# Abstraction-based Model Checking of POMDPs in Motion Planning*

Leonore Winterer[1], Sebastian Junges[2], Ralf Wimmer[1], Nils Jansen[4], Ufuk Topcu[3], Joost-Pieter Katoen[2], and Bernd Becker[1]

[1] BrainLinks-BrainTools Cluster of Excellence
Albert-Ludwigs-Universität Freiburg, Freiburg im Breisgau, Germany
{winterel | wimmer | becker}@informatik.uni-freiburg.de
[2] RWTH Aachen University, Aachen, Germany
{sebastian.junges | katoen}@informatik.rwth-aachen.de
[3] The University of Texas at Austin, Austin, Texas, USA
utopcu@utexas.edu
[4] Radboud University, Nijmegen, The Netherlands
n.jansen@science.ru.nl

**Abstract**

Partially observable Markov decision processes (POMDPs) are a natural model for many applications where one has to deal with incomplete knowledge and random phenomena, including, but not limited to, robotics and motion planning. However, many interesting properties of POMDPs are undecidable or otherwise very expensive to decide in terms of both runtime and memory usage. In our work, we develop abstraction-based methods that can deliver safe bounds and good approximations for certain classes of properties.

## 1 Challenge

In offline motion planning, we aim to find a *strategy* for an agent that ensures certain desired behavior, even in the presence of dynamical obstacles and uncertainties [2]. If random phenomena like uncertainty in the outcome of an action or in the movement of dynamic obstacles need to be taken into account, the natural model for such scenarios are *Markov decision processes* (MDPs). MDPs are are non-deterministic models which allow the agent to perform actions under full knowledge of the current state of the agent and the surrounding environment. In many applications, though, full knowledge cannot be assumed, and we have to deal with *partial observability* [3]. For such scenarios, MDPs are generalized to *partially observable Markov decision processes* (POMDPs). In a POMDP, the agent does not know the exact state of the environment, but only an observation that can be shared between multiple states. Additional information about the likelihood of being in a certain state can be gained by tracking the observations over time. This likelihood is called the belief state. Using an update function mapping a belief state and an action as well as the newly obtained observation to a new belief state, one can construct a (typically infinite) MDP, commonly known as the *belief MDP*.

While model checking and strategy synthesis for MDPs are, in general, well-manageable problems, POMDPs are much harder to handle and, due to the potentially infinite belief space, many problems are actually undecidable [1]. Our aim is to apply different *abstraction and abstraction refinement* techniques to POMDPs in order to get good and safe approximative results for different types of properties.

*This work was (partly) supported by BrainLinks-BrainTools, Cluster of Excellence funded by the German Research Foundation (DFG, grant number EXC 1086) and by the awards ONR # N000141612051, NASA # NNX17AD04G and DARPA # W911NF-16-1-0001.

## 2 Approach

As a case study, we work with a scenario featuring a controllable agent. Within a certain area, the agent needs to traverse a room while avoiding both static obstacles and randomly moving opponents. Only the positions of the opponents cannot always be observed. The area is modeled as a grid, the static obstacles as grid cells that may not be entered. Our detailed assumption for this scenario is that the agent always knows its own position, but the positions of an opponent is only known if its distance from the agent is below a given threshold and if the opponent is not hidden behind a static obstacle. We assume that the opponents move probabilistically. This directly leads to a POMDP model for our case study. For simplification purposes, we are only dealing with one opponent right now, although the same approach would work with an arbitrary number of opponents as well.

The goal is to find a strategy which maximizes the probability to navigate through the grid from an initial to a target location without collision. For a grid size of $n \times n$ cells and one opponent, the number of states in the POMDP is in $O(n^4)$, i. e., the state space grows rapidly with increasing grid size. In order to handle non-trivial grids, we propose an approach using *game-based abstraction* [4].

Intuitively, we lump together all states that induce the same observation; so we can still distinguish between all states in which the opponent's position is known, but states in which the position is not known are merged into one *far away* state [6]. In order to get a safe approximation of the possible behavior of the opponent, for all of these lumped states we add a non-deterministic choice over the potential positions of the opponent. We formalize this as a *2-player probabilistic game* [4], in which one player controls the actions of the agent, and the other player controls the non-determinism added by the abstraction. This allows both players to optimize according to different goals. The abstraction player can create a worst-case scenario to *over-approximate* the realistic behavior, thus ensuring that the obtained bounds are safe and the resulting strategy cannot perform worse when mapped back to the original scenario.

A comparison with the state-of-the-art POMDP model checker PRISM-pomdp [5] indicates that we can handle grids that are considerably larger than what PRISM-pomdp can handle, while still getting schedulers that induce values which are close to the optimum. Table 1 shows a few of our results for verifying a reach-avoid property on a grid without obstacles. As one can see, the abstraction approach is faster by orders of magnitude than solving the POMDP directly, and the game model also is much smaller for large grids while still getting very good approximations for the actual probabilities. The strategies induce even better values when they are mapped back to the original POMDP. Note that this lifting of strategies can currently only be performed for small benchmarks due to technical limitations of the PRISM tool we use.

While being sound, our approach is still targeting an undecidable problem and as such not complete in the sense that in general no strategy with maximum probability for success can be deduced. In particular for cases with few paths to the goal location, the gap between the obtained bounds and the actual maximum can become large. For those cases, we define a scheme to refine the abstraction, which leads to larger games and accordingly longer computation times, but also to better results. In Table 2 we have demonstrated a first, very coarse step of this refinement. We use a benchmark representing a long, narrow tunnel, in which the agent has to pass the opponent once, but, due to the nature of the abstraction, can actually run into him several times. With longer tunnels, the probability to safely arrive in a goal state diminishes. Adding a refinement which remembers the last known position of the opponent and thus restricting the non-deterministic movement keeps the probability steady for arbitrary tunnel length.

Table 1: Comparing the POMDP solution using PRISM-pomdp with the solution of the PG abstraction using PRISM-games on different sized grids without obstacles for a reach-avoid property. The MDP result is the hard upper limit for the probability. All times are given in seconds.

| | POMDP solution | | | | PG solution | | | | Lifting | MDP |
|---|---|---|---|---|---|---|---|---|---|---|
| Grid size | States | Result | Model Time | Sol. Time | States | Result | Model Time | Sol. Time | Result | |
| $3 \times 3$ | 299 | 0.8323 | 0.063 | 0.26 | 400 | 0.8323 | 0.142 | 0.036 | 0.8323 | 0.8323 |
| $4 \times 4$ | 983 | 0.9556 | 0.099 | 1.81 | 1348 | 0.9556 | 0.353 | 0.080 | 0.9556 | 0.9556 |
| $5 \times 5$ | 2835 | 0.9882 | 0.144 | 175.94 | 6124 | 0.9740 | 0.188 | 0.649 | 0.9825 | 0.9882 |
| $5 \times 6$ | 4390 | 0.9945 | 0.228 | 4215.056 | 8058 | 0.9785 | 0.242 | 0.518 | 0.9893 | 0.9945 |
| $6 \times 6$ | 6705 | ?? | 0.377 | – MO – | 10592 | 0.9830 | 0.322 | 1.872 | 0.9933 | 0.9970 |
| $8 \times 8$ | 24893 | ?? | 1.735 | – MO – | 23128 | 0.9897 | 0.527 | 6.349 | 0.9992 | 0.9998 |
| $10 \times 10$ | 66297 | ?? | 9.086 | – MO – | 40464 | 0.9914 | 0.904 | 6.882 | 0.9999 | 0.9999 |
| $20 \times 20$ | – Time out during model construction – | | | | 199144 | 0.9921 | 8.580 | 122.835 | 0.9999 | 0.9999 |
| $30 \times 30$ | – Time out during model construction – | | | | 477824 | 0.9921 | 41.766 | 303.250 | ?? | 0.9999 |
| $40 \times 40$ | – Time out during model construction – | | | | 876504 | 0.9921 | 125.737 | 1480.907 | ?? | 0.9999 |
| $50 \times 50$ | – Time out during model construction – | | | | 1395184 | 0.9921 | 280.079 | 3129.577 | ?? | – MO – |

Table 2: Results of the game-based abstraction for benchmarks describing a long, narrow corridor, with and without rudimentary refinement. All times are given in seconds.

| | Grid | States | Choices | Trans. | Result | Time |
|---|---|---|---|---|---|---|
| no ref. | $4 \times 40$ | 50880 | 93734 | 170974 | 0.9228 | 19 |
| | $4 \times 60$ | 77560 | 143254 | 261534 | 0.8923 | 67 |
| | $4 \times 80$ | 104240 | 192774 | 352094 | 0.8628 | 115 |
| | $4 \times 100$ | 130920 | 242294 | 442654 | 0.8343 | 164 |
| with ref. | $4 \times 40$ | 55300 | 120848 | 198088 | 0.9799 | 63 |
| | $4 \times 60$ | 83820 | 182368 | 300648 | 0.9799 | 220 |
| | $4 \times 80$ | 112340 | 243888 | 403208 | 0.9799 | 265 |
| | $4 \times 100$ | 140860 | 305408 | 505768 | 0.9799 | 746 |

# 3 Conclusion and Further Research

Game-based abstraction has turned out to be a viable way to obtain safe – and in many cases rather tight – bounds on maximal reach-avoid probabilities in our motion planning scenario. It scales much better than the standard analysis algorithms for POMDPs, which are typically based on discretizing the corresponding belief MDP [5].

Currently, our approach is limited to a restricted class of POMDPs based on an agent and its opponents moving inside a connected graph, but we have reason to believe that game-based abstraction is also beneficial for arbitrary POMDPs. We are also going to investigate other classes of properties involving costs and the long-term behavior of agents.

# References

[1] Krishnendu Chatterjee, Martin Chmelík, and Mathieu Tracol. "What is decidable about partially observable Markov decision processes with $\omega$-regular objectives". In: *J. Comput. Syst. Sci.* 82.5 (2016), pp. 878–911. DOI: 10.1016/j.jcss.2016.02.009.

[2] Ronald A. Howard. *Dynamic Programming and Markov Processes*. 1st edition. The MIT Press, 1960.

[3] Leslie Pack Kaelbling, Michael L. Littman, and Anthony R. Cassandra. "Planning and acting in partially observable stochastic domains". In: *Artif. Intell.* 101.1 (1998), pp. 99–134. DOI: 10.1016/S0004-3702(98)00023-X.

[4] Mark Kattenbelt and Michael Huth. "Verification and Refutation of Probabilistic Specifications via Games". In: *FSTTCS*. Vol. 4. LIPIcs. Schloss Dagstuhl, 2009, pp. 251–262. DOI: http://dx.doi.org/10.4230/LIPIcs.FSTTCS.2009.2323.

[5] G. Norman, D. Parker, and Xueyi Zou. "Verification and Control of Partially Observable Probabilistic Systems". In: *Real-Time Systems* 53.3 (2017), pp. 354–402. DOI: 10.1007/s11241-017-9269-4.

[6] Leonore Winterer et al. "Motion Planning under Partial Observability using Game-Based Abstraction". In: *IEEE Conf. on Decision and Control (CDC)*. Melbourne, Australia: IEEE, Dec. 2017.