Learning Structured Reactive Navigation Plans from Executing MDP Navigation Policies

Michael Beetz
Technical University Munich
Dept. of Computer Science IX
Orléansstrasse 34
D-81667 Munich, Germany
beetzm@in.tum.de

Thorsten Belker
University of Bonn
Dept. of Computer Science III
Roemerstr. 164
D-53117 Bonn, Germany
belker@cs.uni-bonn.de

ABSTRACT

Autonomous robots, such as robot office couriers, need navigation routines that support flexible task execution and effective action planning. This paper describes XFRMLEARN, a system that learns structured symbolic navigation plans. Given a navigation task, XFRMLEARN learns to structure continuous navigation behavior and represents the learned structure as compact and transparent plans. The structured plans are obtained by starting with monolithical default plans that are optimized for average performance and adding subplans to improve the navigation performance for the given task. Compactness is achieved by incorporating only subplans that achieve significant performance gains. The resulting plans support action planning and opportunistic task execution. XFRMLEARN is implemented and extensively evaluated on an autonomous mobile robot.

1. INTRODUCTION

Robotic agents operating in human working environments and solving dynamically changing sets of complex tasks are challenging testbeds for autonomous robot control. The dynamic nature of the environments and the nondeterministic effects of actions requires robots to exhibit concurrent, percept-driven behavior to reliably cope with unforeseen events. Moreover, acting competently often requires foresight and weighing alternative courses of action.

Different approaches have been proposed to specify the navigation behavior of such service robots. A number of researchers consider navigation as an instance of Markov decision problems (MDPs). They model the navigation behavior as a finite state automaton in which navigation actions cause stochastic state transitions. The robot is rewarded for reaching its destination quickly and reliably. A solution for such problems is a *policy*, a mapping from discretized robot poses into fine-grained navigation actions.

MDPs form an attractive framework for navigation because they use a uniform mechanism for action selection and a parsimonious problem encoding. The navigation policies computed by MDPs aim at robustness and optimizing the average performance. One of the main problems in the application of MDP planning techniques is to keep the state space small so that the MDPs are still solvable. This limits the number of contingent states that can be considered.

Another approach is the specification of environment- and task-specific navigation plans, such as ***structured reactive navigation plans (*SRNPs*)*** [1]. SRNPs specify a default navigation behavior and employ additional concurrent, percept-driven subplans that overwrite the default behavior while they are active. The default navigation behavior can be generated by an MDP navigation system. The activation and deactivation conditions of the subplans structure the continuous navigation behavior in a task-specific way.

SRNPs are valuable resources for opportunistic task execution and effective action planning because they provide high-level controllers with subplans such as traverse a particular narrow passage or an open area. More specifically, SRNPs (1) can generate qualitative events from continuous behavior, such as entering a narrow passage; (2) support online adaptation of the navigation behavior (drive more carefully while traversing a particular narrow passage), and (3) allow for compact and realistic symbolic predictions of continuous, sensor-driven behavior. The specification of good task- and environment-specific SRNPs, however, requires tailoring their structure and parameterizations to the specifics of the environmental structures and empirically testing them on the robot.

We propose to bridge the gap between both approaches by learning SRNPs from executing MDP navigation policies. Our thesis is that a robot can autonomously learn compact and well-structured SRNPs by using MDP navigation policies as default plans and repeatedly inserting subplans into the SRNPs that significantly improve the navigation performance. This idea works because the policies computed by the MDP path planner are already fairly general and optimized for average performance. If the behavior produced by the default plans were uniformly good, making navigation plans more sophisticated would be of no use. The rationale behind requiring subplans to achieve significant improvements is to keep the structure of the plan simple.

2. AN OVERVIEW ON XFRMLEARN

XFRMLEARN is embedded into a high-level robot control system called ***structured reactive controllers*** (SRCs) [1]. SRCs are are controllers that can revise their intended course of action based on foresight and planning at execution time. SRCs employ and reason about plans that specify and synchronize ***concurrent percept-driven*** behavior. Concurrent plans are represented in a transparent and modular form so that automatic planning techniques can make inferences about them and revise them.

XFRMLEARN is applied to the RHINO navigation system, which

AGENTS'01, May 28-June 1, 2001, Montréal, Quebec, Canada.

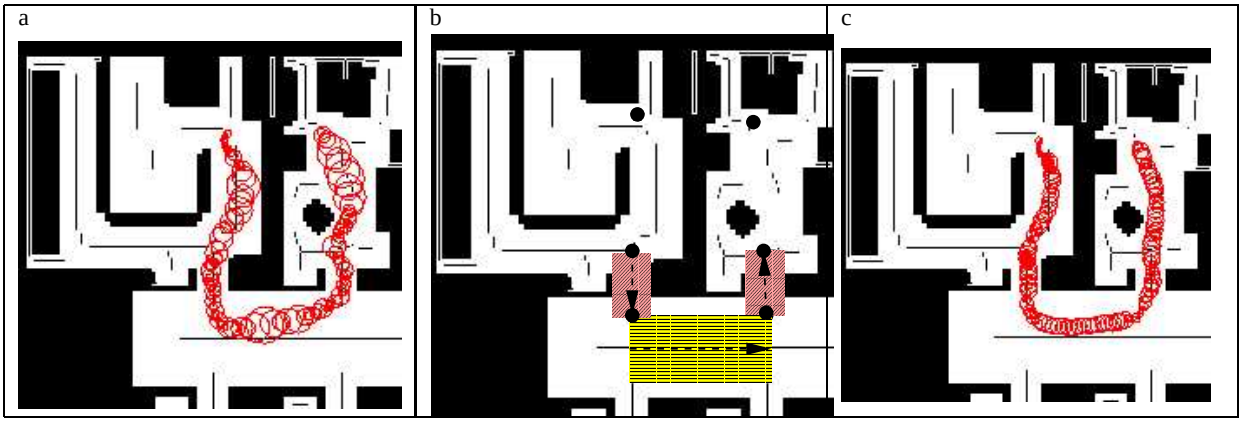

Figure 1: Behavior trace of the default plan (a). Low T-Vel subtraces (b). Learned SRNP (c).

conceptually works as follows [3].. A navigation problem is transformed into a Markov decision problem to be solved by a path planner using a value iteration algorithm. The solution is a policy that maps every possible location into the optimal heading to reach the target. This policy is then given to a reactive collision avoidance module that executes the policy taking the actual sensor readings and the dynamics of the robot into account.

XFRMLEARN executes an "analyze, revise, and test", starting with a default plan that transforms a navigation problem into an MDP problem and passing the MDP problem to RHINO's navigation system. After RHINO's path planner has determined the navigation policy the navigation system executes the resulting policy. XFRMLEARN records the resulting navigation behavior and looks for stretches of behavior that could be possibly improved. XFRMLEARN then tries to explain the improvable behavior stretches using causal knowledge and its knowledge about the environment. These explanations are then used to index promising plan revision methods that introduce and modify subplans. The revisions are then tested in a series of experiments to decide whether they are likely to improve the navigation behavior. Successful subplans are incorporated into the symbolic plan.

3. EXPERIMENTAL RESULTS

To empirically evaluate XFRMLEARN we have performed two long term experiments in which XFRMLEARN has improved the performance of the RHINO navigation system, a state-of-the-art navigation system, for given navigation tasks by up to 44 percent within 6 to 7 hours. A summary of the first session is depicted in Figure 1. Figure 1(a) shows the navigation task (going from the desk in the left room to the one in the right office) and a typical behavior trace generated by the MDP navigation system. Figure 1(b) visualizes the plan that was learned by XFRMLEARN. A typical behavior trace of the learned SRNP is shown in Figure 1(c). We can see that the behavior is much more homogeneous and that the robot travels faster. The t-test for the learned SRNP being at least 21% faster returns a significance of 0.956. A bootstrap test returns the probability of 0.956 that the variance of the performance has been reduced. In the second learning session, the average time needed for performing a navigation task has been reduced by about 44%. The t-test for the revised plan being at least 18% faster has a significance of 0.952 (see [2]).

4. CONCLUSIONS

We have sketched XFRMLEARN, a system that learns SRNPs, symbolic behavior specifications that (a) improve the navigation behavior of an autonomous mobile robot generated by executing MDP navigation policies, (b) make the navigation behavior more predictable, and (c) are structured and transparent so that high-level controllers can exploit them for demanding applications such as office delivery.

XFRMLEARN is capable of learning compact and modular SRNPs that mirror the relevant temporal and spatial structures in the continuous navigation behavior because it starts with default plans that produce flexible behavior optimized for average performance, identifies subtasks, stretches of behavior that look as if they could be improved, and adds subtask specific subplans only if the subplans can improve the navigation behavior significantly.

The learning method builds a synthesis among various subfields of AI: computing optimal actions in stochastic domains, symbolic action planning, learning and skill acquisition, and the integration of symbolic and subsymbolic approaches to autonomous robot control. Our approach also takes a particular view on the integration of symbolic and subsymbolic control processes, in particular MDPs. In our view symbolic representations are resources that allow for more economical reasoning. The representational power of symbolic approaches can enable robot controllers to better deal with complex and changing environmants and achieve changing sets of interacting jobs. This is achieved by making more information explicit and representing behavior specifications symbolically, transparently, and modularly. In our approach, (PO)MDPs are viewed as a way to ground symbolic representations.

5. REFERENCES

[1] M. Beetz. Structured reactive controllers — a computational model of everyday activity. In O. Etzioni, J. Müller, and J. Bradshaw, editors, *Proceedings of the Third International Conference on Autonomous Agents*, pages 228–235, 1999.

[2] M. Beetz and T. Belker. Environment and task adaptation for robotic agents. In W. Horn, editor, *Procs. of the 14th European Conference on Artificial Intelligence (ECAI-2000)*, 2000.

[3] S. Thrun, A. Bücken, W. Burgard, D. Fox, T. Fröhlinghaus, D. Hennig, T. Hofmann, M. Krell, and T. Schmidt. Map learning and high-speed navigation in RHINO. In D. Kortenkamp, R.P. Bonasso, and R. Murphy, editors, *AI-based Mobile Robots: Case studies of successful robot systems*. MIT Press, Cambridge, MA, 1998.