Believing in POMDPs

Felix Richter, Thomas Geier, and Susanne Biundo

Institute of Artificial Intelligence,
Ulm University, D-89069 Ulm, Germany,
email: forename.surname@uni-ulm.de

Abstract. Partially observable Markov decision processes (POMDP) are well-suited for realizing sequential decision making capabilities that respect uncertainty in *Companion* systems that are to naturally interact with and assist human users. Unfortunately, their complexity prohibits modeling the entire *Companion* system as a POMDP. We therefore propose an approach that makes use of abstraction to enable employing POMDPs in *Companion* systems and discuss challenges for applying it.

1 Introduction

Companion systems are cognitive technical systems that live and act in a real-world environment. As they must adapt themselves to their human users, they have to be aware of human-specific states such as emotions and dispositions to support their decisions. They are fitted with a set of sensors that provide them with a multi-modal set of observation channels like speech, video, or even biophysiological signals. Despite their rich sensory fitting, the variables of interest are usually concealed from the *Companion* system, and they can be accessed only indirectly through noisy channels [5,6]. Based on this imperfect perception, decisions have to be taken in a way that maximizes the utility of the system as a whole.

Partially observable Markov decision processes (POMDP) constitute a class of models of sequential decision making under uncertainty that is capable of capturing the described observation processes. POMDPs are an extension of Markov Decision Processes (MDP) that hides the state of the environment from the acting agent, and formalizes an observation model that captures how information can be perceived through incomplete and noisy channels. Extending MDPs by partial observability has the following two consequences: First, the policy followed cannot be a simple mapping of observations to suitable actions, as it is the case for the purely reactive MDP agents. Receiving only partial information about the current state makes past observations informative for estimating the true *current* state of the environment. The second consequence is that a POMDP agent is aware of the value of information, and it has an incentive to execute actions that improve its knowledge/estimate, even if these actions have no influence on the future evolution of the environment. This makes POMDPs particularly suited for applications that involve dialogue [21], as asking is an act with the sole purpose of acquiring information.

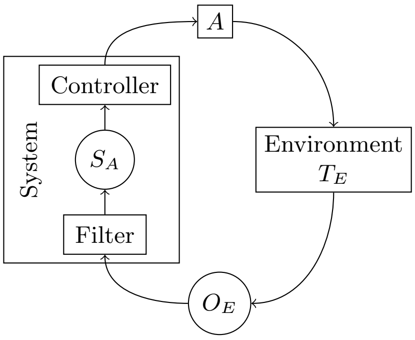

Fig. 1. Illustration of a filter-based controller architecture.

While it is adequate to model a complete *Companion* system as a POMDP, it appears impractical. Despite recent progress in the field [15,14], it is still out of scope for contemporary POMDP solvers to both operate over long time-scales and process high-dimensional video observations, due to the *curse of history* [12]: the number of possible sequences of action-observation pairs grows exponentially in the number of steps considered.

For this reason, it is plausible to preprocess sensory data, such as video, with the goal of producing a more abstract state description. This can be achieved using techniques of machine learning to map low-level, high-dimensional input to high-level, low-dimensional representations via supervised- or unsupervised learning [2,9]. For time-series data it useful to not only map the current observations, but to also take into account previous input. Such prediction of a current hidden variable using all past observations is called *filtering*. Some techniques to perform this task are Hidden Markov Models, Kalman-filters, or particle filters [9, Section 6.2]. If a Bayesian approach to filtering is taken, then the result is a probability distribution over the current state of the abstract state variables (belief state), e.g., a distribution over user dispositions. Although this abstraction appears to be omnipresent in practical fields, such as robotics [13], it is unclear what it truly means to attach a POMDP planner to the output of a Bayesian filtering stage. In the sequel we will discuss the challenges of this approach and sketch some solution options.

2 Problem Statement

Interaction of the technical system with the environment at a physical level is captured by a POMDP $P_E = (S_E, A_E, T_E, O_E, Z_E, r_E)$—which exists, but is unknown. Here, S_E is the set of physical world states, A_E is the set of actions available to the technical system, and $T(s, a, s') = P(s'|s, a)$ denotes the world transition dynamics when the system executes action $a \in A_E$ in world state $s \in S_E$. The resulting world state s' is not visible to the system, it can only see an observation $o \in O_E$ with probability $Z(s', a, o) = P(o|a, s')$ determined by its sensors. The system's goals are given in terms of rewards $r_E(s)$ for being in s. P_E can be accessed by simulation (by running the system in real), or its parameters can be elicited through experimentation (not observation); both is expensive. The observations of the system are processed by a cascade of filters that extract estimates of abstract state features denoted as S_A. The result of this process at time t is a current belief $P^t(S_A^t \mid Z^{0:t})$. Characterization of the filtering process is possible either through simulation, or by quality estimates of the used machine learning algorithms (confusion tables, error rates). The described architecture is depicted in Figure 1.

Our goal is to construct a controller that takes the output of the filter as its input. For this sake, we assume that a useful abstraction T_A of the transition process T_E of P_E can be elicited from a human expert. We also assume that an abstraction $r_A : S_A \to \mathbb{R}$ can be elicited.

To achieve this, we want to either formulate an MDP or POMDP problem P_A over the abstract state space S_A, and describe how it fits into the sketched architecture. A policy for P_A should yield good expected rewards when used as controller for P_E as described in Figure 1.

3 Solution Approaches

There are two major possibilities for representing POMDP polices: as a mapping from histories to actions, i.e., by considering the equivalent history-based MDP of a POMDP [15], or as a mapping from belief states to actions, i.e., by considering the equivalent belief MDP [16].

3.1 History-based Control

Histories are sequences of action-observation pairs corresponding to cycles of executed actions and received observations. Therefore, using a history-based policy necessitates the definition of observations on the abstract level, which introduces apparently arbitrary choices: while it is simple to see what observations are on a primitive level, the case is less clear with abstract observations. It is somewhat reasonable to assume that abstract observation variables O_A and corresponding observation probabilities can be elicited from a human expert. The bigger issue, however, is the integration between the sensor filter cascade and the policy: in each time step, the filter cascade produces a probability distribution over the abstract state space given by S_A. The history-based policy, on the other hand, requires a history, which is a discrete object. What is therefore required in this setting is a method for finding the most likely history for a given distribution over the variables in S_A.

3.2 Belief-MDP-based Control

When the policy of the controller is represented as a mapping from belief states to actions, policy execution is straight-forward: the distribution over the variables in S_A *is* a belief state, and the policy can be used directly. This approach is taken by, e.g., Hoey et al. [8] and results in the filtering task being fully separated from the planning task.

There are two possibilities for constructing a belief MDP on the abstract level. The first is specifying an ordinary POMDP as for the history-based approach, and converting it into its belief MDP. This again requires specifying an abstract observation model over variables O_A. The second possibility is modeling transitions between system beliefs directly. This makes the definition of observations on the abstract level obsolete but requires discretization of the set of belief states, since there are uncountably many belief states. E.g., one can model "state" variables that represent qualitative probability estimates of real state variables of interest [3]. This, in turn, can distort optimal policies.

3.3 Related Work

Although the scenario described above deals with abstractions in the context of POMDPs, it seems that existing approaches to POMDP abstraction are applicable to a limited extent only. One category of approaches employ action abstraction. These approaches do not plan on the abstract level but rather use hierarchical action knowledge to ease planning on the primitive level [10,19,18,11,17]. In particular, all mentioned approaches require a model of the primitive level. In most cases, this means a fully declarative model, except for the MCTS approach of Müller et al. [10], where a generative model suffices. Some authors also consider learning an action hierarchy [4], which also requires access to the primitive model. In any case, generating a policy with these approaches can become prohibitively expensive in our setting where the primitive model corresponds to the true environment.

A further line of research aims at abstracting the observation space of a POMDP [20,1,7]. This seems important in our scenario as well, yet the existing approaches also require access to the primitive POMDP model. A further complication is that the mentioned observation abstraction approaches do not deal with factored observations spaces, so that structure that is certainly present in our multi-sensor environment cannot be leveraged.

Closest in spirit to the setting we deal with is an approach for a multi-modal service robot [13]. Here, a so-called *filterPOMDP* is constructed, which corresponds to what we call Belief-MDP-based control, i.e., a pre-specified POMDP is used for planning, while separate filters are used for maintaining a probability distribution over the world state during execution.

4 Conclusion

We proposed an approach that allows using POMDPs for decision making in *Companion* systems without resorting to modeling the entire *Companion* system as a POMDP. We discussed challenges to overcome for applying the approach as well as options for solving them.

References

1. Atrash, A., Pineau, J.: Efficient planning and tracking in pomdps with large observation spaces. In: AAAI-06 Workshop on Empirical and Statistical Approaches for Spoken Dialogue Systems (2006)
2. Bishop, C.M.: Pattern recognition and machine learning. springer (2006)
3. Boger, J., Hoey, J., Poupart, P., Boutilier, C., Fernie, G., Mihailidis, A.: A planning system based on markov decision processes to guide people with dementia through activities of daily living. Information Technology in Biomedicine, IEEE Transactions on 10(2), 323–333 (2006)
4. Charlin, L., Poupart, P., Shioda, R.: Automated hierarchy discovery for planning in partially observable environments. Advances in Neural Information Processing Systems 19, 225–232 (2007)
5. Geier, T., Reuter, S., Dietmayer, K., Biundo, S.: Goal-based person tracking using a first-order probabilistic model. In: Proceedings of the Ninth UAI Bayesian Modeling Applications Workshop (UAI-AW 2012) (8 2012), `https://www.uni-ulm.de/fileadmin/website_uni_ulm/iui.inst.090/Publikationen/2012/Geier12TrackingGoals.pdf`
6. Glodek, M., Honold, F., Geier, T., Krell, G., Nothdurft, F., Reuter, S., Schüssel, F., Hörnle, T., Dietmayer, K., Minker, W., et al.: Fusion paradigms in cognitive technical systems for human–computer interaction. Neurocomputing 161, 17–37 (2015)
7. Hoey, J., Poupart, P.: Solving pomdps with continuous or large discrete observation spaces. In: IJCAI. pp. 1332–1338 (2005)
8. Hoey, J., Poupart, P., von Bertoldi, A., Craig, T., Boutilier, C., Mihailidis, A.: Automated handwashing assistance for persons with dementia using video and a partially observable Markov decision process. Computer Vision and Image Understanding 114(5), 503–519 (2010)
9. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. MIT Press (2009)
10. Müller, F., Späth, C., Geier, T., Biundo, S.: Exploiting expert knowledge in factored POMDPs. In: ECAI (2012)
11. Pineau, J., Gordon, G., Thrun, S.: Policy-contingent abstraction for robust robot control. In: Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence. pp. 477–484. Morgan Kaufmann Publishers Inc. (2002)
12. Pineau, J., Gordon, G., Thrun, S.: Anytime point-based approximations for large pomdps. Journal of Artificial Intelligence Research pp. 335–380 (2006)
13. Schmidt-Rohr, S.R., Knoop, S., Lösch, M., Dillmann, R.: Bridging the gap of abstraction for probabilistic decision making on a multi-modal service robot. In: Robotics: Science and Systems (2008)
14. Shani, G., Pineau, J., Kaplow, R.: A survey of point-based POMDP solvers. Autonomous Agents and Multi-Agent Systems 27(1), 1–51 (2013)
15. Silver, D., Veness, J.: Monte-carlo planning in large POMDPs. In: NIPS. pp. 2164–2172 (2010)
16. Sondik, E.J.: The optimal control of partially observable markov processes over the infinite horizon: Discounted costs. Operations Research 26(2), 282–304 (1978)
17. Theocharous, G., Mahadevan, S.: Approximate planning with hierarchical partially observable markov decision process models for robot navigation. In: Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on. pp. 1347–1352 (2002)

18. Theocharous, G., Murphy, K., Kaelbling, L.P.: Representing hierarchical pomdps as dbns for multi-scale robot localization. In: Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA'04). pp. 1045–1051 (2004)
19. Toussaint, M., Charlin, L., Poupart, P.: Hierarchical pomdp controller optimization by likelihood maximization. In: UAI. vol. 24, pp. 562–570 (2008)
20. Wolfe, A.P.: Paying attention to what matters: observation abstraction in partially observable environments. Ph.D. thesis, University of Massachusetts Amherst (2010)
21. Young, S., Gasic, M., Thomson, B., Williams, J.D.: POMDP-based statistical spoken dialog systems: A review. Proceedings of the IEEE 101(5), 1160–1179 (2013)