Hierarchical POMDP Decomposition

Joelle Pineau

Problem: Partially Observable Markov Decision Processes (POMDPs) provide a useful framework for decision-making in the presence of uncertainty. They were specifically developed to address the problem of choosing optimal actions in partially observable stochastic domains. Finding solutions to large-scale problems, however, has proven computationally infeasible. We propose a hierarchical approach to POMDPs which takes advantage of structure in the domain to decompose the problem into a collection of smaller POMDPs. These can be solved independently, allowing us to solve larger problems than were previously possible. Furthermore, latest efforts have focused on developing tools to perform automatic decomposition of a given conventional POMDP model. We apply our approach to the problem of human-robot speech dialogues.

Impact: POMDPs are well-suited to a great number of real-world problems where decision-making is required despite prevalent uncertainty, and as such have gained much attention in recent years[4]. Despite the representational power of POMDPs, their use is significantly limited by the great computational cost of finding an optimal policy for the learner. Consequently, research efforts have focused on the development of efficient algorithms to generate near-optimal policies. Current algorithms, however, are still generally unable to handle complex problems. We believe this can be overcome by exploiting the structure present in many real-world domains. This is in contrast to the conventional POMDP approach of treating the problem as a monolithic structure.

The domain currently targeted by our work is that of dialogue management for speech-based interactions between a robot and a human user. The role of our conversational interface is mostly one of information-provider, which has a structure dictated by the information content available in a given application. We believe this domain can be naturally decomposed into a hierarchy and our approach can be used to solve large POMDPs in this context. This would allow the development of a large-scale dialogue system that is more robust to errors in speech recognition and ambiguous statements than previous dialogue management systems.

State of the Art: Robot interactions with users are in many cases governed by manually scripted state-action sequences, which are generally not robust to unexpected events and require significant re-engineering given a change in the environment. Other work on modeling human-machine interactions as Markov processes has focused on using Markov Decision Processes (MDP) and reinforcement learning to generate optimal dialogue strategies [6, 5], thereby assuming full observability at all times. Given the uncertainty inherent in today's speech recognition technology, POMDP models offer a more appropriate framework for managing robot-human dialogues because of their ability to handle uncertainty.

A significant amount of work exists on exploiting hierarchy to reduce the complexity of planning tasks. Hierarchical decomposition of Markov Decision Processes (MDPs) has been the subject of significant research [7, 2, 3], however, there has been little work on extending this to POMDP problems. Preliminary attempts make strict assumptions about prior knowledge of sub-tasks and their ordering [1, 8], which we do not require.

Approach: There are two important sections to this work. In the first part, we focus on developing tools to automatically decompose conventional POMDP models. Secondly, we are concerned with the computation of optimal policies, using hierarchically decomposed POMDPs. Most of the work to date has targetted this second problem.

The hierarchical POMDP approach relies on a division of task which gives higher-level POMDPs direct control over more specialized POMDPs. At the high level, the POMDP policy consists of selecting appropriate sub-modules (i.e. hierarchical actions), while at the lower-level, specialized POMDPs have policies involving direct actions onto the problem domain (i.e. non-hierarchical actions). This division of tasks is not strict, and POMDPs of all levels can have both hierarchical and non-hierarchical actions, with the exception of lowest-level POMDPs which have only non-hierarchical actions. Belief states are maintained over all POMDPs of the hierachy at all times, and are updated after each action/observation pair. When a specific specialized POMDP is selected, it is identified as an "active module", and is responsible for the learner's action selection at that given time step. Other non-active modules act as HMMs during that time step; their belief state is updated according to a default action, thereby ensuring that they capture the entire history and will produce the appropriate behaviour when later selected as an "active module".

We have shown that in a domain such as a conversational speech interface, structure can indeed be exploited to obtain a policy much faster than with a conventional POMDP, and furthermore allowed us to build a larger POMDP-based dialogue manager than was possible with non-hierarchical POMDP representations. Such structure is not present in all domains, however is found in a great variety of applications, ranging from the dialogue task, to robot navigation tasks where the domain could be divided into a set of separate problems according to building topology (e.g. each room gets a separate POMDP). The speedup obtained by applying our hierarchical approach to POMDPs can be remarkable in domains with appropriate structure.

Future Work: Our current approach assumes that a hierarchical structure of POMDPs is provided, possibly hand-crafted, to find an optimal strategy. We are currently looking at techniques to automate the decomposition process, such that a conventional POMDP could be provided and automatically decomposed in an appropriate manner. Furthermore, given that a conventional POMDP can be decomposed in a number of different ways, we need a method to determine which decomposition is better, such that it allows the computation of a maximal reward policy, while most reducing the complexity required to compute this policy. Some loss of performance can be expected when defining a hierarchical structure for a problem, since we use value approximation at the higher nodes of the hierarchy to represent the value of selecting the various specialized lower-level POMDPs. Empirical evidence shows that this loss of performance is minimal in domains where the use of hierarchy is justified by the structure of the domain. We are looking at techniques to predict bounds on the performance loss as a function of the selected hierarchy.

Selected References:

[1] D.A. Castanon. Approximate dynamic programming for sensor management. In *Proc. 1997 Conf. Decision and Control*, 1997.

[2] P. Dayan and G. Hinton. Feudal reinforcement learning. In *NIPS 5*, pages 271–278, San Francisco, CA, 1993. Morgan Kaufmann.

[3] T. G. Dietterich. The MAXQ method for hierarchical reinforcement learning. In *ICML*, 1998.

[4] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101:99–134, 1998.

[5] E. Levin, R. Pieraccini, and W. Eckert. Learning dialogue strategies with the Markov decision process framework. In *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, 1997.

[6] S. Singh, M. Kearns, D. Litman, and M. Walker. Reinforcement learning for spoken dialogue systems. In *Proceedings of NIPS'99*, Denver, CO, December 1999.

[7] S. P. Singh. Transfer of learning by composing solutions of elemental sequential tasks. *Machine Learning*, 8:323–339, 1992.

[8] M. Wiering and J. Schmidhuber. Hq-learning. *Adaptive Behavior*, 1997.