Limit Synchronization in Markov Decision Processes*

Laurent Doyen[1], Thierry Massart[2], and Mahsa Shirmohammadi[1,2]

[1] LSV, ENS Cachan & CNRS, France
[2] Université Libre de Bruxelles, Belgium

Abstract. Markov decision processes (MDP) are finite-state systems with both strategic and probabilistic choices. After fixing a strategy, an MDP produces a sequence of probability distributions over states. The sequence is eventually synchronizing if the probability mass accumulates in a single state, possibly in the limit. Precisely, for $0 \leq p \leq 1$ the sequence is p-synchronizing if a probability distribution in the sequence assigns probability at least p to some state, and we distinguish three synchronization modes: (*i*) *sure* winning if there exists a strategy that produces a 1-synchronizing sequence; (*ii*) *almost-sure* winning if there exists a strategy that produces a sequence that is, for all $\epsilon > 0$, a (1-ϵ)-synchronizing sequence; (*iii*) *limit-sure* winning if for all $\epsilon > 0$, there exists a strategy that produces a (1-ϵ)-synchronizing sequence. We consider the problem of deciding whether an MDP is sure, almost-sure, or limit-sure winning, and we establish the decidability and optimal complexity for all modes, as well as the memory requirements for winning strategies. Our main contributions are as follows: (a) for each winning modes we present characterizations that give a PSPACE complexity for the decision problems, and we establish matching PSPACE lower bounds; (b) we show that for sure winning strategies, exponential memory is sufficient and may be necessary, and that in general infinite memory is necessary for almost-sure winning, and unbounded memory is necessary for limit-sure winning; (c) along with our results, we establish new complexity results for alternating finite automata over a one-letter alphabet.

1 Introduction

Markov decision processes (MDP) are finite-state stochastic processes used in the design of systems that exhibit both controllable and stochastic behavior, such as in planning, randomized algorithms, and communication protocols [2,13,5]. The controllable choices along the execution are fixed by a strategy, and the stochastic choices describe the system response. When a strategy is fixed in an MDP, the *symbolic* semantics is a sequence of probability distributions over states of the MDP, which differs from the *traditional* semantics where a probability measure is considered over sets of sequences of states. This semantics is

* This work has been partly supported by the Belgian Fonds National de la Recherche Scientifique (FNRS).

adequate in many applications, such as systems biology, sensor networks, robot planning, etc. [15,6], where the system consists of several copies of the same process (molecules, sensors, robots, etc.), and the relevant information along the execution of the system is the number of processes in each state, or the relative frequency (i.e., the probability) of each state. In recent works, the verification of quantitative properties of the symbolic semantics was shown undecidable [18]. Decidability is obtained for special subclasses [7], or through approximations [1].

In this paper, we consider a general class of strategies that select actions depending on the full history of the system execution. In the context of several identical processes, the same strategy is used in every process, but the internal state of each process need not be the same along the execution, since probabilistic transitions may have different outcome in each process. Therefore, the execution of the system is best described by the sequence of probability distributions over states along the execution. Previously, the special case of word-strategies have been considered, that at each step select the same control action in all states, and thus only depend on the number of execution steps of the system. Several problems for MDPs with word-strategies (also known as probabilistic automata) are undecidable [4,14,18,12]. In particular the limit-sure reachability problem, which is to decide whether a given state can be reached with probability arbitrarily close to one, is undecidable for probabilistic automata [14].

We establish the decidability and optimal complexity of deciding synchronizing properties for the symbolic semantics of MDPs under general strategies. Synchronizing properties require that the probability distributions tend to accumulate all the probability mass in a single state, or in a set of states. They generalize synchronizing properties of finite automata [21,11]. Formally for $0 \leq p \leq 1$, a sequence $\bar{X} = X_0 X_1 \dots$ of probability distributions $X_i : Q \to [0,1]$ over state space Q of an MDP is *eventually p-synchronizing* if for some $i \geq 0$, the distribution X_i assigns probability at least p to some state. Analogously, it is *always p-synchronizing* if in all distributions X_i, there is a state with probability at least p. For $p = 1$, these definitions are the qualitative analogous for sequences of distributions of the traditional reachability and safety conditions [10]. In particular, an eventually 1-synchronizing sequence witnesses that there is a length ℓ such that all paths of length ℓ in the MDP reach a single state, which is thus reached synchronously no matter the probabilistic choices.

Viewing MDPs as one-player stochastic games, we consider the following traditional winning modes (see also Table 1): (*i*) *sure* winning, if there is a strategy that generates an {eventually, always} 1-synchronizing sequence; (*ii*) *almost-sure* winning, if there exists a strategy that generates a sequence that is, for all $\epsilon > 0$, {eventually, always} $(1-\epsilon)$-synchronizing; (*iii*) *limit-sure* winning, if for all $\epsilon > 0$, there is a strategy that generates an {eventually, always} $(1-\epsilon)$-synchronizing sequence.

We show that the three winning modes form a strict hierarchy for eventually synchronizing: there are limit-sure winning MDPs that are not almost-sure winning, and there are almost-sure winning MDPs that are not sure winning. For always synchronizing, the three modes coincide.

	Always	Eventually
Sure	$\exists \alpha \quad \forall n \ \ \mathcal{M}_n^\alpha(T) = 1$	$\exists \alpha \quad \exists n \ \ \mathcal{M}_n^\alpha(T) = 1$
Almost-sure	$\exists \alpha \ \ \inf_n \mathcal{M}_n^\alpha(T) = 1$	$\exists \alpha \ \ \sup_n \mathcal{M}_n^\alpha(T) = 1$
Limit-sure	$\sup_\alpha \ \inf_n \mathcal{M}_n^\alpha(T) = 1$	$\sup_\alpha \ \sup_n \mathcal{M}_n^\alpha(T) = 1$

Table 1. Winning modes and synchronizing objectives (where $\mathcal{M}_n^\alpha(T)$ denotes the probability that under strategy α, after n steps the MDP $\mathcal{M}$ is in a state of T).

For each winning mode, we consider the problem of deciding if a given initial distribution is winning. We establish the decidability and optimal complexity bounds for all winning modes. Under general strategies, the decision problems have much lower complexity than with word-strategies. We show that all decision problems are decidable, in polynomial time for always synchronizing, and PSPACE-complete for eventually synchronizing. This is also in contrast with almost-sure winning in the traditional semantics of MDPs, which is solvable in polynomial time for both safety and reachability [9]. All complexity results are shown in Table 2.

We complete the picture by providing optimal memory bounds for winning strategies. We show that for sure winning strategies, exponential memory is sufficient and may be necessary, and that in general infinite memory is necessary for almost-sure winning, and unbounded memory is necessary for limit-sure winning.

Some results in this paper rely on insights related to games and alternating automata that are of independent interest. First, the sure-winning problem for eventually synchronizing is equivalent to a two-player game with a synchronized reachability objective, where the goal for the first player is to ensure that a target state is reached after a number of steps that is independent of the strategy of the opponent (and thus this number can be fixed in advance by the first player). This condition is stronger than plain reachability, and while the winner in two-player reachability games can be decided in polynomial time, deciding the winner for synchronized reachability is PSPACE-complete. This result is obtained by turning the synchronized reachability game into a one-letter alternating automaton for which the emptiness problem (i.e., deciding if there exists a word accepted by the automaton) is PSPACE-complete [16,17]. Second, our PSPACE lower bound for the limit-sure winning problem in eventually synchronizing uses a PSPACE-completeness result that we establish for the *universal finiteness problem*, which is to decide, given a one-letter alternating automata, whether from every state the accepted language is finite.

2 Markov Decision Processes and Synchronization

A *probability distribution* over a finite set S is a function $d : S \to [0, 1]$ such that $\sum_{s \in S} d(s) = 1$. The *support* of d is the set $\mathsf{Supp}(d) = \{s \in S \mid d(s) > 0\}$. We denote by $\mathcal{D}(S)$ the set of all probability distributions over S. Given a set $T \subseteq S$, let $d(T) = \sum_{s \in T} d(s)$. For $T \neq \emptyset$, the *uniform distribution* on T assigns probability $\frac{1}{|T|}$ to every state in T. Given $s \in S$, the *Dirac distribution* on s assigns probability 1 to s, and by a slight abuse of notation, we denote it simply by s.

2.1 Markov decision processes

A *Markov decision process* (MDP) $\mathcal{M} = \langle Q, \mathsf{A}, \delta \rangle$ consists of a finite set Q of states, a finite set A of actions, and a probabilistic transition function $\delta : Q \times \mathsf{A} \to \mathcal{D}(Q)$. A state q is *absorbing* if $\delta(q, a)$ is the Dirac distribution on q for all actions $a \in \mathsf{A}$.

We describe the behavior of an MDP as a one-player stochastic game played for infinitely many rounds. Given an initial distribution $\mu_0 \in \mathcal{D}(Q)$, the game starts in the first round in state q with probability $\mu_0(q)$. In each round, the player chooses an action $a \in \mathsf{A}$, and if the game is in state q, the next round starts in the successor state q' with probability $\delta(q, a)(q')$.

Given $q \in Q$ and $a \in \mathsf{A}$, denote by $\mathsf{post}(q, a)$ the set $\mathsf{Supp}(\delta(q, a))$, and given $T \subseteq Q$ let $\mathsf{Pre}(T) = \{q \in Q \mid \exists a \in \mathsf{A} : \mathsf{post}(q, a) \subseteq T\}$ be the set of states from which the player has an action to ensure that the successor state is in T. For $k > 0$, let $\mathsf{Pre}^k(T) = \mathsf{Pre}(\mathsf{Pre}^{k-1}(T))$ with $\mathsf{Pre}^0(T) = T$.

A *path* in $\mathcal{M}$ is an infinite sequence $\pi = q_0 a_0 q_1 a_1 \dots$ such that $q_{i+1} \in \mathsf{post}(q_i, a_i)$ for all $i \geq 0$. A finite prefix $\rho = q_0 a_0 q_1 a_1 \dots q_n$ of a path has length $|\rho| = n$ and last state $\mathsf{Last}(\rho) = q_n$. We denote by $\mathsf{Play}(\mathcal{M})$ and $\mathsf{Pref}(\mathcal{M})$ the set of all paths and finite paths in $\mathcal{M}$ respectively.

For the decision problems considered in this paper, only the support of the probability distributions in the transition function is relevant (i.e., the exact value of the positive probabilities does not matter); therefore, we can encode an MDP as an A-labelled transition system (Q, R) with $R \subseteq Q \times \mathsf{A} \times Q$ such that $(q, a, q') \in R$ is a transition if $q' \in \mathsf{post}(q, a)$.

Strategies. A *randomized strategy* for $\mathcal{M}$ (or simply a strategy) is a function $\alpha : \mathsf{Pref}(\mathcal{M}) \to \mathcal{D}(\mathsf{A})$ that, given a finite path ρ, returns a probability distribution $\alpha(\rho)$ over the action set, used to select a successor state q' of ρ with probability $\sum_{a \in \mathsf{A}} \alpha(\rho)(a) \cdot \delta(q, a)(q')$ where $q = \mathsf{Last}(\rho)$.

A strategy α is *pure* if for all $\rho \in \mathsf{Pref}(\mathcal{M})$, there exists an action $a \in \mathsf{A}$ such that $\alpha(\rho)(a) = 1$; and *memoryless* if $\alpha(\rho) = \alpha(\rho')$ for all ρ, ρ' such that $\mathsf{Last}(\rho) = \mathsf{Last}(\rho')$. We view pure strategies as functions $\alpha : \mathsf{Pref}(\mathcal{M}) \to \mathsf{A}$, and memoryless strategies as functions $\alpha : Q \to \mathcal{D}(\mathsf{A})$, Finally, a strategy α uses *finite-memory* if it can be represented by a finite-state transducer $T = \langle \mathsf{Mem}, m_0, \alpha_u, \alpha_n \rangle$ where Mem is a finite set of modes (the memory of the strategy), $m_0 \in \mathsf{Mem}$ is the initial mode, $\alpha_u : \mathsf{Mem} \times (\mathsf{A} \times Q) \to \mathsf{Mem}$ is an update function, that given the current memory, last action and state updates the memory, and $\alpha_n : \mathsf{Mem} \times Q \to \mathcal{D}(\mathsf{A})$ is a next-move function that selects the probability distribution $\alpha_n(m, q)$ over actions when the current mode is m and the current state of $\mathcal{M}$ is q. For pure strategies, we assume that $\alpha_n : \mathsf{Mem} \times Q \to \mathsf{A}$. The *memory size* of the strategy is the number $|\mathsf{Mem}|$ of modes. For a finite-memory strategy α, let $\mathcal{M}(\alpha)$ be the Markov chain obtained as the product of $\mathcal{M}$ with the transducer defining α. We assume general knowledge of the reader about Markov chains, such as recurrent and transient states, periodicity, and stationary distributions [19].

2.2 Traditional semantics

In the traditional semantics, given an initial distribution $\mu_0 \in \mathcal{D}(Q)$ and a strategy α in an MDP $\mathcal{M}$, a *path-outcome* is a path $\pi = q_0 a_0 q_1 a_1 \dots$ in $\mathcal{M}$ such that $q_0 \in \mathsf{Supp}(\mu_0)$ and $a_i \in \mathsf{Supp}(\alpha(q_0 a_0 \dots q_i))$ for all $i \geq 0$. The probability of a finite prefix $\rho = q_0 a_0 q_1 a_1 \dots q_n$ of π is

$$\mu_0(q_0) \cdot \prod_{j=0}^{n-1} \alpha(q_0 a_0 \dots q_j)(a_j) \cdot \delta(q_j, a_j)(q_{j+1}).$$

We denote by $\mathit{Outcomes}(\mu_0, \alpha)$ the set of all path-outcomes from μ_0 under strategy α. An *event* $\Omega \subseteq \mathsf{Play}(\mathcal{M})$ is a measurable set of paths, and given an initial distribution μ_0 and a strategy α, the probabilities $Pr^{\alpha}(\Omega)$ of events Ω are uniquely defined [20]. In particular, given a set $T \subseteq Q$ of target states, and $k \in \mathbb{N}$, we denote by $\Box T = \{q_0 a_0 q_1 \dots \in \mathsf{Play}(\mathcal{M}) \mid \forall i : q_i \in T\}$ the safety event of always staying in T, by $\Diamond T = \{q_0 a_0 q_1 \dots \in \mathsf{Play}(\mathcal{M}) \mid \exists i : q_i \in T\}$ the event of reaching T, and by $\Diamond^k T = \{q_0 a_0 q_1 \dots \in \mathsf{Play}(\mathcal{M}) \mid q_k \in T\}$ the event of reaching T after exactly k steps. Hence, $\Pr^{\alpha}(\Diamond T)$ is the probability to reach T under strategy α.

We consider the following classical winning modes. Given an initial distribution μ_0 and an event Ω, we say that $\mathcal{M}$ is:

– *sure winning* if there exists a strategy α such that $\mathit{Outcomes}(\mu_0, \alpha) \subseteq \Omega$;
– *almost-sure winning* if there exists a strategy α such that $\Pr^{\alpha}(\Omega) = 1$;
– *limit-sure winning* if $\sup_{\alpha} \Pr^{\alpha}(\Omega) = 1$.

It is known for safety objectives $\Box T$ in MDPs that the three winning modes coincide, and for reachability objectives $\Diamond T$ that an MDP is almost-sure winning if and only if it is limit-sure winning. For both objectives, the set of initial distributions for which an MDP is sure (resp., almost-sure or limit-sure) winning can be computed in polynomial time [9].

2.3 Symbolic semantics

In contrast to the traditional semantics, we consider a symbolic semantics where fixing a strategy in an MDP $\mathcal{M}$ produces a sequence of probability distributions over states defined as follows [18]. Given an initial distribution $\mu_0 \in \mathcal{D}(Q)$ and a strategy α in $\mathcal{M}$, the *symbolic outcome* of $\mathcal{M}$ from μ_0 is the sequence $(\mathcal{M}^{\alpha}_n)_{n \in \mathbb{N}}$ of probability distributions defined by $\mathcal{M}^{\alpha}_k(q) = Pr^{\alpha}(\Diamond^k \{q\})$ for all $k \geq 0$ and $q \in Q$. Hence, $\mathcal{M}^{\alpha}_k$ is the probability distribution over states after k steps under strategy α. Note that $\mathcal{M}^{\alpha}_0 = \mu_0$.

Informally, synchronizing objectives require that the probability of some state (or some group of states) tends to 1 in the sequence $(\mathcal{M}^{\alpha}_n)_{n \in \mathbb{N}}$. Given a set

$T \subseteq Q$, consider the functions $sum_T : \mathcal{D}(Q) \to [0,1]$ and $max_T : \mathcal{D}(Q) \to [0,1]$ that compute $sum_T(X) = \sum_{q \in T} X(q)$ and $max_T(X) = \max_{q \in T} X(q)$. For $f \in \{sum_T, max_T\}$ and $p \in [0,1]$, we say that a probability distribution X is p-synchronized according to f if $f(X) \geq p$, and that a sequence $\bar{X} = X_0 X_1 \dots$ of probability distributions is:

(a) *always p-synchronizing* if X_i is p-synchronized for all $i \geq 0$;
(b) *eventually p-synchronizing* if X_i is p-synchronized for some $i \geq 0$.

For $p = 1$, we view these definitions as the qualitative analogous for sequences of distributions of the traditional safety and reachability conditions for sequences of states [10]. Now, we define the following winning modes. Given an initial distribution μ_0 and a function $f \in \{sum_T, max_T\}$, we say that for the objective of {always, eventually} synchronizing, $\mathcal{M}$ is:

- *sure winning* if there exists a strategy α such that the symbolic outcome of α from μ_0 is {always, eventually} 1-synchronizing according to f;
- *almost-sure winning* if there exists a strategy α such that for all $\epsilon > 0$ the symbolic outcome of α from μ_0 is {always, eventually} $(1-\epsilon)$-synchronizing according to f;
- *limit-sure winning* if for all $\epsilon > 0$, there exists a strategy α such that the symbolic outcome of α from μ_0 is {always, eventually} $(1-\epsilon)$-synchronizing according to f;

We often use $X(T)$ instead of $sum_T(X)$, as in Table 1 where the definitions of the various winning modes and synchronizing objectives for $f = sum_T$ are summarized. In Section 2.4, we present an example to illustrate the definitions.

2.4 Decision problems

For $f \in \{sum_T, max_T\}$ and $\lambda \in \{\text{always}, \text{event(ually)}\}$, the *winning region* $\langle\!\langle 1 \rangle\!\rangle^{\lambda}_{sure}(f)$ is the set of initial distributions such that $\mathcal{M}$ is sure winning for λ-synchronizing (we assume that $\mathcal{M}$ is clear from the context). We define analogously the winning regions $\langle\!\langle 1 \rangle\!\rangle^{\lambda}_{almost}(f)$ and $\langle\!\langle 1 \rangle\!\rangle^{\lambda}_{limit}(f)$. For a singleton $T = \{q\}$ we have $sum_T = max_T$, and we simply write $\langle\!\langle 1 \rangle\!\rangle^{\lambda}_{\mu}(q)$ (where $\mu \in \{\text{sure}, \text{almost}, \text{limit}\}$). We are interested in the algorithmic complexity of the *membership problem*, which is to decide, given a probability distribution μ_0, whether $\mu_0 \in \langle\!\langle 1 \rangle\!\rangle^{\lambda}_{\mu}(f)$. As we show below, it is easy to establish the complexity of the membership problems for always synchronizing, while it is more tricky for eventually synchronizing. The complexity results are summarized in Table 2.

Always synchronizing. We first remark that for always synchronizing, the three winning modes coincide.

Lemma 1. *Let T be a set of states. For all functions $f \in \{max_T, sum_T\}$, we have $\langle\!\langle 1 \rangle\!\rangle^{always}_{sure}(f) = \langle\!\langle 1 \rangle\!\rangle^{always}_{almost}(f) = \langle\!\langle 1 \rangle\!\rangle^{always}_{limit}(f)$.*

	Always		Eventually	
	Complexity	Memory requirement	Complexity	Memory requirement
Sure	PTIME	memoryless	PSPACE-C	exponential
Almost-sure			PSPACE-C	infinite
Limit-sure			PSPACE-C	unbounded

Table 2. Computational complexity of the membership problem, and memory requirement for the strategies (for always synchronizing, the three modes coincide).

Proof. It follows from the definition of winning modes that $\langle\langle 1\rangle\rangle^{always}_{sure}(f) \subseteq \langle\langle 1\rangle\rangle^{always}_{almost}(f) \subseteq \langle\langle 1\rangle\rangle^{always}_{limit}(f)$. Hence it suffices to show that $\langle\langle 1\rangle\rangle^{always}_{limit}(f) \subseteq \langle\langle 1\rangle\rangle^{always}_{sure}(f)$, that is for all μ_0, if $\mathcal{M}$ is limit-sure always synchronizing from μ_0, then $\mathcal{M}$ is sure always synchronizing from μ_0. For $f = max_T$, consider ϵ smaller than the smallest positive probability in the initial distribution μ_0 and in the transitions of the MDP $\mathcal{M} = \langle Q, \mathsf{A}, \delta\rangle$. Then, given an always $(1-\epsilon)$-synchronizing strategy, it is easy to show by induction on k that the distributions $\mathcal{M}^\alpha_k$ are Dirac for all $k \geq 0$. In particular μ_0 is Dirac, and let $q_0 \in T$ be such that $\mu_0(q_0) = 1$. It follows that there is an infinite path from q_0 in the graph $\langle T, E\rangle$ where $(q, q') \in E$ if there exists an action $a \in \mathsf{A}$ such that $\delta(q,a)(q') = 1$. The existence of this path entails that there is a loop reachable from q_0 in the graph $\langle T, E\rangle$, and this naturally defines a sure-winning always synchronizing strategy in $\mathcal{M}$. A similar argument for $f = sum_T$ shows that for sufficiently small ϵ, an always $(1-\epsilon)$-synchronizing strategy α must produce a sequence of distributions with support contained in T, until some support repeats in the sequence. This naturally induces an always 1-synchronizing strategy. □

It follows from the proof of Lemma 1 that the winning region for always synchronizing according to sum_T coincides with the set of winning initial distributions for the safety objective $\Box T$ in the traditional semantics, which can be computed in polynomial time [8]. Moreover, always synchronizing according to max_T is equivalent to the existence of an infinite path staying in T in the transition system $\langle Q, R\rangle$ of the MDP restricted to transitions $(q, a, q') \in R$ such that $\delta(q,a)(q') = 1$, which can also be decided in polynomial time. In both cases, pure memoryless strategies are sufficient.

Theorem 1. *The membership problem for always synchronizing can be solved in polynomial time, and pure memoryless strategies are sufficient.*

Eventually synchronizing. For all functions $f \in \{max_T, sum_T\}$, the following inclusions hold: $\langle\langle 1\rangle\rangle^{event}_{sure}(f) \subseteq \langle\langle 1\rangle\rangle^{event}_{almost}(f) \subseteq \langle\langle 1\rangle\rangle^{event}_{limit}(f)$ and we show that the inclusions are strict in general.

Lemma 2. *There exists an MDP $\mathcal{M}$ and states q_1, q_2 such that (i) $\langle\langle 1\rangle\rangle^{event}_{sure}(q_1) \subsetneq \langle\langle 1\rangle\rangle^{event}_{almost}(q_1)$, and (ii) $\langle\langle 1\rangle\rangle^{event}_{almost}(q_2) \subsetneq \langle\langle 1\rangle\rangle^{event}_{limit}(q_2)$.*

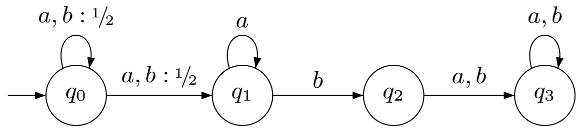

Fig. 1. An MDP $\mathcal{M}$ such that $\langle\!\langle 1 \rangle\!\rangle^{event}_{sure}(q_1) \neq \langle\!\langle 1 \rangle\!\rangle^{event}_{almost}(q_1)$ and $\langle\!\langle 1 \rangle\!\rangle^{event}_{almost}(q_2) \neq \langle\!\langle 1 \rangle\!\rangle^{event}_{limit}(q_2)$.

Proof. Consider the MDP $\mathcal{M}$ with states q_0, q_1, q_2, q_3 and actions a, b as shown in Fig. 1. All transitions are deterministic except from q_0 where on all actions, the successor is q_0 or q_1 with probability $\frac{1}{2}$. Let the initial distribution μ_0 be a Dirac distribution on q_0.

To establish (i), we show that $\mu_0 \in \langle\!\langle 1 \rangle\!\rangle^{event}_{almost}(q_1)$ and $\mu_0 \not\in \langle\!\langle 1 \rangle\!\rangle^{event}_{sure}(q_1)$. To prove that $\mu_0 \in \langle\!\langle 1 \rangle\!\rangle^{event}_{almost}(q_1)$, consider the pure strategy that always plays a. The outcome is such that the probability to be in q_1 after k steps is $1 - \frac{1}{2^k}$, showing that $\mathcal{M}$ is almost-sure winning for the eventually synchronizing objective in q_1 (from μ_0). On the other hand, $\mu_0 \not\in \langle\!\langle 1 \rangle\!\rangle^{event}_{sure}(q_1)$ because for all strategies α, the probability in q_0 remains always positive, and thus in q_1 we have $\mathcal{M}^{\alpha}_n(q_1) < 1$ for all $n \geq 0$, showing that $\mathcal{M}$ is not sure winning for the eventually synchronizing objective in q_1 (from μ_0).

To establish (ii), for all $k \geq 0$ consider a strategy that plays a for k steps, and then plays b. Then the probability to be in q_2 after $k+1$ steps is $1 - \frac{1}{2^k}$, showing that this strategy is eventually $(1 - \frac{1}{2^k})$-synchronizing in q_2. Hence, $\mathcal{M}$ is limit-sure winning for the eventually synchronizing objective in q_2 (from μ_0). Second, for all strategies, since the probability in q_0 remains always positive, the probability in q_2 is always smaller than 1. Moreover, if the probability p in q_2 is positive after n steps $(p > 0)$, then after any number $m > n$ of steps, the probability in q_2 is bounded by $1-p$. It follows that the probability in q_2 is never equal to 1 and cannot tend to 1 for $m \to \infty$, showing that $\mathcal{M}$ is not almost-sure winning for the eventually synchronizing objective in q_2 (from μ_0). $\square$

The rest of this paper is devoted to the solution of the membership problem for eventually synchronizing. We make some preliminary remarks to show that it is sufficient to solve the membership problem according to $f = sum_T$ and for MDPs with a single initial state. Our results will also show that pure strategies are sufficient in all modes.

Remark. For eventually synchronizing and each winning mode, we show that the membership problem with function max_T is polynomial-time equivalent to the membership problem with function $sum_{T'}$ with a singleton T'. First, for $\mu \in \{\text{sure, almost, limit}\}$, we have $\langle\!\langle 1 \rangle\!\rangle^{event}_{\mu}(max_T) = \bigcup_{q \in T} \langle\!\langle 1 \rangle\!\rangle^{event}_{\mu}(q)$, showing that the membership problems for max are polynomial-time reducible to the corresponding membership problem for sum_T with singleton T. The reverse reduction is as follows. Given an MDP $\mathcal{M}$, a state q and an initial distribution μ_0, we can construct an MDP $\mathcal{M}'$ and initial distribution μ'_0 such that

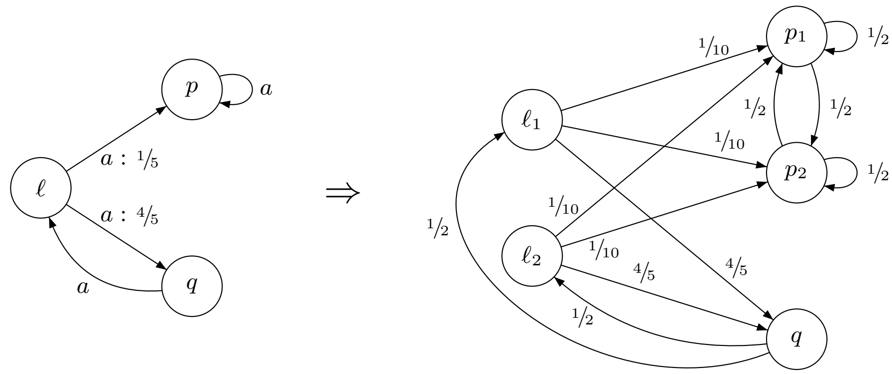

Fig. 2. State duplication ensures that the probability mass can never be accumulated in a single state except in q (we omit action a for readability).

$\mu_0 \in \langle\!\langle 1 \rangle\!\rangle_{\mu}^{event}(q)$ iff $\mu_0' \in \langle\!\langle 1 \rangle\!\rangle_{\mu}^{event}(max_{Q'})$ where Q' is the state space of $\mathcal{M}'$. The idea is to construct $\mathcal{M}'$ and μ_0' as a copy of $\mathcal{M}$ and μ_0 where all states except q are duplicated, and the initial and transition probabilities are equally distributed between the copies (see Fig. 2). Therefore if the probability tends to 1 in some state, it has to be in q.

Remark. To solve the membership problems for eventually synchronizing with function sum_T, it is sufficient to provide an algorithm that decides membership of Dirac distributions (i.e., assuming MDPs have a single initial state), since to solve the problem for an MDP $\mathcal{M}$ with initial distribution μ_0, we can equivalently solve it for a copy of $\mathcal{M}$ with a new initial state q_0 from which the successor distribution on all actions is μ_0. Therefore, it is sufficient to consider initial Dirac distributions μ_0.

3 One-Letter Alternating Automata

In this section, we consider *one-letter alternating automata* (1L-AFA) as they have a structure of alternating graph analogous to MDP (i.e., when ignoring the probabilities). We review classical decision problems for 1L-AFA, and establish the complexity of a new problem, the *universal finiteness problem* which is to decide if from every initial state the language of a given 1L-AFA is finite. These results of independent interest are useful to establish the PSPACE lower bounds for eventually synchronizing in MDPs.

One-letter alternating automata. Let $\mathsf{B}^+(Q)$ be the set of positive Boolean formulas over Q, i.e. Boolean formulas built from elements in Q using $\wedge$ and $\vee$. A set $S \subseteq Q$ *satisfies* a formula $\varphi \in \mathsf{B}^+(Q)$ (denoted $S \models \varphi$) if φ is satisfied when replacing in φ the elements in S by true, and the elements in $Q \setminus S$ by false.

A *one-letter alternating finite automaton* is a tuple $\mathcal{A} = \langle Q, \delta_{\mathcal{A}}, \mathcal{F} \rangle$ where Q is a finite set of states, $\delta_{\mathcal{A}} : Q \to \mathsf{B}^{+}(Q)$ is the transition function, and $\mathcal{F} \subseteq Q$ is the set of accepting states. We assume that the formulas in transition function are in disjunctive normal form. Note that the alphabet of the automaton is omitted, as it has a single letter. In the language of a 1L-AFA, only the length of words is relevant. For all $n \geq 0$, define the set $Acc_{\mathcal{A}}(n, \mathcal{F}) \subseteq Q$ of states from which the word of length n is accepted by $\mathcal{A}$ as follows:

- $Acc_{\mathcal{A}}(0, \mathcal{F}) = \mathcal{F}$;
- $Acc_{\mathcal{A}}(n, \mathcal{F}) = \{q \in Q \mid Acc_{\mathcal{A}}(n-1, \mathcal{F}) \models \delta(q)\}$ for all $n > 0$.

The set $\mathcal{L}(\mathcal{A}_q) = \{n \in \mathbb{N} \mid q \in Acc_{\mathcal{A}}(n, \mathcal{F})\}$ is the *language* accepted by $\mathcal{A}$ from initial state q.

For fixed n, we view $Acc_{\mathcal{A}}(n, \cdot)$ as an operator on 2^{Q} that, given a set $\mathcal{F} \subseteq Q$ computes the set $Acc_{\mathcal{A}}(n, \mathcal{F})$. Note that $Acc_{\mathcal{A}}(n, \mathcal{F}) = Acc_{\mathcal{A}}(1, Acc_{\mathcal{A}}(n-1, \mathcal{F}))$ for all $n \geq 1$. Denote by $\mathsf{Pre}_{\mathcal{A}}(\cdot)$ the operator $Acc_{\mathcal{A}}(1, \cdot)$. Then for all $n \geq 0$ the operator $Acc_{\mathcal{A}}(n, \cdot)$ coincides with $\mathsf{Pre}^{n}_{\mathcal{A}}(\cdot)$, the n-th iterate of $\mathsf{Pre}_{\mathcal{A}}(\cdot)$.

Decision problems. We present classical decision problems for alternating automata, namely the emptiness and finiteness problems, and we introduce a variant of the finiteness problem that will be useful for solving synchronizing problems for MDPs.

- The *emptiness problem* for 1L-AFA is to decide, given a 1L-AFA $\mathcal{A}$ and an initial state q, whether $\mathcal{L}(\mathcal{A}_q) = \emptyset$. The emptiness problem can be solved by checking whether $q \in \mathsf{Pre}^{n}_{\mathcal{A}}(\mathcal{F})$ for some $n \geq 0$. It is known that the emptiness problem is PSPACE-complete, even for transition functions in disjunctive normal form [16,17].
- The *finiteness problem* is to decide, given a 1L-AFA $\mathcal{A}$ and an initial state q, whether $\mathcal{L}(\mathcal{A}_q)$ is finite. The finiteness problem can be solved in (N)PSPACE by guessing $n, k \leq 2^{|Q|}$ such that $\mathsf{Pre}^{n+k}_{\mathcal{A}}(\mathcal{F}) = \mathsf{Pre}^{n}_{\mathcal{A}}(\mathcal{F})$ and $q \in \mathsf{Pre}^{n}_{\mathcal{A}}(\mathcal{F})$. The finiteness problem is PSPACE-complete by a simple reduction from the emptiness problem: from an instance $(\mathcal{A}, q)$ of the emptiness problem, construct $(\mathcal{A}', q')$ where $q' = q$ and $\mathcal{A}' = \langle Q, \delta', \mathcal{F} \rangle$ is a copy of $\mathcal{A} = \langle Q, \delta, \mathcal{F} \rangle$ with a self-loop on q (formally, $\delta'(q) = q \vee \delta(q)$ and $\delta'(r) = \delta(r)$ for all $r \in Q \setminus \{q\}$). It is easy to see that $\mathcal{L}(\mathcal{A}_q) = \emptyset$ iff $\mathcal{L}(\mathcal{A}'_{q'})$ is finite.
- The *universal finiteness problem* is to decide, given a 1L-AFA $\mathcal{A}$, whether $\mathcal{L}(\mathcal{A}_q)$ is finite for all states q. This problem can be solved by checking whether $\mathsf{Pre}^{n}_{\mathcal{A}}(\mathcal{F}) = \emptyset$ for some $n \leq 2^{|Q|}$, and thus it is in PSPACE. Note that if $\mathsf{Pre}^{n}_{\mathcal{A}}(\mathcal{F}) = \emptyset$, then $\mathsf{Pre}^{m}_{\mathcal{A}}(\mathcal{F}) = \emptyset$ for all $m \geq n$.

Given the PSPACE-hardness proofs of the emptiness and finiteness problems, it is not easy to see that the universal finiteness problem is PSPACE-hard.

Lemma 3. *The universal finiteness problem for 1L-AFA is PSPACE-hard.*

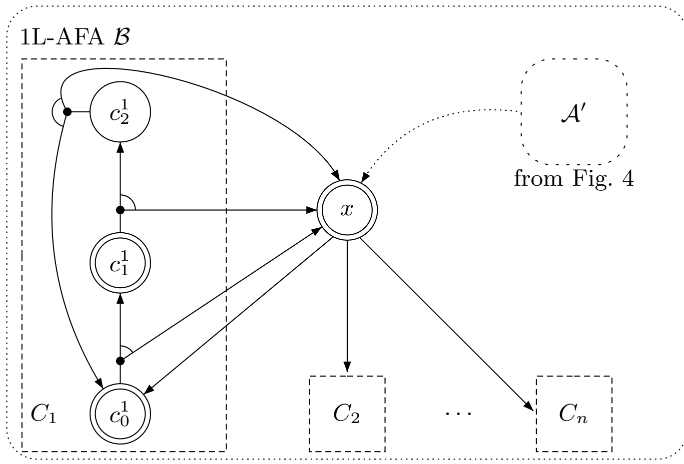

Fig. 3. Sketch of reduction to show PSPACE-hardness of the universal finiteness problem for 1l-AFA.

Proof. The proof is by a reduction from the emptiness problem for 1L-AFA, which is PSPACE-complete [16,17]. The language of a 1L-AFA $\mathcal{A} = \langle Q, \delta, \mathcal{F} \rangle$ is non-empty if $q_0 \in \mathsf{Pre}^i_{\mathcal{A}}(\mathcal{F})$ for some $i \geq 0$. Since the sequence $\mathsf{Pre}^i_{\mathcal{A}}(\mathcal{F})$ is ultimately periodic, it is sufficient to compute $\mathsf{Pre}^i_{\mathcal{A}}(\mathcal{F})$ for all $i \leq 2^{|Q|}$.

From $\mathcal{A}$, we construct a 1L-AFA $B = \langle Q', \delta', \mathcal{F}' \rangle$ with set $\mathcal{F}'$ of accepting states such that the sequence $\mathsf{Pre}^i_B(\mathcal{F}')$ in B mimics the sequence $\mathsf{Pre}^i_{\mathcal{A}}(\mathcal{F})$ in $\mathcal{A}$ for $2^{|Q|}$ steps. The automaton B contains the state space of $\mathcal{A}$, i.e. $Q \subseteq Q'$. The goal is to have $\mathsf{Pre}^i_B(\mathcal{F}') \cap Q = \mathsf{Pre}^i_{\mathcal{A}}(\mathcal{F})$ for all $i \leq 2^{|Q|}$, as long as $q_0 \notin \mathsf{Pre}^i_{\mathcal{A}}(\mathcal{F})$. Moreover, if $q_0 \in \mathsf{Pre}^i_{\mathcal{A}}(\mathcal{F})$ for some $i \geq 0$, then $\mathsf{Pre}^j_B(\mathcal{F}')$ will contain q_0 for all $j \geq i$ (the state q_0 has a self-loop in B), and if $q_0 \notin \mathsf{Pre}\mathcal{A}^i(\mathcal{F})$ for all $i \geq 0$, then B is constructed such that $\mathsf{Pre}^j_B(\mathcal{F}') = \emptyset$ for sufficiently large j (roughly for $j > 2^{|Q|}$). Hence, the language of $\mathcal{A}$ is non-empty if and only if the sequence $\mathsf{Pre}^j_B(\mathcal{F}')$ is not ultimately empty, that is if and only if the language of B is infinite from some state (namely q_0).

The key is to let B simulate $\mathcal{A}$ for exponentially many steps, and to ensure that the simulation stops if and only if q_0 is not reached within $2^{|Q|}$ steps. We achieve this by defining B as the gadget in Fig. 3 connected to a modified copy $\mathcal{A}'$ of $\mathcal{A}$ with the same state space. The transitions in $\mathcal{A}'$ are defined as follows, where x is the entry state of the gadget (see Fig. 4): for all $q \in Q$ let (i) $\delta_B(q) = x \wedge \delta_{\mathcal{A}}(q)$ if $q \neq q_0$, and (ii) $\delta_B(q_0) = q_0 \vee (x \wedge \delta_{\mathcal{A}}(q_0))$. Thus, q_0 has a self-loop, and given a set $S \subseteq Q$ in the automaton $\mathcal{A}$, if $q_0 \notin S$, then $\mathsf{Pre}_{\mathcal{A}}(S) = \mathsf{Pre}_B(S \cup \{x\})$ that is Pre_B mimics $\mathsf{Pre}_{\mathcal{A}}$ when x is in the argument (and q_0 has not been reached yet). Note that if $x \notin S$ (and $q_0 \notin S$), then $\mathsf{Pre}_B(S) = \emptyset$, that is unless q_0 has been reached, the simulation of $\mathcal{A}$ by B stops. Since we need that B mimics $\mathcal{A}$ for $2^{|Q|}$ steps, we define the gadget and the

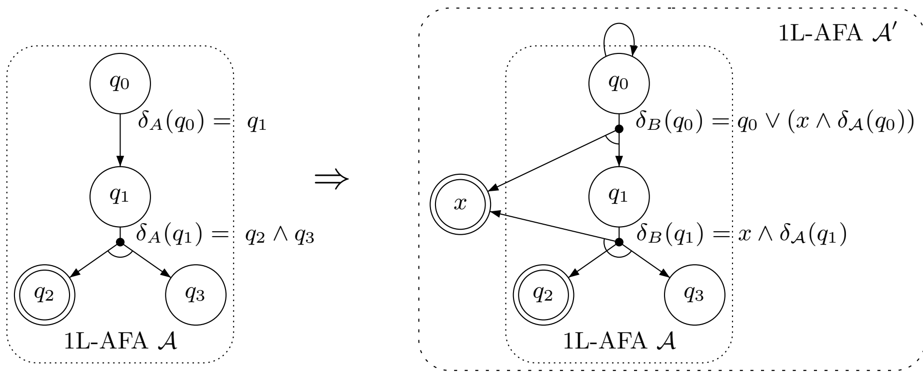

Fig. 4. Detail of the copy $\mathcal{A}'$ obtained from $\mathcal{A}$ in the reduction of Fig. 3.

set $\mathcal{F}'$ to ensure that $x \in \mathcal{F}'$ and if $x \in \mathsf{Pre}_B^i(\mathcal{F}')$, then $x \in \mathsf{Pre}_B^{i+1}(\mathcal{F}')$ for all $i \leq 2^{|Q|}$.

In the gadget, the state x has nondeterministic transitions $\delta_B(x) = c_0^1 \vee c_0^2 \vee \dots \vee c_0^n$ to n components with state space $C_i = \{c_0^i, \dots, c_{p_i-1}^i\}$ where p_i is the $(i+1)$-th prime number, and the transitions[3] $\delta_B(c_j^i) = x \wedge c_{j+1}^i$ form a loop in each component ($i = 1, \dots, n$). We choose n such that $p_n^\# = \prod_{i=1}^n p_i > 2^{|Q|}$ (take $n = |Q|$). Note that the number of states in the gadget is $1 + \sum_{i=1}^n p_i \in O(n^2 \log n)$ [3] and hence the construction is polynomial in the size of $\mathcal{A}$.

By construction, for all sets S, we have $x \in \mathsf{Pre}_B(S)$ whenever the first state c_0^i of some component C_i is in S, and if $x \in S$, then $c_j^i \in S$ implies $c_{j-1}^i \in \mathsf{Pre}_B(S)$. Thus, if $x \in S$, the operator $\mathsf{Pre}_B(S)$ 'shifts' backward the states in each component; and, x is in the next iteration (i.e., $x \in Pre_B(S)$) as long as $c_0^i \in S$ for some component C_i.

Now, define the set of accepting states $\mathcal{F}'$ in B in such a way that all states c_0^i disappear simultaneously only after $p_n^\#$ iterations. Let $\mathcal{F}' = \mathcal{F} \cup \{x\} \cup \bigcup_{1 \leq i \leq n} (C_i \setminus \{c_{p_i-1}^i\})$, thus $\mathcal{F}'$ contains all states of the gadget except the last state of each component. It is easy to check that, irrespective of the transition relation in $\mathcal{A}$, we have $x \in \mathsf{Pre}_B^i(\mathcal{F}')$ if and only if $0 \leq i < p_n^\#$. Therefore, if $q_0 \in \mathsf{Pre}_{\mathcal{A}}^i(\mathcal{F})$ for some i, then $q_0 \in \mathsf{Pre}_B^j(\mathcal{F}')$ for all $j \geq i$ by the self-loop on q_0. On the other hand, if $q_0 \notin \mathsf{Pre}_{\mathcal{A}}^i(\mathcal{F})$ for all $i \geq 0$, then since $x \notin \mathsf{Pre}_B^i(\mathcal{F}')$ for all $i > p_n^\#$, we have $\mathsf{Pre}_B^i(\mathcal{F}') = \emptyset$ for all $i > p_n^\#$. This shows that the language of $\mathcal{A}$ is non-empty if and only if the language of B is infinite from some state (namely q_0), and establishes the correctness of the reduction. $\square$

Relation with MDPs. The underlying structure of a Markov decision process $\mathcal{M} = \langle Q, \mathsf{A}, \delta \rangle$ is an alternating graph, where the successor q' of a state q is obtained by an existential choice of an action a and a universal choice of a state $q' \in \mathsf{Supp}(\delta(q, a))$. Therefore, it is natural that some questions related

[3] In expression c_j^i, we assume that j is interpreted modulo p_i.

to MDPs have a corresponding formulation in terms of alternating automata. We show that such connections exist between synchronizing problems for MDPs and language-theoretic questions for alternating automata, such as emptiness and universal finiteness. Given a 1L-AFA $\mathcal{A} = \langle Q, \delta_{\mathcal{A}}, \mathcal{F} \rangle$, assume without loss of generality that the transition function $\delta_{\mathcal{A}}$ is such that $\delta_{\mathcal{A}}(q) = c_1 \vee \cdots \vee c_m$ has the same number m of conjunctive clauses for all $q \in Q$. From $\mathcal{A}$, construct the MDP $\mathcal{M}_{\mathcal{A}} = \langle Q, \mathsf{A}, \delta_{\mathcal{M}} \rangle$ where $\mathsf{A} = \{a_1, \dots, a_m\}$ and $\delta_{\mathcal{M}}(q, a_k)$ is the uniform distribution over the states occurring in the k-th clause c_k in $\delta_{\mathcal{A}}(q)$, for all $q \in Q$ and $a_k \in \mathsf{A}$. Then, we have $Acc_{\mathcal{A}}(n, \mathcal{F}) = \mathsf{Pre}^n_{\mathcal{M}}(\mathcal{F})$ for all $n \geq 0$. Similarly, from an MDP $\mathcal{M}$ and a set T of states, we can construct a 1L-AFA $\mathcal{A} = \langle Q, \delta_{\mathcal{A}}, \mathcal{F} \rangle$ with $\mathcal{F} = T$ such that $Acc_{\mathcal{A}}(n, \mathcal{F}) = \mathsf{Pre}^n_{\mathcal{M}}(T)$ for all $n \geq 0$ (let $\delta_{\mathcal{A}}(q) = \bigvee_{a \in \mathsf{A}} \bigwedge_{q' \in \mathsf{post}(q,a)} q'$ for all $q \in Q$).

Several decision problems for 1L-AFA can be solved by computing the sequence $Acc_{\mathcal{A}}(n, \mathcal{F})$, and we show that some synchronizing problems for MDPs require the computation of the sequence $\mathsf{Pre}^n_{\mathcal{M}}(\mathcal{F})$. Therefore, the above relation between 1L-AFA and MDPs establishes bridges that we use in Section 4 to transfer complexity results from 1L-AFA to MDPs.

4 Eventually Synchronization

In this section, we show the PSPACE-completeness of the membership problem for eventually synchronizing objectives and the three winning modes. By the remarks at the end of Section 2, we consider the membership problem with function *sum* and Dirac initial distributions (i.e., single initial state).

4.1 Sure eventually synchronization

Given a target set T, the membership problem for sure-winning eventually synchronizing objective in T can be solved by computing the sequence $\mathsf{Pre}^n(T)$ of iterated predecessor. A state q_0 is sure-winning for eventually synchronizing in T if $q_0 \in \mathsf{Pre}^n(T)$ for some $n \geq 0$.

Lemma 4. *Let $\mathcal{M}$ be an MDP and T be a target set. For all states q_0, we have $q_0 \in \langle\langle 1 \rangle\rangle^{event}_{sure}(sum_T)$ if and only if there exists $n \geq 0$ such that $q_0 \in \mathsf{Pre}^n_{\mathcal{M}}(T)$.*

Proof. We prove the following equivalence by induction (on the length i): for all initial states q_0, there exists a strategy α sure-winning in i steps from q_0 (i.e., such that $\mathcal{M}^{\alpha}_i(T) = 1$) if and only if $q_0 \in \mathsf{Pre}^i(T)$. The case $i = 0$ trivially holds since for all strategies α, we have $\mathcal{M}^{\alpha}_0(T) = 1$ if and only if $q_0 \in T$.

Assume that the equivalence holds for all $i < n$. For the induction step, show that $\mathcal{M}$ is sure eventually synchronizing from q_0 (in n steps) if and only if there exists an action a such that $\mathcal{M}$ is sure eventually synchronizing (in $n-1$ steps) from all states $q' \in \mathsf{post}(q_0, a)$ (equivalently, $\mathsf{post}(q_0, a) \subseteq \mathsf{Pre}^{n-1}(T)$ by the induction hypothesis, that is $q_0 \in \mathsf{Pre}^n(T)$). First, if all successors q' of q_0 under some action a are sure eventually synchronizing, then so is q_0 by playing a followed by a winning strategy from each successor q'. For the other direction,

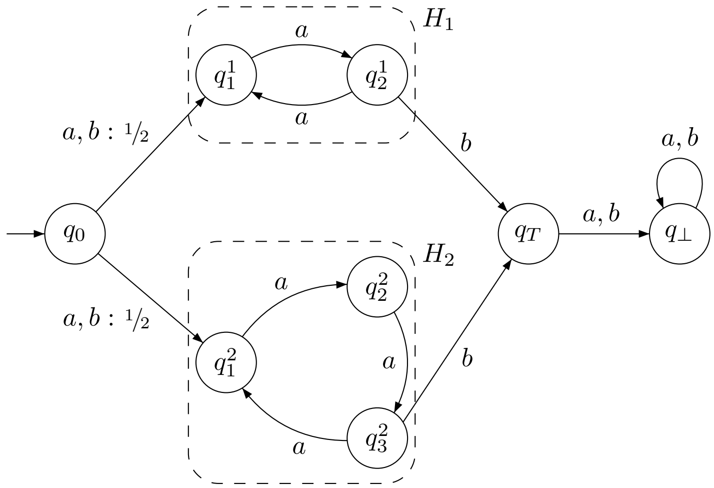

Fig. 5. The MDP $\mathcal{M}_2$.

assume towards contradiction that $\mathcal{M}$ is sure eventually synchronizing from q_0 (in n steps), but for each action a, there is a state $q' \in \mathsf{post}(q_0, a)$ that is not sure eventually synchronizing. Then, from q' there is a positive probability to reach a state not in T after $n-1$ steps, no matter the strategy played. Hence from q_0, for all strategies, the probability mass in T cannot be 1 after n steps, in contradiction with the fact that $\mathcal{M}$ is sure eventually synchronizing from q_0 in n steps. It follows that the induction step holds, and the proof is complete. □

By Lemma 4, the membership problem for sure eventually synchronizing is equivalent to the emptiness problem of 1L-AFA, and thus PSPACE-complete. Moreover if $q_0 \in \mathsf{Pre}^n_{\mathcal{M}}(T)$, a finite-memory strategy with n modes that at mode i in a state q plays an action a such that $\mathsf{post}(q, a) \subseteq \mathsf{Pre}^{i-1}(T)$ is sure winning for eventually synchronizing.

There exists a family of MDPs $\mathcal{M}_n$ ($n \in \mathbb{N}$) over alphabet $\{a, b\}$ that are sure winning for eventually synchronization, and where the sure winning strategies require exponential memory. The MDP $\mathcal{M}_2$ is shown in Fig. 5. The structure of $\mathcal{M}_n$ is an initial uniform probabilistic transition to n components $H_1, \dots, H_n$ where H_i is a cycle of length p_i the ith prime number. On action a, the next state in the cycle is reached, and on action b the target state q_T is reached, only from the last state in the cycles. From other states, the action b leads to $q_\perp$ (transitions not depicted). A sure winning strategy for eventually synchronization in $\{q_T\}$ is to play a in the first $p_n^{\#} = \prod_{i=1}^{n} p_i$ steps, and then play b. This requires memory of size $p_n^{\#} > 2^n$ while the size of $\mathcal{M}_n$ is in $O(n^2 \log n)$ [3]. It can be proved by standard pumping arguments that no strategy of size smaller than $p_n^{\#}$ is sure winning.

The following theorem summarizes the results for sure eventually synchronizing.

Theorem 2. *For sure eventually synchronizing in MDPs:*

1. *(Complexity). The membership problem is PSPACE-complete.*
2. *(Memory). Exponential memory is necessary and sufficient for both pure and randomized strategies, and pure strategies are sufficient.*

4.2 Almost-sure eventually synchronization

We show an example where infinite memory is necessary to win for almost-sure eventually synchronizing. Consider the MDP in Fig. 6 with initial state q_0. We construct a strategy that is almost-sure eventually synchronizing in q_2, showing that $q_0 \in \langle\langle 1 \rangle\rangle^{event}_{almost}(q_2)$. First, observe that for all $\epsilon > 0$ we can have probability at least $1 - \epsilon$ in q_2 after finitely many steps: playing n times a and then b leads to probability $1 - \frac{1}{2^n}$ in q_2. Thus the MDP is limit-sure eventually synchronizing in q_2. Moreover the remaining probability mass is in q_0. It turns out that that from any (initial) distribution with support $\{q_0, q_2\}$, the MDP is again limit-sure eventually synchronizing in q_2, and with support in $\{q_0, q_2\}$. Therefore we can take a smaller value of ϵ and play a strategy to have probability at least $1 - \epsilon$ in q_2, and repeat this for $\epsilon \to 0$. This strategy ensures almost-sure eventually synchronizing in q_2. The next result shows that infinite memory is necessary for almost-sure winning in this example.

Lemma 5. *There exists an almost-sure eventually synchronizing MDP for which all almost-sure eventually synchronizing strategies require infinite memory.*

Proof. Consider the MDP $\mathcal{M}$ shown in Fig. 6. We argued in Section 4.2 that $q_0 \in \langle\langle 1 \rangle\rangle^{event}_{almost}(q_2)$ and we now show that infinite memory is necessary from q_0 for almost-sure eventually synchronizing in q_2.

Assume towards contradiction that there exists a finite-memory strategy α that is almost-sure eventually synchronizing in q_2. Consider the Markov chain $\mathcal{M}(\alpha)$ (the product of the MDP $\mathcal{M}$ with the finite-state transducer defining α). A state (q, m) in $\mathcal{M}(\alpha)$ is called a *q-state*. Since α is almost-sure eventually synchronizing (but is not sure eventually synchronizing) in q_2 , there is a q_2-state in the recurrent states of $\mathcal{M}(\alpha)$. Since on all actions q_0 is a successor of q_2, and q_0 is a successor of itself, it follows that there is a recurrent q_0-state in $\mathcal{M}(\alpha)$, and that all periodic classes of recurrent states in $\mathcal{M}(\alpha)$ contain a q_0-state. Hence, in each stationary distribution there is a q_0-state with a positive probability, and therefore the probability mass in q_0 is bounded away from zero. It follows that the probability mass in q_2 is bounded away from 1 thus α is not almost-sure eventually synchronizing in q_2, a contradiction. □

It turns out that in general, almost-sure eventually synchronizing strategies can be constructed from a family of limit-sure eventually synchronizing strategies if we can also ensure that the probability mass remains in the winning region (as

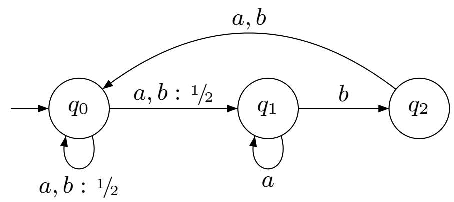

Fig. 6. An MDP where infinite memory is necessary for almost-sure eventually synchronizing strategies.

in the MDP in Fig. 6). We present a characterization of the winning region for almost-sure winning based on an extension of the limit-sure eventually synchronizing objective *with exact support*. This objective requires to ensure probability arbitrarily close to 1 in the target set T, and moreover that after the same number of steps the support of the probability distribution is contained in a given set U. Formally, given an MDP $\mathcal{M}$, let $\langle\langle 1\rangle\rangle^{event}_{limit}(sum_T, U)$ for $T \subseteq U$ be the set of all initial distributions such that for all $\epsilon > 0$ there exists a strategy α and $n \in \mathbb{N}$ such that $\mathcal{M}^\alpha_n(T) \geq 1 - \epsilon$ and $\mathcal{M}^\alpha_n(U) = 1$. We say that α is limit-sure eventually synchronizing in T with support in U.

We will present an algorithmic solution to limit-sure eventually synchronizing objectives with exact support in Section 4.3. Our characterization of the winning region for almost-sure winning is as follows.

Lemma 6. *Let $\mathcal{M}$ be an MDP and T be a target set. For all states q_0, we have $q_0 \in \langle\langle 1\rangle\rangle^{event}_{almost}(sum_T)$ if and only if there exists a set U such that:*

- *$q_0 \in \langle\langle 1\rangle\rangle^{event}_{sure}(sum_U)$, and*
- *$d_U \in \langle\langle 1\rangle\rangle^{event}_{limit}(sum_T, U)$ where d_U is the uniform distribution over U.*

Proof. First, if $q_0 \in \langle\langle 1\rangle\rangle^{event}_{almost}(sum_T)$, then there is a strategy α such that $\sup_{n\in\mathbb{N}} \mathcal{M}^\alpha_n(T) = 1$. Then either $\mathcal{M}^\alpha_n(T) = 1$ for some $n \geq 0$, or $\limsup_{n\to\infty} \mathcal{M}^\alpha_n(T) = 1$. If $\mathcal{M}^\alpha_n(T) = 1$, then q_0 is sure winning for eventually synchronizing in T, thus $q_0 \in \langle\langle 1\rangle\rangle^{event}_{sure}(sum_T)$ and we can take $U = T$. Otherwise, for all $i > 0$ there exists $n_i \in \mathbb{N}$ such that $\mathcal{M}^\alpha_{n_i}(T) \geq 1 - 2^{-i}$, and moreover $n_{i+1} > n_i$ for all $i > 0$. Let $s_i = \mathsf{Supp}(\mathcal{M}^\alpha_{n_i})$ be the support of $\mathcal{M}^\alpha_{n_i}$. Since the state space is finite, there is a set U that occurs infinitely often in the sequence $s_0 s_1 \dots$, thus for all $k > 0$ there exists $m_k \in \mathbb{N}$ such that $\mathcal{M}^\alpha_{m_k}(T) \geq 1 - 2^{-k}$ and $\mathcal{M}^\alpha_{m_k}(U) = 1$. It follows that α is sure eventually synchronizing in U from q_0, hence $q_0 \in \langle\langle 1\rangle\rangle^{event}_{sure}(sum_U)$. Moreover $\mathcal{M}$ with intial distribution $d_1 = \mathcal{M}^\alpha_{m_1}$ is limit-sure eventually synchronizing in T with exact support in U. Since $\mathsf{Supp}(d_1) = U = \mathsf{Supp}(d_U)$, it follows by Corollary 1 that $d_U \in \langle\langle 1\rangle\rangle^{event}_{limit}(sum_T, U)$.

To establish the converse, note that since $d_U \in \langle\langle 1\rangle\rangle^{event}_{limit}(sum_T, U)$, it follows from Corollary 1 that from all initial distributions with support in U, for all $\epsilon > 0$ there exists a strategy α_ϵ and a position n_ϵ such that $\mathcal{M}^{\alpha_\epsilon}_{n_\epsilon}(T) \geq 1 - \epsilon$

and $\mathcal{M}^{\alpha_\epsilon}_{n_\epsilon}(U) = 1$. We construct an almost-sure limit eventually synchronizing strategy α as follows. Since $q_0 \in \langle\!\langle 1 \rangle\!\rangle^{event}_{sure}(sum_U)$, play according to a sure eventually synchronizing strategy from q_0 until all the probability mass is in U. Then for $i = 1, 2, \dots$ and $\epsilon_i = 2^{-i}$, repeat the following procedure: given the current probability distribution, select the corresponding strategy α_{ϵ_i} and play according to α_{ϵ_i} for n_{ϵ_i} steps, ensuring probability mass at least $1-2^{-i}$ in T, and since after that the support of the probability mass is again in U, play according to $\alpha_{\epsilon_{i+1}}$ for $n_{\epsilon_{i+1}}$ steps, etc. This strategy α ensures that $\sup_{n \in \mathbb{N}} \mathcal{M}^\alpha_n(T) = 1$ from q_0, hence $q_0 \in \langle\!\langle 1 \rangle\!\rangle^{event}_{almost}(sum_T)$. □

Note that from Lemma 6, it follows that counting strategies are sufficient to win almost-sure eventually synchronizing objective (a strategy is *counting* if $\alpha(\rho) = \alpha(\rho')$ for all prefixes ρ, ρ' with the same length and $\mathsf{Last}(\rho) = \mathsf{Last}(\rho')$).

As we show in Section 4.3 that the membership problem for limit-sure eventually synchronizing with exact support can be solved in PSPACE, it follows from the characterization in Lemma 6 that the membership problem for almost-sure eventually synchronizing is in PSPACE, using the following (N)PSPACE algorithm: guess the set U, and check that $q_0 \in \langle\!\langle 1 \rangle\!\rangle^{event}_{sure}(sum_U)$, and that $d_U \in \langle\!\langle 1 \rangle\!\rangle^{event}_{limit}(sum_T, U)$ where d_U is the uniform distribution over U (this can be done in PSPACE by Theorem 2 and Theorem 4). We present a matching lower bound.

Lemma 7. *The membership problem for $\langle\!\langle 1 \rangle\!\rangle^{event}_{almost}(sum_T)$ is PSPACE-hard even if T is a singleton.*

Proof. The proof is by a reduction from the membership problem for sure eventually synchronization, which is PSPACE-complete by Theorem 2. Given an MDP $\mathcal{M} = \langle Q, \mathsf{A}, \delta \rangle$, an initial state $q_0 \in Q$, and a state $\hat{q} \in Q$, we construct an MDP $\mathcal{N} = \langle Q', \mathsf{A}', \delta' \rangle$ and a state $\hat{p} \in Q'$ such that $q_0 \in \langle\!\langle 1 \rangle\!\rangle^{event}_{sure}(\hat{q})$ in $\mathcal{M}$ if and only if $q_0 \in \langle\!\langle 1 \rangle\!\rangle^{event}_{almost}(\hat{p})$ in $\mathcal{N}$. The MDP $\mathcal{N}$ is a copy of $\mathcal{M}$ with two new states $\hat{p}$ and sink reachable only by a new action $\sharp$ (see Fig. 7). Formally, $Q' = Q \cup \{\hat{p}, \mathsf{sink}\}$ and $\mathsf{A}' = \mathsf{A} \cup \{\sharp\}$, and the transition function δ' is defined as follows, for all $q \in Q$: $\delta'(q, a) = \delta(q, a)$ for all $a \in \mathsf{A}$, $\delta'(q, \sharp)(\mathsf{sink}) = 1$ if $q \neq \hat{q}$, and $\delta'(\hat{q}, \sharp)(\hat{p}) = 1$; finally, for all $a \in \mathsf{A}'$, let $\delta'(\hat{p}, a)(\mathsf{sink}) = \delta'(\mathsf{sink}, a)(\mathsf{sink}) = 1$.

The goal is that $\mathcal{N}$ simulates $\mathcal{M}$ until the action $\sharp$ is played in $\hat{q}$ to move the probability mass from $\hat{q}$ to $\hat{p}$, ensuring that if $\mathcal{M}$ is sure-winning for eventually synchronizing in $\hat{q}$, then $\mathcal{N}$ is also sure-winning (and thus almost-sure winning) for eventually synchronizing in $\hat{p}$. Moreover, the only way to be almost-sure eventually synchronizing in $\hat{p}$ is to have probability 1 in $\hat{p}$ at some point, because the state $\hat{p}$ is transient under all strategies, thus the probability mass cannot accumulate and tend to 1 in $\hat{p}$ in the long run. It follows that (from all initial states q_0) $\mathcal{M}$ is sure-winning for eventually synchronizing in $\hat{q}$ if and only if $\mathcal{N}$ is almost-sure winning for eventually synchronizing in $\hat{p}$. It follows from this reduction that the membership problem for almost-sure eventually synchronizing objective is PSPACE-hard. □

The results of this section are summarized as follows.

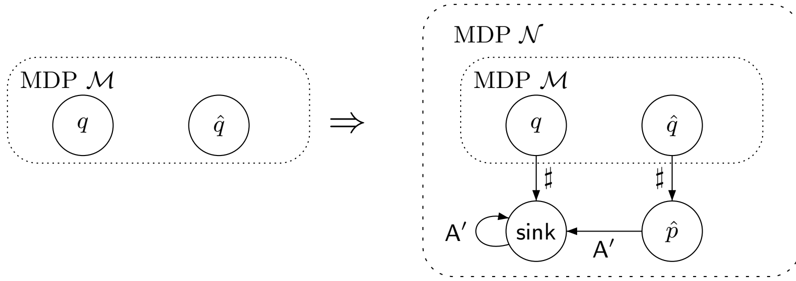

Fig. 7. Sketch of the reduction to show PSPACE-hardness of the membership problem for almost-sure eventually synchronizing.

Theorem 3. *For almost-sure eventually synchronizing in MDPs:*

1. *(Complexity). The membership problem is PSPACE-complete.*
2. *(Memory). Infinite memory is necessary in general for both pure and randomized strategies, and pure strategies are sufficient.*

4.3 Limit-sure eventually synchronization

In this section, we present the algorithmic solution for limit-sure eventually synchronizing with exact support. Note that the limit-sure eventually synchronizing objective is a special case where the support is the state space of the MDP. Consider the MDP in Fig. 1 which is limit-sure eventually synchronizing in $\{q_2\}$, as shown in Lemma 2. For $i = 0, 1, \dots$, the sequence $\mathsf{Pre}^i(T)$ of predecessors of $T = \{q_2\}$ is ultimately periodic: $\mathsf{Pre}^0(T) = \{q_2\}$, and $\mathsf{Pre}^i(T) = \{q_1\}$ for all $i \geq 1$. Given $\epsilon > 0$, a strategy to get probability $1 - \epsilon$ in q_2 first accumulates probability mass in the *periodic* subsequence of predecessors (here $\{q_1\}$), and when the probability mass is greater than $1 - \epsilon$ in q_1, the strategy injects the probability mass in q_2 (through the aperiodic prefix of the sequence of predecessors). This is the typical shape of a limit-sure eventually synchronizing strategy. Note that in this scenario, the MDP is also limit-sure eventually synchronizing in every set $\mathsf{Pre}^i(T)$ of the sequence of predecessors. A special case is when it is possible to get probability 1 in the sequence of predecessors after finitely many steps. In this case, the probability mass injected in T is 1 and the MDP is even sure-winning. The algorithm for deciding limit-sure eventually synchronization relies on the above characterization, generalized in Lemma 8 to limit-sure eventually synchronizing with exact support, saying that limit-sure eventually synchronizing in T with support in U is equivalent to either limit-sure eventually synchronizing in $\mathsf{Pre}^k(T)$ with support in $\mathsf{Pre}^k(U)$ (for arbitrary k), or sure eventually synchronizing in T (and therefore also in U).

Lemma 8. *For $T \subseteq U$ and $k \geq 0$, let $R = \mathsf{Pre}^k(T)$ and $Z = \mathsf{Pre}^k(U)$. Then, $\langle\!\langle 1 \rangle\!\rangle^{event}_{limit}(sum_T, U) = \langle\!\langle 1 \rangle\!\rangle^{event}_{sure}(sum_T) \cup \langle\!\langle 1 \rangle\!\rangle^{event}_{limit}(sum_R, Z)$.*

Proof. The proof is in two parts. First we show that $\langle\!\langle 1\rangle\!\rangle^{event}_{sure}(sum_T) \cup \langle\!\langle 1\rangle\!\rangle^{event}_{limit}(sum_R, Z) \subseteq \langle\!\langle 1\rangle\!\rangle^{event}_{limit}(sum_T, U)$: since $T \subseteq U$, it follows from the definitions that $\langle\!\langle 1\rangle\!\rangle^{event}_{sure}(sum_T) \subseteq \langle\!\langle 1\rangle\!\rangle^{event}_{limit}(sum_T, U)$; to show that $\langle\!\langle 1\rangle\!\rangle^{event}_{limit}(sum_R, Z) \subseteq \langle\!\langle 1\rangle\!\rangle^{event}_{limit}(sum_T, U)$ in an MDP $\mathcal{M}$, let $\epsilon > 0$ and consider an initial distribution μ_0 and a strategy α such that for some $i \geq 0$ we have $\mathcal{M}^{\alpha}_i(R) \geq 1 - \epsilon$ and $\mathcal{M}^{\alpha}_i(Z) = 1$. We construct a strategy β that plays like α for the first i steps, and then since $R = \mathsf{Pre}^k(T)$ and $Z = \mathsf{Pre}^k(U)$ plays from states in R according to a sure eventually synchronizing strategy with target T, and from states in $Z \setminus R$ according to a sure eventually synchronizing strategy with target U (such strategies exist by the proof of Lemma 4). The strategy β ensures from μ_0 that $\mathcal{M}^{\beta}_{i+k}(T) \geq 1 - \epsilon$ and $\mathcal{M}^{\beta}_{i+k}(U) = 1$, showing that $\mathcal{M}$ is limit-sure eventually synchronizing in T with support in U.

Second we show the converse inclusion, namely that $\langle\!\langle 1\rangle\!\rangle^{event}_{limit}(sum_T, U) \subseteq \langle\!\langle 1\rangle\!\rangle^{event}_{sure}(sum_T) \cup \langle\!\langle 1\rangle\!\rangle^{event}_{limit}(sum_R, Z)$. Consider an initial distribution $\mu_0 \in \langle\!\langle 1\rangle\!\rangle^{event}_{limit}(sum_T, U)$ in the MDP $\mathcal{M}$ and for $\epsilon_i = \frac{1}{i}$ ($i \in \mathbb{N}$) let α_i be a strategy and $n_i \in \mathbb{N}$ such that $\mathcal{M}^{\alpha_i}_{n_i}(T) \geq 1 - \epsilon_i$ and $\mathcal{M}^{\alpha_i}_{n_i}(U) = 1$. We consider two cases. (a) If the set $\{n_i \mid i \geq 0\}$ is bounded, then there exists a number n that occurs infinitely often in the sequence $(n_i)_{i\in\mathbb{N}}$, and such that for all $i \geq 0$, there exists a strategy β_i such that $\mathcal{M}^{\beta_i}_n(T) \geq 1 - \epsilon$ and $\mathcal{M}^{\beta_i}_n(U) = 1$. Since n is fixed, we can assume w.l.o.g. that the strategies β_i are pure, and since there is a finite number of pure strategies over paths of length at most n, it follows that there is a strategy β that occurs infinitely often among the strategies β_i and such that for all $\epsilon > 0$ we have $\mathcal{M}^{\beta}_n(T) \geq 1 - \epsilon$, hence $\mathcal{M}^{\beta}_n(T) = 1$, showing that $\mathcal{M}$ is sure winning for eventually synchronizing in T, that is $\mu_0 \in \langle\!\langle 1\rangle\!\rangle^{event}_{sure}(sum_T)$. ($b$) otherwise, the set $\{n_i \mid i \geq 0\}$ is unbounded and we can assume w.l.o.g. that $n_i \geq k$ for all $i \geq 0$. We claim that the family of strategies α_i ensures limit-sure synchronization in $R = \mathsf{Pre}^k(T)$ with support in $Z = \mathsf{Pre}^k(U)$. Essentially this is because if the probability in T is close to 1 after n_i steps, then k steps before the probability in $\mathsf{Pre}^k(T)$ must be close to 1 as well. Formally, we show that α_i is such that $\mathcal{M}^{\alpha_i}_{n_i-k}(R) \geq 1 - \frac{\epsilon}{\eta^k}$ and $\mathcal{M}^{\alpha_i}_{n_i-k}(Z) = 1$ where η is the smallest positive probability in the transitions of $\mathcal{M}$. Towards contradiction, assume that $\mathcal{M}^{\alpha_i}_{n_i-k}(R) < 1 - \frac{\epsilon}{\eta^k}$, then $\mathcal{M}^{\alpha_i}_{n_i-k}(Q \setminus R) > \frac{\epsilon}{\eta^k}$ and from every state $q \in Q \setminus R$, no matter which sequence of actions is played by α_i for the next k steps, there is a path from q to a state outside of T, thus with probability at least η^k. Hence the probability in $Q \setminus T$ after n_i steps is greater than $\frac{\epsilon}{\eta^k} \cdot \eta^k$, and it follows that $\mathcal{M}^{\alpha_i}_{n_i}(T) < 1 - \epsilon$, in contradiction with the definition of α_i. This shows that $\mathcal{M}^{\alpha_i}_{n_i-k}(R) \geq 1 - \frac{\epsilon}{\eta^k}$, and an argument analogous to the proof of Lemma 4 shows that $\mathcal{M}^{\alpha_i}_{n_i-k}(Z) = 1$. It follows that $\mu_0 \in \langle\!\langle 1\rangle\!\rangle^{event}_{limit}(sum_R, Z)$ and the proof is complete. □

Thanks to Lemma 8, since sure-winning is already solved in Section 4.1, it suffices to solve the limit-sure eventually synchronizing problem for target $R = \mathsf{Pre}^k(T)$ and support $Z = \mathsf{Pre}^k(U)$ with arbitrary k, instead of T and U. We can choose k such that both $\mathsf{Pre}^k(T)$ and $\mathsf{Pre}^k(U)$ lie in the periodic part of the sequence of pairs of predecessors $(\mathsf{Pre}^i(T), \mathsf{Pre}^i(U))$. We can assume that $k \leq 3^{|Q|}$ since $\mathsf{Pre}^i(T) \subseteq \mathsf{Pre}^i(U) \subseteq Q$ for all $i \geq 0$. For such value of k

the limit-sure problem is conceptually simpler: once some probability is injected in $R = \mathsf{Pre}^k(T)$, it can loop through the sequence of predecessors and visit R infinitely often (every r steps, where $r \leq 3^{|Q|}$ is the period of the sequence of pairs of predecessors). It follows that if a strategy ensures with probability 1 that the set R can be reached by finite paths whose lengths are congruent modulo r, then the whole probability mass can indeed synchronously accumulate in R in the limit.

Therefore, limit-sure eventually synchronizing in R reduces to standard limit-sure reachability with target set R and the additional requirement that the numbers of steps at which the target set is reached be congruent modulo r. In the case of limit-sure eventually synchronizing with support in Z, we also need to ensure that no mass of probability leaves the sequence $\mathsf{Pre}^i(Z)$. In a state $q \in \mathsf{Pre}^i(Z)$, we say that an action $a \in \mathsf{A}$ is *Z-safe* at position i if[4] $\mathsf{post}(q, a) \subseteq \mathsf{Pre}^{i-1}(Z)$. In states $q \notin \mathsf{Pre}^i(Z)$ there is no Z-safe action at position i.

To encode the above requirements, we construct an MDP $\mathcal{M}_Z \times [r]$ that allows only Z-safe actions to be played (and then mimics the original MDP), and tracks the position (modulo r) in the sequence of predecessors, thus simply decrementing the position on each transition since all successors of a state $q \in \mathsf{Pre}^i(Z)$ on a safe action are in $\mathsf{Pre}^{i-1}(Z)$.

Formally, if $\mathcal{M} = \langle Q, \mathsf{A}, \delta \rangle$ then $\mathcal{M}_Z \times [r] = \langle Q', \mathsf{A}, \delta' \rangle$ where

– $Q' = Q \times \{r-1, \dots, 1, 0\} \cup \{\mathsf{sink}\}$; intuitively, we expect that $q \in \mathsf{Pre}^i(Z)$ in the reachable states $\langle q, i \rangle$ consisting of a state q of $\mathcal{M}$ and a *position* i in the predecessor sequence;
– δ' is defined as follows (assuming an arithmetic modulo r on positions) for all $\langle q, i \rangle \in Q'$ and $a \in \mathsf{A}$: if a is a Z-safe action in q at position i, then $\delta'(\langle q, i \rangle, a)(\langle q', i-1 \rangle) = \delta(q, a)(q')$, otherwise $\delta'(\langle q, i \rangle, a)(\mathsf{sink}) = 1$ (and sink is absorbing).

Note that the size of the MDP $\mathcal{M}_Z \times [r]$ is exponential in the size of $\mathcal{M}$ (since r is at most $3^{|Q|}$).

Lemma 9. *Let $\mathcal{M}$ be an MDP and $R \subseteq Z$ be two sets of states such that $\mathsf{Pre}^r(R) = R$ and $\mathsf{Pre}^r(Z) = Z$ where $r > 0$. Then a state q_0 is limit-sure eventually synchronizing in R with support in Z ($q_0 \in \langle\langle 1 \rangle\rangle^{event}_{limit}(sum_R, Z)$) if and only if there exists $0 \leq t < r$ such that $\langle q_0, t \rangle$ is limit-sure winning for the reachability objective $\Diamond(R \times \{0\})$ in the MDP $\mathcal{M}_Z \times [r]$.*

Proof. For the first direction of the lemma, assume that q_0 is limit-sure eventually synchronizing in R with support in Z, and for $\epsilon > 0$ let β be a strategy such that $\mathcal{M}^\beta_k(Z) = 1$ and $\mathcal{M}^\beta_k(R) \geq 1 - \epsilon$ for some number k of steps. Let $0 \leq t \leq r$ such that $t = k \mod r$. We show that from initial state (q_0, t) the strategy α in $\mathcal{M}_Z \times [r]$ that mimics (copies) the strategy β is limit-sure winning for the reachability objective $\Diamond R_0$: it follows from Lemma 4 that α plays only

[4] Since $\mathsf{Pre}^r(Z) = Z$ and $\mathsf{Pre}^r(R) = R$, we assume a modular arithmetic for exponents of Pre, that is $\mathsf{Pre}^x(\cdot)$ is defined as $\mathsf{Pre}^{x \bmod r}(\cdot)$. For example $\mathsf{Pre}^{-1}(Z)$ is $\mathsf{Pre}^{r-1}(Z)$.

Z-safe actions, and since $Pr^{\alpha}(\Diamond R_0) \geq Pr^{\alpha}(\Diamond^k R_0) = \mathcal{M}_k^{\beta}(R) \geq 1 - \epsilon$, the result follows.

For the converse direction, let $R_0 = R \times \{0\}$ and assuming that there exists $0 \leq t < r$ such that $\langle q_0, t\rangle$ is limit-sure winning for the reachability objective $\Diamond R_0$ in $\mathcal{M}_Z \times [r]$, show that q_0 is limit-sure synchronizing in target set R with exact support in Z. Since the winning region of limit-sure and almost-sure reachability coincide for MDPs [10], there exists a (pure) strategy α in $\mathcal{M}_Z \times [r]$ with initial state $\langle q, t\rangle$ such that $\Pr^{\alpha}(\Diamond R_0) = 1$.

Given $\epsilon > 0$, we construct from α a pure strategy β in $\mathcal{M}$ that is $(1 - \epsilon)$-synchronizing in R with support in Z. Given a finite path $\rho = q_0 a_0 q_1 a_1 \dots q_n$ in $\mathcal{M}$ (with $q_0 = q$), there is a corresponding path $\rho' = \langle q_0, k_0\rangle a_0 \langle q_1, k_1\rangle a_1 \dots \langle q_n, k_n\rangle$ in $\mathcal{M}_Z \times [r]$ where $k_0 = t$ and $k_{i+1} = k_i - 1$ for all $i \geq 0$. Since the sequence $k_0, k_1, \dots$ is uniquely determined from ρ, there is a clear bijection between the paths in $\mathcal{M}$ and the paths in $\mathcal{M}_Z \times [r]$ that we often omit to apply and mention. Define the strategy β as follows: if $q_n \in \mathsf{Pre}^{k_n}(R)$, then there exists an action a such that $\mathsf{post}(q_n, a) \subseteq \mathsf{Pre}^{k_n - 1}(R)$ and we define $\beta(\rho) = a$, otherwise let $\beta(\rho) = \alpha(\rho')$. Thus β mimics α (thus playing only Z-safe actions) unless a state q is reached at step n such that $q \in \mathsf{Pre}^{t-n}(R)$, and then β switches to always playing actions that are R-safe (and thus also Z-safe since $R \subseteq Z$). We now prove that β is limit-sure eventually synchronizing in target set R with support in Z. First since β plays only Z-safe actions, it follows for all k such that $t - k = 0$ (modulo r), all states reached from q_0 with positive probability after k steps are in Z. Hence $\mathcal{M}_k^{\beta}(Z) = 1$ for all such k. Second, we show that given $\epsilon > 0$ there exists k such that $t - k = 0$ and $\mathcal{M}_k^{\beta}(R) \geq 1 - \epsilon$, thus also $\mathcal{M}_k^{\beta}(Z) = 1$ and β is limit-sure eventually synchronizing in target set R with support in Z. To show this, recall that $\Pr^{\alpha}(\Diamond R_0) = 1$, and therefore $\Pr^{\alpha}(\Diamond^{\leq k} R_0) \geq 1 - \epsilon$ for all sufficiently large k. Without loss of generality, consider such a k satisfying $t - k = 0$ (modulo r). For $i = 1, \dots, r-1$, let $R_i = \mathsf{Pre}^i(R) \times \{i\}$. Then trivially $\Pr^{\alpha}(\Diamond^{\leq k} \bigcup_{i=0}^{r} R_i) \geq 1 - \epsilon$ and since β agrees with α on all finite paths that do not (yet) visit $\bigcup_{i=0}^{r} R_i$, given a path ρ that visits $\bigcup_{i=0}^{r} R_i$ (for the first time), only R-safe actions will be played by β and thus all continuations of ρ in the outcome of β will visit R after k steps (in total). It follows that $\Pr^{\beta}(\Diamond^{=k} R_0) \geq 1 - \epsilon$, that is $\mathcal{M}_k^{\beta}(R) \geq 1 - \epsilon$. Note that we used the same strategy β for all $\epsilon > 0$ and thus β is also almost-sure eventually synchronizing in R. □

Since deciding limit-sure reachability is PTIME-complete, it follows from Lemma 9 that limit-sure synchronization (with exact support) can be decided in EXPTIME. We show that the problem can be solved in PSPACE by exploiting the special structure of the exponential MDP in Lemma 9. We conclude this section by showing that limit-sure synchronization with exact support is PSPACE-complete (even in the special case of a trivial support).

Lemma 10. *The membership problem for limit-sure eventually synchronization with exact support is in PSPACE.*

Proof. We present a (nondeterministic) PSPACE algorithm to decide, given an MDP $\mathcal{M} = \langle Q, \mathsf{A}, \delta \rangle$, a state q_0, and two sets $T \subseteq U$, whether q_0 is limit-sure eventually synchronizing in T with support in U.

First, the algorithm computes numbers $k \geq 0$ and $r > 0$ such that for $R = \mathsf{Pre}^k(T)$ and $Z = \mathsf{Pre}^k(U)$ we have $\mathsf{Pre}^r(R) = R$ and $\mathsf{Pre}^r(Z) = Z$. As discussed before, this can be done by guessing $k, r \leq 3^{|Q|}$. By Lemma 8, we have $\langle\langle 1 \rangle\rangle^{event}_{limit}(sum_T, U) = \langle\langle 1 \rangle\rangle^{event}_{limit}(sum_R, Z) \cup \langle\langle 1 \rangle\rangle^{event}_{sure}(sum_T)$, and since sure eventually synchronizing in T can be decided in PSPACE (by Theorem 2), it suffices to decide limit-sure eventually synchronizing in R with support in Z in PSPACE. According to Lemma 9, it is therefore sufficient to show that deciding limit-sure winning for the (standard) reachability objective $\Diamond(R \times \{0\})$ in the MDP $\mathcal{M}_Z \times [r]$ can be done in polynomial space. As we cannot afford to construct the exponential-size MDP $\mathcal{M}_Z \times [r]$, the algorithm relies on the following characterization of the limit-sure winning set for reachability objectives in MDPs. It is known that the winning region for limit-sure and almost-sure reachability coincide [10], and pure memoryless strategies are sufficient. Therefore, we can see that the almost-sure winning set W for the reachability objective $\Diamond(R \times \{0\})$ satisfies the following property: there exists a memoryless strategy $\alpha : W \to \mathsf{A}$ such that (1) W is closed, that is $\mathsf{post}(q, \alpha(q)) \subseteq W$ for all $q \in W$, and (2) in the graph of the Markov chain $M(\alpha)$, for every state $q \in W$, there is a path (of length at most $|W|$) from q to $R \times \{0\}$.

This property ensures that from every state in W, the target set $R \times \{0\}$ is reached within $|W|$ steps with positive (and bounded) probability, and since W is closed it ensures that $R \times \{0\}$ is reached with probability 1 in the long run. Thus any set W satisfying the above property is almost-sure winning.

Our algorithm will guess and explore on the fly a set W to ensure that it satisfies this property, and contains the state $\langle q_0, t \rangle$ for some $t < r$. As we cannot afford to explicitly guess W (remember that W could be of exponential size), we decompose W into *slices* $W_0, W_1, \dots$ such that $W_i \subseteq Q$ and $W_i \times \{-i \mod r\} = W \cap (Q \times \{-i \mod r\})$. We start by guessing W_0, and we use the property that in $\mathcal{M}_Z \times [r]$, from a state (q, j) under all Z-safe actions, all successors are of the form $(\cdot, j-1)$. It follows that the successors of the states in $W_i \times \{-i\}$ should lie in the slice $W_{i+1} \times \{-i-1\}$, and we can guess on the fly the next slice $W_{i+1} \subseteq Q$ by guessing for each state q in a slice W_i an action a_q such that $\bigcup_{q \in W_i} \mathsf{post}(q, a_q) \subseteq W_{i+1}$. Moreover, we need to check the existence of a path from every state in W to $R \times \{0\}$. As W is closed, it is sufficient to check that there is a path from every state in $W_0 \times \{0\}$ to $R \times \{0\}$. To do this we guess along with the slices $W_0, W_1, \dots$ a sequence of sets $P_0, P_1, \dots$ where $P_i \subseteq W_i$ contains the states of slice W_i that belong to the guessed paths. Formally, $P_0 = W_0$, and for all $i \geq 0$, the set P_{i+1} is such that $\mathsf{post}(q, a_q) \cap P_{i+1} \neq \emptyset$ for all $q \in P'_i$ (where $P'_i = P_i \setminus R$ if i is a multiple of r, and $P'_i = P_i$ otherwise), that is P_{i+1} contains a successor of every state in P_i that is not already in the target R (at position 0 modulo r).

We need polynomial space to store the first slice W_0, the current slice W_i and the set P_i, and the value of i (in binary). As $\mathcal{M}_Z \times [r]$ has $|Q| \cdot r$ states,

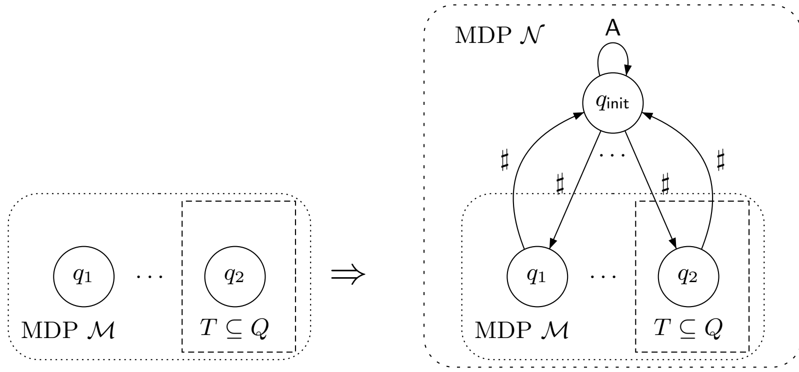

Fig. 8. Sketch of reduction to show PSPACE-hardness of the membership problem for limit-sure eventually synchronizing.

the algorithm runs for $|Q| \cdot r$ iterations and then checks that (1) $W_{|Q|\cdot r} \subseteq W_0$ to ensure that $W = \bigcup_{i \leq |Q|\cdot r} W_i \times \{i \mod r\}$ is closed, (2) $P_{|Q|\cdot r} = \emptyset$ showing that from every state in $W_0 \times \{0\}$ there is a path to $R \times \{0\}$ (and thus also from all states in W), and (3) the state q_0 occurs in some slice W_i. The correctness of the algorithm follows from the characterization of the almost-sure winning set for reachability in MDPs: if some state $\langle q_0, t\rangle$ is limit-sure winning, then the algorithm accepts by guessing (slice by slice) the almost-sure winning set W and the paths from $W_0 \times \{0\}$ to $R \times \{0\}$ (at position 0 modulo r), and otherwise any set (and paths) correctly guessed by the algorithm would not contain q_0 in any slice.

□

It follows from the proof of Lemma 9 that all winning modes for eventually synchronizing are independent of the numerical value of the positive transition probabilities.

Corollary 1. *Let $\mu \in \{sure, almost, limit\}$ and $T \subseteq U$ be two sets. For two distributions d, d' with $\mathsf{Supp}(d) = \mathsf{Supp}(d')$, we have $d \in \langle\!\langle 1 \rangle\!\rangle_{\mu}^{event}(sum_T, U)$ if and only if $d' \in \langle\!\langle 1 \rangle\!\rangle_{\mu}^{event}(sum_T, U)$.*

To establish the PSPACE-hardness for limit-sure eventually synchronizing in MDPs, we use a reduction from the universal finiteness problem for 1L-AFAs.

Lemma 11. *The membership problem for $\langle\!\langle 1 \rangle\!\rangle_{limit}^{event}(sum_T)$ is PSPACE-hard even if T is a singleton.*

Proof. The proof is by a reduction from the universal finiteness problem for one-letter alternating automata (1L-AFA), which is PSPACE-complete (by Lemma 3). It is easy to see that this problem remains PSPACE-complete even if the set T of accepting states of the 1L-AFA is a singleton, and given the tight relation between 1L-AFA and MDP (see Section 3), it follows from the definition of the universal finiteness problem that deciding, in an MDP $\mathcal{M}$, whether the sequence $\mathsf{Pre}^n_{\mathcal{M}}(T) \neq \emptyset$ for all $n \geq 0$ is PSPACE-complete.

The reduction is as follows (see also Fig. 8). Given an MDP $\mathcal{M} = \langle Q, \mathsf{A}, \delta \rangle$ and a singleton $T \subseteq Q$, we construct an MDP $\mathcal{N} = \langle Q', \mathsf{A}', \delta' \rangle$ with state space $Q' = Q \uplus \{q_{\mathsf{init}}\}$ such that $\mathsf{Pre}^n_{\mathcal{M}}(T) \neq \emptyset$ for all $n \geq 0$ if and only if q_{init} is limit-sure eventually synchronizing in T. The MDP $\mathcal{N}$ is essentially a copy of $\mathcal{M}$ with alphabet $\mathsf{A} \uplus \{\sharp\}$ and the transition function on action $\sharp$ is the uniform distribution on Q from q_{init}, and the Dirac distribution on q_{init} from the other states $q \in Q$. There are self-loops on q_{init} for all other actions $a \in \mathsf{A}$. Formally, the transition function δ' is defined as follows, for all $q \in Q$:

- $\delta'(q, a) = \delta(q, a)$ for all $a \in \mathsf{A}$ (copy of $\mathcal{M}$), and $\delta'(q, \sharp)(q_{\mathsf{init}}) = 1$;
- $\delta'(q_{\mathsf{init}}, a)(q_{\mathsf{init}}) = 1$ for all $a \in \mathsf{A}$, and $\delta'(q_{\mathsf{init}}, \sharp)(q) = \frac{1}{|Q|}$.

We establish the correctness of the reduction as follows. For the first direction, assume that $\mathsf{Pre}^n_{\mathcal{M}}(T) \neq \emptyset$ for all $n \geq 0$. Then since $\mathcal{N}$ embeds a copy of $\mathcal{M}$ it follows that $\mathsf{Pre}^n_{\mathcal{N}}(T) \neq \emptyset$ for all $n \geq 0$ and there exist numbers $k_0, r \leq 2^{|Q|}$ such that $\mathsf{Pre}^{k_0+r}_{\mathcal{N}}(T) = \mathsf{Pre}^{k_0}_{\mathcal{N}}(T) \neq \emptyset$. Using Lemma 8 with $k = k_0$ and $R = \mathsf{Pre}^{k_0}_{\mathcal{N}}(T)$ (and $U = Z = Q'$ is the trivial support), it is sufficient to prove that $q_{\mathsf{init}} \in \langle\!\langle 1 \rangle\!\rangle^{event}_{limit}(R)$ to get $q_{\mathsf{init}} \in \langle\!\langle 1 \rangle\!\rangle^{event}_{limit}(T)$ (in $\mathcal{N}$). We show the stronger statement that q_{init} is actually almost-sure eventually synchronizing in R with the pure strategy α defined as follows, for all play prefix ρ (let $m = |\rho| \bmod r$):

- if $\mathsf{Last}(\rho) = q_{\mathsf{init}}$, then $\alpha(\rho) = \sharp$;
- if $\mathsf{Last}(\rho) = q \in Q$, then
  - if $q \in \mathsf{Pre}^{r-m}_{\mathcal{N}}(R)$, then $\alpha(\rho)$ plays a R-safe action at position $r - m$;
  - otherwise, $\alpha(\rho) = \sharp$.

The strategy α ensures that the probability mass that is not (yet) in the sequence of predecessors $\mathsf{Pre}^n_{\mathcal{N}}(R)$ goes to q_{init}, where by playing $\sharp$ at least a fraction $\frac{1}{|Q|}$ of it would reach the sequence of predecessors (at a synchronized position). It follows that after $2i$ steps, the probability mass in q_{init} is $(1 - \frac{1}{|Q|})^i$ and the probability mass in the sequence of predecessors is $1 - (1 - \frac{1}{|Q|})^i$. For $i \to \infty$, the probability in the sequence of predecessors tends to 1 and since $\mathsf{Pre}^n_{\mathcal{N}}(R) = R$ for all positions n that are a multiple of r, we get $\sup_n \mathcal{M}^{\alpha}_n(R) = 1$ and $q_{\mathsf{init}} \in \langle\!\langle 1 \rangle\!\rangle^{event}_{almost}(R)$.

For the converse direction, assume that $q_{\mathsf{init}} \in \langle\!\langle 1 \rangle\!\rangle^{event}_{limit}(T)$ is limit-sure eventually synchronizing in T. By Lemma 8, either (1) q_{init} is limit-sure eventually synchronizing in $\mathsf{Pre}^n_{\mathcal{N}}(T)$ for all $n \geq 0$, and then it follows that $\mathsf{Pre}^n_{\mathcal{N}}(T) \neq \emptyset$ for all $n \geq 0$, or (2) q_{init} is sure eventually synchronizing in T, and then since only the action $\sharp$ leaves the state q_{init} (and $\mathsf{post}(q_{\mathsf{init}}, \sharp) = Q$), the characterization of Lemma 4 shows that $Q \subseteq \mathsf{Pre}^k_{\mathcal{N}}(T)$ for some $k \geq 0$, and since $Q \subseteq \mathsf{Pre}_{\mathcal{N}}(Q)$ and $\mathsf{Pre}_{\mathcal{N}}(\cdot)$ is a monotone operator, it follows that $Q \subseteq \mathsf{Pre}^n_{\mathcal{N}}(T)$ for all $n \geq k$ and thus $\mathsf{Pre}^n_{\mathcal{N}}(T) \neq \emptyset$ for all $n \geq 0$. We conclude the proof by noting that $\mathsf{Pre}^n_{\mathcal{M}}(T) = \mathsf{Pre}^n_{\mathcal{N}}(T) \cap Q$ and therefore $\mathsf{Pre}^n_{\mathcal{M}}(T) \neq \emptyset$ for all $n \geq 0$.

□

The example in the proof of Lemma 5 can be used to show that the memory needed by a family of strategies to win limit-sure eventually synchronizing objective (in target $T = \{q_2\}$) is unbounded.

The following theorem summarizes the results for limit-sure eventually synchronizing.

Theorem 4. *For limit-sure eventually synchronizing (with or without exact support) in MDPs:*

1. *(Complexity). The membership problem is PSPACE-complete.*
2. *(Memory). Unbounded memory is required for both pure and randomized strategies, and pure strategies are sufficient.*

Acknowledgment We are grateful to Winfried Just and German A. Enciso for helpful discussions on Boolean networks and for the gadget in the proof of Lemma 3.

References

1. M. Agrawal, S. Akshay, B. Genest, and P. S. Thiagarajan. Approximate verification of the symbolic dynamics of Markov chains. In *Proc. of LICS*, pages 55–64. IEEE, 2012.
2. J. Aspnes and M. Herlihy. Fast randomized consensus using shared memory. *J. Algorithm*, 11(3):441–461, 1990.
3. E. Bach and J. Shallit. *Algorithmic Number Theory, Vol. 1: Efficient Algorithms*. MIT Press, 1996.
4. C. Baier, N. Bertrand, and M. Größer. On decision problems for probabilistic Büchi automata. In *Proc. of FoSSaCS*, LNCS 4962, pages 287–301. Springer, 2008.
5. C. Baier, N. Bertrand, and P. Schnoebelen. On computing fixpoints in well-structured regular model checking, with applications to lossy channel systems. In *Proc. of LPAR*, volume 4246 of *LNCS*, pages 347–361. Springer, 2006.
6. R. Baldoni, F. Bonnet, A. Milani, and M. Raynal. On the solvability of anonymous partial grids exploration by mobile robots. In *Proc. of OPODIS*, LNCS 5401, pages 428–445. Springer, 2008.
7. R. Chadha, V. A. Korthikanti, M. Viswanathan, G. Agha, and Y. Kwon. Model checking MDPs with a unique compact invariant set of distributions. In *Proc. of QEST*, pages 121–130. IEEE Computer Society, 2011.
8. K. Chatterjee and T. A. Henzinger. A survey of stochastic ω-regular games. *J. Comput. Syst. Sci.*, 78(2):394–413, 2012.
9. L. de Alfaro. *Formal Verification of Probabilistic Systems*. PhD thesis, Stanford University, 1997.
10. L. de Alfaro, T. A. Henzinger, and O. Kupferman. Concurrent reachability games. *Theor. Comput. Sci.*, 386(3):188–217, 2007.
11. L. Doyen, T. Massart, and M. Shirmohammadi. Infinite synchronizing words for probabilistic automata. In *Proc. of MFCS*, LNCS 6907, pages 278–289. Springer, 2011.
12. L. Doyen, T. Massart, and M. Shirmohammadi. Infinite synchronizing words for probabilistic automata (Erratum). *CoRR*, abs/1206.0995, 2012.

13. W. Fokkink and J. Pang. Variations on Itai-Rodeh leader election for anonymous rings and their analysis in PRISM. *Journal of Universal Computer Science*, 12(8):981–1006, 2006.
14. H. Gimbert and Y. Oualhadj. Probabilistic automata on finite words: Decidable and undecidable problems. In *Proc. of ICALP (2)*, LNCS 6199, pages 527–538. Springer, 2010.
15. T. A. Henzinger, M. Mateescu, and V. Wolf. Sliding window abstraction for infinite Markov chains. In *Proc. of CAV*, LNCS 5643, pages 337–352. Springer, 2009.
16. M. Holzer. On emptiness and counting for alternating finite automata. In *Developments in Language Theory*, pages 88–97, 1995.
17. P. Jancar and Z. Sawa. A note on emptiness for alternating finite automata with a one-letter alphabet. *Inf. Process. Lett.*, 104(5):164–167, 2007.
18. V. A. Korthikanti, M. Viswanathan, G. Agha, and Y. Kwon. Reasoning about MDPs as transformers of probability distributions. In *Proc. of QEST*, pages 199–208. IEEE Computer Society, 2010.
19. M.A. Pinsky and S. Karlin. *An Introduction to Stochastic Modeling*. Academic Press. Academic Press, 2011.
20. M. Y. Vardi. Automatic verification of probabilistic concurrent finite-state programs. In *Proc. of FOCS*, pages 327–338, 1985.
21. M. V. Volkov. Synchronizing automata and the Cerny conjecture. In *Proc. of LATA*, LNCS 5196, pages 11–27. Springer, 2008.