of the U.S. government, and no official endorsement should be inferred.

References

Agre, P. E., and Chapman, D. 1987. Pengi: An implementation of a theory of activity. In *Proceedings of the National Conference on Artificial Intelligence*, 268–272. Los Altos, California: Morgan Kaufmann.

Bellingham, J. G.; Consi, T. R.; Beaton, R. M.; and Hall, W. 1990. Keeping layered control simple. In *Proceedings of the Symposium on Autonomous Underwater Vehicle Technology (AUV '90)*, 3–9. IEEE.

Bellingham, J.; Goudey, C.; Consi, T.; Bales, J.; and Atwood, D. 1994. A second-generation survey AUV. In *Proceedings of the 1994 IEEE Symposium on Autonomous Underwater Vehicle Technology (AUV'94)*, 148–155.

Blidberg, D. R., and Chappell, S. G. 1986. Guidance and control architecture for the EAVE vehicle. *IEEE Journal of Oceanic Engineering* OE–11(4):449–461.

Blythe, J. 1999. Decision-theoretic planning. *AI Magazine* 20(2):37–54.

Bonasso, R. P., and Barratt, J. 1993. A reactive robot system for find and visit tasks in a dynamic ocean environment. In *Proceedings of the Eigth International Symposium on Unmanned Untethered Submersible Technology (AUV'93)*, 69–80. MSEL, Northeastern University.

Bonasso, R. P.; Firby, J.; Gat, E.; Kortenkamp, D.; Miller, D. P.; and Slack, M. G. 1997. Experiences with an architecture for intelligent, reactive agents. volume 9, 237–256.

Choi, D.; Kaufman, M.; Langley, P.; Nejati, N.; and Shapiro, D. 2004. An architecture for persistent reactive behavior. In *AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, 988–995. Washington, DC, USA: IEEE Computer Society.

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

Firby, R. J. 1987. An investigation into reactive planning in complex domains. In *Proceedings of the Sixth National Conference on Artificial Intelligence*, 202–206.

Freed, M. 1998. Managing multiple tasks in complex, dynamic environments. In *AAAI '98/IAAI '98: Proceedings* *of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence*, 921–927. Menlo Park, CA, USA: American Association for Artificial Intelligence.

Gat, E. 1991. Integrating reaction and planning in a heterogeneous asynchronous architecture for mobile robot navigation. *SIGART Bull.* 2(4):70–74.

Georgeff, M. P., and Lansky, A. L. 1987. Reactive reasoning and planning. In *Proceedings of the National Conference on Artificial Intelligence*, 677–682. Los Altos, California: Morgan Kaufmann.

Ghallab, M.; Nau, D.; and Traverso, P. 2004. *Automated Planning: Theory and Practice*. Morgan Kaufmann. chapter 11.

Horling, B.; Lesser, V.; Vincent, R.; Wagner, T.; Raja, A.; Zhang, S.; Decker, K.; and Garvey, A. 1999. The TAEMS White Paper.

Jensen, R., and Veloso, M. 1998. Interleaving deliberative and reactive planning in dynamic multi-agent domains. In *Proceedings of the AAAI Fall Symposium on on Integrated Planning for Autonomous Agent Architectures*. AAAI Press.

Komerska, R.; Chappell, S. G.; Peng, L.; and Blidberg, R. 1999. Generic behaviors as an interface for communication, command and monitoring between AUVs. Technical Report 9904-01, Autonomous Undersea Systems Institute, 86 Old Concord Turnpike, Lee, NH.

McAllester, D., and Rosenblitt, D. 1991. Systematic nonlinear planning. In *Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI-91)*, volume 2, 634–639. Anaheim, California, USA: AAAI Press/MIT Press.

Schneider, W., and Detweiler, M. 1988. The role of practice in dual-task performance: toward workload modeling in a connectionist/control architecture. *Human Factors* 30(5):539–566.

Smith, S.; Ganesan, K.; Dunn, S.; and An, P. 1996. Strategies for simultaneous multiple AUV operation and control. In *IARP'96*.

Tate, A. 1975. Interacting goals and their use. In *Proceedings of the Fourth International Joint Conference on Artificial Intelligence (IJCAI-75)*, 215–218. Tbilisi, Georgia: IJCAI.

Thangarajah, J.; Padgham, L.; and Winikoff, M. 2003. Detecting & exploiting positive goal interaction in intelligent agents. In *AAMAS '03: Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, 401–408. New York, NY, USA: ACM.

Turner, R. M.; Blidberg, D. R.; Chappell, S. G.; and Jalbert, J. C. 1993. Generic behaviors: An approach to modularity in intelligent systems control. In *Proceedings of the 8th International Symposium on Unmanned Untethered Submersible Technology (AUV'93)*.

Turner, R. M. 1994. *Adaptive Reasoning for Real-World Problems: A Schema-Based Approach*. Hillsdale, NJ: Lawrence Erlbaum Associates.

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.