# Combinations of Qualitative Winning for Stochastic Parity Games

**Krishnendu Chatterjee**
IST Austria, Klosterneuburg, Austria

**Nir Piterman**
University of Gothenburg, Sweden | University of Leicester, UK

## Abstract

We study Markov decision processes and turn-based stochastic games with parity conditions. There are three qualitative winning criteria, namely, sure winning, which requires all paths to satisfy the condition, almost-sure winning, which requires the condition to be satisfied with probability 1, and limit-sure winning, which requires the condition to be satisfied with probability arbitrarily close to 1. We study the combination of two of these criteria for parity conditions, e.g., there are two parity conditions one of which must be won surely, and the other almost-surely. The problem has been studied recently by Berthon et al. for MDPs with combination of sure and almost-sure winning, under infinite-memory strategies, and the problem has been established to be in NP∩co-NP. Even in MDPs there is a difference between finite-memory and infinite-memory strategies. Our main results for combination of sure and almost-sure winning are as follows: (a) we show that for MDPs with finite-memory strategies the problem is in NP ∩ co-NP; (b) we show that for turn-based stochastic games the problem is co-NP-complete, both for finite-memory and infinite-memory strategies; and (c) we present algorithmic results for the finite-memory case, both for MDPs and turn-based stochastic games, by reduction to non-stochastic parity games. In addition we show that all the above complexity results also carry over to combination of sure and limit-sure winning, and results for all other combinations can be derived from existing results in the literature. Thus we present a complete picture for the study of combinations of two qualitative winning criteria for parity conditions in MDPs and turn-based stochastic games.

**2012 ACM Subject Classification** Theory of computation → Logic and verification

**Keywords and phrases** Two Player Games, Stochastic Games, Parity Winning Conditions

**Digital Object Identifier** 10.4230/LIPIcs.CONCUR.2019.2

**Related Version** A full version of the paper is available [18], `https://arxiv.org/abs/1804.03453`.

**Funding** *Krishnendu Chatterjee*: This research is funded by Austrian Science Fund (FWF) NFN Grant S11407-N23 (RiSE/SHiNE), and Vienna Science and Technology Fund (WWTF) Project ICT15-003.
*Nir Piterman*: This research is funded by the ERC consolidator grant D-SynMA under the European Union's Horizon 2020 research and innovation programme (grant agreement No 772459).

## 1 Introduction

*Stochastic games and parity conditions.* Two-player games on graphs are an important model to reason about reactive systems, such as, reactive synthesis [21, 32] and open reactive systems [2]. To reason about probabilistic behaviors of reactive systems, such games are enriched with stochastic transitions, and this gives rise to models such as Markov decision processes (MDPs) [25, 33] and turn-based stochastic games [22]. While these games provide the model for stochastic reactive systems, the specifications for such systems that describe the desired non-terminating behaviors are typically $\omega$-regular conditions [35]. The class of parity winning conditions can express all $\omega$-regular conditions, and has emerged as a convenient and canonical specification for algorithmic studies in the analysis of stochastic reactive systems.

**Table 1** Summary of Results for Sure-Almost-sure as well as Sure-Limit-sure Winning for Parity Conditions. New results are boldfaced. The reductions give algorithmic results from algorithms for non-stochastic games.

| Model | Finite-memory | Infinite-memory |
|---|---|---|
| MDPs | **NP ∩ co-NP**<br>**Reduction to non-stochastic parity games** | NP ∩ co-NP [5] |
| Turn-based stochastic game | **co-NP-complete**<br>**Reduction to non-stochastic games with conjunction of parity conditions** | **co-NP-complete** |

**Table 2** Conjunctions of various qualitative winning criteria.

| Criterion 1 | Criterion 2 | Solution Method |
|---|---|---|
| Sure $\psi_1$ | Sure $\psi_2$ | Sure $(\psi_1 \wedge \psi_2)$ |
| Sure $\psi_1$ | Almost-sure $\psi_2$ | This work |
| Sure $\psi_1$ | Limit-sure $\psi_2$ | This work |
| Almost-sure $\psi_1$ | Almost-sure $\psi_2$ | Almost-sure $(\psi_1 \wedge \psi_2)$ |
| Almost-sure $\psi_1$ | Limit-sure $\psi_2$ | Almost-sure $(\psi_1 \wedge \psi_2)$ |
| Limit-sure $\psi_1$ | Limit-sure $\psi_2$ | Almost-sure $(\psi_1 \wedge \psi_2)$ |

*Qualitative winning criteria.* In the study of stochastic games with parity conditions, there are three basic qualitative winning criteria, namely, (a) *sure winning*, which requires all possible paths to satisfy the parity condition; (b) *almost-sure winning*, which requires the parity condition to be satisfied with probability 1; and (c) *limit-sure winning*, which requires the parity condition to be satisfied with probability arbitrarily close to 1. For MDPs and turn-based stochastic games with parity conditions, almost-sure winning coincides with limit-sure winning, however, almost-sure winning is different from sure winning [9]. Moreover, for all the winning criteria above, if a player can ensure winning, she can do so with memoryless strategies, that do not require to remember the past history of the game. All the above decision problems belong to NP ∩ co-NP, and the existence of polynomial-time algorithm is a major open problem.

*Combination of multiple conditions.* While traditionally MDPs and stochastic games have been studied with a single condition with respect to different winning criteria, in recent studies combinations of winning criteria has emerged as an interesting problem. An example is the *beyond worst-case synthesis* problem that combines the worst-case adversarial requirement with probabilistic guarantee [7]. Consider the scenario that there are two desired conditions, one of which is critical and cannot be compromised at any cost, and hence sure winning must be ensured, whereas for the other condition the probabilistic behavior can be considered. Since almost-sure and limit-sure provide the strongest probabilistic guarantee, this gives rise to stochastic games where one condition must be satisfied surely, and the other almost-surely (or limit-surely). The setting of two objectives have been considered in several prior works; such as in [1], where the primary objective is parity objective and the secondary objective is a quantitative mean-payoff objective; and in [5], where both the primary and the secondary objectives are different parity objectives, but for MDPs.

*Previous results and open questions.* While MDPs and turn-based stochastic games with parity conditions have been widely studied in the literature (e.g., [23, 24, 3, 14, 15, 9]), the study of combination of different qualitative winning criteria is recent. The problem has been studied only for MDPs with sure winning criteria for one parity condition, and almost-sure winning criteria (also probabilistic threshold guarantee) for another parity condition, and it has been established that even in MDPs infinite-memory strategies are required, and the decision problem lies in NP ∩ co-NP [5]. While the existence of infinite-memory strategies represent the general theoretical problem, many important questions have been left open for the problem where both objectives are parity objectives. For example, (i) the analysis for games, which is relevant in reactive synthesis, and (ii) finite-memory strategy synthesis, which represents the synthesis of practical controllers (such as Mealy or Moore machines). an equally important problem is the existence of finite-memory strategies, as the class of finite-memory strategies correspond to realizable finite-state transducers (such as Mealy or Moore machines). In this work we present answers to these open questions, with optimal complexity results.

*Our results.* In this work our main results are as follows:

1. For MDPs with finite-memory strategies, we show that the combination of sure winning and almost-sure winning for parity conditions also belong to NP ∩ co-NP, and we present a linear reduction to parity games. Our reduction implies a quasi-polynomial time algorithm, and also polynomial time algorithm as long as the number of indices for the sure winning parity condition is logarithmic. Note that no such algorithmic result is known for the infinite-memory case for MDPs.
2. For turn-based stochastic games, we show that the combination of sure and almost-sure winning for parity conditions is a co-NP-complete problem, both for finite-memory as well as infinite-memory strategies. For the finite-memory strategy case we present a reduction to non-stochastic games with conjunction of parity conditions, which implies a fixed-parameter tractable algorithm, as well as a polynomial-time algorithm as long as the number of indices of the parity conditions are logarithmic.
3. Finally, while for turn-based stochastic parity games almost-sure and limit-sure winning coincide, we show that in contrast, while ensuring one parity condition surely, limit-sure winning does not coincide with almost-sure winning even for MDPs. However, we show that all the above complexity results established for combination of sure and almost-sure winning also carry over to sure and limit-sure winning.

Our main results are summarized in Table 1. In addition to our main results, we also argue that our results complete the picture of all possible conjunctions of two qualitative winning criteria as follows: (a) conjunctions of sure (or almost-sure) winning with conditions $\psi_1$ and $\psi_2$ is equivalent to sure (resp., almost-sure) winning with the condition $\psi_1 \wedge \psi_2$ (the conjunction of the conditions); (b) by determinacy and since almost-sure and limit-sure winning coincide for $\omega$-regular conditions, if the conjunction of $\psi_1 \wedge \psi_2$ cannot be ensured almost-surely, then the opponent can ensure that at least one of them is falsified with probability bounded away from zero; and thus conjunction of almost-sure winning with limit-sure winning, or conjunctions of limit-sure winning coincide with conjunction of almost-sure winning. This is illustrated in Table 2 and shows that we present a complete picture of conjunctions of two qualitative winning criteria in MDPs and turn-based stochastic games. Full proofs are available in a technical report [18].

**Related work.** We have already mentioned the most important related works above. We discuss other related works here. MDPs with multiple Boolean as well as quantitative objectives have been widely studied in the literature [17, 24, 26, 6, 16]. For non-stochastic games combination of various Boolean objectives is conjunction of the objectives, and such games with multiple quantitative objectives have been studied in the literature [36, 11]. For turn-based stochastic games, the general analysis of multiple quantitative objectives is intricate, and they have been only studied for special cases, such as, reachability objectives [20] and almost-sure winning [4, 10]. However none of these above works consider combinations of qualitative winning criteria. The problem of beyond worst-case synthesis has been studied for MDPs with various quantitative objectives [7, 34], such as long-run average, shortest path, and for parity objectives [5]. In particular [5] studies the problem of satisfying one parity objective surely and maximizing the probability of satisfaction of another parity objective in MDPs with infinite-memory strategies. We extend the literature of the study of beyond worst-case synthesis problem for parity objectives by considering combinations of qualitative winning in both MDPs and turn-based stochastic games, and the distinction between finite-memory and infinite-memory strategies. Thus in contrast to [5] we do not consider optimal probability of satisfaction, but consider turn-based stochastic games as well as finite-memory strategies.

## 2 Background

For a countable set $S$ let $\mathcal{D}(S) = \{d : S \to [0,1] \mid \exists T \subseteq S$ such that $|T| \in \mathbb{N}, \forall s \notin T \, . \, d(s) = 0$ and $\Sigma_{s\in T} d(s) = 1\}$ be the set of discrete probability distributions with finite support over $S$. A distribution $d$ is *pure* if there is some $s \in S$ such that $d(s) = 1$.

A stochastic turn-based game is $G = (V, (V_0, V_1, V_p), E, \kappa)$, where $V$ is a finite set of configurations, $V_0$, $V_1$, and $V_p$ form a partition of $V$ to Player 0, Player 1, and stochastic configurations, respectively, $E \subseteq V \times V$ is the set of edges, and $\kappa : V_p \to \mathcal{D}(V)$ is a probabilistic transition for configurations in $V_p$ such that $\kappa(v, v') > 0$ implies $(v, v') \in E$. If either $V_0 = \emptyset$ or $V_1 = \emptyset$ then $G$ is a Markov Decision Process (MDP). If both $V_0 = \emptyset$ and $V_1 = \emptyset$ then $G$ is a Markov Chain (MC). If $V_p = \emptyset$ then $G$ is a turn-based game (non-stochastic). For an MC $M$, an initial configuration $v$, and a measurable set of paths $W \subseteq V^\omega$, let $\mathsf{Prob}_{M_v}(W)$ denote the measure of $W$.

A set of plays $W \subseteq V^\omega$ is a *parity* condition if there is a parity priority function $\alpha : V \to \{0, \ldots, d\}$, with $d$ as its *index*, such that a play $\pi = v_0, v_1, \ldots$ is in $W$ iff $\min\{c \in \{0, \ldots, d\} \mid \exists^\infty i \, . \, \alpha(v_i) = c\}$ is even. A parity condition with $d = 1$ is a Büchi condition identified with the set $B = \alpha^{-1}(0)$. A parity condition with $d = 2$ and $\alpha^{-1}(0) = \emptyset$ is a co-Büchi condition identified with the set $C = \alpha^{-1}(1)$.

A *strategy* $\sigma$ for Player 0 is $\sigma : V^* \cdot V_0 \to \mathcal{D}(V)$, such that $\sigma(w \cdot v)(v') > 0$ implies $(v, v') \in E$. A strategy $\pi$ for Player 1 is defined similarly. A strategy is *pure* if it uses only pure distributions. Let $w$ range over $V^*$ and $v$ over $V$. A strategy for Player 0 uses memory $m$ if there is a domain $M$ of size $m$ with an initial value $m_0 \in M$ and two functions $\sigma_s : M \times V_0 \to \mathcal{D}(V)$ and $\sigma_u : M \times V \to M$ such that for $v \in V_0$ we have $\sigma(v) = \sigma_s(m_0, v_0)$ and $\sigma(w \cdot v) = \sigma_s(m_w, v)$, where $m_{v_0} = \sigma_m(v_0, m_0)$ and $m_{w \cdot v} = \sigma_m(m_w, v)$. Two strategies $\sigma$ and $\pi$ for both players and an initial configuration $v \in V$ induce a Markov chain $v(\sigma, \pi) = (S(v), (\emptyset, \emptyset, S(v)), E', \kappa')$, where $S(v) = \{v\} \cdot V^*$, $E' = \{(w, w \cdot v)\}$, and if $v \in V_0$ we have $\kappa'(wv) = \sigma(wv)$, if $v \in V_1$ we have $\kappa'(wv) = \pi(wv)$ and if $v \in V_p$ then for every $w \in V^*$ and $v' \in V$ we have $\kappa'(wv, wvv') = \kappa(v, v')$. We denote the set of strategies for Player 0 by $\Sigma$ and the set of strategies for Player 1 by $\Pi$.

For a game $G$, an $\omega$-regular set of plays $W$, and a configuration $v$, the value of $W$ from $v$ for Player 0, denoted $\mathsf{val}_0(W, v)$, and for Player 1, denoted $\mathsf{val}_1(W, v)$, are $\mathsf{val}_0(W, v) = \sup_{\sigma\in\Sigma} \inf_{\pi\in\Pi} \mathsf{Prob}_{v(\sigma,\pi)}(W)$ and $\mathsf{val}_1(W, v) = \sup_{\pi\in\Pi} \inf_{\sigma\in\Sigma} \left(1 - \mathsf{Prob}_{v(\sigma,\pi)}(W)\right)$.

We say that Player 0 wins $W$ *surely* from $v$ if $\exists\sigma \in \Sigma \, . \, \forall\pi \in \Pi \, . \, v(\sigma, \pi) \subseteq W$, where by $v(\sigma, \pi) \subseteq W$ we mean that *all* paths in $v(\sigma, \pi)$ are in $W$. We say that Player 0 wins $W$ *almost surely* from $v$ if $\exists\sigma \in \Sigma \, . \, \forall\pi \in \Pi \, . \, \mathsf{Prob}_{v(\sigma,\pi)}(W) = 1$. We say that Player 0 wins $W$ *limit surely* from $v$ if $\forall r < 1 \, . \, \exists\sigma \in \Sigma \, . \, \forall\pi \in \Pi \, . \, \mathsf{Prob}_{v(\sigma,\pi)}(W) \geq r$. In a given setup (e.g, almost-sure) if Player 0 cannot win we say that Player 1 wins. A strategy $\sigma$ for Player 0 is optimal if $\mathsf{val}_0(W, v) = \inf_{\pi\in\Pi} \mathsf{Prob}_{v(\sigma,\pi)}(W)$. Optimality for Player 1 is defined similarly.

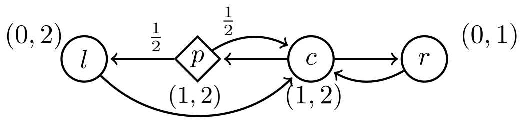


**Figure 1** An SAS MDP where Player 0 requires infinite memory to win [5]. Configuration $p$ is probabilistic and configurations $l$, $c$, and $r$ are Player 0 configurations. The parity is induced by the following priorities $\alpha_s(l) = \alpha_s(r) = 0$, $\alpha_s(p) = \alpha_s(c) = 1$, and $\alpha_{as}(r) = 1$ and $\alpha_{as}(l) = \alpha_{as}(c) = \alpha_{as}(p) = 2$.

A game with condition $W$ is *determined* if for every configuration $v$ we have $\mathsf{val}_0(W, v) + \mathsf{val}_1(W, v) = 1$.

## 3 Sure-Almost-Sure MDPs

Berthon et al. considered the case of MDPs with two parity conditions and finding a strategy that has to satisfy one of the conditions surely and satisfy a given probability threshold with respect to the other [5]. Here we consider the case that the second condition has to hold with probability 1. We consider winning conditions composed of two parity conditions. The goal of Player 0 is to have one strategy such that she can win *surely* for the *sure* winning condition and *almost-surely* for the *almost-sure* winning condition. The authors of [5] show that optimal strategies exist in this case and that it can be decided whether Player 0 can win. Here we revisit their claim that Player 0 may need infinite memory in order to win in such an MDP. We then show that checking whether she can win using a finite-memory strategy is simpler than deciding if there is a general winning strategy.

Given a set of configurations $V$, a sure-almost-sure winning condition is $\mathcal{W} = (W_s, W_{as})$, where $W_s \subseteq V^\omega$ and $W_{as} \subseteq V^\omega$ are two parity winning conditions. A sure-almost-sure (SAS) MDP is $G = (V, (V_0, V_p), E, \kappa, \mathcal{W})$, where all components are as before and where $\mathcal{W}$ is a sure-almost-sure winning condition. Strategies for Player 0 are defined as before. We say that Player 0 wins from configuration $v$ if the same strategy $\sigma$ is winning surely with respect to $W_s$ and almost-surely with respect to $W_{as}$.

▶ **Theorem 1.** [5] *In a finite SAS parity MDP deciding whether a configuration $v$ is winning for Player 0 is in NP ∩ co-NP. Furthermore, there exists an optimal infinite-state strategy for the joint goal.*

There exist SAS MDPs where Player 0 wins but not with finite-memory.

▶ **Theorem 2.** [5] *For SAS MDPs finite-memory strategies do not capture winning.*

In the proof (in [18]) we revisit the MDP in Figure 1 (due to [5]) and repeat their argument showing that there is an infinite-memory strategy that can win both the sure (visit $\{l, r\}$ infinitely often) and almost-sure (visit $\{r\}$ finitely often) winning conditions. Intuitively, longer and longer attempts to reach $l$ at $c$ ensure infinitely many visits to $\{l, r\}$ and finitely many visits to $r$ with probability 1. We present a detailed proof that every finite-memory strategy winning almost-surely is losing with respect to the sure winning condition.

In the proof (in [18]) we prove the following theorem by a chain of reductions. First, reduce the winning in an SAS MDP to the winning in an SAS MDP where the almost-sure winning condition is a Büchi condition. Second, we reduce the winning in an SAS MDP with a Büchi almost-sure winning condition to the winning in a (non-stochastic) game with the winning condition a conjunction of parity and Büchi. This is a special case of Theorem 8. Third, we reduce the winning in a game with a winning condition that is the conjunciton of parity and Büchi to winning in a parity game. Formally, we have the following.

▶ **Theorem 3.** *In order to decide whether it is possible to win an SAS MDP with $n$ locations and indices $d_s$ and $d_{as}$ with finite memory it is sufficient to solve a (non-stochastic) parity game with $O(n \cdot d_s \cdot d_{as})$ configurations and index $d_s$. Furthermore, $d_s$ is a bound on the size of the required memory in case of a win.*

▶ **Corollary 4.** *Consider an SAS MDP with $n$ configurations, sure winning condition of index $d_s$, and almost-sure winning condition of index $d_{as}$. Checking whether Player 0 can win with finite-memory can be computed in quasi-polynomial time. In case that $d_s \leq \log n$ it can be decided in polynomial time.*

**Proof.** This is a direct result of Theorem 3 and the quasi-polynomial algorithm for solving parity games in [8, 30]. ◀

## 4 Sure-Almost-Sure Parity Games

We now turn our attention to sure-almost-sure parity games.

A sure-almost-sure (SAS) parity game is $G = (V, (V_0, V_1, V_p), E, \kappa, \mathcal{W})$, where all components are as before and $\mathcal{W}$ consists of *two* parity conditions $W_s \subseteq V^\omega$ and $W_{as} \subseteq V^\omega$. Strategies and the resulting Markov chains are as before. We say that Player 0 wins $G$ from configuration $v$ if she has a strategy $\sigma$ such that for every strategy $\pi$ of Player 1 we have $v(\sigma, \pi) \subseteq W_s$ and $\mathsf{Prob}_{v(\sigma,\pi)}(W_{as}) = 1$. That is, Player 0 has to win for sure (on all paths) with respect to $W_s$ and with probability 1 with respect to $W_{as}$. Otherwise, Player 1 wins.

### 4.1 Determinacy

We start by showing that SAS parity games are determined.

▶ **Theorem 5.** *SAS parity games are determined.*

In the proof (in [18]) we use a reduction similar to Martin's proof that Blackwell games are determined [31]. We reduce SAS games to turn-based two-player games in a way that preserves winning.

### 4.2 General Winning

We show that determining whether Player 0 has a (general) winning strategy in an SAS parity game is co-NP-complete and that for Player 1 memoryless strategies are sufficient and that deciding her winning is NP-complete.

▶ **Theorem 6.** *In an SAS parity game Player 1 has optimal memoryless strategies.*

The proof (in [18]) is by an inductive argument over the number of configurations of Player 1 (similar to that done in [28, 27, 10]).

▶ **Corollary 7.** *Consider an SAS parity game. Deciding whether Player 1 wins is NP-complete and whether Player 0 wins is co-NP-complete.*

**Proof.** Consider the case of Player 1. The optimal strategy for Player 1 is memoryless. Fixing Player 1's strategy in the game results in an SAS MDP. According to Theorem 1, the winning for Player 0 in SAS MDPs is in NP∩co-NP. The NP algorithm is as follows: it guess the memoryless strategy of Player 1 in the game, and the required polynomial witness of the

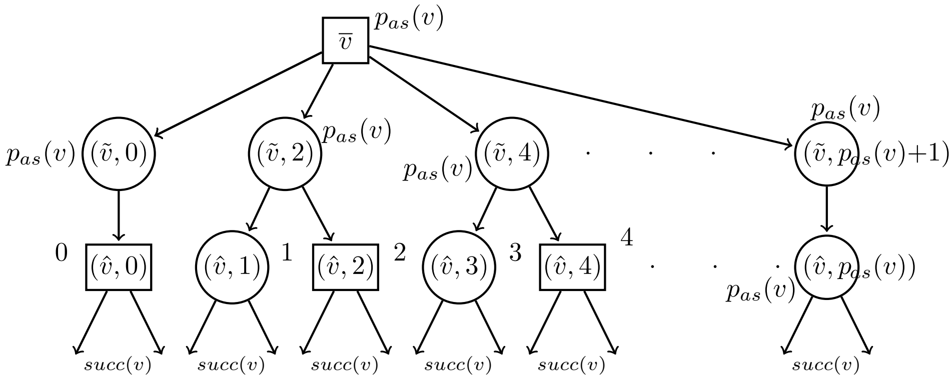


**Figure 2** Gadget replacing probabilistic configurations for a configuration with odd parity.

SAS MDP, and use the polynomial-time verification procedure of the SAS MDP given the witness.[1] Hardness is by considering SAS games with no stochastic configurations [13].

Consider the case of Player 0. Membership in co-NP follows from dualizing the previous argument about membership in NP and determinacy. Hardness follows from considering SAS games with no stochastic configurations [13]. ◀

## 4.3 Winning with Finite Memory

We show that in order to check whether Player 0 can win with finite memory it is enough to use the standard reduction from almost-sure winning in two-player stochastic parity games to sure winning in two-player parity games [15].

▶ **Theorem 8.** *In a finite SAS parity game with $n$ locations and $d_{as}$ almost-sure index deciding whether a node $v$ is winning for Player 0 with finite memory can be decided by a reduction to a two-player (non-stochastic) game with $O(n \cdot d_{as})$ locations, where the winning condition is the intersection of two parity conditions of indices $d_s$ and $d_{as}$.*

The proof has the following steps: Given an SAS parity game $G$, we construct a non-stochastic game $G'$ with conjunction of two objectives with a mapping between configurations of $G$ and $G'$. We show that we can win from a configuration in $G$ if and only if we can win from its mapped configuration in $G'$. In one direction, we show that given winning strategy in $G'$, we can construct winning strategy in $G$ (from the mapped configurations). The construction of the winning strategy is based on the translation of a ranking function in $G'$ to an almost-sure ranking function in $G$. Such a ranking function ensures winning the SAS objective in $G$. . In the other direction, we show that given a winning strategy in $G$, we can construct a winning strategy in $G'$ (from the mapped configurations). As before, the construction of the winning strategy is based on the translation of a ranking function in $G$ to a ranking function in $G'$.

**Proof.** Let $G = (V, (V_0, V_1, V_p), E, \kappa, \mathcal{W})$. Let $p_{as} : V \to [0..d_{as}]$ be the parity priority function that induces $W_{as}$ and $p_s : V \to [0..d_s]$ be the parity priority function that induces $W_s$. Without loss of generality assume that both $d_s$ and $d_{as}$ are even.

[1] Note that we do not require a general NP algorithm with NP ∩co-NP oracle (such algorithms can make polynomially many queries to the oracle, as well as adaptive queries where queries can depend on answers of previous queries). Instead we have a NP algorithm with a single query to a NP∩co-NP oracle, and outputs the answer of the oracle.

Given $G$ we construct the game $G'$ where every configuration $v \in V_p$ is replaced by the gadget in Figure 2. That is, $G' = (V', (V'_0, V'_1), E', \kappa', \mathcal{W}')$, with the following components:

- $V'_0 = V_0 \cup \left\{ (\tilde{v}, 2i), (\hat{v}, 2j-1) \;\middle|\; \begin{array}{l} v \in V_p,\ 2i \in [0..p_{as}(v)+1], \\ \text{and } 2j-1 \in [1..p_{as}(v)] \end{array} \right\}$
- $V'_1 = V_1 \cup \{\overline{v}, (\hat{v}, 2i) \mid v \in V_p \text{ and } 2i \in [0..p_{as}(v)]\}$
- $E' = \{(v,w) \mid (v,w) \in E \cap (V_0 \cup V_1)^2\} \quad \cup \quad \{(v, \overline{w}) \mid (v,w) \in E \cap (V_0 \cup V_1) \times V_p\} \quad \cup$ $\{((\hat{v}, j), w) \mid (v,w) \in E \cap V_p \times (V_0 \cup V_1)\} \quad \cup \quad \{((\hat{v}, j), \overline{w}) \mid (v,w) \in E \cap V_p^2\} \quad \cup$ $\{(\overline{v}, (\tilde{v}, 2i)) \mid v \in V_p\} \quad \cup \quad \{((\tilde{v}, 2i), (\hat{v}, j)) \mid v \in V_p \text{ and } j \in \{2i, 2i-1\}\}$
- $\mathcal{W}' = W'_s \cap W'_{as}$, where $W'_s$ and $W'_{as}$ are the parity winning sets that are induced by the following priority functions.

$$p'_{as}(t) \quad = \quad \begin{cases} p_{as}(t) & t \in V_0 \cup V_1 \\ p_{as}(v) & t \in \{\overline{v}, (\tilde{v}, 2i)\} \\ j & t = (\hat{v}, j) \end{cases} \qquad p'_s(t) \quad = \quad \begin{cases} p_s(t) & t \in V_0 \cup V_1 \\ p_s(v) & t \in \{\overline{v}, (\tilde{v}, 2i), (\hat{v}, j)\} \end{cases}$$

We show that Player 0 surely wins from a configuration $v \in V_0 \cup V_1$ in $G'$ iff she wins from $v$ in $G$ with a pure finite-memory strategy and she wins from $\overline{v} \in V'$ in $G'$ iff she wins from $v$ in $G$ with a pure finite-memory strategy.

The game $G'$ is a linear game whose winning condition (for Player 0) is an intersection of two parity conditions. It is known that such games are determined and that the winning sets can be computed in NP $\cap$ co-NP [13]. Indeed, the winning condition for Player 0 can be expressed as a Streett condition, and hence her winning can be decided in co-NP. The winning condition for Player 1 can be expressed as a Rabin condition, and hence her winning can be decided in NP. It follows that $V'$ can be partitioned to $W'_0$ and $W'_1$, the winning regions of Player 0 and Player 1, respectively. Furthermore, Player 0 has a pure finite-memory winning strategy for her from every configuration in $W'_0$ and Player 1 has a pure memoryless winning strategy for her from every configuration in $W'_1$. Let $\sigma'_0$ denote the winning strategy for Player 0 on $W'_0$ and $\pi'_1$ denote the winning strategy for Player 1 on $W'_1$. Let $M$ be the memory domain used by $\sigma'_0$. As $\sigma'_0$ is pure, we can think about it as $\sigma'_0 \subseteq V' \times M \to V' \times M$, where for every $m \in M$ and $v \in V'_0$ there is a unique $w \in V$ and $m' \in M$ such that $((v,m),(w,m')) \in \sigma'$ and for every $m \in M$ and $v \in V'_1$ and $w$ such that $(v,w) \in E'$ there is a unique $m'$ such that $((v,m),(w,m')) \in \sigma'_0$. We freely say $\sigma'_0$ chooses $v'$ from $(v,m)$ for the unique $v'$ such that $(v,m,v',m') \in \sigma'_0$ for some $m'$ and $\sigma'_0$ updates the memory to $m'$. Similarly, a pure strategy in $G$ can be described as $\sigma \subseteq (V \times M)^2$ where stochastic configurations are handled like Player 1 configuration in term of memory update for all successors as above. By abuse of notation we refer to the successor of a configuration $v$ in $G'$ and mean either $w$ or $\overline{w}$ according to the context.

$\Leftarrow$ We show that every configuration $v \in W'_0$ that is winning for Player 0 in $G'$ is in the winning region $W_0$ of Player 0 in $G$. Consider the strategy $\sigma'_0 \subseteq (V' \times M)^2$. We construct a winning strategy $\sigma_0 \subseteq (V \times M)^2$, induced by $\sigma'_0$ as follows:

- For a configuration-memory (cm) pair $(v,m) \in V_0 \times M$ there is a unique cm pair $(v',m')$ such that $(v,m,v',m') \in \sigma'_0$. We set $(v,m,v',m') \in \sigma_0$.
- For a cm pair $(v,m) \in V_1 \times M$ and for every successor $w$ of $v$ there is a unique memory value $m'$ such that $(v,m,w,m') \in \sigma'_0$. We set $(v,m,w,m') \in \sigma_0$.
- Consider a cm pair $(v,m) \in V_p \times M$. As $\overline{v}$ is a Player 1 configuration in $G'$, for every configuration $(\tilde{v}, 2i)$ there is a uniqye $m'$ such that $(v,m,(\tilde{v},2i),m') \in \sigma'_0$.
  - If for some $i$ we have that the choice from $(\tilde{v}, 2i)$ according to $\sigma'_0$ is $(\hat{v}, 2i-1)$. Then, let $i_0$ be the minimal such $i$ and let $w_0$ be the successor of $v$ such that the choice of $\sigma'_0$ from $(\hat{v}, 2i_0 - 1)$ is $w_0$. We update in $\sigma_0$ the tuple $(v,m,w_0,m')$, where $m'$ is the

memory resulting from taking the path $\overline{v}$, $(\tilde{v}, 2i_0)$, $(\hat{v}, 2i_0-1)$, $w_0$ in $G'$ based on $\sigma_0'$. We update in $\sigma_0$ the tuple $(v, m, w', m_{w'})$ for $w' \neq w_0$, where $m_{w'}$ is the memory resulting from taking the path $\overline{v}$, $(\tilde{v}, 2i_0-2)$, $(\hat{v}, 2i_0-2)$, $w'$. Notice that as $i_0$ is chosen to be the minimal the choice from $(\tilde{v}, 2i_0-2)$ to $(\hat{v}, 2i_0-2)$ is compatible with $\sigma_0'$, where $2i_0-2$ could be 0.

* If for all $i$ we have that the choice from $(\tilde{v}, 2i)$ according to $\sigma_0'$ is $(\hat{v}, 2i)$. Then, for every $w$ successor of $v$ we update in $\sigma_0$ the tuple $(v, m, w, m')$, where $m'$ is the memory resulting from taking the path $\overline{v}$, $(\tilde{v}, p_{as}(v))$, $(\hat{v}, p_{as})$, $w$.

Notice that if $p_{as}(v)$ is odd then the first case always holds as the only successor of $(\tilde{v}, p_{as}(v)+1)$ is $(\hat{v}, p_{as}(v))$.

The resulting strategy $\sigma_0$ includes no further decisions for Player 0. Consider the winning condition $W_s$. Every path in $G$ that is consistent with $\sigma_0$ (with proper memory updates) corresponds to a path in $G'$ that is consistent with $\sigma_0'$ (with the same memory updates) and agrees on the parities of all configurations according to $p_s$. Indeed, every configuration of the form $(\tilde{v}, 2i)$ or $(\hat{v}, j)$ in $G'$ has the same priority according to $p_s$ as $\overline{v}$ (and $v$ in $G$). As every path consistent with $\sigma_0'$ is winning according to $W_s'$ then every path in $G$ consistent with $\sigma$ is winning according to $W_s$.

We turn our attention to consider only the parity condition $p_{as}$ in both $G'$ and $G$. We think about $G'$ as a parity game with the winning condition $W_{as}'$ and about $G$ as a stochastic parity game with the winning condition $W_{as}$. As $\sigma_0'$ is winning, all paths in $G'$ (with proper memory updates) are winning for Player 0 according to $W_{as}'$.

We recall some definitions and results from [15]. For $k \leq d_{as}$, let $\underline{k}$ denote $k$ if $k$ is odd and $k-1$ if $k$ is even. A *parity ranking* for Player 0 is $\vec{r} : V' \times M \to [n]^{d_{as}/2} \cup \{\infty\}$ for some $n \in \mathbb{N}$, where $[n]$ denotes $\{0, \ldots, n\}$. For a configuration $v$, Let $\vec{r}(v) = (r_1, \ldots, r_d)$ and $\vec{r}(v') = (r_1', \ldots, r_d')$, where $d = d_{as}/2$. For $v$, we denote by $\vec{r}^k(v)$ the prefix $(r_1, r_3, \ldots, r_{\underline{k}})$ of $\vec{r}(v)$. We write $\vec{r}(v) \leq_k \vec{r}(v')$ if the prefix $(r_1, \ldots r_{\underline{k}})$ is at most $(r_1', \ldots, r_{\underline{k}}')$ according to the lexicographic ordering. Similarly, we write $\vec{r}(v) <_k \vec{r}(v')$ if $(r_1, \ldots r_{\underline{k}})$ is less than $(r_1', \ldots, r_{\underline{k}}')$ according to the lexicographic ordering.

A parity ranking is good if (i) for every vertex $v \in V_0$ and memory $m \in M$ there is a vertex $w \in succ(v)$ and $m' \in M$ such that $\vec{r}(w, m') \leq_{p(v)} \vec{r}(v, m)$ and if $p(v)$ is odd then $\vec{r}(w, m') <_{p(v)} \vec{r}(v, m)$ and (ii) for every vertex $v \in V_1$, memory $m \in M$, and vertex $w \in succ(v)$ it holds that there is a $m' \in M$ such that $\vec{r}(w, m') \leq_{p(v)} \vec{r}(v, m)$ and if $p(v)$ is odd then $\vec{r}(w, m') <_{p(v)} \vec{r}(v, m)$. It is well known that in a parity game (here $G'$ combined with the strategy $\sigma_0'$) there is a *good* parity ranking such that for every $v \in W_0'$ and memory $m \in M$ we have $\vec{r}(v, m) \neq \infty$ [29]. Let $\vec{r}$ be the good parity ranking for $G'$. Consider the same ranking for $G$ with the same memory $M$. For a cm pair $(v, m) \in V_p \times M$, we write $\mathsf{Prob}_{v,m}(\vec{r}_{\leq_k})$ for the probability (according to $\kappa$) of successors $w$ of $v$ such that for some memory values $m_w$ we have $\vec{r}(w, m_w) \leq_k \vec{r}(v, m)$ and $\mathsf{Prob}_{v,m}(\vec{r}_{<_k})$ for the probability of successors $w$ of $v$ such that for some memory values $m_w$ we have $\vec{r}(w, m_w) <_k \vec{r}(v, m)$.

▶ **Definition 9** (Almost-sure ranking [14]). *A ranking function $\vec{r} : V \times M \to [n]^{d_{as}/2} \cup \{\infty\}$ for Player 0 is an almost-sure ranking if there is an $\epsilon \geq 0$ such that for every pair $(v, m)$ with $r(v, m) \neq \infty$, the following conditions hold:*

- *If $v \in V_0$ there exists a successor $w$ and memory $m'$ such that $\vec{r}(w, m') \leq_{p(v)} \vec{r}(v, m)$ and if $p(v)$ is odd then $\vec{r}(w, m') <_{p(v)} \vec{r}(v, m)$.*
- *If $v \in V_1$ then for every successor $w$ of $v$ there is a memory $m'$ such that $\vec{r}(w, m') \leq_{p(v)} \vec{r}(v, m)$ and if $p(v)$ is odd then $\vec{r}(w, m') <_{p(v)} \vec{r}(v, m)$.*

- *If $v \in V_p$ and $p(v)$ is even then either* $\mathsf{Prob}_{v,m}(\vec{r}_{\leq_{p(v)-1}}) = 1$ *or*

$$\bigvee_{j=2i+1\in[1..p(v)]} (\mathsf{Prob}_{v,m}(\vec{r}_{\leq_{j-2}}) = 1 \wedge \mathsf{Prob}_{v,m}(\vec{r}_{<_j}) \geq \epsilon)$$

- *If $v \in V_p$ and $p(v)$ is odd then* $\bigvee_{j=2i+1\in[1..p(v)]} \big(\mathsf{Prob}_{v,m}(\vec{r}_{\leq_{j-2}}) = 1 \wedge \mathsf{Prob}_{v,m}(\vec{r}_{<_j}) \geq \epsilon\big)$

▶ **Lemma 10.** [14] *A stochastic parity game has an almost-sure ranking iff Player 0 can win for the parity objective with probability 1 from every configuration $v$ such that for some $m$ we have $\vec{r}(v,m) \neq \infty$.*

The following lemma specializes a similar lemma in [14] for our needs.

▶ **Lemma 11.** *The good ranking of $G'$ with $M$ induces an almost-sure ranking of $G$ with $M$.*

**Proof.** Let $\epsilon$ be the minimal probability of a transition in $G$. As $G$ is finite $\epsilon$ exists. For configurations in $V_0 \cup V_1$ the definitions of good parity ranking and almost-sure ranking coincide.

Consider a configuration $v \in V_p$ a memory $m \in M$ and the matching configuration $\overline{v}$. Let $p = p_{as}(v)$. Consider the pair $(v,m)$ in $V \times M$ and $(\overline{v},m)$ in $V' \times M$. We consider the cases where $p$ is even and when $p$ is odd.

- Suppose that $p$ is even. If there is some minimal $i$ such that the choice of $\sigma_0'$ from $((\tilde{v},2i),m')$ in $G'$ is $((\hat{v},2i-1),m'')$. Then, there is some $w \in succ(v)$ and some $m'''$ such that $\vec{r}(w,m''') <_{2i-1} \vec{r}((\hat{v},2i-1),m'') \leq_p \vec{r}((\tilde{v},2i),m') \leq_p \vec{r}(\overline{v},m)$. It follows that $\mathsf{Prob}_{v,m}(\vec{r}_{<_{2i-1}}) \geq \epsilon$. Furthermore, as $i$ is minimal it follows that $i \neq 0$ and that the choice of $\sigma_0'$ from $((\tilde{v},2i-2),n)$ is $((\hat{v},2i-2),n')$ and $(\tilde{v},2i-2)$ belongs to Player 1 in $G'$. Then, for every successor $w$ of $(\hat{v},2i-2)$ and for every memory value $n'$ there is a memory value $n'''$ such that

$$\vec{r}(w,n''') \leq_{2i-2} \vec{r}((\hat{v},2i-2),n'') \leq_p ((\tilde{v},2i-2),n') \leq_p (\overline{v},m).$$

  It follows that $\mathsf{Prob}_{v,m}(\vec{r}_{\leq_{2i-2}}) = 1$.
  If there is no such $i$, then the choice of $\sigma_0'$ from $((\tilde{v},p),m')$ in $G'$ is $((\hat{v},p),m'')$ and for every $w \in \mathsf{succ}(v)$ there is some $m'''$ such that

$$\vec{r}(w,m''') \leq_p \vec{r}((\hat{v},p),m'') \leq_p \vec{r}((\tilde{v},p),m') \leq_p \vec{r}(\overline{v},m).$$

  If follows that $\mathsf{Prob}_{v,m}(\vec{r}_{\leq_p}) = 1$.
- Suppose that $p$ is odd. In this case there must be some minimal $i$ such that the choice of $\sigma_0'$ from $((\tilde{v},2i),m')$ is $((\hat{v},2i-1),m'')$. We can proceed as above.

◀

As Player 0 has no further choices in $G$, it follows that the strategy $\sigma_0$ defined above is winning in $G$. That is, sure winning w.r.t. $W_s$ and almost-sure winning w.r.t. $W_{as}$.

⇒ In the proof (in [18]) we show how to use a winning finite-memory strategy in $G$ to induce a strategy in $G'$ and use a ranking argument to show that this strategy is winning.

◀

▶ **Corollary 12.** *Consider an SAS turn-based stochastic parity game. Deciding whether Player 0 can win with finite-memory is co-NP-complete. Deciding whether Player 1 can win against finite-memory is NP-complete.*

**Proof.** Upper bounds follow from the reductions to Streett and Rabin winning conditions. Completeness follows from the case where the game has no stochastic configurations [13]. ◀

▶ Remark 13. The complexity established above in the case of finite-memory is the same as that established for the general case in Corollary 7. However, this reduction gives us a clear algorithmic approach to solve the case of finite-memory strategies. Indeed, in the general case, the proof of the NP upper bound requires enumeration of all memoryless strategies, and does not present an algorithmic approach, regardless of the indices of the different winning conditions. In contrast our reduction for the finite-memory case to non-stochastic games with conjunction of parity conditions and recent algorithmic results on non-stochastic games with $\omega$-regular conditions of [8] imply the following:

- For the finite-memory case, we have a fixed parameter tractable algorithm that is polynomial in the number of the game configurations and exponential only in the indices to compute the SAS winning region.
- For the finite-memory case, if both indices are constant or logarithmic in the number of configurations, we have a polynomial time algorithm to compute the SAS winning region.

# 5 Sure-Limit-Sure Parity Games

In this section we extend our results to the case where the unsure goal is required to be met with limit-sure certainty, rather than almost-sure certainty.

*Sure-limit-sure parity games.* A *sure-limit-sure (SLS) parity game* is, as before, $G = (V, (V_0, V_1, V_p), E, \kappa, \mathcal{W})$. We denote the second winning condition with the subscript $ls$, i.e., $W_{ls}$. We say that Player 0 wins $G$ from configuration $v$ if she has a sequence of strategies $\sigma_i \in \Sigma$ such that for every $i$ for every strategy $\pi$ of Player 1 we have $v(\sigma_i, \pi) \subseteq W_s$ and $\mathsf{Prob}_{v(\sigma_i,\pi)}(W_{ls}) \geq 1 - \frac{1}{i}$. That is, Player 0 has a sequence of strategies that are sure winning (on all paths) with respect to $W_s$ and ensure satisfaction probabilities approaching 1 with respect to $W_{ls}$.

## 5.1 Limit-Sure vs Almost-Sure

In MDPs and stochastic turn-based games with parity conditions almost-sure and limit-sure winning coincide [9]. In contrast to the above result we present an example MDP where in addition to surely satisfying one parity condition limit-sure winning with another parity condition can be ensured, but almost-sure winning cannot be ensured. In other words, in conjunction with sure winning, limit-sure winning does not coincide with almost-sure winning even for MDPs. Such a result was established in [5] for MDPs with infinite-memory strategies. We show the same holds for finite-memory strategies.

▶ **Theorem 14.** *While satisfying one parity condition surely, the almost-sure winning set for another parity condition is a strict subset of limit-sure winning set, even in the context of MDPs with finite-memory strategies.*

**Proof.** Consider the MDP in Figure 3. Clearly, Player 0 wins surely with respect to both parity conditions in configuration $r$ and Player 0 cannot win the condition $W_{ls}$ on $l$. In order to win $W_s$ the cycle between $p$ and $c$ has to be taken finitely often. Then, the edge from $c$ to $l$ must be taken eventually. However, $l$ is a sink that is losing with respect to $W_{ls}$. It follows, that Player 0 cannot win almost-surely with respect to $W_{ls}$ while winning surely with respect to $W_s$.

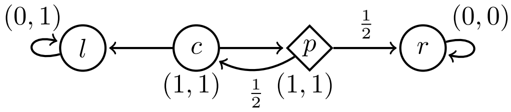


**Figure 3** An MDP where Player 0 can ensure sure winning and win limit-surely but cannot win almost-surely. Configuration $p$ is probabilistic and configurations $l$, $c$, and $r$ are Player 0 configurations. The winning conditions are induced by the following priorities $\alpha_s(l) = \alpha_s(r) = 0$, $\alpha_s(p) = \alpha_s(c) = 1$, and $\alpha_{ls}(r) = 0$ and $\alpha_{ls}(l) = \alpha_{ls}(c) = \alpha_{ls}(p) = 1$.

On the other hand, for every $\epsilon > 0$ there is a finite-memory strategy that is sure winning with respect to $W_s$ and wins with probability at least $1 - \epsilon$ with respect to $W_{ls}$. Indeed, Player 0 has to choose the edge from $c$ to $p$ at least $N$ times, where $N$ is large enough such that $\frac{1}{2^N} < \epsilon$, and then choose the edge from $c$ to $l$. Then, Player 0 wins surely with respect to $W_s$ (every play eventually reaches either $l$ or $r$) and with probability more than $1 - \epsilon$ with respect to $W_{ls}$.

To summarize, Player 0 wins surely w.r.t. $W_s$ and limit-surely w.r.t. $W_{ls}$ from both $c$ and $p$ but cannot win almost-surely w.r.t. $W_{ls}$ from $c$ and $p$. ◀

## 5.2 Solving SLS MDPs and Games

We first note that Player 1 has optimal memoryless strategies similar to the SAS case. The proof (in [18]) reuses the proof of Theorem 6.

▶ **Theorem 15.** *In an SLS parity game Player 1 has optimal memoryless strategies.*

**SLS MDPs.** We now present the solution to winning in SLS MDPs. Given an SLS MDP $G$ with winning conditions $W_s$ and $W_{ls}$, we call the *induced* SAS MDP the MDP with winning conditions $W_s$ and $W_{ls}$, where the latter is interpreted as an almost-sure winning condition. We use the induced SAS MDP in the solution of the SLS MDP. The memory used in the SLS part has to match the memory used for winning in the SAS part. That is, if Player 0 is restricted to finite-memory in the SLS part of the game she has to consider finite-memory strategies in the induced SAS MDP.

▶ **Theorem 16.** *In a finite SLS parity MDP deciding whether a node $v$ is winning for Player 0 can be reduced to the limit-sure reachability while maintaining sure-parity. The target of the limit-sure reachability is the winning region of the induced SAS partiy MDP.*

**Proof.** *SAS winning region $A$.* Consider an MDP $G = (V, (V_0, V_p), E, \kappa, \mathcal{W})$, where $\mathcal{W} = (W_s, W_{ls})$. Consider $G$ as an SAS MDP and compute the set of configurations from which Player 0 can win $G$. Let $A \subseteq V$ denote this winning region and $B = V \setminus A$ be the complement region. Clearly, $A$ is *closed* under probabilistic moves. That is, if $v \in V_p \cap A$ then for every $v'$ such that $(v, v') \in E$ we have $v' \in A$. Furthermore, under Player 0's winning strategy, Player 0 does not use edges going back from $A$ to $B$. It follows that we can consider $A$ as a sink in $G$.

*Reduction to limit-sure reachability.* We present the argument for finite-memory strategies for Player 0, and the argument for infinite-memory strategies is similar. Consider an arbitrary finite-memory strategy $\sigma \in \Sigma$, and consider the Markov chain that is the result of restricting Player 0 moves according to $\sigma$.

- *Bottom SCC property.* Let $S$ be a bottom SCC (SCC that is only reachable from itself) that intersects with $B$ in the Markov chain. As explained above, it cannot be the case that this SCC intersects $A$ (since we consider $A$ as sink due to the closed property). Thus the SCC $S$ must be contained in $B$. Thus, either $S$ must be losing according to $W_s$ or the minimal parity in $S$ according to $W_{ls}$ is odd, as otherwise in the region $S$ Player 0 ensures sure winning wrt $W_s$ and almost-sure winning wrt $W_{ls}$, which means that $S$ belongs to the SAS winning region $A$. This contradicts that $S$ is contained in $B$.

- *Reachability to A.* In a Markov chain bottom SCCs are reached with probability 1, and from the above item it follows that the probability to satisfy the $W_{ls}$ goal along with ensuring $W_s$ while reaching bottom SCCs in $B$ is zero. Hence, the probability to satisfy $W_{ls}$ along with ensuring $W_s$ is at most the probability to reach $A$. On the other hand, after reaching $A$, the SAS goal can be ensured by switching to an appropriate SAS strategy in the winning region $A$, which implies that the SLS goal is ensured. Hence it follows that the SLS problem reduces to limit-sure reachability to $A$, while ensuring the sure parity condition $W_s$.

◀

▶ Remark 17. Note that for finite-memory strategies the argument above is based on bottom SCCs. The SAS region for MDPs wrt to infinite-memory strategies is achieved by characterizing certain strongly connected components (called Ultra-good end-components [5, Definition 5]), and hence a similar argument as above also works for infinite-memory strategies to show that SLS for infinite-memory strategies for two parity conditions reduces to limit-sure reachability to the SAS region while ensuring the sure parity condition (however, in this case the SAS region has to be computed for infinite-memory strategies).

**Limit-sure reachability and sure parity in games.** We consider the problem of Player 0 ensuring limit-sure reachability to target set $A$ while preserving sure parity. We present the solution for games (which subsumes the case of MDPs).

▶ **Theorem 18.** *Consider an SLS Game, where the limit-sure condition is to reach a target set $A$ that is also winning for the sure condition. Player 0's winning region is the limit-sure reachability region to $A$ within the winning region of the sure parity condition.*

In one direction, in the limit-sure reachability to $A$ within the sure winning region, the limit-sure reachability strategy can be played to enforce high probability of winning for the limit-sure winning condition and then revert to the sure-winning strategy. The combination delivers an arbitrarily high probability of reaching $A$ as well as sure winning. In the other direction, a strategy that wins limit-sure reachability to $A$ and sure-winning with respect to the sure condition is clearly restricted to the sure-winning region. At the same time, it ensures limit-sure reachability to A. Hence, the analysis of such games is simplified into two steps; first compute the sure winning region for the sure objective, and in this subgame only consider reachability to the limit-sure target set.

**Proof.** WLOG we replace the region $A$ by a single configuration $t$ with a self loop and an even priority with respect to $W_s$. Consider an SLS game, with a configuration $t$ of sink target state, such that the limit-sure goal is to reach $t$, and $t$ has even priority with respect to $W_s$. We now present solution to this limit-sure reachability with sure parity problem. The computational steps are as follows:

- First, compute the sure winning region w.r.t the parity condition in the game. Let $X$ be this winning region. Note that $t \in X$ as $t$ is a sink state with even priority for $W_s$.
- Second, restrict the game to $X$ and compute limit-sure reachability region to $t$, and let the region be $Y$. Note that the game restricted to $X$ is a turn-based stochastic game where almost-sure and limit-sure reachability coincide.

Let us denote by $Z$ the desired winning region (i.e., from where sure parity can be ensured along with limit-sure reachability to $t$). We argue that $Y$ computes the desired winning region $Z$ as follows:

- First, note that since the sure parity condition $W_s$ must be ensured, the sure winning region $X$ must never be left. Thus without loss of generality, we can restrict the game to $X$. By definition $Y$ is the region in $X$ to ensure limit-sure reachability to $t$. As $Z$ ensures both limit-sure reachability to $t$ as well as sure parity, it follows that $Z$ is a subset of $Y$.
- Second, for any $\epsilon > 0$, there is a strategy in $Y$ to ensure that $t$ is reached with probability at least $1 - \epsilon$ within $N_\epsilon$ steps staying in $X$ (since in the subgame restricted to $X$, almost-sure reachability to $t$ can be ensured). Consider a strategy that plays the above strategy for $N_\epsilon$ steps, and if $t$ is not reached, then switches to a sure winning strategy for $W_s$ (such a strategy exists since $X$ is never left, and parity conditions are independent of finite prefixes). It follows that from $Y$ both limit-sure reachability to $t$ as well as sure parity condition $W_s$ can be ensured. Hence $Y \subseteq Z$.

Thus, $Y = Z$ as required. ◀

▶ **Corollary 19.** *Consider an SLS turn-based stochastic parity game. Deciding whether Player 0 wins is co-NP-complete. Deciding whether Player 1 wins is NP-complete. Consider an SLS turn-based MDP with $n$ locations and indices $d_s$ and $d_{ls}$. Checking whether Player 0 can win with finite-memory can be computed in quasi-polynomial time. In case that $d_s \leq \log n$ it can be decided in polynomial time.*

**Proof.** It follows from above that to solve SLS MDPs, the following computation steps are sufficient: (a) solve SAS MDP, (b) compute sure winning region for parity condition, and (c) compute almost-sure (=limit-sure) reachability in MDPs. The second step is a special case of the first step, and the third step can be achieved in polynomial time [12, 19]. Hence it follows that all the complexity and algorithmic upper bounds we established for the SAS MDPs carry over to SLS MDPs. For games, since Player 1 has memoryless optimal strategies (Theorem 15) and the complexity of SAS MDPs and SLS MDPs coincide, the complexity upper bounds for SAS games carry over to SLS games. Finally, since the complexity lower bound results for SAS parity games follow from games with no stochastic transitions, they apply to SLS parity games as well. ◀

## 6 Conclusions and Future Work

In this work we consider MDPs and turn-based stochastic games with two parity winning conditions, with combinations of qualitative winning criteria. In particular, we study the case where one winning condition must be satisfied surely, and the other almost-surely (or limit-surely). We present results for MDPs with finite-memory strategies, and turn-based stochastic games with finite-memory and infinite-memory strategies. Our results establish complexity results, as well as algorithmic results for finite-memory strategies by reduction to non-stochastic games. Some interesting directions for future work are as follows. First, while our results establish algorithmic results for finite-memory strategies, whether similar results can be established for infinite-memory strategies is an interesting open question. Second, the study of the synthesis problem for turn-based stochastic games with combinations of quantitative objectives is another interesting direction of future work. If we consider more than two conjuncts with only two types, i.e., sure and almost-sure, or sure and limit-sure, then solution of the game reduces to a conjunction of two conditions. The problem of conjunctions with more than two types and general Boolean combinations of winning conditions are interesting directions for future work.

## References


1. S. Almagor, O. Kupferman, and Y. Velner. Minimizing expected cost under hard boolean constraints, with applications to quantitative synthesis. In *27th International Conference on Concurrency Theory*, volume 59 of *LIPIcs*, pages 9:1–9:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.
2. R. Alur, T.A. Henzinger, and O. Kupferman. Alternating-time temporal logic. *Journal of the ACM*, 49(5):672–713, 2002.
3. C. Baier and J.-P. Katoen. *Principles of Model Checking*. MIT Press, 2008.
4. N. Basset, M.Z. Kwiatkowska, U. Topcu, and C. Wiltsche. Strategy synthesis for stochastic games with multiple long-run objectives. In *Proc. 21st Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems*, volume 9035 of *Lecture Notes in Computer Science*, pages 256–271. Springer, 2015.
5. R. Berthon, M. Randour, and J.-F. Raskin. Threshold constraints with guarantees for parity objectives in markov decision processes. In *44th International Colloquium on Automata, Languages, and Programming*, volume 80 of *LIPIcs*, pages 121:1–121:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017.
6. T. Brázdil, V. Brozek, K. Chatterjee, V. Forejt, and A. Kucera. Two views on multiple mean-payoff objectives in Markov decision processes. In *Proc. 26rd IEEE Symp. on Logic in Computer Science*, pages 33–42. IEEE Computer Society Press, 2011.
7. V. Bruyère, E. Filiot, M. Randour, and J.-F. Raskin. Meet your expectations with guarantees: Beyond worst-case synthesis in quantitative games. *Inf. Comput.*, 254:259–295, 2017.
8. C. Calude, S. Jain, B. Khoussainov, W. Li, and F. Stephan. Deciding parity games in quasipolynomial time. In *Proc. 49th ACM Symp. on Theory of Computing*, pages 252–263. ACM Press, 2017.
9. K. Chatterjee. *Stochastic $\omega$-regular Games*. PhD thesis, University of California at Berkeley, 2007.
10. K. Chatterjee and L. Doyen. Perfect-information stochastic games with generalized mean-payoff objectives. In *Proc. 31st IEEE Symp. on Logic in Computer Science*, pages 247–256. ACM Press, 2016.
11. K. Chatterjee, L. Doyen, M. Randour, and J.-F. Raskin. Looking at mean-payoff and total-payoff through windows. *Information and Computation*, 242:25–52, 2015.
12. K. Chatterjee and M. Henzinger. Faster and dynamic algorithms for maximal end-component decomposition and related graph problems in probabilistic verification. In *SODA'11*. SIAM, 2011.
13. K. Chatterjee, T.A. Henzinger, and N. Piterman. Generalized parity games. In *Proc. 10th Int. Conf. on Foundations of Software Science and Computation Structures*, volume 4423 of *Lecture Notes in Computer Science*, pages 153–167, Braga, Porgugal, 2007. Springer.
14. K. Chatterjee, M. Jurdziński, and T.A. Henzinger. Simple stochastic parity games. In *Proc. 12th Annual Conf. of the European Association for Computer Science Logic*, Lecture Notes in Computer Science, pages 100–113. Springer, 2003.
15. K. Chatterjee, M. Jurdziński, and T.A. Henzinger. Quantitative stochastic parity games. In *Symposium on Discrete Algorithms*, pages 114–123. SIAM, 2004.
16. K. Chatterjee, Z. Komárková, and J. Kretínský. Unifying two views on multiple mean-payoff objectives in Markov Decision Processes. In *Proc. 30th IEEE Symp. on Logic in Computer Science*, pages 244–256. IEEE Computer Society Press, 2015.
17. K. Chatterjee, R. Majumdar, and T.A. Henzinger. Markov Decision Processes with multiple objectives. In *Proc. 23rd Symp. on Theoretical Aspects of Computer Science*, volume 3884 of *Lecture Notes in Computer Science*, pages 325–336. Springer, 2006.
18. K. Chatterjee and N. Piterman. Combinations of qualitative winning for stochastic parity games. Technical report, arXiv:1804.03453, 2018.
19. Krishnendu Chatterjee and Monika Henzinger. Efficient and dynamic algorithms for alternating büchi games and maximal end-component decomposition. *J. ACM*, 61(3):15:1–15:40, 2014.

**20** T. Chen, V. Forejt, M.Z. Kwiatkowska, A. Simaitis, and C. Wiltsche. On stochastic games with multiple objectives. In *38th Int. Symp. on Mathematical Foundations of Computer Science*, volume 8087 of *Lecture Notes in Computer Science*, pages 266–277. Springer, 2013.

**21** A. Church. Logic, arithmetics, and automata. In *Proc. Int. Congress of Mathematicians, 1962*, pages 23–35. Institut Mittag-Leffler, 1963.

**22** A. Condon. The complexity of stochastic games. *Information and Computation*, 96:203–224, 1992.

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

**24** K. Etessami, M.Z. Kwiatkowska, M.Y. Vardi, and M. Yannakakis. Multi-objective model checking of markov decision processes. *Logical Methods in Computer Science*, 4(4), 2008.

**25** Jerzy Filar and Koos Vrieze. *Competitive Markov decision processes*. Springer, 1996.

**26** V. Forejt, M.Z. Kwiatkowska, G. Norman, D. Parker, and H. Qu. Quantitative multi-objective verification for probabilistic systems. In *Proc. 17th Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems*, volume 6605 of *Lecture Notes in Computer Science*, pages 112–127. Springer, 2011.

**27** H. Gimbert and E. Kelmendi. Two-player perfect-information shift-invariant submixing stochastic games are half-positional. *CoRR*, abs/1401.6575, 2014. URL: `http://arxiv.org/abs/1401.6575`.

**28** H. Gimbert and W. Zielonka. Games where you can play optimally without any memory. In *16th Int. Conf. on Concurrency Theory*, volume 3653 of *Lecture Notes in Computer Science*, pages 428–442. Springer, 2005.

**29** M. Jurdziński. Small progress measures for solving parity games. In *Proc. 17th Symp. on Theoretical Aspects of Computer Science*, volume 1770 of *Lecture Notes in Computer Science*, pages 290–301, Lille, France, 2000. Springer.

**30** M. Jurdzinski and R. Lazic. Succinct progress measures for solving parity games. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 1–9, Reykjavik, Iceland, 2017. IEEE Computer Society Press.

**31** D.A. Martin. The determinacy of Blackwell games. *The Journal of Symbolic Logic*, 63(4):1565–1581, 1998.

**32** A. Pnueli and R. Rosner. On the synthesis of a reactive module. In *Proc. 16th ACM Symp. on Principles of Programming Languages*, pages 179–190. ACM Press, 1989.

**33** Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley, 1st edition, 1994.

**34** M. Randour, J.-F. Raskin, and O. Sankur. Variations on the stochastic shortest path problem. In *Proc. 16th Int. Conf. on Verification, Model Checking, and Abstract Interpretation*, volume 8931 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2015.

**35** W. Thomas. Automata on infinite objects. *Handbook of Theoretical Computer Science*, pages 133–191, 1990.

**36** Y. Velner, K. Chatterjee, L. Doyen, T.A. Henzinger, A. Rabinovich, and J.-F. Raskin. The complexity of multi-mean-payoff and multi-energy games. *Information and Computation*, 241:177–196, 2015.