ORGANIZATIONAL STRATEGIES FOR INFORMED COMMITMENT IN A REACTIVE MISSION PLANNER

Erik Albert
Department of Computer Science
University of Maine
Orono, ME 04469–5752
albert17@umcs.maine.edu

Abstract

In this paper I evaluate an approach to autonomous, reactive mission planning that uses organizational features of the situation to inform commitment to future actions and goals. By organizing actions around these commitments, we have created a planner that is able to react quickly to unanticipated events, operate opportunistically, and quickly incorporate new goals without the need to replan. I have built a reference implementation of the planner that operates on simple domains that can be used to evaluate the feasibility and scalability of the approach.

Introduction

The task of autonomous mission planning in the underwater vehicles domain is a complicated task that is only exacerbated by the dynamic nature of the environment, a lack of complete knowledge of the environment and the mission, and uncertainty about the actual tasks and goals that will need to be accomplished.

Of special interest are long-term, collaborative missions of self-organizing agents. One such type of mission is an autonomous oceanographic sampling network (AOSN) [8], which can be used for a variety of tasks such as scientific monitoring and long-term surveillance. An AOSN is an open system, meaning that equipment (agents) may be added to, or removed from, the AOSN at any time. Agents in the AOSN may need to take on additional tasks when another leaves (or fails), and new agents are assigned tasks only after they join the mission.

Agents operating autonomously in real world environments continuously have to deal with the problem of their plans becoming invalidated. Irregardless of the robustness of any plan, there remains a large chance that the commitments made during planning are inappropriate during execution. The arrival of new goals (through collaboration, in response to equipment failure, etc.) can also invalidate an agent's plan.

Many approaches to flexible control of AUVs (e.g., [6, 14, 21, 17, 3, 4]) have attempted to alleviate the over-commitment problem by employing a *reactive planning* [1, 9, 11] approach. These planners, while avoiding unnecessary over-commitments, suffer from not being able to make any commitments to future actions, such as those required for collaboration. Being completely reactive, it is difficult to bound or predict the behavior of an agent utilizing this type of approach.

Another common approach is to use a fast classical planner (such as FastForward [13] or Graphplan [5]) and to replan whenever the plan becomes invalidated. In this approach, the planner completely commits to future actions, following the assumption that when a commitment becomes invalidated that a new plan can be generated quickly. While this approach may work well in simple domains, these approaches do not scale well as the complexity of the domain increases.

What is needed is an approach that allows for *appropriate* commitment to future actions and goals while otherwise remaining reactive to the current situation. We consider appropriate commitments to be those that are based on features of the situation that are important enough to justify the risk of the

commitment becoming invalidated [2]. A planner can use these appropriate commitments to organize actions and inform additional commitments in the mission plan.

In the next section, I will give an overview of our method for planning in dynamic environments. I will then show the experimental results obtained by running our planner, and a fast replanning system, in simulated environments. Finally, I will give my conclusions and outline areas of future work.

The Appropriate Commitment Reactive Planning Approach

Our approach, called *appropriate commitment reactive planning*, is a mission-level reactive planning system. We have based our approach on Orca [20], a schema-based reactive mission planner. A schema is an explicit 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 goals. A schema-based planner solves a mission by applying schemas from its knowledge base to achieve the goals in its agenda.

Our work extends the schema-based planning approach to allow for informed selection between alternatives, improved interleaving of actions from disjoint schemas, the quick addition of new goals into the mission plan, and an increased efficiency in mission execution. To meet these ends, we have first introduced an explicit representation for the intentions and commitments of the planner that is flexible enough to allow for dynamic reorganization and goal addition. This representation makes explicit important interactions between goals, so that reasoning about these interactions can be automatic and inexpensive. Secondly, we have created technique for focusing attention (deciding what to do next) that can utilize all of the information encoded in the plan representation.

The Reactive Plan Network

The reactive plan network is an explicit representation of an agent's current plan, including goals, pro-

cedural schemas, alternatives currently under consideration, and organizational information. Components in a reactive plan network are connected to the other components to which they are related. For example, a goal will be connected to the procedural schema that has been chosen for its achievement. There is never any effort made to totally order the plan components in a reactive plan network – a planner may decide to work on any area of the plan at any time[1].

An *organizational feature*[2] of the plan or the environment is something that is costly to acquire, expensive or limited, can be predicted with a great deal of certainty [19], or otherwise directly influences the appropriateness of commitments in the reactive plan network. In real world domains, such as the AUV domain, these features include location and quantitative and qualitative resources [15].

Each organizational feature in a domain is modeled by an *organizational node* in the reactive plan network. Plan components are connected to the features that are known to influence them with *use links* and to features that may influence them, or influence a possible expansion of them, using *examine links*.

An example reactive plan network (from the student experiment, described below) is shown in Figure 1. The organizational node representing the location of the agent is depicted at the center of the network, and is connected through organizational links (solid lines are use links and dashed lines are examine links) to all of the goals that require the agent to be at any specific location.

Focus of Attention

In order to determine what to do next (perform an action, select a p-schema for a goal, choose between alternatives, add information to a schema in the plan, etc.), the ACRP planner uses a simple activation metaphor. Each component in the plan receives two types of activation: *intentional activation*, which is based on the priority of goals, and *organizational activation*, based on the organizational strategies specific to each type of organiza-

---

[1]There is support for the specification of uninterruptible sequences, such as those used to implement protocols.

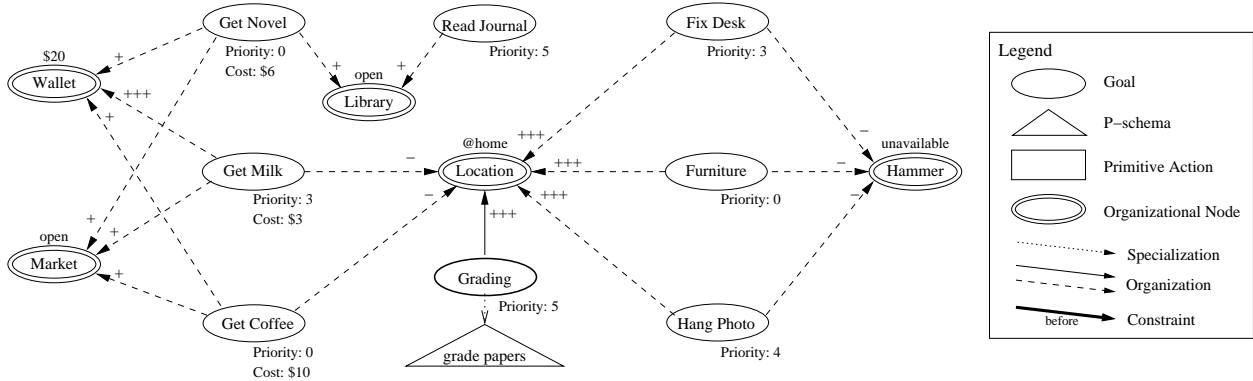[2]Previously called *predictive feature* [2].

Figure 1: An example reactive plan network

tional feature.

In the reactive plan network of Figure 1, organizational activations are depicted as a series of pluses (+) or minuses (-). The pluses represent positive activation, or activation that encourages the selection of a particular plan component, while the minuses represent negative activation – activation that delays the selection of a plan component, causes a plan component to be shed from the mission plan, or causes one plan alternative to be selected over another. For example, the location organizational node is giving a large amount of positive activation to those plan components that require the agent to be at the location `@home`, the current location of the agent. Other plan components receive a slight negative activation.

We are currently evaluating a single value activation metaphor, where the activations from all sources are compiled into a single activation value. Each component is ranked by their compiled activation value in order to determine where the planner will focus its attention next. This approach is efficient, as we can treat all sources of activation as equivalent "advisors," and in order to focus attention, we only need to take into consideration one activation value per component.

A planner operating with the reactive plan network of Figure 1 would likely focus its attention on the `grade papers` procedural schema that achieves the Grading goal, due to its large amount of both positive organizational activation (from the location organizational node) and positive intentional activation (from the high priority of the Grad-

ing goal).

Experiments

In order to evaluate the feasibility of our approach, we have designed two experimental domains: the student domain and the islands domain. These two domains stress the ability of the ACRP approach to handle the types of resource- and location-constrained missions that exist in the AUV domain. We have kept each of the domains very simple in order to be able to quickly implement an experimental version of the ACRP planner, to evaluate specific features of the ACRP approach in isolation, and to be able to encode the domains using the planning domain definition language (PDDL) [16, 10] to allow for comparison with existing planners.

In addition, we have built a discrete-event domain simulator that allows for the experimental simulation of our domains. The simulator also loads a simulation file (encoded in a PDDL-like language) that allows the specification of real action durations, future events, and scenarios.

For these experiments, we decided to specify goal priorities as an integer value between zero and five, with zero meaning that a goal is optional and five meaning that a goal is critical to the success of a mission. Activation values are specified as real numbers in the range [-3, 3] so that it is easy to differentiate positive and negative activation. Goal priorities are converted into intentional activation in the range [1, 2].

For comparison, we have also run our experi-

ments with the Metric-FF planner [12], an extension of the FastForward planner that allows for numeric state variables. In the experiments that we designed, there are often situations in which a plan that satisfies all goals is not possible. The Metric-FF planner (along with many other classical planners), is not guaranteed to terminate when there is no possible plan. We devised the following strategies that allow planning for subsets of the goal set:

Contract In the contract strategy, the planner first attempts to find a plan that solves all of the goals in the plan. If that fails, all of the goals at the lowest priority level are removed and Metric-FF is re-invoked. This continues either until a plan is found or all goals have been removed.

A timeout parameter limits the length of time that the Metric-FF planner is allowed to run during each invocation.

Contract with Individuals In this strategy, the planner first attempts to find a plan using the contract technique. If this fails, the planner attempts to find a plan for a single goal, starting with the goal with the highest priority.

Expand In the expand strategy, the planner first orders the goals in the mission from highest to lowest priority. The planner then attempts to find the first goal for which a plan can be derived. If no plan is found within the time limit, the goal is removed from consideration; otherwise, it is added to a set of included goals. The planner continues to attempt to add each of the remaining mission goals to the inclusion set. Because of the initial ordering of the goals, priority is given to the goals with high priority values.

This approach has two parameters: a timeout that limits the amount of time that the Metric-FF planner will be allowed to run, and an optional time limit that specifies the total amount of time that the expand technique is allowed to run while generating a single plan.

If, at the end of a planning cycle, a plan could not be generated, the agent will perform a wait action before attempting to replan. Hopefully, after waiting, the situation will have changed in such a way that a plan will now be possible.

Organizational Features of the Experimental Domains

In the experimental domains, we identified a number of organizational features that directly influence the appropriateness of commitments. These features, while simple in these experimental domains, map directly to features of the AUV domain.

Simple Discrete Locations

This organizational features is a simplification of the complex location feature into several distinct locations that are fully connected. Travel between locations may take time or be otherwise costly (fuel, money, etc.). This feature is used when movement is abstracted away to very high-level movement functions such as "drive to the library."

The activation strategy used by the simple discrete locations organization node is to give positive activation to components that require the current location, and negative activation to components that require another (in the student experiment, these values are -1 and 1.5, respectively).

Two-Dimensional Discrete Locations with Naturally Bounded Areas

The two-dimensional discrete location feature is a model of the world in which location is abstracted into a two-dimensional grid, where each cell in the grid is the same size and travel between cells is relatively uniform. In this abstraction, the agent can only move from one cell to an adjacent cell and cannot move diagonally.

A naturally bounded area (NBA) is a permanent, or extremely long-duration, feature of the environment that is costly or dangerous for an AUV to cross [18, 7], such as a strong current.

The organizational node that represents this feature uses a simple strategy for assigning activation to connected plan components. If a component requires the agent to be in a different NBA, the component receives negative activation; otherwise, the component receives activation inversely proportional to the distance of the required location from

the agent's current location.

If there are components that require the agent to change NBAs, the organizational node will add a goal to change NBA to the reactive plan network that has a priority equal to the maximum priority of all of the components of which it is required.

Quantitative Resources

A quantitative resource is a resource that is consumed when used. In the AUV domain, example quantitative resources include electricity, fuel, and sensors with limited charges (such as biological tests). There are many factors that determine how an organizational node distributes activation to connected components, including: the amount of resource remaining, the amount of resource to be consumed by a component, the priorities of the plan components, and whether or not there is a method of replenishing the resource.

Qualitative Resources

Qualitative resources can generally be considered as "tools," meaning that as they are used they are not consumed (e.g., a hammer), but their state may be changed (i.e., they may be unavailable while in use). For example, an AUV might possess an acoustic modem for communication; however, while the modem is in use for transmitting a message, it is unavailable for the transmission of other messages.

The strategy that an qualitative resource organizational node uses for dispersing activation is to give a large amount of positive activation while the resource is available, and a small amount of negative activation otherwise.

If a qualitative resources is not available and its associated organizational node has an component connected by a use organizational link, or if there is a connected component with a sufficiently high priority, the organizational node may add a goal to acquire the resource (if one does not already exist) to the reactive plan network.

Time-Limited Resources

A time-limited resource is a type of qualitative resource in which the availability of the resource is determined by the current time. An example time-limited resource in the AUV domain is the sun; a solar-panel equipped AUV can only recharge its batteries during the daylight hours.

The activation strategy employed by time-limited resource organizational nodes is to give a small amount of negative activation while the resource is unavailable, and to give an amount of positive activation inversely proportional to the amount of time the resource will remain available. The actual activation is given by the formula:

$$A_F + (A_M - A_F)\frac{T_{NOW} - T_A}{T_U - T_A}$$

where T_{NOW} is the current time, T_A and T_U are the times at which the resource becomes available and unavailable, A_F is the minimum amount (the fixed) activation that the organizational node will give, and A_M is the maximum amount of activation that will be given. A_F and A_M are parameters that can be specified per resource.

The Student Experiment

The student domain was designed to evaluate the ability of our planning approach to manage scarce resources. One criterion that we had when designing this experiment was to make it approachable to non-specialists in order to get experimental data from human participants. To that end, the goals and tasks in this experiment are based on common tasks that a graduate student may need to accomplish on a Sunday in order to be prepared for work and classes on Monday.

There are three discrete locations in the student domain: @home, @market, and @library. An agent in this domain can travel between locations at any time; however, this takes time and consumes fuel.

The mission involves completing a number of goals that require the use of the agent's resources (e.g., money) and the domain's resources (e.g., the library). Many of the goals are given at the start of the mission, although two more are added later during the mission time period. The complete set of goals are:

- Mission critical goals (priority 5):
    - Read journal (available from the library)

- Grade papers

- Required goals:

  - Hang photograph (priority 4)
  - Fix desk (priority 3)
  - Buy milk (priority 3)
  - Buy juice (priority 3; new at 10:38 AM)

- Optional goals (priority 0):

  - Get novel (available from the market and the library)
  - Buy coffee
  - Assemble furniture
  - Buy chocolate (new at 11:10 AM)

Additionally, there is a preservation goal to return any object that has been borrowed.

In order to evaluate performance in this domain, each mission is scored based on which goals have been completed. Each optional goal is worth ten points and each required goal is worth points equal to ten times its priority. If any of the mission critical goals have not been completed, a 200 point penalty is applied. There is a 100 point bonus if all of the required goals have been completed, and an additional 100 point bonus if all of the optional goals listed above have also been completed. The status of the preservation goal to return objects is not considered when calculating the bonus. The highest possible mission score is 480.

This domain consists of a number of organizational features, including the library and market (time-limited resources), money (a quantitative resource), a hammer and a neighbor (qualitative resources), and the agent's location (a simple discrete location feature). The neighbor is a resource that, when available, can lend the agent a hammer. Several of the required actions require the use of a hammer, and if the agent does not borrow the hammer from the neighbor, it will have to spend half of its money to purchase a hammer from the market.

The student domain is completely deterministic and fully observable; however, the planners are given no information about future events and only estimates for action durations. There are also three separate scenarios which vary in the availability of the neighbor:

no-ted The neighbor is never at home.

early-ted The neighbor arrives home at 8:30 AM but leaves again at 10:05 AM.

late-ted The neighbor arrives home at 12:12 PM.

For the human trials, I created a web-based interface that allows the user to select the next action to perform from the set of all actions that are currently applicable. The interface displays the results of each action, the location of the player, and the current time. The user is responsible for keeping track of all other details.

The ACRP experimental planner and the Fast-Forward planning strategies were also run on each of the three simulation scenarios[3]. The contract and contract with individuals strategies were run with a thirty second timeout. The expand strategy was run with a ten second timeout and no total time limit for plan generation. Figures 2 and 3 show the results of the student experiment.

In Figure 2, you can see the average human score compared with the scores obtained by the automated planners. The error bars show the standard error of the mean (because the domain is deterministic, each automated planner scored the same score on each of its trial runs). In each of the scenarios, the ACRP planner and the FF / Expand planner achieve the best scores. In the early-ted scenario, the ACRP planner borrows the hammer from the neighbor early in the mission and is able to accomplish all goals aside from returning the hammer (for a mission score of 470).

In figure 3, you can see the average time spent by each planner. The ACRP experimental planner performed at least as well as the Metric-FF planner using the expand strategy, while only taking a fraction of the total time (less than two seconds during each of its trials).

The Islands Experiment

The islands experiment was designed to evaluate the ability of the Appropriate Commitment Reactive Planning approach to effectively manage movement in a two-dimensional domain with naturally

---

[3]The trials were run on a computer with a 2GHz Intel Core2 Duo CPU and 756MB of memory.
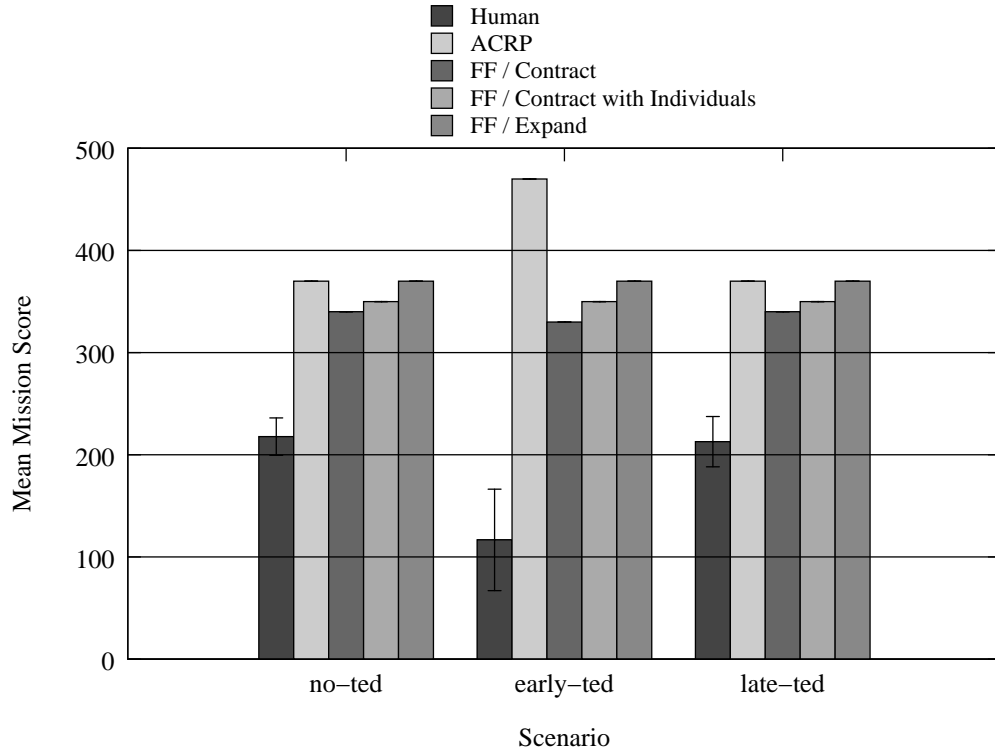
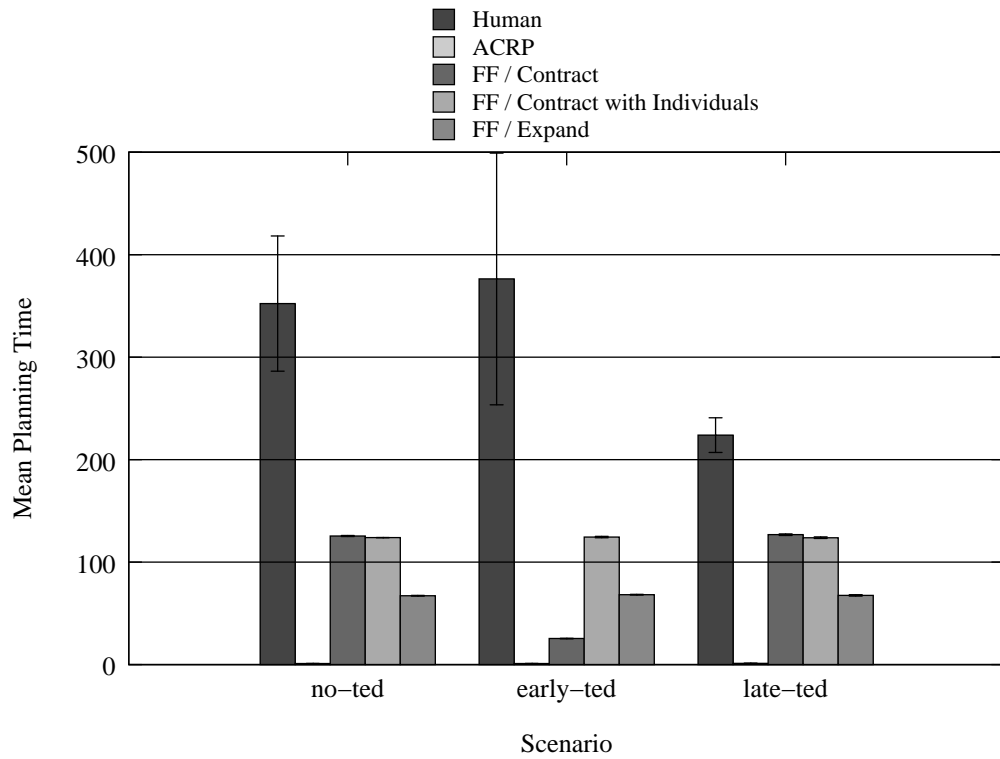Figure 2: Student experiment: mean mission score by scenario

Figure 3: Student experiment: mean planning time by scenario

bounded areas. In this simulation, all of the goals in the mission plan require the agent to collect and deliver packages. Each mission begins with a random number of goals (drawn from a normal distribution with mean twenty and variance six) and new goals are added throughout the mission following a Poisson process with a parameterized rate.
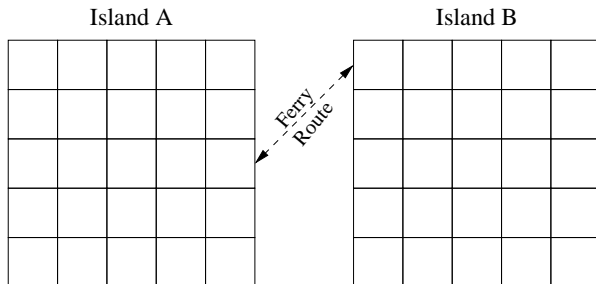

Figure 4: The twin islands

There are two islands in this domain, each divided into $n \times n$ discrete locations (Figure 4). The agent can move between any two adjacent locations, and movement takes on average three minutes (the actual time is drawn from a normal distribution with mean three and variance 0.25). In order to travel between islands, the agent must take a ferry. There are two ferries that travel back and forth between the islands. They are scheduled to board fifteen minutes before the hour, to depart on the hour, and to arrive (on average) at thirty-eight minutes after the hour. If a ferry arrives late (the actual transit time is drawn from a normal distribution with mean thirty-eight and variance ten), the ferry will still allow fifteen minutes for boarding and will depart late.

The agent using the ACRP approach does not explicitly plan to visit specific locations in the domain. When the agent focuses on an action, and the action requires the agent to be in a different location, the agent moves one position closer to the location for the action. After moving, the agent refocuses on a new component in the reactive plan network.

We ran the ACRP planner and the Metric-FF planner[4] (using the contract strategy with a thirty second timeout, and the expand strategy with a ten

second timeout and two minute time limit) on ten sets of 5x5 islands, while varying the rate at which new goals are added to the mission. The experimental results are shown in Figures 5 and 6.

Figure 5 shows the mean delivery rate separated by the rate of new goals. The ACRP planner and the Metric-FF planner using the expand strategy perform similarly, while the Metric-FF planner using the contract strategy performs relatively poorly. In Figure 6, you can compare the average planning time taken by each of these methods. While both the ACRP approach and the FF / Expand approach score similarly with regards to delivery rate, the amount of time taken by the FF / Expand approach grows much more quickly as the rate of new goals increases. The longest time taken by the ACRP approach in all sixty of its trials was 36 seconds, compared with one hour and twenty-seven minutes for the FF / Expand approach).

Conclusions and Future Work

Through experimentation, we have shown that a planner using the appropriate commitment planning approach remains reactive to changes in the situation, such as the introduction of new resources and goals, by dynamically refocusing based on the activation values generated by the organizational nodes in a reactive plan network. As the situation changes, the planner quickly adjusts without having to spend time replanning.

Even when utilizing a very simple activation metaphor, the ACRP approach performs very well in these experimental domains, scoring at least as well as the FF / Expand technique, while exhibiting a much slower run time growth as the complexity of the problem increases.

Having shown the plausibility of our approach, we will now extend the approach to operate on a more comprehensive knowledge base of the autonomous underwater vehicle domain, such as the one used by the Orca mission planner, and a more expressive domain definition language. Through simulation, we will evaluate how well our approach scales from these simple domains, and reevaluate our activation metaphor if necessary.

In this paper I have addressed only domain-

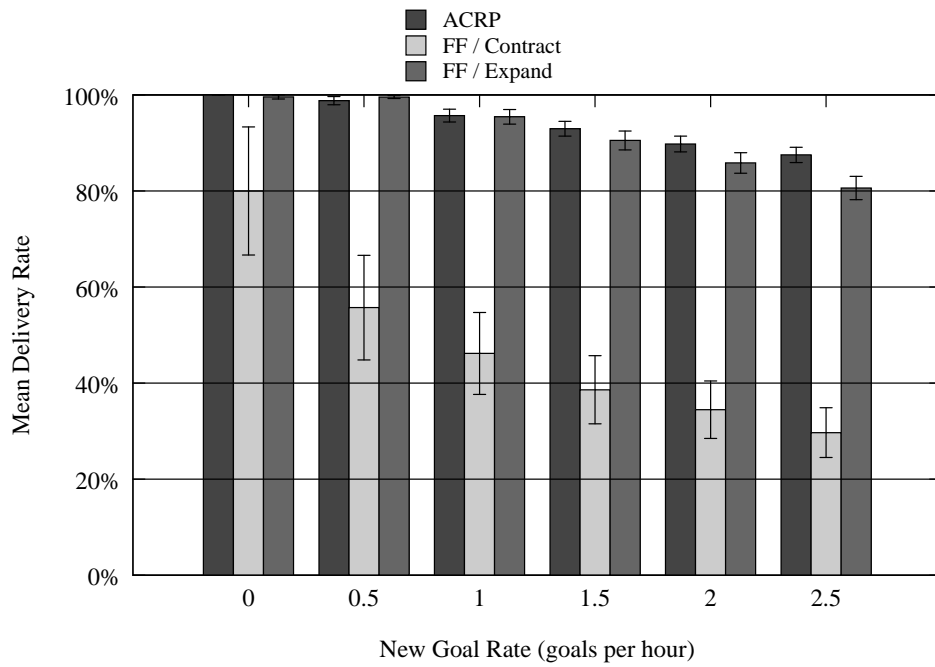[4]The trials were run on a computer with two AMD Opteron 240 CPUs and 2GB of memory.

Figure 5: Islands experiment: mean delivery completion rate by goal arrival rate
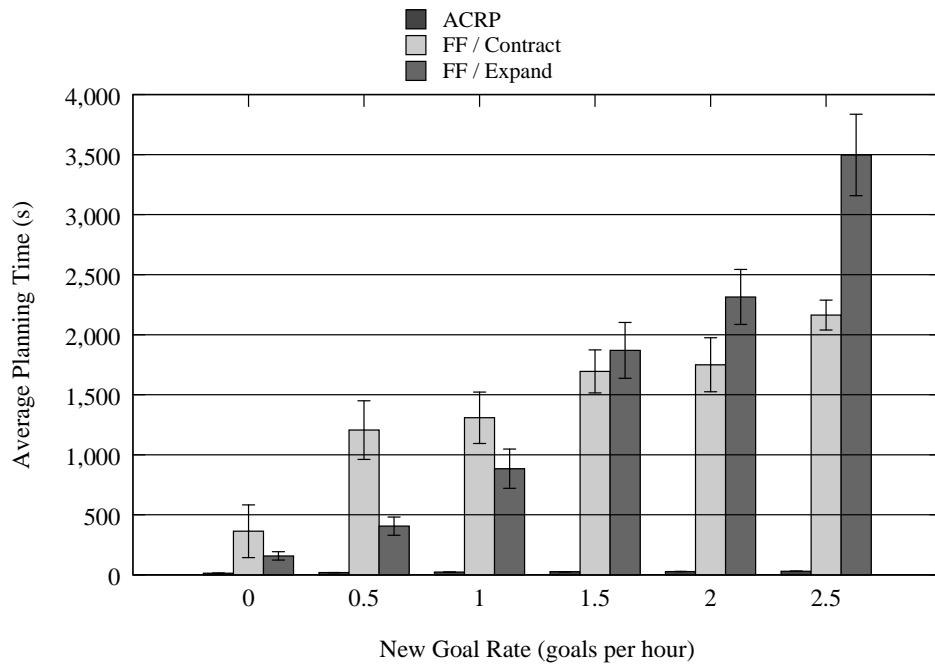

Figure 6: Islands experiment: mean planning time by goal arrival rate

dependent organizational features; however, not all organizational features are necessarily related to the domain. In the future, we would like to study domain-independent organizational features related to recognizable patterns in the reactive plan network, or domain-independent strategies for manipulating the reactive plan network (such as techniques for ordering non-mutually-exclusive actions that can be completed in parallel).

Many of the strategies employed by the organizational nodes are parameterized, such as the A_F and A_M parameters of the time-limited resource node. The experimental implementation uses a fixed value for each of these parameters, but in the future we would like to use machine learning techniques to allow the planner to learn the best values of these parameters over time.

References

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

[2] E. Albert, E. H. Turner, and R. M. Turner. Appropriate commitment planning for AUV control. In *Proceedings of the International Symposium on Unmanned Untethered Submersible Technology (UUST)*, Durham, NH, 2007. The Autonomous Undersea Systems Institute.

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

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

[5] A. L. Blum and M. L. Furst. Fast planning through planning graph analysis. *Artificial Intelligence*, (90):281–300, 1997.

[6] R. P. Bonasso and J. Barratt. 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)*, pages 69–80. MSEL, Northeastern University, 1993.

[7] T. L. Briggs. Exploiting natural boundaries in the autonomous underwater vehicle domain to limit resource utilization. Technical Report 91-15, University of New Hampshire, December 1991. M.S. thesis.

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

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

[10] M. Fox and D. Long. Pddl2.1: An extension to pddl for expressing temporal planning domains. *Journal of Artificial Intelligence Research (JAIR)*, 20:61–124, 2003.

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

[12] J. Hoffmann. Extending FF to numerical state variables. In *Proceedings of the 15th European Conference on Artificial Intelligence (ECAI-02)*, pages 571–575, Lyon, France, July 2002.

[13] J. Hoffmann and B. Nebel. The FF planning system: Fast plan generation through heuristic search. 14:253–302, 2001.

[14] R. Komerska, S. G. Chappell, L. Peng, and R. Blidberg. 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, 1999.

[15] D. Long, M. Fox, L. Sebastia, and A. Coddington. An examination of resources in planning, 2000.

[16] D. McDermott. The planning domain definition lanuage manual. Technical Report 1165, Yale Computer Science, 1998. (CVC Report 98-003).

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

[18] E. H. Turner and T. L. Briggs. Responding to unanticipated goals when planning travel for autonomous underwater vehicles. In *Proceedings of of the Tenth IEEE Conference on Artificial Intelligence Applications (CAIA-94)*, pages 493–494, 1994.

[19] E. H. Turner, R. M. Turner, and E. Albert. Placing newly-arising goals in the proper context. In *Workshop notes for Cooperative Systems and Context Workshop of the Fifth International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT-05)*, Paris, France, July 5–8 2005.

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

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