What is Decidable about Partially Observable Markov Decision Processes with omega-Regular Objectives*

Krishnendu Chatterjee, Martin Chmelik, and Mathieu Tracol

IST Austria
Klosterneuburg, Austria

Abstract

We consider partially observable Markov decision processes (POMDPs) with ω-regular conditions specified as parity objectives. The qualitative analysis problem given a POMDP and a parity objective asks whether there is a strategy to ensure that the objective is satisfied with probability 1 (resp. positive probability). While the qualitative analysis problems are known to be undecidable even for very special cases of parity objectives, we establish decidability (with optimal EXPTIME-complete complexity) of the qualitative analysis problems for POMDPs with all parity objectives under finite-memory strategies. We also establish optimal (exponential) memory bounds.

1998 ACM Subject Classification D.2.4 Formal methods

Keywords and phrases POMDPs, Omega-regular objectives, Parity objectives, Qualitative analysis

Digital Object Identifier 10.4230/LIPIcs.CSL.2013.165

1 Introduction

Partially observable Markov decision processes (POMDPs). *Markov decision processes (MDPs)* are standard models for probabilistic systems that exhibit both probabilistic and nondeterministic behavior [16]. MDPs have been used to model and solve control problems for stochastic systems [13]: nondeterminism represents the freedom of the controller to choose a control action, while the probabilistic component of the behavior describes the system response to control actions. In *perfect-observation (or perfect-information) MDPs (PIMDPs)* the controller can observe the current state of the system to choose the next control actions, whereas in *partially observable MDPs (POMDPs)* the state space is partitioned according to observations that the controller can observe i.e., given the current state, the controller can only view the observation of the state (the partition the state belongs to), but not the precise state [22]. POMDPs provide the appropriate model to study a wide variety of applications such as in computational biology [12], speech processing [21], software verification [6], robot planning [17], to name a few. In verification of probabilistic systems, MDPs have been adopted as models for concurrent probabilistic systems [10], under-specified probabilistic systems [4], and applied in diverse domains [3, 18]. POMDPs also subsume many other powerful computational models such as probabilistic automata [25, 23] (since probabilistic automata are a special case of POMDPs with a single observation).

* The research was supported by Austrian Science Fund (FWF) Grant No P 23499- N23, FWF NFN Grant No S11407-N23 (RiSE), ERC Start grant (279307: Graph Games), and Microsoft faculty fellows award.

© Krishnendu Chatterjee and Martin Chmelik and Mathieu Tracol;
licensed under Creative Commons License CC-BY

Computer Science Logic 2013 (CSL'13).
Editor: Simona Ronchi Della Rocca; pp. 165–180
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

The class of ω-regular objectives. An objective specifies the desired set of behaviors (or paths) for the controller. In verification and control of stochastic systems an objective is typically an ω-regular set of paths. The class of ω-regular languages extends classical regular languages to infinite strings, and provides a robust specification language to express all commonly used specifications [28]. In a parity objective, every state of the MDP is mapped to a non-negative integer priority (or color) and the goal is to ensure that the minimum priority (or color) visited infinitely often is even. Parity objectives are a canonical way to define such ω-regular specifications. Thus POMDPs with parity objectives provide the theoretical framework to study problems such as the verification and control of stochastic systems.

Qualitative and quantitative analysis. The analysis of POMDPs with parity objectives can be classified into qualitative and quantitative analysis. Given a POMDP with a parity objective and a start state, the *qualitative analysis* asks whether the objective can be ensured with probability 1 (*almost-sure winning*) or positive probability (*positive winning*); whereas the *quantitative analysis* asks whether the objective can be satisfied with probability at least λ for a given threshold $\lambda \in (0, 1)$.

Importance of qualitative analysis. The qualitative analysis of MDPs is an important problem in verification that is of interest independent of the quantitative analysis problem. There are many applications where we need to know whether the correct behavior arises with probability 1. For instance, when analyzing a randomized embedded scheduler, we are interested in whether every thread progresses with probability 1 [11]. Even in settings where it suffices to satisfy certain specifications with probability $\lambda < 1$, the correct choice of λ is a challenging problem, due to the simplifications introduced during modeling. For example, in the analysis of randomized distributed algorithms it is quite common to require correctness with probability 1 (see, e.g., [24, 27]). Furthermore, in contrast to quantitative analysis, qualitative analysis is robust to numerical perturbations and modeling errors in the transition probabilities. Thus qualitative analysis of POMDPs with parity objectives is a fundamental theoretical problem in verification and analysis of probabilistic systems.

Previous results. On one hand POMDPs with parity objectives provide a rich framework to model a wide variety of practical problems, on the other hand, most theoretical results established for POMDPs are *negative* (undecidability) results. There are several deep undecidability results established for the special case of probabilistic automata (that immediately imply undecidability for the more general case of POMDPs). The basic undecidability results are for probabilistic automata over finite words (that can be considered as a special case of parity objectives). The quantitative analysis problem is undecidable for probabilistic automata over finite words [25, 23]; and it was shown in [19] that even the following approximation version is undecidable: for any fixed $0 < \epsilon < \frac{1}{2}$, given a probabilistic automaton and the guarantee that either (a) there is a word accepted with probability at least $1 - \epsilon$; or (ii) all words are accepted with probability at most ϵ; decide whether it is case (i) or case (ii). The almost-sure (resp. positive) problem for probabilistic automata over finite words reduces to the non-emptiness question of universal (resp. non-deterministic) automata over finite words and is PSPACE-complete (resp. solvable in polynomial time). However, another related decision question whether for every $\epsilon > 0$ there is a word that is accepted with probability at least $1 - \epsilon$ (the value 1 problem) is undecidable for probabilistic automata over finite words [14]. Also observe that all undecidability results for probabilistic automata over finite words carry over to POMDPs where the controller is restricted to finite-memory strategies. In [20], the authors consider POMDPs with finite-memory strategies under expected rewards, but the general problem remains undecidable. For qualitative analysis of

POMDPs with parity objectives, deep undecidability results were shown for very special cases of parity objectives (even in the special case of probabilistic automata). It was shown in [2] that the almost-sure (resp. positive) problem is undecidable for probabilistic automata with coBüchi (resp. Büchi) objectives which are special cases of parity objectives that use only two priorities. In summary the most important theoretical results are negative (they establish undecidability results).

Our contributions. The undecidability proofs for the qualitative analysis of POMDPs with parity objectives crucially require the use of *infinite-memory* strategies for the controller. In all practical applications, the controller must be a *finite-state* controller to be implementable. Thus for all practical purposes the relevant question is the existence of finite-memory controllers. The quantitative analysis problem remains undecidable even under finite-memory controllers as the undecidability results are established for probabilistic automata over finite words. In this work we study the most prominent remaining theoretical open question (that is also of practical relevance) for POMDPs with parity objectives that whether the qualitative analysis of POMDPs with parity objectives is decidable or undecidable for finite-memory strategies (i.e., finite-memory controllers). Our main result is the *positive* result that the qualitative analysis of POMDPs with parity objectives is *decidable* under finite-memory strategies. Moreover, for qualitative analysis of POMDPs with parity objectives under finite-memory strategies, we establish optimal complexity bounds both for strategy complexity as well as computational complexity. Our contributions are as follows (summarized in Table 1):

1. *(Strategy complexity).* Our first result shows that *belief-based stationary* strategies are not sufficient (where a belief-based stationary strategy is based on the subset construction that remembers the possible set of current states): we show that there exist POMDPs with coBüchi objectives where finite-memory almost-sure winning strategy exists but there exists no randomized belief-based stationary almost-sure winning strategy. All previous results about decidability for almost-sure winning in sub-classes of POMDPs crucially relied on the sufficiency of randomized belief-based stationary strategies that allowed standard techniques like subset construction to establish decidability. However, our counter-example shows that previous techniques based on simple subset construction (to construct an exponential size PIMDP) are not adequate to solve the problem. Before the result for parity objectives, we consider a slightly more general form of objectives, called Muller objectives. For a Muller objective a set $\mathcal{F}$ of subsets of colors is given and the set of colors visited infinitely often must belong to $\mathcal{F}$. We show our main result that given a POMDP with $|S|$ states and a Muller objective with d colors (priorities), if there is a finite-memory almost-sure (resp. positive) winning strategy, then there is an almost-sure (resp. positive) winning strategy that uses at most $\mathsf{Mem}^* = 2^{2\cdot|S|} \cdot (2^{2^d})^{|S|}$ memory. Developing on our result for Muller objectives, for POMDPs with parity objectives we show that if there is a finite-memory almost-sure (resp. positive) winning strategy, then there is an almost-sure (resp. positive) winning strategy that uses at most $2^{3\cdot d\cdot|S|}$ memory. Our exponential memory upper bound for parity objectives is optimal as it is shown in [8] that almost-sure winning strategies require at least exponential memory even for the very special case of reachability objectives in POMDPs.

2. *(Computational complexity).* We present an exponential time algorithm for the qualitative analysis of POMDPs with parity objectives under finite-memory strategies, and thus obtain an EXPTIME upper bound. The EXPTIME-hardness follows from [8] for

Table 1 Strategy and computational complexity for POMDPs. UB:Upper bound; LB: Lower bound. New results in bold fonts.

Objectives		Almost-sure		Positive	
		Inf. Mem.	Fin. Mem.	Inf. Mem.	Fin. Mem.
Büchi	Strategy	Exp . (belief)	Exp . (belief)	Inf. mem.	UB: **Exp.** $\mathbf{2^{6\cdot\lvert S\rvert}}$ LB: Exp. **(belief not suf.)**
	Complexity	EXP-c.	EXP-c.	Undec.	**EXP-c.**
coBüchi	Strategy	Inf. mem	UB: **Exp.** $\mathbf{2^{6\cdot\lvert S\rvert}}$ LB: Exp. **(belief not suf.)**	UB: Exp. LB: Exp. **(belief not suf.)**	UB: Exp. LB: Exp. **(belief not suf.)**
	Complexity	Undec.	**EXP-c.**	EXP-c.	EXP-c.
Parity	Strategy	Inf. mem	UB: **Exp.** $\mathbf{2^{3\cdot d\cdot\lvert S\rvert}}$ LB: Exp. **(belief not suf.)**	Inf. mem	UB: **Exp.** $\mathbf{2^{3\cdot d\cdot\lvert S\rvert}}$ LB: Exp. **(belief not suf.)**
	Complexity	Undec.	**EXP-c.**	Undec.	**EXP-c.**

the special case of reachability and safety objectives, and thus we obtain the optimal EXPTIME-complete computational complexity result.[1]

Technical contributions. The key technical contribution for the decidability result is as follows. Since belief-based stationary strategies are not sufficient, standard subset construction techniques do not work. For an arbitrary finite-memory strategy we construct a projected strategy that collapses memory states based on a projection graph construction given the strategy. The projected strategy at a collapsed memory state plays uniformly over actions that were played at all the corresponding memory states of the original strategy. The projected strategy thus plays more actions with positive probability. The key challenge is to show the bound on the size of the projection graph, and to show that the projected strategy, even though plays more actions, does not destroy the structure of the recurrent classes of the original strategy. For parity objectives, we show a reduction from general parity objectives to parity objectives with two priorities on a polynomially larger POMDP and from our general result for Muller objectives obtain the optimal memory complexity bounds for parity objectives. For the computational complexity result, we show how to construct an exponential size special class of POMDPs (which we call belief-observation POMDPs where the belief is always the current observation) and present polynomial time algorithms for the qualitative analysis of the special belief-observation POMDPs of our construction. Full proofs are available as technical report, Feb 20, 2013, `https://repository.ist.ac.at/109/`.

2 Definitions

In this section we present the basic definitions of POMDPs, strategies (policies), ω-regular objectives, and the winning modes.

Notations. For a finite set X, we denote by $\mathcal{P}(X)$ the set of subsets of X (the power set of X). A probability distribution f on X is a function $f : X \to [0,1]$ such that $\sum_{x\in X} f(x) = 1$, and we denote by $\mathcal{D}(X)$ the set of all probability distributions on X. For $f \in \mathcal{D}(X)$ we denote by $\mathrm{Supp}(f) = \{x \in X \mid f(x) > 0\}$ the support of f.

▶ **Definition 1** (POMDPs). A *Partially Observable Markov Decision Process (POMDP)* is a tuple $G = (S, A, \delta, \mathcal{O}, \gamma, s_0)$ where: (i) S is a finite set of states; (ii) A is a finite alphabet of

[1] Recently, Nain and Vardi (personal communication, to appear LICS 2013) considered the finite-memory strategies problem for one-sided partial-observation games and established 2EXPTIME upper bound. Our work is independent and establishes optimal (EXPTIME-complete) complexity bounds for POMDPs.

actions; (iii) $\delta : S \times A \to \mathcal{D}(S)$ is a *probabilistic transition function* that given a state s and an action $a \in A$ gives the probability distribution over the successor states, i.e., $\delta(s,a)(s')$ denotes the transition probability from state s to state s' given action a; (iv) $\mathcal{O}$ is a finite set of *observations*; (v) $\gamma : S \to \mathcal{O}$ is an *observation function* that maps every state to an observation; and (vi) s_0 is the initial state.

Given $s, s' \in S$ and $a \in A$, we also write $\delta(s'|s,a)$ for $\delta(s,a)(s')$. For an observation o, we denote by $\gamma^{-1}(o) = \{s \in S \mid \gamma(s) = o\}$ the set of states with observation o. For a set $U \subseteq S$ of states and $O \subseteq \mathcal{O}$ of observations we denote $\gamma(U) = \{o \in \mathcal{O} \mid \exists s \in U.\ \gamma(s) = o\}$ and $\gamma^{-1}(O) = \bigcup_{o \in O} \gamma^{-1}(o)$. For technical convenience we consider that the initial state s_0 has a unique observation.

Plays, cones and belief-updates. A *play* (or a path) in a POMDP is an infinite sequence $(s_0, a_0, s_1, a_1, s_2, a_2, \ldots)$ of states and actions such that for all $i \geq 0$ we have $\delta(s_i, a_i)(s_{i+1}) > 0$. We write Ω for the set of all plays. For a finite prefix $w \in (S \cdot A)^* \cdot S$ of a play, we denote by $\mathsf{Cone}(w)$ the set of plays with w as the prefix (i.e., the cone or cylinder of the prefix w), and denote by $\mathsf{Last}(w)$ the last state of w. For a finite prefix $w = (s_0, a_0, s_1, a_1, \ldots, s_n)$ we denote by $\gamma(w) = (\gamma(s_0), a_0, \gamma(s_1), a_1, \ldots, \gamma(s_n))$ the observation and action sequence associated with w. For a finite sequence $\rho = (o_0, a_0, o_1, a_1, \ldots, o_n)$ of observations and actions, the *belief* $\mathcal{B}(\rho)$ after the prefix ρ is the set of states in which a finite prefix of a play can be after the sequence ρ of observations and actions, i.e., $\mathcal{B}(\rho) = \{s_n = \mathsf{Last}(w) \mid w = (s_0, a_0, s_1, a_1, \ldots, s_n), w \text{ is a prefix of a play, and for all } 0 \leq i \leq n.\ \gamma(s_i) = o_i\}$. The belief-updates associated with finite-prefixes are as follows: for prefixes w and $w' = w \cdot a \cdot s$ the belief update is defined inductively as $\mathcal{B}(\gamma(w')) = \left(\bigcup_{s_1 \in \mathcal{B}(\gamma(w))} \mathrm{Supp}(\delta(s_1, a))\right) \cap \gamma^{-1}(s)$.

Strategies. A *strategy (or a policy)* is a recipe to extend prefixes of plays and is a function $\sigma : (S{\cdot}A)^*{\cdot}S \to \mathcal{D}(A)$ that given a finite history (i.e., a finite prefix of a play) selects a probability distribution over the actions. Since we consider POMDPs, strategies are *observation-based*, i.e., for all histories $w = (s_0, a_0, s_1, a_1, \ldots, a_{n-1}, s_n)$ and $w' = (s'_0, a_0, s'_1, a_1, \ldots, a_{n-1}, s'_n)$ such that for all $0 \leq i \leq n$ we have $\gamma(s_i) = \gamma(s'_i)$ (i.e., $\gamma(w) = \gamma(w')$), we must have $\sigma(w) = \sigma(w')$. In other words, if the observation sequence is the same, then the strategy cannot distinguish between the prefixes and must play the same. We now present an equivalent definition of strategies such that the memory is explicit.

▶ **Definition 2** (Strategies with memory and memoryless strategies)**.** A *strategy* with memory is a tuple $\sigma = (\sigma_u, \sigma_n, M, m_0)$ where: (i) *(Memory set).* M is a denumerable set (finite or infinite) of memory elements (or memory states). (ii) *(Action selection function).* The function $\sigma_n : M \to \mathcal{D}(A)$ is the *action selection function* that given the current memory state gives the probability distribution over actions. (iii) *(Memory update function).* The function $\sigma_u : M \times \mathcal{O} \times A \to \mathcal{D}(M)$ is the *memory update function* that given the current memory state, the current observation and action, updates the memory state probabilistically. (iv) *(Initial memory).* The memory state $m_0 \in M$ is the initial memory state. A strategy is a *finite-memory* strategy if the set M of memory elements is finite. A strategy is *pure (or deterministic)* if the memory update function and the action selection function are deterministic. A strategy is *memoryless (or stationary)* if it is independent of the history but depends only on the current observation, and can be represented as a function $\sigma : \mathcal{O} \to \mathcal{D}(A)$.

▶ Remark. It was shown in [7] that in POMDPs pure strategies are as powerful as randomized strategies, hence in sequel we omit discussions about pure strategies.

Probability measure. Given a strategy σ, the unique probability measure obtained given σ is denoted as $\mathbb{P}^\sigma(\cdot)$. We first define the measure $\mu^\sigma(\cdot)$ on cones. For $w = s_0$ we have

$\mu^\sigma(\mathsf{Cone}(w)) = 1$, and for $w = s$ where $s \neq s_0$ we have $\mu^\sigma(\mathsf{Cone}(w)) = 0$; and for $w' = w \cdot a \cdot s$ we have $\mu^\sigma(\mathsf{Cone}(w')) = \mu^\sigma(\mathsf{Cone}(w)) \cdot \sigma(w)(a) \cdot \delta(\mathsf{Last}(w), a)(s)$. By Caratheódary's extension theorem, the function $\mu^\sigma(\cdot)$ can be uniquely extended to a probability measure $\mathbb{P}^\sigma(\cdot)$ over Borel sets of infinite plays [5].

Objectives. An *objective* in a POMDP G is a measureable set $\varphi \subseteq \Omega$ of plays. For a play $\rho = (s_0, a_0, s_1, a_1, s_2 \ldots)$, we denote by $\mathrm{Inf}(\rho) = \{s \in S \mid \forall i \geq 0 \cdot \exists j \geq i : s_j = s\}$ the set of states that occur infinitely often in ρ. We consider the following objectives.

- *Reachability and safety objectives.* Given a set $\mathcal{T} \subseteq S$ of target states, the *reachability* objective $\mathsf{Reach}(\mathcal{T}) = \{(s_0, a_0, s_1, a_1, s_2 \ldots) \in \Omega \mid \exists k \geq 0 : s_k \in \mathcal{T}\}$ requires that a target state in $\mathcal{T}$ is visited at least once. Dually, the *safety* objective $\mathsf{Safe}(\mathcal{T}) = \{(s_0, a_0, s_1, a_1, s_2 \ldots) \in \Omega \mid \forall k \geq 0 : s_k \in \mathcal{T}\}$ requires that only states in $\mathcal{T}$ are visited.
- *Büchi and coBüchi objectives.* Given a set $\mathcal{T} \subseteq S$ of target states, the *Büchi* objective $\mathsf{Buchi}(\mathcal{T}) = \{\rho \in \Omega \mid \mathrm{Inf}(\rho) \cap \mathcal{T} \neq \emptyset\}$ requires that a state in $\mathcal{T}$ is visited infinitely often. Dually, the *coBüchi* objective $\mathsf{coBuchi}(\mathcal{T}) = \{\rho \in \Omega \mid \mathrm{Inf}(\rho) \subseteq \mathcal{T}\}$ requires that only states in $\mathcal{T}$ are visited infinitely often.
- *Parity objectives.* For $d \in \mathbb{N}$, let $p : S \to \{0, 1, \ldots, d\}$ be a *priority function* that maps each state to a non-negative integer priority. The *parity* objective $\mathsf{Parity}(p) = \{\rho \in \Omega \mid \min\{p(s) \mid s \in \mathrm{Inf}(\rho)\}$ is even$\}$ requires that the smallest priority that appears infinitely often is even.
- *Muller objectives.* Let D be a set of colors, and $\mathsf{col} : S \to D$ be a color mapping function that maps every state to a color. A Muller objective $\mathcal{F}$ consists of a set of subsets of colors and requires that the set of colors visited infinitely often belongs to $\mathcal{F}$, i.e., $\mathcal{F} \in \mathcal{P}(\mathcal{P}(D))$ and $\mathsf{Muller}(\mathcal{F}) = \{\rho \in \Omega \mid \{\mathsf{col}(s) \mid s \in \mathrm{Inf}(\rho)\} \in \mathcal{F}\}$.

Given a set $U \subseteq S$ we will denote by $p(U)$ the set of priorities of the set U given by the priority function p, i.e., $p(U) = \{p(s) \mid s \in U\}$, and similarly $\mathsf{col}(U) = \{\mathsf{col}(s) \mid s \in U\}$. Büchi and coBüchi objectives are parity objectives with two priorities; and parity objectives are a special case of Muller objectives. However, given a POMDP with a Muller objective with color set D, an equivalent POMDP with $|S| \cdot |D|!$ states and a parity objective with $|D|^2$ priorities can be constructed with the latest appearance record (LAR) construction of [15].

Winning modes. Given a POMDP, an objective φ, and a class $\mathcal{C}$ of strategies, we say that: a strategy $\sigma \in \mathcal{C}$ is *almost-sure winning* (resp. *positive winning*) if $\mathbb{P}^\sigma(\varphi) = 1$ (resp. $\mathbb{P}^\sigma(\varphi) > 0$); and a strategy $\sigma \in \mathcal{C}$ is *quantitative winning*, for a threshold $\lambda \in (0, 1)$, if $\mathbb{P}^\sigma(\varphi) \geq \lambda$. We first precisely summarize related works in the following Theorem.

▶ **Theorem 3** (Previous results [25, 23, 2, 26, 8]). *The following assertions hold for POMDPs with the class $\mathcal{C}$ of all infinite-memory (randomized or pure) strategies: (1) The quantitative winning problem is undecidable for safety, reachability, Büchi, coBüchi, parity, and Muller objectives. (2) The almost-sure winning problem is EXPTIME-complete for safety, reachability, and Büchi objectives; and undecidable for coBüchi, parity, and Muller objectives. (3) The positive winning problem is PTIME-complete for reachability objectives, EXPTIME-complete for safety and coBüchi objectives; and undecidable for Büchi, parity, and Muller objectives.*

Explanation of the previous results and implications under finite-memory strategies. All the undecidability results follow from the special case of probabilistic automata: the undecidability of the quantitative problem for probabilistic automata follows from [25, 23, 9]. The undecidability for positive winning for Büchi and almost-sure winning for coBüchi objectives was established in [1, 2]. For the decidable results, the optimal complexity results for safety objectives can be obtained from the results of [26] and all the other results follow from [8, 2]. If the classes of strategies are restricted to finite-memory strategies, then the undecidability results for quantitative winning still hold, as they are established for reachability objectives and for reachability objectives finite-memory suffices. The most prominent and important open question is whether the almost-sure and positive winning problems are decidable for parity and Muller objectives in POMDPs under finite-memory strategies.

3 Strategy Complexity

In this section we will first show that belief-based stationary strategies are not sufficient for finite-memory almost-sure winning strategies in POMDPs with coBüchi objectives; and then present the upper bound on memory size required for finite-memory almost-sure and positive winning strategies in POMDPs with Muller objectives, and finally for parity objectives. We start with some basic results about Markov chains.

Markov chains, recurrent classes, and reachability. A Markov chain $\overline{G} = (\overline{S}, \overline{\delta})$ consists of a *finite* set $\overline{S}$ of states and a probabilistic transition function $\overline{\delta} : \overline{S} \to \mathcal{D}(\overline{S})$. Given the Markov chain, we consider the graph $(\overline{S}, \overline{E})$ where $\overline{E} = \{(\overline{s}, \overline{s}') \mid \delta(\overline{s}' \mid \overline{s}) > 0\}$. A *recurrent class* $\overline{C} \subseteq \overline{S}$ of the Markov chain is a bottom strongly connected component (scc) in the graph $(\overline{S}, \overline{E})$ (a bottom scc is an scc with no edges out of the scc). We denote by $\mathsf{Rec}(\overline{G})$ the set of recurrent classes of the Markov chain, i.e., $\mathsf{Rec}(\overline{G}) = \{\overline{C} \mid \overline{C} \text{ is a recurrent class}\}$. Given a state $\overline{s}$ and a set $\overline{U}$ of states, we say that $\overline{U}$ is reachable from $\overline{s}$ if there is a path from $\overline{s}$ to some state in $\overline{U}$ in the graph $(\overline{S}, \overline{E})$. Given a state $\overline{s}$ of the Markov chain we denote by $\mathsf{Rec}(\overline{G})(\overline{s}) \subseteq \mathsf{Rec}(\overline{G})$ the subset of the recurrent classes reachable from $\overline{s}$ in $\overline{G}$. A state is recurrent if it belongs to a recurrent class.

▶ **Lemma 4.** *For a Markov chain $\overline{G} = (\overline{S}, \overline{\delta})$ with Muller objective $\mathsf{Muller}(\mathcal{F})$ (or parity objective $\mathsf{Parity}(p)$), a state $\overline{s}$ is almost-sure winning (resp. positive winning) if for all recurrent classes $\overline{C} \in \mathsf{Rec}(\overline{G})(\overline{s})$ (resp. for some recurrent class $\overline{C} \in \mathsf{Rec}(\overline{G})(\overline{s})$) reachable from $\overline{s}$ we have $\mathsf{col}(\overline{C}) \in \mathcal{F}$ ($\min(p(\overline{C}))$ is even for the parity objective).*

Markov chains $G \upharpoonright \sigma$ under finite-memory strategies σ. We now define Markov chains obtained by fixing finite-memory strategies in a POMDP G. A finite-memory strategy $\sigma = (\sigma_u, \sigma_n, M, m_0)$ induces a finite-state Markov chain $(S \times M, \delta_\sigma)$, denoted $G \upharpoonright \sigma$, with the probabilistic transition function $\delta_\sigma : S \times M \to \mathcal{D}(S \times M)$: given $s, s' \in S$ and $m, m' \in M$, the transition $\delta_\sigma\big((s', m') \mid (s, m)\big)$ is the probability to go from state (s, m) to state (s', m') in one step under the strategy σ. The probability of transition can be decomposed as follows: (i) First an action $a \in A$ is sampled according to the distribution $\sigma_n(m)$; (ii) then the next state s' is sampled according to the distribution $\delta(s, a)$; and (iii) finally the new memory m' is sampled according to the distribution $\sigma_u(m, \gamma(s'), a)$ (i.e., the new memory is sampled according to σ_u given the old memory, new observation and the action). More formally, we have: $\delta_\sigma\big((s', m') \mid (s, m)\big) = \sum_{a \in A} \sigma_n(m)(a) \cdot \delta(s, a)(s') \cdot \sigma_u(m, \gamma(s'), a)(m')$.

Belief-based stationary strategies not sufficient. For all previous decidability results for almost-sure winning in POMDPs, the key was to show that *belief-based stationary* strategies are sufficient. In POMDPs with Büchi objectives, belief-based stationary strategies are sufficient for almost-sure winning, and we now show that in POMDPs with coBüchi objectives finite-memory almost-sure winning strategies may exist whereas no belief-based stationary ones.

▶ **Example 5.** We consider a POMDP G with state space $\{s_0, X, X', Y, Y', Z, Z'\}$ and action set $\{a, b\}$, and let $U = \{X, X', Y, Y', Z, Z'\}$. From the initial state s_0 all the other states

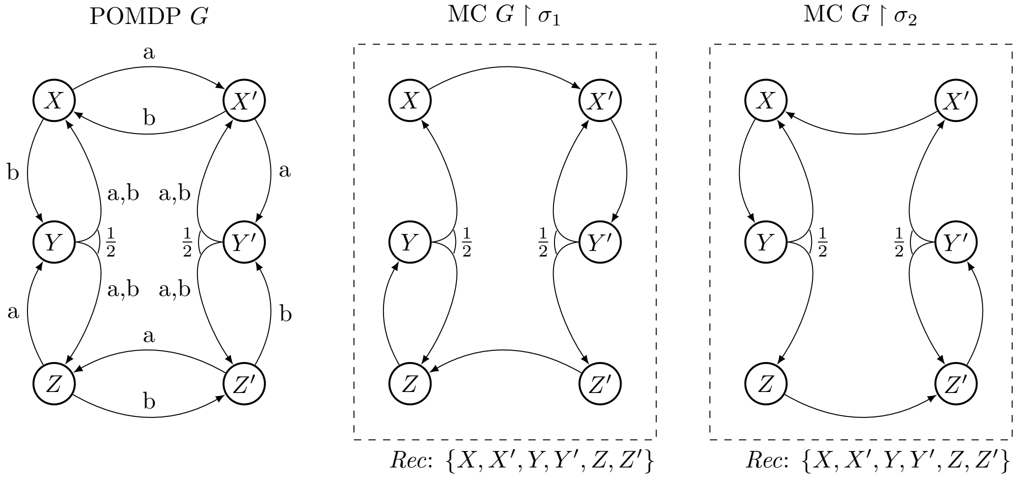
POMDP G
MC $G \restriction \sigma_1$
MC $G \restriction \sigma_2$
X
X'
Y
Y'
Z
Z'
a
b
a,b
$\frac{1}{2}$
Rec: $\{X, X', Y, Y', Z, Z'\}$
Rec: $\{X, X', Y, Y', Z, Z'\}$

Figure 1 Belief is not sufficient.

are reached with uniform probability in one-step, i.e., for all $s' \in U = \{X, X', Y, Y', Z, Z'\}$ we have $\delta(s_0, a)(s') = \delta(s_0, b)(s') = \frac{1}{6}$. The transitions from the other states (shown in Figure 1) are as follows: (i) $\delta(X, a)(X') = 1$ and $\delta(X, b)(Y) = 1$; (ii) $\delta(X', a)(Y') = 1$ and $\delta(X', b)(X) = 1$; (iii) $\delta(Z, a)(Y) = 1$ and $\delta(Z, b)(Z') = 1$; (iv) $\delta(Z', a)(Z) = 1$ and $\delta(Z', b)(Y') = 1$; (v) $\delta(Y, a)(X) = \delta(Y, b)(X) = \delta(Y, a)(Z) = \delta(Y, b)(Z) = \frac{1}{2}$; and (vi) $\delta(Y', a)(X') = \delta(Y', b)(X') = \delta(Y', a)(Z') = \delta(Y', b)(Z') = \frac{1}{2}$. All states in U have the same observation. The coBüchi objective is given by the target set $\{X, X', Z, Z'\}$, i.e., Y and Y' must be visited only finitely often. The belief initially after one-step is the set U since from s_0 all of them are reached with positive probability. The belief is always the set U since every state has an input edge for every action, i.e., if the current belief is U (i.e., the set of states that the POMDP is currently in with positive probability is U), then irrespective of whether a or b is chosen all states of U are reached with positive probability and hence the belief set is again U. There are three belief-based stationary strategies: (i) σ_1 that plays always a; (ii) σ_2 that plays always b; or (iii) σ_3 that plays both a and b with positive probability. For all the three strategies, the Markov chains obtained have the whole set U as the recurrent class (see Figure 1 for the Markov chains $G \restriction \sigma_1$ and $G \restriction \sigma_2$), and hence both Y and Y' are visited infinitely often with probability 1 violating the coBüchi objective. The strategy σ_4 that plays action a and b alternately gives rise to the Markov chain $G \restriction \sigma_4$ shown in Figure 2 (i.e., σ_4 has two memory states a and b, in memory state a it plays action a and switches to memory state b, and in memory state b it plays action b and switches to memory state a). The recurrent classes do not intersect with (Y, m) or (Y', m), for memory state $m \in \{a, b\}$, and hence σ_4 is a finite-memory almost-sure winning strategy. ◀

Upper bound on memory. For the following of the section, we fix a POMDP $G = (S, A, \delta, \mathcal{O}, \gamma, s_0)$, with a Muller objective $\mathsf{Muller}(\mathcal{F})$ with the set D of colors and a color mapping function col. We will denote by $\mathfrak{D}$ the powerset of the powerset of the set D of colors, i.e., $\mathfrak{D} = \mathcal{P}(\mathcal{P}(D))$; and note that $|\mathfrak{D}| = 2^{2^d}$, where $d = |D|$. Our goal is to prove the following fact: given a finite-memory almost-sure (resp. positive) winning strategy σ on G there exists a finite-memory almost-sure (resp. positive) winning strategy σ' on G, of memory size at most $\mathsf{Mem}^* = 2^{|S|} \cdot 2^{|S|} \cdot |\mathfrak{D}|^{|S|}$.

Overview of the proof. We first present an overview of our proof. (i) Given an arbitrary finite-memory strategy σ we will consider the Markov chain $G \restriction \sigma$ arising by fixing the

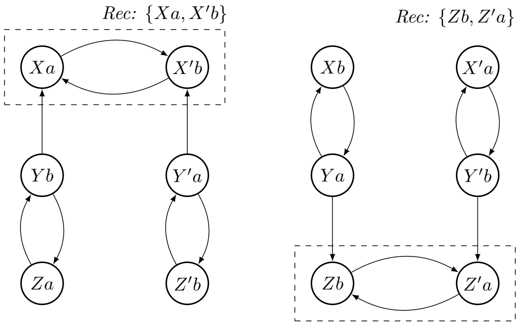

Figure 2 The Markov chain $G \upharpoonright \sigma_4$.

strategy. (ii) Given the Markov chain we will define a projection graph that depends on the recurrent classes of the Markov chain. The projection graph is of size at most Mem^*. (iii) Given the projection graph we will construct a projected strategy with memory size at most Mem^* that preserves the recurrent classes of the Markov chain $G \upharpoonright \sigma$.

Notations. Given $Z \in \mathfrak{D}^{|S|}$ and given $s \in S$, we write $Z(s)$ (which is in $\mathfrak{D} = \mathcal{P}(\mathcal{P}(D))$) for the s-component of Z. For two sets U_1 and U_2 and $U \subseteq U_1 \times U_2$, we denote by $\mathsf{Proj}_i(U)$ for $i \in \{1, 2\}$ the projection of U on the i-th component.

Basic definitions for the projection graph. We now introduce notions associated with the finite Markov chain $G \upharpoonright \sigma$ that will be essential in defining the projection graph.

▶ **Definition 6** (Recurrence set functions). Let σ be a finite-memory strategy with memory M on G for the Muller objective with the set D of colors, and let $m \in M$.

- *(Function set recurrence).* The function $\mathsf{SetRec}_\sigma(m) : S \to \mathfrak{D}$ maps every state $s \in S$ to the projections of colors of recurrent classes reachable from (s, m) in $G \upharpoonright \sigma$. Formally, $\mathsf{SetRec}_\sigma(m)(s) = \{\mathsf{col}(\mathsf{Proj}_1(U)) \mid U \in \mathsf{Rec}(G \upharpoonright \sigma)((s, m))\}$, i.e., we consider the set $\mathsf{Rec}(G \upharpoonright \sigma)((s, m))$ of recurrent classes reachable from the state (s, m) in $G \upharpoonright \sigma$, obtain the projections on the state space S and consider the colors of states in the projected set. We will in sequel consider $\mathsf{SetRec}_\sigma(m) \in \mathfrak{D}^{|S|}$.
- *(Function boolean recurrence).* The function $\mathsf{BoolRec}_\sigma(m) : S \to \{0, 1\}$ is such that for all $s \in S$, we have $\mathsf{BoolRec}_\sigma(m)(s) = 1$ if there exists $U \in \mathsf{Rec}(G \upharpoonright \sigma)((s, m))$ such that $(s, m) \in U$, and 0 if not. Intuitively, $\mathsf{BoolRec}_\sigma(m)(s) = 1$ if (s, m) belongs to a recurrent class in $G \upharpoonright \sigma$ and 0 otherwise. In sequel we will consider $\mathsf{BoolRec}_\sigma(m) \in \{0, 1\}^{|S|}$.

▶ **Lemma 7.** *Let $s, s' \in S$ and $m, m' \in M$ be such that (s', m') is reachable from (s, m) in $G \upharpoonright \sigma$. Then $\mathsf{SetRec}_\sigma(m')(s') \subseteq \mathsf{SetRec}_\sigma(m)(s)$.*

▶ **Definition 8** (Projection graph). Let σ be a finite-memory strategy. We define the *projection graph* $\mathsf{PrGr}(\sigma) = (V, E)$ associated to σ as follows:

- *(Vertex set).* The set of vertices is $V = \{(U, \mathsf{BoolRec}_\sigma(m), \mathsf{SetRec}_\sigma(m)) \mid U \subseteq S \text{ and } m \in M\}$.
- *(Edge labels).* The edges are labeled by actions in A.
- *(Edge set).* Let $U \subseteq S$, $m \in M$ and $a \in \mathrm{Supp}(\sigma_n(m))$. Let $\overline{U} = \bigcup_{s \in U} \mathrm{Supp}(\delta(s, a))$ denote the set of possible successors of states in U given action a. We add the following set of edges in E: Given (U', m') such that there exists $o \in \mathcal{O}$ with $\gamma^{-1}(o) \cap \overline{U} = U'$ and $m' \in \mathrm{Supp}(\sigma_u(m, o, a))$, we add the edge $(U, \mathsf{BoolRec}_\sigma(m), \mathsf{SetRec}_\sigma(m)) \xrightarrow{a} (U', \mathsf{BoolRec}_\sigma(m'), \mathsf{SetRec}_\sigma(m'))$ to E. Intuitively, the update from U to U' is the update of the belief, i.e., if the previous belief is the set U of states, and the current observation

is o, then the new belief is U'; the update of m to m' is according to the support of the memory update function; and the $\mathsf{BoolRec}$ and SetRec functions for the memories are given by σ.

- *(Initial vertex).* The *initial vertex* of $\mathsf{PrGr}(\sigma)$ is the vertex $(\{s_0\}, \mathsf{BoolRec}_\sigma(m_0), \mathsf{SetRec}_\sigma(m_0))$.

Note that $V \subseteq \mathcal{P}(S) \times \{0,1\}^{|S|} \times \mathfrak{D}^{|S|}$, and hence $|V| \leq \mathsf{Mem}^*$. For the rest of the section we fix a finite-memory strategy σ that uses memory M. We now define projected strategies: intuitively the projected strategy collapses memory with same $\mathsf{BoolRec}$ and SetRec functions, and at a collapsed memory state plays uniformly the union of the actions played at the corresponding memory states.

▶ **Definition 9** (Projected strategy $\mathit{proj}(\sigma)$)**.** Let $\mathsf{PrGr}(\sigma) = (V, E)$ be the projection graph of σ. We define the following projected strategy $\sigma' = \mathit{proj}(\sigma) = (\sigma'_u, \sigma'_n, M', m'_0)$:

- *(Memory set).* The memory set of $\mathit{proj}(\sigma)$ is $M' = V = \{(U, \mathsf{BoolRec}_\sigma(m), \mathsf{SetRec}_\sigma(m)) \mid U \subseteq S \text{ and } m \in M\}$.
- *(Initial memory).* The initial memory state of $\mathit{proj}(\sigma)$ is $m'_0 = (\{s_0\}, \mathsf{BoolRec}_\sigma(m_0), \mathsf{SetRec}_\sigma(m_0))$.
- *(Memory update).* Let $m = (U, B, L) \in M'$, $o \in \mathcal{O}$ and $a \in A$. Then $\sigma'_u(m, o, a)$ is the uniform distribution over the set $\{m' = (U', B', L') \in M' \mid m \overset{a}{\to} m' \in E \text{ and } U' \subseteq \gamma^{-1}(o)\}$.
- *(Action selection).* Given $m \in M'$, the action selection function $\sigma'_n(m)$ is the uniform distribution over $\{a \in A \mid \exists m' \in M' \ s.t.\ m \overset{a}{\to} m' \in E\}$.

Let $(V, E) = \mathsf{PrGr}(\sigma)$ be the projection graph, and let $\sigma' = \mathit{proj}(\sigma)$ be the projected strategy. The chain $G \restriction \sigma'$ is a finite-state Markov chain, with state space $S \times M'$, which is a subset of $S \times \mathcal{P}(S) \times \{0,1\}^{|S|} \times \mathfrak{D}^{|S|}$.

Random variable notations. For all $n \geq 0$ we write X_n, Y_n, C_n, Z_n, W_n for the random variables which correspond respectively to the projection of the n-th state of the Markov chain $G \restriction \sigma'$ on the S component, the $\mathcal{P}(S)$ component, the $\{0,1\}^{|S|}$ component, the $\mathfrak{D}^{|S|}$ component, and the n-th action, respectively.

Run of the Markov chain $G \restriction \sigma'$. A *run* on $G \restriction \sigma'$ is a sequence $r = (X_0, Y_0, C_0, Z_0) \overset{W_0}{\to} (X_1, Y_1, C_1, Z_1) \overset{W_1}{\to} \ldots$ such that each finite prefix of r is generated with positive probability on the chain, i.e., for all $i \geq 0$, we have (i) $W_i \in \mathrm{Supp}(\sigma'_n(Y_i, C_i, Z_i))$; (ii) $X_{i+1} \in \mathrm{Supp}(\delta(X_i, W_i))$; and (iii) $(Y_{i+1}, C_{i+1}, Z_{i+1}) \in \mathrm{Supp}(\sigma'_u((Y_i, C_i, Z_i), \gamma(X_{i+1}), W_i))$.

In the following lemma we show that reachability in the Markov chain $G \restriction \sigma$ implies reachability in the Markov chain $G \restriction \sigma'$. Intuitively, the result follows from the fact that the projected strategy σ' plays in the collapsed memory state uniformly all actions that were played at all the corresponding memory states of the original strategy σ.

▶ **Lemma 10.** *Let $\sigma' = \mathit{proj}(\sigma)$ be the projected strategy of σ. Given $s, s' \in S$ and $m, m' \in M$, if (s', m') is reachable from (s, m) in $G \restriction \sigma$, then for all $Y \subseteq S$ such that $(s, Y, \mathsf{BoolRec}_\sigma(m), \mathsf{SetRec}_\sigma(m))$ is a state of $G \restriction \sigma'$, there exists $Y' \subseteq S$ such that $(s', Y', \mathsf{BoolRec}_\sigma(m'), \mathsf{SetRec}_\sigma(m'))$ is reachable from $(s, Y, \mathsf{BoolRec}_\sigma(m), \mathsf{SetRec}_\sigma(m))$ in $G \restriction \sigma'$.*

Proof. Suppose first that (s', m') is reachable from (s, m) in $G \restriction \sigma$ in one step. Let $Y \subseteq S$ be such that $(s, Y, \mathsf{BoolRec}_\sigma(m), \mathsf{SetRec}_\sigma(m))$ is a state of $G \restriction \sigma'$. Then there exists an edge in the projection graph of σ from $(Y, \mathsf{BoolRec}_\sigma(m), \mathsf{SetRec}_\sigma(m))$ to another vertex $(Y', \mathsf{BoolRec}_\sigma(m'), \mathsf{SetRec}_\sigma(m'))$. As a consequence, there exists $Y' \subseteq S$ such that $(s', Y', \mathsf{BoolRec}_\sigma(m'), \mathsf{SetRec}_\sigma(m'))$ is reachable from $(s, Y, \mathsf{BoolRec}_\sigma(m), \mathsf{SetRec}_\sigma(m))$ in $G \restriction \sigma'$.

We conclude the proof by induction: if (s', m') is reachable from (s, m) in $G \restriction \sigma$, then there exists a sequence of couples $(s_1, m_1), (s_2, m_2), ..., (s_i, m_i)$ such that $(s_1, m_1) = (s, m)$, $(s_i, m_i) = (s', m')$, and for all $j \in \{1, ..., i-1\}$ we have that (s_{j+1}, m_{j+1}) is reachable from (s_j, m_j) in one step. Using the proof for an elementary step (or one step) inductively on such a sequence, we get the result. ◀

In the following lemma we establish the crucial properties of the Markov chain obtained from the projected strategy.

▶ **Lemma 11.** *Let $X_0 \in S$, $Y_0 \in \mathcal{P}(S)$, $C_0 \in \{0,1\}^{|S|}$ and $Z_0 \in \mathfrak{D}^{|S|}$, and let $r = (X_0, Y_0, C_0, Z_0) \overset{W_0}{\to} (X_1, Y_1, C_1, Z_1) \overset{W_1}{\to} \ldots$ be a run on $G \restriction \sigma'$ with a starting state (X_0, Y_0, C_0, Z_0). Then for all $n \geq 0$ the following assertions hold:*

- $X_{n+1} \in \mathrm{Supp}(\delta(X_n, W_n))$.
- *$Z_n(X_n)$ is not empty.*
- $Z_{n+1}(X_{n+1}) \subseteq Z_n(X_n)$.
- *$(Y_n, C_n, Z_n) \overset{W_n}{\to} (Y_{n+1}, C_{n+1}, Z_{n+1})$ is an edge in E, where $(V, E) = \mathsf{PrGr}(\sigma)$.*
- *If $C_n(X_n) = 1$, then $C_{n+1}(X_{n+1}) = 1$.*
- *If $C_n(X_n) = 1$, then $|Z_n(X_n)| = 1$; and if $\{Z\} = Z_n(X_n)$, then for all $j \geq 0$ we have $\mathsf{col}(X_{n+j}) \in Z$.*

Proof. We prove the last point. Suppose (X_n, Y_n, C_n, Z_n) is such that $C_n(X_n) = 1$. Let $m \in M$ be an arbitrary memory state such that $C_n = \mathsf{BoolRec}_\sigma(m)$ and $Z_n = \mathsf{SetRec}_\sigma(m)$. By hypothesis, since $C_n(X_n) = 1$, it follows that (X_n, m) is a recurrent state in the Markov chain $G \restriction \sigma$. As a consequence, only one recurrent class $R \subseteq S \times M$ of $G \restriction \sigma$ is reachable from (X_n, m), and (X_n, m) belongs to this class. Hence $Z_n(X_n) = \{\mathsf{col}(\mathsf{Proj}_1(R))\}$, and thus $|Z_n(X_n)| = 1$. It also follows that all states (X', m') reachable in one step from (X_n, m) also belong to the recurrent class R. It follows that $X_{n+1} \in \mathsf{Proj}_1(R)$ and hence $\mathsf{col}(X_{n+1}) \in \mathsf{col}(\mathsf{Proj}_1(R))$. By induction for all $j \geq 0$ we have $\mathsf{col}(X_{n+j}) \in \mathsf{col}(\mathsf{Proj}_1(R))$. The desired result follows. ◀

We now introduce the final notion that is required to complete the proof. The notion is that of a pseudo-recurrent state. Intuitively a state (X, Y, C, Z) is pseudo-recurrent if Z contains exactly one recurrent subset, X belongs to the subset and it will follow that for some memory $m \in M$ (of certain desired property) (X, m) is a recurrent state in the Markov chain $G \restriction \sigma$. The important property that is useful is that once a pseudo-recurrent state is reached, then C and Z remain invariant (follows from Lemma 11).

▶ **Definition 12** (Pseudo-recurrent states). Let $X \in S$, $Y \subseteq S$, $C \in \{0,1\}^{|S|}$, and $Z \in \mathfrak{D}^{|S|}$. Then the state (X, Y, C, Z) is called *pseudo-recurrent* if there exists $Z_\infty \subseteq D$ such that: (i) $Z(X) = \{Z_\infty\}$, (ii) $\mathsf{col}(X) \in Z_\infty$, and (iii) $C(X) = 1$.

▶ **Lemma 13.** *Let (X, Y, C, Z) be a pseudo-recurrent state. If (X', Y', C', Z') is reachable from (X, Y, C, Z) in $G \restriction \sigma'$, then (X', Y', C', Z') is also a pseudo-recurrent state and $Z'(X') = Z(X)$.*

We establish the following key properties of pseudo-recurrent states with the aid of the properties of Lemma 11. Firstly, with probability 1 a run of a Markov chain $G \restriction \sigma'$ reaches a pseudo-recurrent state.

▶ **Lemma 14.** *Let $X \in S$, $Y \in \mathcal{P}(S)$, $C \in \{0,1\}^{|S|}$, and $Z \in \mathfrak{D}^{|S|}$. Then almost-surely (with probability 1) a run on $G \restriction \sigma'$ from any starting state (X,Y,C,Z) reaches a pseudo-recurrent state.*

Proof. We show that given (X,Y,C,Z) there exists a pseudo-recurrent state (X',Y',C',Z') which is reachable from (X,Y,C,Z) in $G \restriction \sigma'$. First let us consider the Markov chain $G \restriction \sigma$ obtained from the original finite-memory strategy σ with memory M. Let $m \in M$ be such that $C = \mathsf{BoolRec}_{\sigma}(m)$ and $Z = \mathsf{SetRec}_{\sigma}(m)$. We will now show that the result is a consequence of Lemma 10. First we know that there exists $t \in S$ and $m' \in M$ such that (t,m') is recurrent and reachable from (X,m) with positive probability in $G \restriction \sigma$. Let $R \subseteq S \times M$ be the unique recurrent class such that $(t,m') \in R$, and $Z_\infty = \{\mathsf{col}(\mathsf{Proj}_1(R))\}$. By Lemma 10, this implies that from (X,Y,C,Z) we can reach a state (X',Y',C',Z') such that (i) $X' = t$; (ii) $Z'(X') = \{Z_\infty\}$; (iii) $\mathsf{col}(X') \in Z_\infty$; and (iv) $C'(X') = 1$. Hence (X',Y',C',Z') is a pseudo-recurrent state. This shows that from all states with positive probability a pseudo-recurrent state is reached, and since it holds for all states with positive probability, it follows that it holds for all states with probability 1. ◀

Moreover, for every projection Z_B of a reachable recurrent class in the Markov chain $G \restriction \sigma$, there exists a pseudo-recurrent state (X',Y',C',Z') reachable in $G \restriction \sigma'$ such that $Z'(X') = \{Z_B\}$.

▶ **Lemma 15.** *Let (X,Y,C,Z) be a state of $G \restriction \sigma'$, and let $Z_B \in Z(X)$. Then there exists a pseudo-recurrent state (X',Y',C',Z') which is reachable from (X,Y,C,Z) and such that $Z'(X') = \{Z_B\}$.*

Finally, if we consider a pseudo-recurrent state, and consider the projection on the state space of the POMDP G of the recurrent classes reachable and consider the colors, then they coincide with $Z(X)$.

▶ **Lemma 16.** *Let (X,Y,C,Z) be a pseudo-recurrent state, then we have $Z(X) = \mathsf{SetRec}_{\sigma'}(m')(X)$, where $m' = (Y,C,Z)$.*

Proof. Let (X,Y,C,Z) be a pseudo-recurrent state, and let Z_∞ be such that $Z(X) = \{Z_\infty\}$. First, by Lemma 13, we know that if (X',Y',C',Z') is reachable from (X,Y,C,Z) in $G \restriction \sigma'$, then $\mathsf{col}(X') \in Z_\infty$. This implies that for all $Z_B \in \mathsf{SetRec}_{\sigma'}(m')(X)$, where $m' = (Y,C,Z)$, we have $Z_B \subseteq Z_\infty$. Second, by Lemma 10, if (X',Y',C',Z') is reachable from (X,Y,C,Z) in $G \restriction \sigma'$ and $\ell \in Z_\infty$, then there exists (X'',Y'',C'',Z'') reachable from (X',Y',C',Z') such that $\mathsf{col}(X'') = \ell$. This implies that for all $Z_B \in \mathsf{SetRec}_{\sigma'}(m')(X)$, where $m' = (Y,C,Z)$, we have $Z_\infty \subseteq Z_B$. Thus, $\mathsf{SetRec}_{\sigma'}(m')(X) = \{Z_\infty\} = Z(X)$. ◀

With the key properties we prove the main lemma (Lemma 17) which shows that the color sets of the projections of the recurrent classes on the state space of the POMDP coincide for σ and $\sigma' = proj(\sigma)$. Lemma 17 and Lemma 4 yield Theorem 18.

▶ **Lemma 17.** *Consider a finite-memory strategy $\sigma = (\sigma_u, \sigma_n, M, m_0)$ and the projected strategy $\sigma' = proj(\sigma) = (\sigma'_u, \sigma'_n, M', m'_0)$. Then we have $\mathsf{SetRec}_{\sigma'}(m'_0)(s_0) = \mathsf{SetRec}_{\sigma}(m_0)(s_0)$; i.e., the colors of the projections of the recurrent classes of the two strategies on the state space of the POMDP G coincide.*

Proof. For the proof, let $X = s_0$, $Y = \{s_0\}$, $C = \mathsf{BoolRec}_{\sigma}(m_0)$, $Z = \mathsf{SetRec}_{\sigma}(m_0)$. We need to show that $\mathsf{SetRec}_{\sigma'}(m'_0)(X) = Z(X)$, where $m'_0 = (Y,C,Z)$. We show inclusion in both directions.

First inclusion:$(Z(X) \subseteq \mathsf{SetRec}_{\sigma'}(m'_0)(X))$. Let $Z_B \in Z(X)$. By Lemma 15, there exists (X', Y', C', Z') which is reachable in $G \restriction \sigma'$ from (X, Y, C, Z), which is pseudo-recurrent, and such that $Z'(X') = \{Z_B\}$. By Lemma 16, we have $Z'(X') = \mathsf{SetRec}_{\sigma'}(m')(X')$ where $m' = (Y', C', Z')$. By Lemma 7, we have $\mathsf{SetRec}_{\sigma'}(m')(X') \subseteq \mathsf{SetRec}_{\sigma'}(m'_0)(X)$. This proves that $Z_B \in \mathsf{SetRec}_{\sigma'}(m'_0)(X)$.
Second inclusion: $(\mathsf{SetRec}_{\sigma'}(m'_0)(X) \subseteq Z(X))$. Conversely, let $Z_B \in \mathsf{SetRec}_{\sigma'}(m'_0)(X)$. Since $G \restriction \sigma'$ is a finite Markov chain, there exists (X', Y', C', Z') which is reachable from (X, Y, C, Z) in $G \restriction \sigma'$ and such that:

- $\{Z_B\} = \mathsf{SetRec}_{\sigma'}(m')(X')$, where $m' = (Y', C', Z')$.
- For all (X'', Y'', C'', Z'') reachable from (X', Y', C', Z') in $G \restriction \sigma'$ we have $\{Z_B\} = \mathsf{SetRec}_{\sigma'}(m'')(X'')$ where $m'' = (Y'', C'', Z'')$.

The above follows from the following property of a finite Markov chain: given a state s of a finite Markov chain and a recurrent class R reachable from s, from all states t of R the recurrent class reachable from t is R only. The condition is preserved by a projection on colors of states in R. By Lemma 14, there exists a pseudo-recurrent state (X'', Y'', C'', Z'') which is reachable from (X', Y', C', Z', W') in $G \restriction \sigma'$. By Lemma 16, we know that $Z''(X'') = \mathsf{SetRec}_{\sigma'}(m'')(X'')$ where $m'' = (Y'', C'', Z'')$. Since $\mathsf{SetRec}_{\sigma'}(m'')(X'') = \{Z_B\}$, and since by Lemma 11 (third point) we have $Z''(X'') \subseteq Z'(X') \subseteq Z(X)$, we get that $Z_B \in Z(X)$. ◀

▶ **Theorem 18.** *Given a POMDP G and a Muller objective* $\mathsf{Muller}(\mathcal{F})$ *with the set D of colors, if there is a finite-memory almost-sure (resp. positive) winning strategy σ, then the projected strategy proj(σ), with memory of size at most* $\mathsf{Mem}^* = 2^{2\cdot|S|} \cdot |\mathfrak{D}|^{|S|}$ *(where $\mathfrak{D} = \mathcal{P}(\mathcal{P}(D))$), is also an almost-sure (resp. positive) winning strategy.*

Büchi and coBüchi objectives are parity (thus Muller) objectives with 2 priorities (or colors) (i.e., $d = 2$), and from Theorem 18 we obtain an upper bound of $2^{6\cdot|S|}$ on memory size for them. However, applying the result of Theorem 18 for Muller objectives to parity objectives we obtain a double exponential bound. We establish Theorem 19: for item (1), we present a reduction (details in appendix) that for almost-sure (resp. positive) winning given a POMDP with $|S|$ states and a parity objective with $2 \cdot d$ priorities constructs an equivalent POMDP with $d \cdot |S|$ states with coBüchi (resp. Büchi) objectives (and thus applying Theorem 18 we obtain the $2^{3\cdot d\cdot|S|}$ upper bound); and item (2) follows from Example 5 (and [8] for lower bounds for reachability and safety objectives).

▶ **Theorem 19.** *Given a POMDP G and a parity objective* $\mathsf{Parity}(p)$ *with the set D of d priorities, the following assertions hold: (1) If there is a finite-memory almost-sure (resp. positive) winning strategy, then there is an almost-sure (resp. positive) winning strategy with memory of size at most $2^{3\cdot d\cdot|S|}$. (2) Finite-memory almost-sure (resp. positive) winning strategies require exponential memory in general, and belief-based stationary strategies are not sufficient in general for finite-memory almost-sure (resp. positive) winning strategies.*

4 Computational Complexity

We will present an exponential time algorithm to solve almost-sure winning in POMDPs with coBüchi objectives under finite-memory strategies (and our polynomial time reduction for parity objectives to coBüchi objectives for POMDPs allows our results to carry over to parity objectives). The results for positive Büchi is similar. The naive algorithm would be to enumerate over all finite-memory strategies with memory bounded by $2^{6\cdot|S|}$, this leads to an algorithm that runs in double-exponential time. Instead our algorithm consists of two steps: (1) given a POMDP G we first construct a special kind of a POMDP $\widehat{G}$ such that there is a finite-memory winning strategy in G iff there is a randomized memoryless winning strategy in $\widehat{G}$; and (2) then show how to solve the special kind of POMDPs under randomized memoryless strategies in time polynomial in the size of $\widehat{G}$. We introduce the special kind of POMDPs which we call belief-observation POMDPs which satisfy that the current belief is always the set of states with current observation.

▶ **Definition 20.** A POMDP $G = (S, A, \delta, \mathcal{O}, \gamma, s_0)$ is a *belief-observation POMDP* iff for every finite prefix $w = (s_0, a_0, s_1, a_1, \ldots, s_n)$ with the observation sequence $\rho = \gamma(w)$, the belief $\mathcal{B}(\rho)$ is equal to the set of states with the observation $\gamma(s_n)$, i.e., $\mathcal{B}(\rho) = \{s \in S \mid \gamma(s) = \gamma(s_n)\}$.

POMDPs to belief-observation POMDPs. We will construct a belief-observation POMDP $\widehat{G}$ from a POMDP G for almost-sure winning with coBüchi objectives. Since we are interested in coBüchi objectives, for the sequel of this section we will denote by $M = 2^S \times \{0,1\}^{|S|} \times \mathfrak{D}^{|S|}$, i.e., all the possible beliefs $\mathcal{B}$, $\mathsf{BoolRec}$ and SetRec functions (recall that $\mathfrak{D}$ is $\mathcal{P}(\mathcal{P}(\{1,2\}))$ for coBüchi objectives). If there exists a finite-memory almost-sure winning strategy σ, then the projected strategy $\sigma' = \mathit{proj}(\sigma)$ is also a finite-memory almost-sure winning strategy (by Theorem 18) and will use memory $M' \subseteq M$. The size of the constructed POMDP $\widehat{G}$ will be exponential in the size of the original POMDP G and polynomial in the size of the memory set M (and $|M| = 2^{6 \cdot |S|}$ is exponential in the size of the POMDP G). We define the set $M_{\mathsf{coBuchi}} \subseteq M$ as the memory elements, where for all states s in the belief component of the memory, the set $\mathsf{SetRec}(s)$ contains only a set with priority two, i.e., there is no state with priority 1 in the reachable recurrent classes according to SetRec. Formally, $M_{\mathsf{coBuchi}} = \{(Y, B, L) \in M \mid \forall s \in Y, L(s) = \{\{2\}\}\}$. The POMDP $\widehat{G}$ is constructed such that it allows all possible ways that a projected strategy of a finite-memory almost-sure winning strategy could play in G. Informally, since beliefs are part of states of $\widehat{G}$ it is belief-observation; and since possible memory states of projected strategies are part of the state space, we only need to consider memoryless strategies. We will now present a polynomial time algorithm for the computation of the almost-sure winning set for the belief-observation POMDP $\widehat{G}$ with state space $\widehat{S}$ for coBüchi objectives under randomized memoryless strategies.

Almost-sure winning observations. For an objective φ, we denote by $\mathsf{Almost}(\varphi) = \{o \in \mathcal{O} \mid$ there exists a randomized memoryless strategy σ such that for all $s \in \gamma^{-1}(o)$. $\mathbb{P}_s^{\sigma}(\varphi) = 1\}$ the set of observations such that there is a randomized memoryless strategy to ensure winning with probability 1 from all states of the observation. Also note that since we consider belief-observation POMDPs we can only consider beliefs that correspond to all states of an observation.

Almost-sure winning for coBüchi objectives. We show that the computation can be achieved by computing almost-sure winning regions for safety and reachability objectives. The steps of the computation are as follows: *(Step 1).* Let $F \subseteq \widehat{S}$ be the set of states of $\widehat{G}$ where some actions can be played consistent with a projected strategy of a finite-memory strategy, and we first compute $\mathsf{Almost}(\mathsf{Safe}(F))$. *(Step 2).* Let $\widehat{S}_{wpr} \subseteq \widehat{S}$ denote the subset of states that intuitively correspond to *winning pseudo-recurrent (wpr)* states, i.e., formally it is defined as follows: $\widehat{S}_{wpr} = \{(s, (Y, B, L)) \mid B(s) = 1, L(s) = \{\{2\}\}$ and $\widehat{p}(s) = 2\}$. In the POMDP restricted to $\mathsf{Almost}(\mathsf{Safe}(F))$ we compute the set of observations $W_2 = \mathsf{Almost}(\mathsf{Reach}(\widehat{S}_{wpr}))$. We show that $W_2 = \mathsf{Almost}(\mathsf{coBuchi}(\widehat{p}^{-1}(2)))$, and then show that in belief-observation POMDPs almost-sure safety and reachability sets can be computed in polynomial time (and thus obtain Theorem 23).

▶ **Lemma 21.** $\mathsf{Almost}(\mathsf{coBuchi}(\widehat{p}^{-1}(2))) = W_2$.

Proof. We prove the inclusion $W_2 \subseteq \mathsf{Almost}(\mathsf{coBuchi}(\widehat{p}^{-1}(2)))$. Let $o \in W_2$ be an observation in W_2, and we show how to construct a randomized memoryless almost-sure winning strategy ensuring that $o \in \mathsf{Almost}(\mathsf{coBuchi}(\widehat{p}^{-1}(2)))$. Let σ be the strategy produced by the computation of $\mathsf{Almost}(\mathsf{Reach}(\widehat{S}_{wpr}))$. We will show that the same strategy ensures also $\mathsf{Almost}(\mathsf{coBuchi}(\widehat{p}^{-1}(2)))$. As in every observation o the strategy σ plays only a subset of actions that are available in the POMDP restricted to $\mathsf{Almost}(\mathsf{Safe}(F)$ (to ensure safety in F), where $F = \widehat{S} \setminus \widehat{s}_b$, the loosing absorbing state $\widehat{s}_b$ is not reachable. Intuitively, the state $\widehat{s}_b$ is reached whenever a strategy violates the structure of a projected strategy. Also with probability 1 the set $\widehat{S}_{wpr}$ is reached. We show that for all states $(s,(Y,B,L)) \in \widehat{S}_{wpr}$ that all the states reachable from $(s,(Y,B,L))$ have priority 2 according to $\widehat{p}$. Therefore ensuring that all recurrent classes reachable from $\widehat{S}_{wpr}$ have minimal priority 2. In the construction of the POMDP $\widehat{G}$, the only actions allowed in a state $(s,(Y,B,L))$ satisfy that for all states $\widehat{s} \in Y$ if $B(\widehat{s}) = 1$, $L(\widehat{s}) = \{Z_\infty\}$ and $\widehat{p}(s) \in Z_\infty$ for some $Z_\infty \subseteq \{1,2\}$, then for all states $\widehat{s}' \in \mathrm{Supp}(\delta(s,a))$ we have that $p(\widehat{s}') \in Z_\infty$. As all states in $(s,(Y,B,L)) \in \widehat{S}_{wpr}$ have $L(s) = \{\{2\}\}$, it follows that any state reachable in the next step has priority 2. Let $(s',Y',(Y,B,L),a)$ be an arbitrary state reachable from $(s,(Y,B,L))$ in one step. By the previous argument we have that the priority $\widehat{p}((s',Y',(Y,B,L),a)) = 2$. Similarly the only allowed memory-update actions (Y',B',L') from state $(s',Y',(Y,B,L),a)$ satisfy that whenever $\widehat{s} \in Y$ and $B(\widehat{s}) = 1$, then for all $\widehat{s}' \in \mathrm{Supp}(\delta(\widehat{s},a))$, we have that $B'(\widehat{s}') = 1$ and similarly we have that $L'(s')$ is a non-empty subset of $L(s)$, i.e., $L'(s') = \{\{2\}\}$. Therefore the next reachable state $(s',(Y',B',L'))$ is again in $\widehat{S}_{wpr}$. In other words, from states $(s,(Y,B,L))$ in $\widehat{S}_{wpr}$ in all future steps only states with priority 2 are visited, i.e., $\mathsf{Safe}(\widehat{p}^{-1}(2))$ is ensured which ensures the coBüchi objective. As the states in $\widehat{S}_{wpr}$ are reached with probability 1 and from them all recurrent classes reachable have only states that have priority 2, the desired result follows. ◀

▶ **Lemma 22.** *For $T \subseteq S$ and $F \subseteq S$, the set $Y^* = \mathsf{Almost}(\mathsf{Safe}(F))$ can be computed in linear time; and the set $Z^* = \mathsf{Almost}(\mathsf{Buchi}(T))$ and $\mathsf{Almost}(\mathsf{Reach}(T))$ can be computed in quadratic time for belief-observation POMDPs.*

▶ **Theorem 23.** *(1) Given a POMDP G with $|S|$ states and a parity objective with d priorities, the decision problem of the existence (and the construction if one exists) of a finite-memory almost-sure (resp. positive) winning strategy can be solved in $2^{O(|S|\cdot d)}$ time. (2) The decision problem of given a POMDP and a parity objective whether there exists a finite-memory almost-sure (resp. positive) winning strategy is EXPTIME-complete.*

Concluding remarks. Our EXPTIME-algorithm for parity objectives, and the LAR reduction of Muller objectives to parity objectives [15] give an $2^{O(d!\cdot d^2\cdot|S|)}$ time algorithm for Muller objectives with d colors for POMDPs with $|S|$ states, i.e., exponential in $|S|$ and double exponential in d. Note that the Muller objective specified by the set $\mathcal{F}$ maybe in general itself double exponential in d.

Acknowledgement. We thank Sumit Nain and Moshe Vardi for sharing their work with us.

References

1 C. Baier, N. Bertrand, and M. Größer. On decision problems for probabilistic Büchi automata. In *FoSSaCS*, LNCS 4962, pages 287–301. Springer, 2008.

2 C. Baier, M. Größer, and N. Bertrand. Probabilistic omega-automata. *J. ACM*, 59(1), 2012.

3 C. Baier and J-P. Katoen. *Principles of Model Checking.* MIT Press, 2008.

4 A. Bianco and L. de Alfaro. Model checking of probabilistic and nondeterministic systems. In *FSTTCS 95*, volume 1026 of *LNCS*, pages 499–513. Springer-Verlag, 1995.

5 P. Billingsley, editor. *Probability and Measure.* Wiley-Interscience, 1995.

6 P. Cerný, K. Chatterjee, T. A. Henzinger, A. Radhakrishna, and R. Singh. Quantitative synthesis for concurrent programs. In *Proc. of CAV*, LNCS 6806, pages 243–259. Springer, 2011.

7 K. Chatterjee, L. Doyen, H. Gimbert, and T. A. Henzinger. Randomness for free. In *MFCS*, 2010.

8 K. Chatterjee, L. Doyen, and T. A. Henzinger. Qualitative analysis of partially-observable Markov decision processes. In *MFCS*, pages 258–269, 2010.

9 A. Condon and R. J. Lipton. On the complexity of space bounded interactive proofs. In *FOCS*, pages 462–467, 1989.

10 C. Courcoubetis and M. Yannakakis. The complexity of probabilistic verification. *Journal of the ACM*, 42(4):857–907, 1995.

11 L. de Alfaro, M. Faella, R. Majumdar, and V. Raman. Code-aware resource management. In *EMSOFT 05.* ACM, 2005.

12 R. Durbin, S. Eddy, A. Krogh, and G. Mitchison. *Biological sequence analysis: probabilistic models of proteins and nucleic acids.* Cambridge Univ. Press, 1998.

13 J. Filar and K. Vrieze. *Competitive Markov Decision Processes.* Springer-Verlag, 1997.

14 H. Gimbert and Y. Oualhadj. Probabilistic automata on finite words: Decidable and undecidable problems. In *Proc. of ICALP*, LNCS 6199, pages 527–538. Springer, 2010.

15 Y. Gurevich and L. Harrington. Trees, automata, and games. In *STOC'82*, pages 60–65, 1982.

16 H. Howard. *Dynamic Programming and Markov Processes.* MIT Press, 1960.

17 H. Kress-Gazit, G. E. Fainekos, and G. J. Pappas. Temporal-logic-based reactive mission and motion planning. *IEEE Transactions on Robotics*, 25(6):1370–1381, 2009.

18 M. Kwiatkowska, G. Norman, and D. Parker. PRISM: Probabilistic symbolic model checker. In *TOOLS' 02*, pages 200–204. LNCS 2324, Springer, 2002.

19 O. Madani, S. Hanks, and A. Condon. On the undecidability of probabilistic planning and related stochastic optimization problems. *Artif. Intell.*, 147(1-2):5–34, 2003.

20 N. Meuleau, K-E. Kim, L. P. Kaelbling, and A.R. Cassandra. Solving pomdps by searching the space of finite policies. In *UAI*, pages 417–426, 1999.

21 M. Mohri. Finite-state transducers in language and speech processing. *Computational Linguistics*, 23(2):269–311, 1997.

22 C. H. Papadimitriou and J. N. Tsitsiklis. The complexity of Markov decision processes. *Mathematics of Operations Research*, 12:441–450, 1987.

23 A. Paz. *Introduction to probabilistic automata.* Academic Press, 1971.

24 A. Pogosyants, R. Segala, and N. Lynch. Verification of the randomized consensus algorithm of Aspnes and Herlihy: a case study. *Distributed Computing*, 13(3):155–186, 2000.

25 M.O. Rabin. Probabilistic automata. *Information and Control*, 6:230–245, 1963.

26 J. H. Reif. The complexity of two-player games of incomplete information. *JCSS*, 29, 1984.

27 M.I.A. Stoelinga. Fun with FireWire: Experiments with verifying the IEEE1394 root contention protocol. In *Formal Aspects of Computing*, 2002.

28 W. Thomas. Languages, automata, and logic. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume 3, Beyond Words, chapter 7, pages 389–455. Springer, 1997.