Project Thesis
PRO-XXX

Mechanik und Meerestechnik

Partially Observable Markov Decision Processes for Planning in Uncertain Environments

by
Lasse Peters

Supervisors at TUHH	**Supervisors at UC Berkeley**
Prof. Dr.-Ing. Robert Seifried	Prof. Claire J. Tomlin
Daniel Duecker, M.Sc.	Zachary Sunberg, PhD.

Hamburg University of Technology
Institute of Mechanics and Ocean Engineering
Prof. Dr.-Ing. R. Seifried

Hamburg, May 2019

Contents

List of Abbreviations

MDP Markov Decision Process. 4–6

POMDP Partially Observable Markov Decision Process. 2–4, 6, 7, 11

Chapter 1

Introduction

Many decision making problems are subject to inherent uncertainty. Examples of such problems range from aircraft collision avoidance and robotic navigation tasks to to applications in health care and medical treatment [KochenderferHollandChryssanthacopoulos12, SchaeferEtAl05]. While humans have developed good intuition for decision making problems present in their day to day lives, many of the same tasks – like planning in autonomous driving – pose difficult problems for robotic agents [LevinsonEtAl11].

cite

Actively considering uncertainty in planning promises to improve robustness, safety and performance of the system. In conventional control theory a typical approach is to model uncertainty in terms of a bounded disturbance, robustifying the controller through reasoning over worst case disturbance sequences. However, modelling the disturbance as an adversarial player is often impractical, since long tail distributions may cause the controller to come up with overly conservative, thus poorly performing plans.

cite

Therefore, a tremendous amount of research has focused on incorporating uncertainty in decision making through probabilistic models, rather than adversarial game type approaches [RoyEtAl99, AmatoEtAl15, FisacEtAl18, ChoudhuryKochenderfer19].

One of the most general framework for modeling uncertainty in sequential decision making in a probabilistic fashion is provided by the Partially Observable Markov Decision Process (POMDP). Formulating and solving a planning problem as a POMDP allows the planner to reason over both, state and outcome uncertainty. Also, by taking into account future observations, solving a POMDP gives rise to behavior through computation that actively performs information gathering.

While POMDPs provide a comprehensive way of taking into account uncertainty in the planning procedure, finding the optimal solution to these problems is in practice often intractable since in worst case it can not be done in polynomial time [PapadimitriouTsitsiklis87]. Therefore, in robotics applications, characterized by limited compute and real time constraints, POMDP solution methods are traditionally avoided. Instead, simplifications are made that neglect the partial observability or make other assumptions about the problem structure [SadighEtAl16, FisacEtAl18].

still not happy with this paragraph

This work aims to provide insight into the use of POMDPs for modelling and solving planning problems in robotics. This results in robust policies for optimized interaction with uncertain environments. We show how this methodology provides solutions to problems where uncertainty otherwise, either keeps engineers from solving them in a principled manner or forces them to make simplifications that compromise performance and robustness.

We look at two application domains where uncertainty affects decision making in different ways. First, we focus on simultaneous localization and planning, a problem characterized by inherent uncertainty in the physical state of the robot. Subsequently, we examine motion planning in a shared space with a human agent where uncertainty forces the robot to reason over the latent human behavior model. We compare the performance of our provided solution to a domain specific approximation provided by [FisacEtAl18] and discuss the mechanisms through which POMDPs improve the agent performance in this domain.

- we discuss the mechanisms which make considering uncertainty important
- . . . and discuss the differences between the two problems.

Outline

write, when structure has converged

Theory Theoretical properties and background of POMDPs. Solution methods: POMCPOW, DESPOT.

Experiments Applications of POMDPs to two problem domains: Simultaneous localization an planning, Motion planning with latent human intentions.

Summary The summary of this work.

Chapter 2

Theory

This work showcases the use of POMDPs to model and solve decision making problems in robotics with inherent uncertainty. In order to provide a baseline for further discussions of specific application domains (chapter 3) we first introduce some of the throughout this work. It should be noted that in the interest of conciseness we focus this theoretical introduction on only the main tools use. Fundamental concepts of statistical inference and tools like *Monte Carlo Integration* are assumed to be known. For a thorough discussion of these underlying concept the reader may refer to [Kochenderfer15, Bertsekas05, ThrunBurgardFox05].

In the following, we first introduce the theoretical framework of POMDPs used for sequential decision making under uncertainty (section 2.1). This section discusses modelling assumptions, structure of solutions as well as theoretical properties of POMDPs. Thereafter, section 2.2 describes two state-of-the-art online solution methods for problems of this domain.

2.1 Sequential Decision Making Problems Under Uncertainty

The Partially Observable Markov Decision Process (POMDP) is a principled mathematical formalism capable of representing a broad range of sequential decision making problems under uncertainty. As the name suggests, this framework is a generalization of the more popular Markov Decision Process (MDP) to the partially observable case. Thus, before proceeding with the full complexity of a POMDP let us first examine it's fully observable version.

2.1.1 MDP

MDPs are sequential decision making problems in which an agent takes *actions* a that affect the *state* s of the environment and receives *rewards* r based the state-transition and the action taken [Kochenderfer15, Bertsekas05]. The state evolves according to a stochastic transition model $\mathcal{T}$ that and obeys the Markov property. That is, future states are independent of past states given the current state and action. By this means, MDPs allow to model outcome uncertainty. As common in literature, we denote quantities at time time t with an according subscript. When examining only a single step in a context where time does not explicitly matter, we may also refer to states before and after the transition as s and s' rather than s_t and s_{t+1}.

Formally, an MDP is fully characterized by the following quantities:

State Space $\mathcal{S}$. The set of all possible states.

Action Space $\mathcal{A}$. The set of all possible actions.

Transition Model $\mathcal{T}$. A model to represent the likelihood of each transition. This model provides $\mathcal{T}(s' \mid s, a)$, the probability of state s' given that previously the environment was in state s and the agent took action a.

Reward Function $\mathcal{R} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$. A deterministic mapping that assigns a real-valued reward r to each transition (s, a, s') with finite transition probability, $\mathcal{T}(s' \mid s, a) > 0$.

Discount Factor γ. A scalar that governs how rewards are discounted in the future.

The objective of the agent in the MDP is to maximize the expected cumulative rewards. Formally, this translates to finding a *policy* $\pi : \mathcal{S} \to \mathcal{A}$, that maps each encountered s to an action $a = \pi(s)$, such that following this policy maximizes the objective,

$$J(\pi) = E\left[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, \pi(s_t), s_{t+1})\right]. \tag{2.1}$$

An optimal solution to an MDP can always be formulated as a deterministic Markov policy, even though the reward that a policy achieves may be randomized through $\mathcal{T}$. Consequently, there always exists a maximizer π^* of eq. (2.1), where π^* assigns a single action to every state. Also, since the state obeys the Markov property, no additional information beyond the current state at every time is necessary to maximize J [Altman99].

2.1.2 Partially Observable MDP

A Partially Observable Markov Decision Process (POMDP) is an MDP where the agent cannot directly observe the state, but instead at every time step t receives an *observation* o_t, emission of the latent state s_t and action a_t. By this means, a POMDP is able to encode state uncertainty in addition to the outcome uncertainty present in the underlying MDP.

Having introduced the properties of an MDP in the previous section, a generalization of this formalism to the partially observable case is obtained by augmenting the 5-tuple $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma)$ describing the underlying MDP with the following quantities:

Observation Space $\mathcal{O}$. The set of all possible observations.

Observation Model $\mathcal{Z}$. A model to represent the likelihood of each observation o given the current state of the environment and the action taken at that time. Formally, this model provides the conditional probability $\mathcal{Z}(o \mid s, a)$ for each $(o, s, a) \in (\mathcal{O} \times \mathcal{S} \times \mathcal{A})$.

The dynamic decision network representing the information structure for a finite time horizon of a POMDP is depicted in fig. 2.1. As evident from this figure, the decision of the agent at time t is informed by the sequence of all previous actions and observations as well as some prior knowledge, b_0. In contrast to the fully observable case where the policy is only a function of quantities at the current time, the policy in a POMDP maps each possible *history*, $h_t = (b_0, a_0, o_1, a_1, \ldots, a_{t-1}, o_t)$ to an action, a_t.

@zach is information structure the right word here? Or is this only valid for game theory?

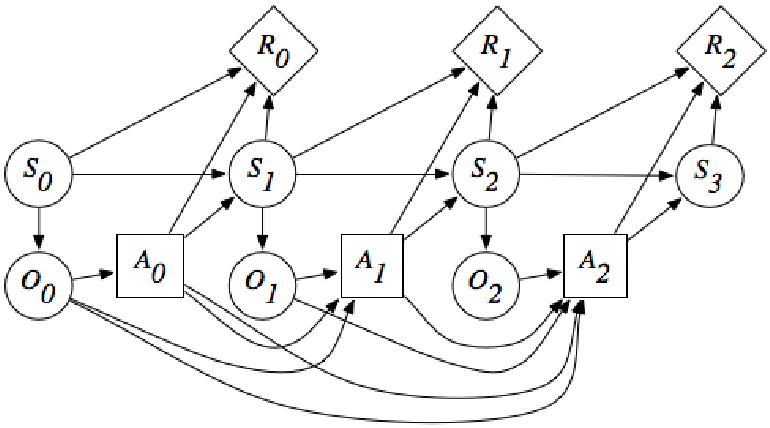

Figure 2.1: The POMDP as a dynamic decision network.

Replace with vector graphics and add initial information $b0$

In cases where this history can be compressed to a probability for every state at time t, we will refer to this distribution as the *belief*, $b_t(s)$. This belief is a sufficient statistic for optimal decision making. Consequently, there exists a policy $\pi^*(b_t)$ only dependent on the belief at the current time step, such that choosing $a_t = \pi^*(b_t)$ maximizes eq. (2.1) subject to the constraints on the information pattern imposed by the POMDP, [KaelblingLittmanCassandra98, Kochenderfer15]. Loosely speaking, b_t preserves all relevant information contained in the action-observation-history, h_t necessary to compute the optimal action. The belief is maintained by recursively performing a Bayesian update,

$$b'(s') = \frac{\int_{s\in\mathcal{S}} \mathcal{Z}(o \mid s, a, s')\mathcal{T}(s' \mid s, a)b(s)\,ds}{\int_{s'\in\mathcal{S}}\int_{s\in\mathcal{S}} \mathcal{Z}(o \mid s, a, s')\mathcal{T}(s' \mid s, a)b(s)\,ds\,ds'}. \tag{2.2}$$

with incoming observations, o. In the case of a discrete state space, $\mathcal{S}$ integrals a to be replaced with sums. In cases where this update rule is too complex to be evaluated analytically, Monte Carlo integration may be used for approximate inference [Kochenderfer15, ThrunBurgardFox05].

Solutions to a POMDP are Markov policies on belief states. That is, the optimal policy $\pi^*(b(s)) : \mathcal{B} \to \mathcal{A}$ selects action $a = \pi(b(s))$ for each belief in the *belief space*, $\mathcal{B}$ under the objective of maximizing eq. (2.1). Alternatively, when thinking of this procedure in terms of action-observation-histories, the policy may be envisioned as a conditional plan reasoning over the optimal action to take for every possible sequence of actions and observations starting from the root belief, b_0. This perspective reveals that the search space of possible policies rapidly grows with increasing size of $\mathcal{A}$ and $\mathcal{O}$. In fact, it has been shown that even in the case of finite-horizon problems POMDPs remain PSPACE-complete. Hence, it is reasonable to assume to that no efficient general algorithm for solving large problems of this class can be found [PapadimitriouTsitsiklis87]. However, recent research has made significant progress on approximate solution methods, some of which we will use in this work [SilverVeness10, SomaniEtAl13, SunbergKochenderfer18].

visualizatio of conditional plan?

@zach: should this already contain some of the theoretical advantages that one can expect?

2.2 Online POMDP Solvers

The solvers described here were chosen as they are state of the art and show good performance over a range of problems. Point to Zach's paper for solver performance comparison and DESPOT paper.

2.2.1 POMCPOW

write:

- explain the basic idea of POMCPOW as Monte Carlo with DPW.
- is extension of POMCP with weighted particle beliefs.

Missing figure

Pseudo Code with formal description in text.

Missing figure

graphical model of POMCP-Tree with weighted scenarios.

2.2.2 DESPOT

write:

- formal, high-level idea of a DESPOT (idea of scenarios etc.)
- the search algorithm on a DESPOT with bounds
- explain the difference to POMCPOW (or Monte Carlo methods in general)
- point to literature for convergence guarantees

Missing figure

DESPOT tree visualization

Missing figure

DESPOT algorithm Pseudo Code

Chapter 3

Experiments

3.1 Tools and Software Framework

Some insight into which tools are used and why:

- briefly point to speed comparison results (own application specific comparison + official charts)
- prototyping speed C++ vs Julia (Python already out of the game: too slow, see first point)
- Julia: good, general framework provided: `POMDPs.jl`, an interface for defining, solving, and simulating discrete and continuous, fully and partially observable Markov decision processes. [EgorovEtAl17] There also exists a C++ framework, but this very verbose and not suitable for rapid prototyping (`http://bigbird.comp.nus.edu.sg/pmwiki/farm/appl/`)
- conclusion: Julia [BezansonEtAl17]

3.2 Simultaneous Localization and Planning

3.3 Motion Planning with Latent Human Intentions

3.4 Comparison

This section will mainly contain the ”lessons learned” + a comparison between the problems, explaining the mechanisms which lead to the improvement in each case.

- POMDPs provide an elegant way of closing the loop between prediction (and perception models) and planning.
- This provides benefits through various mechanisms:
- Simultaneous localization and planning: Using a POMDP leads to active information gathering and provides a principled, optimization based approach to solving this problem (even under massive uncertainty). Solving it without such procedure is not straightforward, since planning on expectation with these highly multi-modal belief topologies is impractical.
- Planning with latent human intentions: The POMDP based solution allows the robot to consider future observations, making the robots plans less conservative because the agent knows that uncertainty about the humans intentions will be reduced in the future. For this very problem structure, neglecting future observations provides a principled way of solving this problem. Through this approximation, the problem is simplified to a problem that is still NP-hard, but well studied. Good heuristics exist, such that search is well guided to find results withing reasonable planning horizons. However, sacrificing performance (in particular for models with unbounded uncertainty like constant velocity).
    - Future work: POMDP solution method is safer than planning with probabilistic obstacles while reaching the goal in fewer number of steps. However, probabilistic obstacles can provide a-priory estimate of safety while for POMDPs this can only be shown empirically through large scale simulations. It would be nice to provide a safety assurance for this special type of POMDPs (Mixed Observability Markov Decision Process where the safety only depends on some unactuated decoupled part of the state.)

Chapter 4

Summary

Bibliography

[Altman99] Altman, E.: Constrained Markov decision processes. CRC Press, 1999.

[AmatoEtAl15] Amato, C.; Konidaris, G.; Cruz, G.; Maynor, C.A.; How, J.P.; Kaelbling, L.P.: Planning for decentralized control of multiple robots under uncertainty. In 2015 IEEE International Conference on Robotics and Automation (ICRA), pp. 1241–1248, IEEE, 2015.

[Bertsekas05] Bertsekas, D.: Dynamic Programming and Optimal Control. Athena Scientific, 2005.

[BezansonEtAl17] Bezanson, J.; Edelman, A.; Karpinski, S.; Shah, V.B.: Julia: A fresh approach to numerical computing. SIAM review, Vol. 59, No. 1, pp. 65–98, 2017.

[ChoudhuryKochenderfer19] Choudhury, S.; Kochenderfer, M.J.: Dynamic real-time multimodal routing with hierarchical hybrid planning. CoRR, Vol. abs/1902.01560, 2019.

[EgorovEtAl17] Egorov, M.; Sunberg, Z.N.; Balaban, E.; Wheeler, T.A.; Gupta, J.K.; Kochenderfer, M.J.: Pomdps. jl: A framework for sequential decision making under uncertainty. The Journal of Machine Learning Research, Vol. 18, No. 1, pp. 831–835, 2017.

[FisacEtAl18] Fisac, J.F.; Bajcsy, A.; Herbert, S.L.; Fridovich-Keil, D.; Wang, S.; Tomlin, C.J.; Dragan, A.D.: Probabilistically safe robot planning with confidence-based human predictions. arXiv preprint arXiv:1806.00109, 2018.

[KaelblingLittmanCassandra98] Kaelbling, L.P.; Littman, M.L.; Cassandra, A.: Planning and acting in partially observable stochastic domains. Artificial Intelligence, Vol. 101, pp. 99–134, 1998.

[Kochenderfer15] Kochenderfer, M.: Decision Making Under Uncertainty: Theory and Application. MIT Lincoln Laboratory Series. MIT Press, 2015.

[KochenderferHollandChryssanthacopoulos12] Kochenderfer, M.J.; Holland, J.E.; Chryssanthacopoulos, J.P.: Next-generation airborne collision avoidance system. Tech. rep., Massachusetts Institute of Technology-Lincoln Laboratory Lexington United States, 2012.

[LevinsonEtAl11] Levinson, J.; Askeland, J.; Becker, J.; Dolson, J.; Held, D.; Kammel, S.; Kolter, J.Z.; Langer, D.; Pink, O.; Pratt, V.; et al.: Towards fully autonomous driving: Systems and algorithms. In 2011 IEEE Intelligent Vehicles Symposium (IV), pp. 163–168, IEEE, 2011.

[PapadimitriouTsitsiklis87] Papadimitriou, C.H.; Tsitsiklis, J.N.: The complexity of Markov decision processes. Mathematics of Operations Research, Vol. 12, No. 3, pp. 441–450, 1987.

[RoyEtAl99] Roy, N.; Burgard, W.; Fox, D.; Thrun, S.: Coastal navigation-mobile robot navigation with uncertainty in dynamic environments. In Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C), Vol. 1, pp. 35–40, IEEE, 1999.

[SadighEtAl16] Sadigh, D.; Sastry, S.S.; Seshia, S.A.; Dragan, A.: Information gathering actions over human internal state. In 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 66–73, IEEE, 2016.

[SchaeferEtAl05] Schaefer, A.J.; Bailey, M.D.; Shechter, S.M.; Roberts, M.S.: Modeling medical treatment using markov decision processes. In Operations research and health care, pp. 593–612. Springer, 2005.

[SilverVeness10] Silver, D.; Veness, J.: Monte-Carlo planning in large POMDPs. In Advances in Neural Information Processing Systems (NIPS), 2010.

[SomaniEtAl13] Somani, A.; Ye, N.; Hsu, D.; Lee, W.S.: DESPOT: Online POMDP planning with regularization. pp. 1772–1780, 2013.

[SunbergKochenderfer18] Sunberg, Z.N.; Kochenderfer, M.J.: Online algorithms for pomdps with continuous state, action, and observation spaces. In Twenty-Eighth International Conference on Automated Planning and Scheduling, 2018.

[ThrunBurgardFox05] Thrun, S.; Burgard, W.; Fox, D.: Probabilistic Robotics. MIT Press, 2005.

Appendix

A.1 Contents Archive

There is a folder **PRO_XXX_Peters/** in the archive. The main folder contains the entries

- **PRO_XXX_Peters.pdf**: the pdf-file of the thesis PRO-XXX.
- **Data/**: a folder with all the relevant data, programs, scripts and simulation environments.
- **Latex/**: a folder with the *.tex documents of the thesis PRO-XXX written in Latex and all figures (also in *.svg data format if available).
- **Presentation/**: a folder with the relevant data for the presentation including the presentation itself, figures and videos.

Erklärung

Ich, Lasse Peters (Student der Mechatronik an der Technischen Universität Hamburg, Matrikelnummer 21486931), versichere, dass ich die vorliegende Projektarbeit selbstständig verfasst und keine anderen als die angegebenen Hilfsmittel verwendet habe. Die Arbeit wurde in dieser oder ähnlicher Form noch keiner Prüfungskommission vorgelegt.

____________________	____________________
Unterschrift	Datum