Good policies for partially-observable Markov decision processes are hard to find *

Martin Mundhenk † Judy Goldsmith ‡ Christopher Lusena ‡

March 19,1997

Abstract

Optimal policy computation in finite-horizon Markov decision processes is a classical problem in optimization with lots of pratical applications. For stationary policies and *infinite* horizon it is known to be solvable in polynomial time by linear programming, whereas for *finite-horizon* it is a longstanding open problem. We consider this problem for a slightly generalized model, namely partially-observable Markov decision processes (POMDPs). We show that it is NP-complete and that – unless P = NP – the optimal policy cannot be polynomial time ε-approximated for any $\varepsilon < 1$. A similar result is shown for the average policy. The problem of whether the average performance is positive is shown to be PP-complete for stationary policies, and PL-complete for time-dependent policies, and of whether the median performance is positive is PP-complete for both types of policies. Furthermore, we systematically investigate the complexity of stationary, time- and history-dependent policy existence problems for POMDPs in different compressed representations.

1 Introduction

Consider the Robot Location Problem: a robot has a map of its environment, but has just regained power without knowing its location. It has sensors that give it possibly incomplete information about its actual location; it has a control program that issues commands ("roll forward 1 unit," and "rotate 90 degrees,") that have costs and pay-offs for each location, and succeed with known probabilities. The problems of interest are several: Does there exist a control program (i.e., a *policy*) that will get it where it's going (the *policy existence problem*)? Do most possible policies succeed (the *median policy problem*)? What is the optimal policy? Is there an efficient algorithm that gives a good approximation of the optimal policy? We analyze the computational complexity of all of these problems, with constraints on resources such as the amount of memory the controller has, the completeness of the sensor information, and the problem representation.

The robot's world can be modeled by a partially-observable Markov Decision Process. Markov Decision Processes model controllable stochastic processes: there is a set of *states;* a controller chooses among some number of *actions,* each of which has an associated probability matrix of state transitions; associated with each state and action pair is a *reward.* In the case of *fully-observable* processes (MDPs), the controller can observe the current state of the system. With partially-observable Markov Decision Processes (POMDPs), the controller may not get complete information about the system's current state; i.e. the state space is partitioned into *observations*.

*Correspondence address: Martin Mundhenk, Universität Trier, FB 4 - Informatik, D-54286 Trier (Germany), mundhenk@ti.uni-trier.de

†Supported in part by the Office of the Vice Chancellor for Research and Graduate Studies at the University of Kentucky, and by the Deutsche Forschungsgemeinschaft (DFG), grant Mu 1226/2-1.

‡Supported in part by NSF grant CCR-9315354.

Note that MDPs are special cases of POMDPs, where each observation consists of exactly one state. The basic goal for any POMDP is to find a strategy or *policy* for choosing actions that maximizes the total expected reward over some fixed time horizon. Policies may consider the current observation only (stationary), the observation and time (time-dependent), or the full history of observations including the current one (history-dependent). The problem of finding optimal policies for MDPs is well studied for finite and infinite horizon. Dynamic and linear programming techniques are well-known to efficiently find optimal policies for infinite and finite horizon MDPs. But there is one longstanding open problem, the complexity of computing the optimal stationary policy for an MDP with finite horizon, or even deciding whether a stationary policy with positive performance for finite horizon exists (called the *stationary policy existence problem*).

In this paper, we systematically investigate a variety of types of POMDPs:

- fully-observable and partially-observable,
- performance under stationary, time-dependent and history-dependent policies,
- rewards restricted to be nonnegative, or unrestricted, and
- straightforward and compressed representations.

Our main interest have the policy existence, average and median problems for POMDPs under stationary policies. We show the stationary policy existence problem for POMDPs to be NP-complete, and obtain the same complexity for the time-dependent problem. The average problem is shown to be PP-complete for stationary policies, and PL-complete for time-dependent policies, and the median problem is PP-complete for both types of policies.

We consider the problem of computing an optimal policy. For stationary policies for fully-observable MDPs, Tseng gives polynomial-time approximation algorithms [16]. For POMDPs, we show that this is not possible unless P = NP, improving results from Burago et al. [5].

We consider both straightforward and *compressed* representations of POMDPs. Our interest in compressed representations is motivated by work in AI/planning [4, 6, 10]. These representations arise when one can make use of structures in the state space to provide small descriptions of very large systems [4]. Whereas those systems are not tractable by classical methods, there is some hope – expressed in different algorithmic approaches – that many special cases of these structured POMDPs can be solved more efficiently. Boutillier, et al., in [4] conjecture that finding optimal policies for structured POMDPs is infeasible. We prove this conjecture by showing that in many cases the complexity of our decision problems increases exponentially if succinct descriptions [8, 18] for POMDPs are considered. E.g. policy existence problems for POMDPs with nonnegative rewards are NL-complete under straightforward descriptions. Using succinct descriptions, the completeness increases to PSPACE. We also consider a new intermediate notion of compressed representations and get intermediate complexity results, e.g. NP-completeness for the above example.

The complexity of partially-observable MDPs was firstly considered by Papadimitriou and Tsitsiklis [13, 14]. They considered POMDPs with nonpositive rewards and investigated the complexity of the decision problem: given a POMDP, does there exist a time-dependent or history-dependent policy with expected sum of rewards *equal to* 0? Note that this decision problem cannot be used in a binary search to determine the maximal performance of a POMDP. Our work extends theirs significantly, both in the problems considered, and in the scope of the results. Because of the subtle differences between the decision problems, our complexity results on similar problems differ greatly from the results in [14]. For example, our decision problem for POMDPs with nonnegative rewards under history-dependent policy is NL-complete. (Cf. Theorem 3.2, compare to PSPACE-completeness in [14, Theorem 6].)

The AI/planning community has considered the complexity of several versions of controlled *deterministic* processes [6, 7]. These problems are generally computationally simpler than those considered here. The closest results from that literature are by Littman, et al. [10, 9].

We give formal definitions for POMDPs in Section 2; the results are divided into four categories: policy existence problems in Sections 3, median and average problems in Section 4, non-approximation results in Section 5, and compressed represented POMDPs in Section 6.

2 Definitions and Preliminaries

For definitions of standard complexity classes, reductions, and results from complexity theory we refer to [12].

2.1 Markov Decision Processes

A partially-observable Markov decision process (POMDP) describes a controlled stochastic system by its states and the consequences of actions on the system. It is denoted as a tuple $M = (\mathcal{S}, s_0, \mathcal{A}, \mathcal{O}, t, o, r)$, where

- $\mathcal{S}$, $\mathcal{A}$ and $\mathcal{O}$ are finite sets of *states*, *actions* and *observations*,
- $s_0 \in \mathcal{S}$ is the *initial state*,
- $t : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the *state transition function*, where $t(s, a, s')$ is the probability to reach state s' from state s on action a (where $\Sigma_{s' \in \mathcal{S}} t(s, a, s') \in \{0, 1\}$ for $s \in \mathcal{S}, a \in \mathcal{A}$),
- $o : \mathcal{S} \to \mathcal{O}$ is the *observation function*, where $o(s)$ is the observation made in state s,
- $r : \mathcal{S} \times \mathcal{A} \to \mathbf{Z}$ is the *reward function*, where $r(s, a)$ is the reward gained by taking action a in state s.

We distinguish two special types of observability of a POMDP. If states and observations are identical, i.e. $\mathcal{O} = \mathcal{S}$ and o is the identity function (or a bijection), then the POMDP is called *fully-observable*. (We use *MDP* to denote fully-observable POMDPs.) It is called *unobservable*, if the set of observations contains only one element or the observation function is constant, i.e. in every state the same observation is made.[1] If the reward function maps to nonnegative integers, we say that the POMDP has *nonnegative rewards*.

2.2 Policies and Performances

A policy describes how to act depending on observations. We distinguish three types of policies.

- A *stationary policy* π_s (for M) is a function $\pi_s : \mathcal{O} \to \mathcal{A}$, mapping observations to actions.
- A *time-dependent policy* π_t is a function $\pi_t : \mathcal{O} \times \mathbf{N} \to \mathcal{A}$, mapping pairs $\langle$observation, time$\rangle$ to actions.
- A *history-dependent policy* π_h is a function $\pi_h : \mathcal{O}^* \to \mathcal{A}$, mapping finite sequences of observations to actions.

Let $M = (\mathcal{S}, s_0, \mathcal{A}, \mathcal{O}, t, o, r)$ be a POMDP. A *trajectory* θ *for* M is a finite sequence of states $\theta = \sigma_1, \sigma_2, \ldots, \sigma_m$ ($m \geq 0$, $\sigma_i \in \mathcal{S}$). The probability $\text{prob}_\pi(M, \theta)$ of a trajectory $\theta = \sigma_1, \sigma_2, \ldots, \sigma_m$ under policy π is

- $\text{prob}_\pi(M, \theta) = \Pi_{i=1}^{m-1} t(\sigma_i, \pi(o(\sigma_i)), \sigma_{i+1})$, if π is a stationary policy,
- $\text{prob}_\pi(M, \theta) = \Pi_{i=1}^{m-1} t(\sigma_i, \pi(o(\sigma_i), i), \sigma_{i+1})$, if π is a time-dependent policy.

[1] Making observations probabilistically does not add any power to POMDPs. Any probabilistically observable POMDP can be turned into one with deterministic observations with only a polynomial increase in its size.

The *reward* $\mathrm{rew}_\pi(M,\theta)$ of trajectory θ under π is the sum of its rewards, i.e. $\mathrm{rew}_\pi(M,\theta) = \Sigma_{i=1}^{m-1} r(\sigma_i, \pi(\cdot))$ dependent on the type of policy. The *performance of a policy* π for finite-horizon k with initial state σ is the expected sum of rewards received on the next k steps by following the policy π, i.e. $\mathrm{perf}(M,\sigma,k,\pi) = \sum_{\theta\in\Theta(\sigma,k)} \mathrm{prob}_\pi(M,\theta)\cdot \mathrm{rew}_\pi(M,\theta)$, where $\Theta(\sigma,k)$ is the set of all length k trajectories beginning with state σ. If σ is the initial state of M and $k=|\mathcal{S}|$, then we use $\mathrm{perf}(M,\pi) := \mathrm{perf}(M,\sigma_0,|\mathcal{S}|,\pi)$. The *$\alpha$-maximum* $\max_\alpha(M,k)$ of M for horizon k is M's maximal performance under a policy π of type α for horizon k when started in its initial state, i.e. $\mathrm{val}_\alpha(M,k) = \max_{\pi\in\Pi_\alpha^k} \mathrm{perf}(M,\sigma_0,k,\pi)$, where Π_α^k is the set of all α policies for horizon k. The *α-average* $\mathrm{ave}_\alpha(M,k)$ is the average performance of α policies for M with horizon k, i.e. $\mathrm{ave}_\alpha(M,k) = \frac{1}{|\Pi_\alpha^k|}\sum_{\pi\in\Pi_\alpha^k} \mathrm{perf}(M,\pi)$. If all the possible performances of α policies for M and horizon k are sorted in increasing order, the median value is denoted by $\mathrm{med}_\alpha(M,k)$.

We consider the following decision problems for POMDPs M with state space $\mathcal{S}$. The α policy existence problem asks whether $\max_\alpha(M,|\mathcal{S}|)>0$. The α policy average problem asks whether $\mathrm{ave}_\alpha(M,|\mathcal{S}|)>0$. The α policy median problem asks whether $\mathrm{med}_\alpha(M,|\mathcal{S}|)>0$. Whenever we mention explicitly a horizon k in the problem statement, we consider the respective problem with horizon k instead of $|\mathcal{S}|$.

For simplicity, we assume that the size of a POMDP is determined by the size $|\mathcal{S}|$ of its state space. We assume that there are no more actions than states, and that each state transition probability is given as binary fraction with $|\mathcal{S}|$ bits and each reward is an integer of $|\mathcal{S}|$ bits. This is no real restriction, since adding unreachable "dummy" states allows one to use more bits for transition probabilities and rewards. Also, it is straightforward to transform a POMDP M with non-integer rewards to M' with integer rewards such that $\mathrm{val}_\alpha(M,k) = c\cdot \mathrm{val}_\alpha(M',k)$ for some constant c.

3 Policy existence problems

For fully-observable MDPs with nonnegative rewards the existence problem is not very hard. Because negative rewards do not exist, the expected reward for a policy π is greater than 0 if and only if at least one trajectory $\sigma_0,\ldots,\sigma_m$ consistent with π exists for which the action chosen by π for σ_m yields a reward greater than 0. It is clear that each such π can be changed to a policy for which there exists a consistent trajectory with positive reward, where no state appears twice in that trajectory. The latter bounds the length of the trajectories to be considered by the number of the states of the MDP. Therefore, the policy existence problems can be shown to be equivalent to graph reachability problems, from which we can conclude NL-completeness.

Theorem 3.1 *The stationary, time-dependent and history-dependent policy existence problems for fully-observable MDPs with nonnegative rewards are* NL*-complete.*

For partially-observable MDPs the same argument works for time- or history-dependent policy existence problems. In order to create a consistent trajectory, consistent with any policy, at any step of the process the next action can be chosen without taking former observations and actions into account.

Theorem 3.2 *The time-dependent and history-dependent policy existence problems for POMDPs with nonnegative rewards are* NL*-complete.*

For stationary policies, the same action must be chosen whenever the same observation is made. Logspace is not sufficient to handle this problem. On the other hand, we can use observations to store information. This allows us to express NP-complete problems.

Theorem 3.3 *The stationary policy existence problem for POMDPs with nonnegative rewards is* NP*-complete.*

Proof Membership in NP is straightforward, because a policy can be guessed and evaluated in polynomial time. To show NP-hardness, we polynomial time reduce the NP-complete satisfiability problem 3SAT to it. Let $\phi(x_1, \ldots, x_n)$ be such a formula with variables $x_1, \ldots, x_n$ and clauses $C_1, \ldots, C_m$, where clause $C_j = (l_{v(1,j)} \vee l_{v(2,j)} \vee l_{v(3,j)})$ for $l_i \in \{x_i, \neg x_i\}$. We say that variable x_i *appears in* C_j *with signum* 0 *(resp.* 1*)* if $\neg x_i$ (resp. x_i) is a literal in C_j. W.l.o.g. we assume that every variable appears at most once in each clause. The idea is to construct an POMDP $M(\phi)$ having one state for each appearance of a variable in a clause. Each observation class consists of all appearances of one variable. Each action corresponds to an assignment of a value to a variable. The transition function is deterministic. The process starts with the first variable in the first clause. If the action chosen in a certain state satisfies the corresponding literal, the process proceeds to the first variable of the next clause, or with reward 1 to the final state, if all clauses were considered. If the action does not satisfy the literal, the process proceeds to the next variable of the clause, or with reward 0 to the final state. The partition of the state space into obsrvation classes guarantees that the same assignment is made for every appearance of the same variable. Therefore, the value of $M(\phi)$ equals 1 iff ϕ is satisfiable.

Formally, from ϕ, we construct a POMDP $M(\phi) = (\mathcal{S}, s_0, \mathcal{A}, \mathcal{O}, t, o, r)$ with

$$\mathcal{S} = \{(i,j) \mid 1 \le i \le n, 1 \le j \le m\} \cup \{F, T\}$$

$$s_0 = (v(1,1),1), \qquad \mathcal{A} = \{0,1\}, \qquad \mathcal{O} = \{x_1, \ldots, x_n, F, T\}$$

$$t(s,a,s') = \begin{cases} 1, & \text{if } s = (v(i,j),j), s' = (1, j+1), j < m, 1 \le i \le 3, \\ & \qquad \text{and } x_{v(i,j)} \text{ appears in } C_j \text{ with signum } a \\ 1, & \text{if } s = (v(i,m),m), s' = T, 1 \le i \le 3, \\ & \qquad \text{and } x_{v(i,m)} \text{ appears in } C_m \text{ with signum } a \\ 1, & \text{if } s = (v(i,j),j), s' = (v(i+1,j),j), 1 \le i < 3, \\ & \qquad \text{and } x_{v(i,j)} \text{ appears in } C_j \text{ with signum } 1-a \\ 1, & \text{if } s = (v(3,j),j), s' = F, \\ & \qquad \text{and } x_{v(3,j)} \text{ appears in } C_j \text{ with signum } 1-a \\ 1, & \text{if } s = s' = F \text{ or } s = s' = T \\ 0, & \text{otherwise} \end{cases}$$

$$r(s,a) = \begin{cases} 1, & \text{if } t(s,a,T) = 1, s \neq T \\ 0, & \text{otherwise} \end{cases}, \qquad o(s) = \begin{cases} x_i, & \text{if } s = (i,j) \\ T, & \text{if } s = T \\ F, & \text{if } s = F \end{cases}.$$

Note that all transitions in $M(\phi)$ are deterministic, and every trajectory has value 0 or 1. There is a correspondence between policies for $M(\phi)$ and assignments of values to the variables of ϕ, such that policies under which $M(\phi)$ has value 1 correspond to satisfying assignments for ϕ, and vice versa.

Claim 1 *If $\phi(b_1, \ldots, b_n)$ is true for an assignment $b_1 \cdots b_n \in \{0,1\}$, then there exists a trajectory of length $\le |\mathcal{S}|$ consistent to π with $\pi(x_i) = b_i$ for $M(\phi)$ with reward* 1.

Let i_j be the smallest i such that $x_{v(i,j)}$ appears in C_j with signum b_i, for $j = 1, 2, \ldots, m$. Since $\phi(b_1, \ldots, b_n)$ is true, such an i_j exists for every C_j. Then,

$$(v(1,1),1), \ldots, (v(i_1,1),1), (v(1,2),2), \ldots, (v(i_2,2),2), \cdots, (v(1,m),m), \ldots, (v(i_m,m),m), T$$

is a trajectory for π. Because $t((v(i_m,m),m), \pi(x_m), T) = 1$, this trajectory has reward 1. □

Claim 2 *If a trajectory for a policy π for $M(\phi)$ has reward 1, then $\phi(\pi(x_1), \ldots, \pi(x_n))$ is true.*

Let θ be a trajectory for π for $M(\phi)$ with reward 1. Then by the definition of $M(\phi)$, θ has the form

$$(v(1,1),1), \ldots, (v(i_1,1),1), \cdots, (v(1,m),m), \ldots, (v(i_m,m),m), T, \ldots, T$$

Therefore, $x_{v(i_j,j)}$ appears in C_j with signum $\pi(x_{i_j})$ for every $j = 1, 2, \ldots, m$. This means that every clause C_j contains a literal that is satisfied by the assignment $\pi(x_1) \cdots \pi(x_n)$, and therefore $\pi(x_1) \cdots \pi(x_n)$ is a satisfying assignment for ϕ. □

From the above claims follows that ϕ is satisfiable if and only if there exists a policy under which $M(\phi)$ has value greater than 0. By the description of the reduction it also follows that $M(\phi)$ is computable from ϕ in polynomial time, which completes the proof. ■

To decide the performance of a policy for a POMDP with positive and negative rewards, a full "tree" of trajectories has to be evaluated, instead of one trajectory, and the weighted average of their values computed.

The computation of optimal policies for fully-observable MDPs is a well-studied optimization problem. The maximal performance of any infinite-horizon stationary policy for a fully-observable Markov decision process can be solved by linear or dynamic programming techniques in polynomial time. These techniques also work for the finite horizon history- and time-dependent policy cases.

Theorem 3.4 *The time-dependent and history-dependent policy existence problems for fully-observable MDPs are* P*-complete.*

Because the proof is a straightforward modification of a proof in [14], we omit it here.

The exact complexity of the policy existence problem for stationary, fully-observable MDPs with finite horizon is not known. From [14] it follows that it is P-hard, and it is easily seen to be in NP. But completeness results are not known. For POMDPs with unrestricted rewards we obtain completeness results for all policy existence problems. Contrary to the results for time- and history-dependent policies, surprisingly, the complexity of the existence problem for stationary partially-observable MDPs does not depend on whether the rewards are nonnegative or unrestricted. Instead, POMDPs seem to obtain their expressive power by history-dependent policies.

Theorem 3.5 *The stationary and time-dependent policy existence problems for POMDPs are* NP*-complete.*

Proof That the existence problems for both types of policies are in NP, follows using the guess-and-check argument. A policy can be guessed and its value calculated in polynomial time.

To show NP-hardness, we have to consider both types of policies separately. For stationary policies, the reduction from 3SAT as in the proof of Theorem 3.3 works here too. For time-dependent policies, we sketch a proof similar to one in [14]. In the proof of Theorem 3.3, we constructed from a formula a POMDP which searches through the literals of every clause, one clause after another. Here, we search in parallel through every clause, independent of whether the variable appears in that clause. At the first step, a clause is chosen randomly. At step $i+1$, the assignment of variable x_i is determined. If a clause was satisfied by this assignment, it will gain reward 1, if not, the reward will be $-m$, where m is the number of clauses of the formula. Therefore, if all clauses are satisfied, the value of the POMDP is positive, otherwise negative. ■

Whereas for fully-observable MDPs the existence problems for time-dependent and history-dependent policies are equally hard to solve, for POMDPs the complexity is different.

Theorem 3.6 *The history-dependent policy existence problem for POMDPs is* PSPACE*-complete.*

The proof of Theorem 3.6 is a straightforward modification of the proof by Papadimitriou and Tsitsiklis [14, Theorem 6], where the rewards for reaching the satisfying resp. unsatisfying final state are changed appropriately.

4 Median and average performances

We now ask whether most policies for a POMDP have positive performance, and whether we can expect to get a positive performance if we pick a policy randomly. The first of these problems, called the policy median problem, is intuitively harder than the policy existence problem, because we need to count policies with positive performance. We show that it is complete for probabilistic polynomial time PP both for stationary and time-dependent policies. The second problem (the policy average problem) turns out to be harder than the policy existence problem for stationary policies, and less hard for time-dependent policies.

Theorem 4.1 *The stationary policy median problem for POMDPs is* PP*-complete.*

Proof We show that the problem is in PP. Given a POMDP M, guess a policy, calculate the value of M under that policy, and accept iff that value is greater than 0. Because every policy is guessed with same probability, this nondeterministic computation has more accepting than rejecting paths iff the median of M is greater than 0.

To show PP-completeness, we give a reduction from the PP-complete set MAJSAT. A Boolean formula ϕ is in MAJSAT iff more than half of all assignments satisfy ϕ. Given ϕ, we construct a POMDP $M(\phi)$ as in the proof of Theorem 3.3. Because of the bijection between stationary policies and assignments, we obtain that ϕ is in MAJSAT iff $M(\phi)$ has value 1 under more than half of all policies. Since $M(\phi)$ has value 0 or 1 under any policy, we obtain that the median of $M(\phi)$ is greater than 0 iff ϕ is in MAJSAT. ∎

Theorem 4.2 *The stationary policy average problem for POMDPs is* PP*-complete.*

Proof In order to prove PP-hardness, we apply the same technique as in the proof of Theorem 4.1, but we change the reward function of the POMDP $M(\phi)$ constructed from the formula ϕ in a way such that M has value 1 under every stationary policy corresponding to a satisfying assignment, and value -1 (instead of 0) under every other policy. Then the average of $M(\phi)$ is greater than 0 iff ϕ is in MAJSAT.

In [11] it is shown that for every POMDP M and policy π there exists a logspace polynomial-time nondeterministic computation such that the number of accepting paths minus the number of rejecting paths in this computation equals the value of M under π multiplied by some constant dependent only on M. To show that the average problem is in PP, instead of generating one computation path for each policy, we now generate such a computation tree for every policy. (Note that it is necessary to write down each policy, requiring polynomial rather than logarithmic space.) Then this whole computation has more accepting than rejecting paths iff the average value of all policies is greater than 0. ∎

Theorem 4.3 *The time-dependent policy median problem for POMDPs is* PP*-complete.*

Proof Containment in PP can be shown using the same idea as for the stationary case. To show PP-hardness, we use a similar argument as for the stationary case. We showed how to reduce 3SAT to the existence problem for time-dependent policies for POMDPs by giving a transformation from formulae ϕ to POMDPs $M(\phi)$ such that ϕ is satisfiable iff there exists a time-dependent policy under which $M(\phi)$ has performance greater than 0. Because we have a bijective correspondence between time-dependent policies and assignments such that $M(\phi)$ has positive value under a policy iff ϕ is satisfied by the correponding assignment, it follows that ϕ is in MAJSAT iff the median of $M(\phi)$ under time-dependent policy is positive. ∎

The time-dependent policy average problem for POMDPs has complexity even lower than the existence problem – we locate it in PL. The difference between time-dependent and stationary policies is that it is not necessary, in a trajectory for a time-dependent policy, to remember the previous history. Thus, the space requirements drop dramatically.

The class PL consists of sets decidable probabilistically in logarithmic space. In apparent contrast to P-complete sets, sets in PL can be decided using very fast parallel computations (see [3]).

Theorem 4.4 *The time-dependent policy average problem for POMDPs is* PL*-complete.*

Proof We start with showing that the problem is in PL. Let M be a POMDP. Imagine M as a graph with vertices labeled by states of M and edges labeled by actions between vertices corresponding to the state transition function of M. Every path $\sigma_1 \xrightarrow{a_1} \sigma_2 \xrightarrow{a_2} \cdots \xrightarrow{a_{n-1}} \sigma_n$ with $t(\sigma_i, a_i, \sigma_{i+1}) > 0$ $(i = 1, \ldots, n-1)$ is consistent with some time-dependent policy. Actually, every such path is consistent with exactly the same number of time-dependent policies. Because every policy has equal probability, the average performance can be computed by computing the average performance of all paths as above. Let $av(i, m)$ denote the average performance of all paths through the graph for M starting from state i after m steps. This can be calculated as

$$av(i,m) = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \left(r(i,a) + \sum_{j \in \mathcal{S}} t(i,a,j) \cdot av(j, m-1) \right) .$$

Because we are interested only in whether $av(i, m)$ greater than 0, we can multiply both sides with $|\mathcal{A}|2^{|\mathcal{S}|m}$ without changing the sign of the average. Then we obtain

$$av(i,m) \cdot |\mathcal{A}|2^{|\mathcal{S}|m} = \sum_{a \in \mathcal{A}} \left(2^{|\mathcal{S}|m} r(i,a) + \sum_{j \in \mathcal{S}} 2^{|\mathcal{S}|} t(i,a,j) \cdot 2^{|\mathcal{S}|(m-1)} av(j, m-1) \right) .$$

Note that the right hand side of the equation now only deals with integer values. We argue that $av(i, m)$ is in GapL (defined in [17]), i.e. the class of integer functions f for which exist logspace nondeterministic Turing machines N_f such that the number of accepting paths minus the number of rejectin paths N_f on input x equals $f(x)$. The functions r and t are part of the input, and GapL is closed under all operations used on the right hand side of the equation [1]. Therefore it follows that $av(i, m) \cdot |\mathcal{A}|2^{|\mathcal{S}|m}$ is computable in GapL. Thus the problem whether $av(i, m) \cdot |\mathcal{A}|2^{|\mathcal{S}|m}$ is greater than 0 is decidable in PL (follows using a characterization of PL by GapL from [1]).

Hardness for PL follows using the standard simulating technique, where rejecting computations have reward -1 and accepting computations have reward 1 (weighted depending on the length of the computation). ∎

Note that the time-dependent policy average problem for fully-observable MDPs can be shown to have the same complexity using the same arguments as above.

5 Nonapproximability

How hard is it to find the policy with maximal performance for a given POMDP? Given a policy and a POMDP, one can compute the performance of the policy in polynomial time. Therefore, computing an optimal policy is at least as hard as deciding the existence problem for the POMDP. Instead of asking for an optimal policy, one might settle for a large fraction of the optimal value. A polynomial time algorithm computing such a nearly optimal policy is called an ε-approximation (for $0 < \varepsilon < 1$), where ε indicates the quality of the approximation in the following way. Let A be a polynomial time algorithm which for any POMDP M computes a policy π_α^M. The algorithm A is called an ε-approximation, if for any POMDP M,

$$\mathrm{perf}(M, \pi_\alpha^M) > (1-\varepsilon) \cdot \mathrm{val}_\alpha(M, |\mathcal{S}|) .$$

(See e.g. [12] for more detailed definitions.) Approximability distinguishes NP-complete problems: there are problems which are ε-approximable for all ε, for certain ε, or for no ε (unless $\mathrm{P} = \mathrm{NP}$).

We consider the question whether the optimal stationary policy can be ε-approximated for POMDPs with nonnegative rewards. Remember that the related decision problem is NP-complete (Theorem 3.3).

Theorem 5.1 *The optimal stationary policy for POMDPs with nonnegative rewards can be ε-approximated for some $\varepsilon < 1$ if and only if* $\mathrm{P} = \mathrm{NP}$.

The proof relies on the fact that the POMDP constructed from a formula ϕ in the proof of Theorem 3.3 has stationary value either 0 or 1. Immediately we get the same nonapproximability result for POMDPs with unrestricted rewards.

Corollary 5.2 *The optimal stationary policy for POMDPs can be ε-approximated for some $\varepsilon < 1$ if and only if* $\mathrm{P} = \mathrm{NP}$.

A similar argument can be used to show that a policy with performance at least the average of all performances for a POMDP cannot be computed in polynomial time, unless $\mathrm{P} = \mathrm{NP}$. Remember that in the proof of Theorem 3.5, NP-hardness followed from a reduction to a POMDP with nonnegative rewards. Such a POMDP has a positive-value policy iff the average value of policies is positive.

Theorem 5.3 *The following are equivalent.*

1. *There exists a polynomial-time algorithm which for a given POMDP M computes a stationary policy under which M has performance greater or equal the average stationary performance of M.*
2. $\mathrm{P} = \mathrm{NP}$.

6 Compressed representations of POMDPs

Compressed representations of MDPs arise when the system being modeled has sufficient structure, e.g., when a state is modeled by the states of n variables, and for each action, the state of each variable depends in some simple way on the states of other variables. This is true, for instance, when a system is modeled as a Bayes belief net, or when actions are given in the STRIPS model [6, 9, 10].

Changing the way in which POMDPs are represented may change the complexities of the considered problems too. We focus on two concepts, on *succinct representations*, which was introduced independently in [8, 18], and on what we call *small representations*. For succinct representations, the transition tables of a POMDP with n states and actions can be represented by a Boolean circuit C with $4\lceil \log n \rceil$ input bits such that $C(s, a, s', l)$ outputs the l-th bit of $t(s, a, s')$. Encodings of those circuits are no larger than the straightforward table encodings. But for POMDPs with sufficient structure, the circuit encodings may be much smaller, namely of size order of the logarithm of the state space. It is easy to see that in going from straightforward to succinct encodings, the complexity of the corresponding decision problems increases at most exponentially. More importantly (and less obviously), there are many problems for which this exponential increase in complexity is inherent (see [2] for conditions). Put succinctly, all results on policy existence problems translate to succinctly represented POMDPs. Note that we still take the size of the state space as our finite horizon, which may be exponential in the size of the representation of the POMDP.

Theorem 6.1 *The existence problem for stationary, time-dependent, or history-dependent policies for fully-observable succinct MDPs with nonnegative rewards is* PSPACE*-complete.*

Theorem 6.2 *The time- or history-dependent policy existence problems for fully-observable succinct MDPs are* EXP*-complete.*

Theorem 6.3 *The stationary policy existence problem for succinct POMDPs with nonnegative rewards is* NEXP*-complete. The time- or history-dependent policy existence problems for succinct POMDPs with nonnegative rewards are* PSPACE*-complete.*

Theorem 6.4 *The stationary and time-dependent policy existence problems for succinct POMDPs are* NEXP*-complete. The history-dependent policy existence problem for succinct POMDPs is* EXPSPACE*-complete.*

Another case of compressed represented POMDPs are those where the circuits representing the transition tables have many output gates from which, on input s, a, s', the transition probability $t(s, a, s')$ can be read. Because the number of output gates is bounded by the size of the circuit, for a POMDP with n states, which is encoded by such circuits of size order of $\log n$, the transition probabilities can use only order of $\log n$ binary bits to be written down (in their standard representation); i.e. they can be represented by a small number of bits. Our above completeness proof techniques do not apply for these *small* POMDPs. But we show that for POMDPs represented by small circuits we get completeness results intermediate between those of straightforward and succinct POMDPs, if we restrict the horizon to be logarithmic in the size of the state space.

Theorem 6.5 *The* $\lceil \log |\mathcal{S}| \rceil$*-horizon stationary, time-dependent and history-dependent policy existence problems for fully-observable small MDPs with nonnegative rewards are* NP*-complete.*

Proof Membership in NP follows using similar arguments as for Theorem 3.1. Hardness for NP follows from the fact that an NP computation tree can be represented as an MDP, where nondeterministic computation steps are probabilistic transitions between states. Reaching an accepting state yields a positive reward. ■

The same argument works for small POMDPs with nonnegative rewards.

Theorem 6.6 *The* $\lceil \log |\mathcal{S}| \rceil$*-horizon stationary, time-dependent and history-dependent policy existence problems for small POMDPs with nonnegative rewards are* NP*-complete.*

Turning to POMDPs with unrestricted rewards, the complexity of the policy existence problems increases drastically. Actually, we show that the stationary policy existence problem has equal complexity for small and for succinct POMDPs (cf. Theorem 6.4), and for history-dependent policy the existence problems for straightforward and small POMDPs have equal complexity (cf. Theorem 3.6).

Theorem 6.7 *The* $\lceil \log |\mathcal{S}| \rceil$*-horizon stationary policy existence problem for small POMDPs is* NEXP*-complete.*

Proof Membership in NEXP follows from Theorem 6.4. To show NEXP-hardness, we sketch a reduction from the succinct version of 3SAT, shown to be NEXP-complete in [15]. The reduction is similar to that of Theorem 3.5, where a formula was transformed into a POMDP which "checks" all clauses of the formula in parallel, works here. We put all states which stand for appearances of the same variable into one observation class and leave out considerations of variables in a clause where they do not appear. Then each trajectory of length at least 4 checks a clause, which satisfies the horizon bound. ■

Theorem 6.8 *The* $\lceil \log |\mathcal{S}| \rceil$*-horizon stationary, time-dependent and history-dependent policy existence problems for fully-observable small MDPs are* PSPACE*-hard.*

Proof We consider the case for stationary policies at first. To prove hardness, we show a polynomial time reduction from QBF, the validity problem for quantified Boolean formulae.

Informally, from a formula Φ with n variables we construct a fully-observable MDP with $2^{n+1} - 1$ states, where every state represents an assignment of Boolean values to the first i variables ($0 \leq i \leq n$) of Φ. Transitions from state s can reach the two states representing assignments that extend s by assigning a value to the next unassigned variable. If this variable is bound by an existential quantifier, then the action taken in s assigns a value to that variable; otherwise the transition is random and independent of the action. Reward 1 is gained for every action after a state representing a satisfying assignment for the formula is reached. If a state representing an unsatisfying assignment is reached, reward $-(2^n)$ is gained. Then the value of this MDP is > 0 iff the formula is true. A succinct representation of the MDP can be computed in time polynomial in the size of the formula.

This construction proves the same lower bound for the other types of policies, since every state except the last one appears at most once in every (consistent) trajectory. ■

The history-dependent case can additionally be shown to be in PSPACE.

Theorem 6.9 *The* $\lceil \log |\mathcal{S}| \rceil$*-horizon history-dependent policy existence problem for fully-observable small MDPs is* PSPACE*-complete.*

Theorem 6.10 *The* $\lceil \log |\mathcal{S}| \rceil$*-horizon history-dependent policy existence problem for small POMDPs is* PSPACE*-complete.*

The exact complexity of the time-dependent policy existence problem for small POMDPs remains open.

Theorem 6.11 *The* $\lceil \log |\mathcal{S}| \rceil$*-horizon time-dependent policy existence problem for small POMDPs is* PSPACE*-hard and is in* NEXP*.*

Proof Hardness follows from Theorem 6.8. Containment in NEXP follows from the standard guess-and-check approach. ■

7 Conclusion

We have systematically investigated the complexity of policy existence problems for finite-horizon partially-observable Markov decision processes, showing how the complexity depends on the observability, the type of policy, the rewards and the representation of the process. For the stationary policy existence problem, whose complexity is unsolved for fully-observable processes, we investigated the complexity of the existence, average and median problem, and showed that a good policy cannot be approximated in polynomial time.

Acknowledgements

We would like to thank Anne Condon, Matthew Levy, Andy Klapper, and especially Michael Littman for discussions and suggestions on this material.

References

[1] E. Allender and M. Ogihara. Relationships among PL, #L, and the determinant. *Theoretical Informatics and Applications*, 30(1):1–21, 1996.

[2] J.L. Balcázar, A. Lozano, and J. Torán. The complexity of algorithmic problems on succinct instances. In R. Baeza-Yates and U. Manber, editors, *Computer Science*, pages 351–377. Plenum Press, 1992.

[3] A. Borodin, S. Cook, and N. Pippenger. Parallel computation for well-endowed rings and space-bounded probabilistic machines. Technical Report 162/83, University of Toronto, Department of Computer Science, 1983.

[4] C. Boutilier, R. Dearden, and M. Goldszmidt. Exploiting structure in policy construction. In *14th International Conference on AI*, 1995.

[5] D. Burago, M. de Rougemont, and A. Slissenko. On the complexity of partially observed Markov decision processes. *Theoretical Computer Science*, 157(2):161–183, 1996.

[6] T. Bylander. The computational complexity of propositional STRIPS planning. *Artificial Intelligence*, 69:165–204, 1994.

[7] K. Erol, D. Nau, and V. S. Subrahmanian. Complexity, decidability and undecidability results for domain-independent planning. *Artificial Intelligence*, 76:75–88, 1995.

[8] H. Galperin and A. Wigderson. Succinct representation of graphs. *Information and Control*, 56:183–198, 1983.

[9] J. Goldsmith, M. Littman, and M. Mundhenk. The complexity of plan existence and evaluation in probabilistic domains. Technical report, Duke University Department of Computer Science, 1997.

[10] M.L. Littman. Probabilistic propositional planning: Representations and complexity. Technical Report CS-1996-18, Department of Computer Science, Duke University, 1996.

[11] M. Mundhenk, J. Goldsmith, and E. Allender. The complexity of policy evaluation for finite-horizon partially-observable Markov decision processes. Technical Report 269-96, University of Kentucky Department of Computer Science, 1996.

[12] C.H. Papadimitriou. *Computational Complexity.* Addison-Wesley, 1994.

[13] C.H. Papadimitriou and J.N. Tsitsiklis. Intractable problems in control theory. *SIAM Journal of Control and Optimization*, pages 639–654, 1986.

[14] C.H. Papadimitriou and J.N. Tsitsiklis. The complexity of Markov decision processes. *Mathematics of Operations Research*, 12(3):441–450, 1987.

[15] C.H. Papadimitriou and M. Yannakakis. A note on succinct representations of graphs. *Information and Control*, 71:181–185, 1986.

[16] P. Tseng. Solving h-horizon, stationary Markov decision problems in time proportional to $\log h$. *Operations Research Letters*, pages 287–297, September 1990.

[17] V. Vinay. Counting auxiliary pushdown automata and semi-unbounded arithmetic circuits. In *Proc. 6th Structure in Complexity Theory Conference*, pages 270–284. IEEE, 1991.

[18] K. W. Wagner. The complexity of combinatorial problems with succinct input representation. *Acta Informatica*, 23:325–356, 1986.