The Value 1 Problem Under Finite-memory Strategies for Concurrent Mean-payoff Games

Krishnendu Chatterjee (IST Austria) Rasmus Ibsen-Jensen (IST Austria)

Abstract

We consider concurrent mean-payoff games, a very well-studied class of two-player (player 1 vs player 2) zero-sum games on finite-state graphs where every transition is assigned a reward between 0 and 1, and the payoff function is the long-run average of the rewards. The value is the maximal expected payoff that player 1 can guarantee against all strategies of player 2. We consider the computation of the set of states with value 1 under finite-memory strategies for player 1, and our main results for the problem are as follows: (1) we present a polynomial-time algorithm; (2) we show that whenever there is a finite-memory strategy, there is a stationary strategy that does not need memory at all; and (3) we present an optimal bound (which is double exponential) on the patience of stationary strategies (where patience of a distribution is the inverse of the smallest positive probability and represents a complexity measure of a stationary strategy).

1 Introduction

Concurrent mean-payoff games. Concurrent mean-payoff games are played on finite-state graphs by two players (player 1 and player 2) for infinitely many rounds. In each round, the players simultaneously choose moves (or actions), and the current state along with the two chosen moves determine a probability distribution over the successor states. The outcome of the game (or a *play*) is an infinite sequence of states and action pairs. Every transition is associated with a reward between 0 and 1, and the mean-payoff (or limit-average payoff) of a play is the limit-inferior (or limit-superior) average of the rewards of the play. Concurrent games were introduced in a seminal work of Shapley [26], where *discounted* sum objectives (or games that halt with probability 1) were considered. The generalization to concurrent games with mean-payoff objectives (or games that have zero stop probabilities) was presented by Gillette in [19]. The player-1 *value* $\mathrm{val}(s)$ of the game at a state s is the supremum value of the expectation that player 1 can guarantee for the mean-payoff objective against all strategies of player 2. The games are zero-sum where the objective of player 2 is the opposite.

Important previous results. Many celebrated results have been established for concurrent mean-payoff games and its sub-classes: (1) the existence of values (or determinacy or equivalence of switching of strategy quantifiers for the players as in von-Neumann's min-max theorem) for concurrent discounted games was established in [26]; (2) the result of Blackwell and Ferguson established existence of values for the celebrated game of Big-Match [2] (the celebrated Big-Match example is from [19])[1]; and (3) developing on the results of [2] and of Bewley and Kohlberg on Puisuex series [1] the existence of values for concurrent mean-payoff games was established in [25]. The decision problem of whether the value $\mathrm{val}(s)$ is at least a rational constant λ can be decided in PSPACE [6, 21]; and the results of [21] present an algorithm for approximation which is polynomial in the number of actions and double exponential in the size of the state space (hence if the number of states is constant then the value can be approximated in polynomial time). Several special cases of concurrent mean-payoff games have been widely studied, for example, (a) concurrent reachability games [13] where reachability objectives are the very special case of mean-payoff objectives where reward zero is assigned to all transitions other than a set of sink terminal states which are assigned reward 1; (b) turn-based deterministic mean-payoff games [14, 28], where in each state at most one of the players have the choice of more than one action and the transition function is deterministic; and (c) turn-based (stochastic) reachability games [12]. The decision problem of whether the value $\mathrm{val}(s)$ is at least a rational constant λ is *square-root sum* hard even for concurrent reachability games [15], and even for the special case of turn-based stochastic reachability games [12] or turn-based deterministic mean-payoff games [28] the existence of a polynomial-time algorithm is a major and long-standing open problem.

Value 1 problem and its potential significance. While the decision problem for value computation is notoriously hard for concurrent mean-payoff games, an important special case of the problem is to compute the set of states with value 1. We refer to this problem as the value-1 set computation problem. We discuss the potential significance of the value 1 problem for mean-payoff objectives. It was shown in [10] that reliability requirements can be specified as a mean-payoff condition, where in every step a computation is done, and if the computation succeeds a reward 1 is assigned, and if the computation might fail, then reward 0 is assigned. The reliability is the long-run average reward. The value 1 problem asks whether there exists a strategy to ensure that reliability arbitrarily close to 1 can be achieved. Note that this problem cannot naturally be modeled as a reachability objective.

Strategies. A strategy in a concurrent game, considers the past history of the game (the finite sequence of

[1] note that even showing existence of a value for the specific Big-Match game was open for years, which shows the hardness of analysis of such games

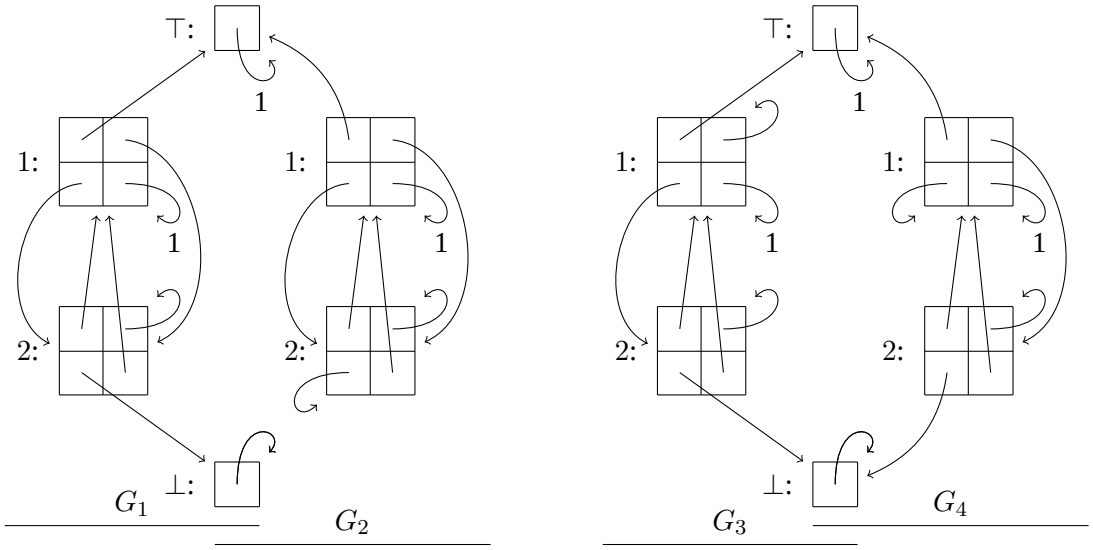

Figure 1: The games G_1 to G_4

states and actions played so far), and specifies a probability distribution over the next moves. Thus a strategy requires memory to remember the past history of the game. A strategy is *stationary* if it is independent of the past history and only depends on the current state. The complexity of a stationary strategy is described by its *patience* which is the inverse of the minimum non-zero probability assigned to a move. The notion of patience was introduced in [16] and also studied in the context of concurrent reachability games [22, 20]. A strategy is *finite-memory* if the memory set used by the strategy is finite. Note that for implementability of a strategy (such as by an automata), we need a finite-memory strategy.

Examples. We now illustrate concurrent mean-payoff games with a few examples. Consider the four games ($G_1, G_2, G_3,$ and G_4) shown in Figure 1: the transition functions are deterministic and shown as arrows; and transition with rewards 1 are annotated, and all other rewards are 0. Each game has four states, namely, 1, 2, $\top$ and $\bot$; and since $\top$ and $\bot$ remain the same, in the figures G_1 and G_2 (also G_3 and G_4) are drawn such that they share $\top$ and $\bot$. The state $\top$ has value 1 and state $\bot$ has value 0. In the first game G_1, both state 1 and state 2 have value $1/2$ (because of symmetry). The other three example games, G_2, G_3 and G_4, are minor variants of G_1 (only one successor is changed).

1. In G_2, the edge from state 2 to $\bot$ is changed to a self-loop. In G_2, there exists an infinite-memory strategy to ensure that the mean-payoff is 1, and for every $\epsilon > 0$ there is a stationary strategy to ensure mean-payoff $1 - \epsilon$. The witness stationary strategy is as follows: in state 1 play the action pairs with probability $(\epsilon/4, 1 - \epsilon/4)$ and in state 2 with probability $(1/2, 1/2)$.
2. In G_3, the top edge from state 1 to state 2 is changed to a self-loop. In G_3, there is no strategy to ensure that the mean-payoff is 1, but for every $\epsilon > 0$ there is a stationary strategy to ensure mean-payoff $1 - \epsilon$. The witness stationary strategy is as follows: in state 1 play the action pairs with probability $(\epsilon/2, 1 - \epsilon/2)$ and in state 2 with probability $(1 - \epsilon^2/2, \epsilon^2/2)$.
3. In G_4, the bottom edge from state 1 to state 2 is changed to a self-loop. In G_4, there exists no stationary strategy that can ensure positive mean-payoff value; however, for every $\epsilon > 0$ there exists an infinite-memory strategy to ensure mean-payoff $1 - \epsilon$.

Details regarding the analysis of the values of the above games and in depth discussion on the strategy constructions for them are available in [23, Section 1.6.2].

Our contributions. Our main contributions are related to the computation of the value 1 problem for concurrent mean-payoff games where player 1 is restricted to finite-memory strategies[2]. Our main results are as follows: (1) We present a polynomial-time algorithm to compute the value 1 set. (2) We show that stationary strategies are sufficient, i.e., whenever finite-memory strategies exist, then there is a stationary strategy. (3) We establish an optimal double exponential patience bound for the witness stationary strategies (our contribution for patience is the upper bound, and the matching lower bound follows from [20, 22] for the special case of reachability objectives). A key and novel insight of our polynomial-time algorithm is that we establish that we can use local operators and iterate them to compute the value 1 set; this is perhaps counter-intuitive for concurrent mean-payoff games as no strategy-iteration algorithm is known to exist. In addition we also establish a robustness result, which shows that for concurrent games, if the support of the transition probabilities match (but the precise transition probabilities may differ), then the value 1 set remains unchanged.

Related works. The problem of value-1 set computation has been extensively studied in many different contexts; such as, concurrent games with reachability objectives [13] as well as with ω-regular and prefix independent objectives [5, 4, 8], probabilistic automata [7, 17], and probabilistic systems with counters [3]. However, the value-1 set computation was not considered for concurrent mean-payoff games which we consider in this work. A related problem of computing the set of states where there exists an optimal strategy that ensures mean-payoff 1 (almost-sure winning) has been considered in [11].

2 Definitions

In this section we present the definitions of game structures, strategies, mean-payoff objectives, the value and value 1 problem, and other basic notions.

Probability distributions. For a finite set A, a *probability distribution* on A is a function $\delta\colon A \to [0,1]$ such that $\sum_{a\in A} \delta(a) = 1$. We denote the set of probability distributions on A by $\mathcal{D}(A)$. Given a distribution $\delta \in \mathcal{D}(A)$, we denote by $\mathrm{Supp}(\delta) = \{x \in A \mid \delta(x) > 0\}$ the *support* of the distribution δ. For a distribution, the *patience* of the distribution is the inverse of the minimum non-zero probability assigned to an element: formally, the patience $\mathsf{pat}(\delta)$ is $\max_{a\in A}\{\frac{1}{\delta(a)} \mid \delta(a) > 0\}$.

Concurrent game structures. A (two-player) *concurrent stochastic game structure* $G = (S, A, \Gamma_1, \Gamma_2, \delta)$ consists of the following components.

- A finite state space S and a finite set A of actions (or moves).
- Two move assignments $\Gamma_1, \Gamma_2 : S \to 2^A \setminus \emptyset$. For $i \in \{1,2\}$, assignment Γ_i associates with each state $s \in S$ the non-empty set $\Gamma_i(s) \subseteq A$ of moves available to player i at state s. For technical convenience, we assume that $\Gamma_i(s) \cap \Gamma_j(t) = \emptyset$ unless $i = j$ and $s = t$, for all $i, j \in \{1,2\}$ and $s, t \in S$. If this assumption is not met, then the moves can be trivially renamed to satisfy the assumption.
- A probabilistic transition function $\delta\colon S \times A \times A \to \mathcal{D}(S)$, which associates with every state $s \in S$ and moves $a_1 \in \Gamma_1(s)$ and $a_2 \in \Gamma_2(s)$ a probability distribution $\delta(s, a_1, a_2) \in \mathcal{D}(S)$ for the successor state.

[2] note that once a finite-memory strategy for player 1 is fixed, then there always exists a finite-memory optimal counter-strategy for player 2, and thus the strategies for player 2 are not restricted

For a set $Q \subseteq S$ of states we will denote by $\overline{Q} = S \setminus Q$ the complement of Q. We will denote by $\delta_{\min}$ the minimum non-zero transition probability, i.e., $\delta_{\min} = \min_{s,t \in S} \min_{a_1 \in \Gamma_1(s), a_2 \in \Gamma_2(s)} \{\delta(s,a_1,a_2)(t) \mid \delta(s,a_1,a_2)(t) > 0\}$. We will denote by n the number of states (i.e., $n = |S|$), and by m the maximal number of actions available for a player at a state (i.e., $m = \max_{s \in S} \max\{|\Gamma_1(s)|, |\Gamma_2(s)|\}$). We will later define Markov chains as games where $m = 1$. Since finding the mean-payoff of Markov chains can be done in polynomial time, we will only consider the case where $m \geq 2$. For all states $s \in S$, moves $a_1 \in \Gamma_1(s)$ and $a_2 \in \Gamma_2(s)$, let $\mathrm{Succ}(s,a_1,a_2) = \mathrm{Supp}(\delta(s,a_1,a_2))$ denote the set of possible successors of s when moves a_1 and a_2 are selected. The size of the transition relation of a game structure is defined as $|\delta| = \sum_{s \in S} \sum_{a_1 \in \Gamma_1(s)} \sum_{a_2 \in \Gamma_2(s)} |\mathrm{Succ}(s,a_1,a_2)|$.

One step probabilities. Given a concurrent game structure G, a state s, two distributions $\xi_1 \in \mathcal{D}(\Gamma_1(s))$ and $\xi_2 \in \mathcal{D}(\Gamma_2(s))$, the one step probability transition for a set U of states, denoted as $\delta(s,\xi_1,\xi_2)(U)$ is $\sum_{a_1 \in \Gamma_1(s), a_2 \in \Gamma_2(s), t \in U} \delta(s,a_1,a_2)(t) \cdot \xi_1(a_1) \cdot \xi_2(a_2)$. Often we will consider the distribution of player 2 to be a single action, i.e., $\xi_2(a_2) = 1$ for an action a_2, and then use the notation $\delta(s,\xi_1,a_2)$. We will also write $\mathrm{Succ}(s,\xi_1,\xi_2) = \bigcup_{a_1 \in \mathrm{Supp}(\xi_1), a_2 \in \mathrm{Supp}(\xi_2)} \mathrm{Succ}(s,a_1,a_2)$ for the set of possible successors under the distributions.

Turn-based stochastic games, turn-based deterministic games and MDPs. A game structure G is *turn-based stochastic* if at every state at most one player can choose among multiple moves; that is, for every state $s \in S$ there exists at most one $i \in \{1,2\}$ with $|\Gamma_i(s)| > 1$. A turn-based stochastic game with a deterministic transition function is a turn-based deterministic game. A game structure is a player-2 *Markov decision process (MDP)* if for all $s \in S$ we have $|\Gamma_1(s)| = 1$, i.e., only player 2 has choice of actions in the game, and player-1 MDPs are defined analogously.

Plays. At every state $s \in S$, player 1 chooses a move $a_1 \in \Gamma_1(s)$, and simultaneously and independently player 2 chooses a move $a_2 \in \Gamma_2(s)$. The game then proceeds to the successor state t with probability $\delta(s,a_1,a_2)(t)$, for all $t \in S$. A *path* or a *play* of G is an infinite sequence $\omega = \big((s_0,a_1^0,a_2^0),(s_1,a_1^1,a_2^1),(s_2,a_1^2,a_2^2)\ldots\big)$ of states and action pairs such that for all $k \geq 0$ we have (1) $s_{k+1} \in \mathrm{Succ}(s_k,a_1^k,a_2^k)$; and (2) $a_1^k \in \Gamma_1(s_k)$; and (3) $a_2^k \in \Gamma_2(s_k)$. We denote by Ω the set of all paths.

Strategies. A *strategy* for a player is a recipe that describes how to extend prefixes of a play. Formally, a strategy for player $i \in \{1,2\}$ is a mapping $\sigma_i : (S \times A \times A)^* \times S \to \mathcal{D}(A)$ that associates with every finite sequence $x \in (S \times A \times A)^*$ of state and action pairs, and the current state s in S, representing the past history of the game, a probability distribution $\sigma_i(x \cdot s)$ used to select the next move. The strategy σ_i can prescribe only moves that are available to player i; that is, for all sequences $x \in (S \times A \times A)^*$ and states $s \in S$, we require that $\mathrm{Supp}(\sigma_i(x \cdot s)) \subseteq \Gamma_i(s)$. We denote by Σ_i the set of all strategies for player $i \in \{1,2\}$. Once the starting state s and the strategies σ_1 and σ_2 for the two players have been chosen, the probabilities of events are uniquely defined [27], where an *event* $\mathcal{A} \subseteq \Omega$ is a measurable set of paths. For an event $\mathcal{A} \subseteq \Omega$, we denote by $\Pr_s^{\sigma_1,\sigma_2}(\mathcal{A})$ the probability that a path belongs to $\mathcal{A}$ when the game starts from s and the players use the strategies σ_1 and σ_2. We denote by $\mathbb{E}_s^{\sigma_1,\sigma_2}[\cdot]$ the associated expectation measure. We will consider the following special classes of strategies:

1. *Stationary (memoryless) and positional strategies.* A strategy σ_i is *stationary* (or memoryless) if it is independent of the history but only depends on the current state, i.e., for all $x, x' \in (S \times A \times A)^*$ and all $s \in S$, we have $\sigma_i(x \cdot s) = \sigma_i(x' \cdot s)$, and thus can be expressed as a function $\sigma_i : S \to \mathcal{D}(A)$. For stationary strategies, the complexity of the strategy is described by the *patience* of the strategy, which is the inverse of the minimum non-zero probability assigned to an action [16]. Formally, for a stationary strategy $\sigma_i : S \to \mathcal{D}(A)$ for player i, the patience is $\max_{s \in S} \mathsf{pat}(\sigma_i(s))$,

where $\mathsf{pat}(\sigma_i(s))$ is the patience of the distribution $\sigma_i(s)$. A strategy is *pure (deterministic)* if it does not use randomization, i.e., for any history there is always some unique action a that is played with probability 1. A pure stationary strategy σ_i is also called a *positional* strategy, and represented as a function $\sigma_i : S \to A$. We denote by Σ_i^S the set of stationary strategies for player i.

2. *Strategies with memory and finite-memory strategies.* A strategy σ_i can be equivalently defined as a pair of functions (σ_i^u, σ_i^n), along with a set Mem of memory states, such that (i) the next move function $\sigma_i^n : S \times \mathsf{Mem} \to \mathcal{D}(A)$ given the current state of the game and the current memory state specifies the probability distribution over the actions; and (ii) the memory update function $\sigma_i^u : S \times A \times A \times \mathsf{Mem} \to \mathsf{Mem}$ given the current state of the game, the action pairs, and the current memory state updates the memory state. Any strategy can be expressed with an infinite set Mem of memory states, and a strategy is a *finite-memory* strategy if the set Mem of memory states is finite, otherwise it is an *infinite-memory* strategy. We denote by Σ_i^F the set of finite-memory strategies for player i.

Absorbing states. A state s is *absorbing* if for all actions $a_1 \in \Gamma_1(s)$ and all actions $a_2 \in \Gamma_2(s)$ we have $\mathrm{Succ}(s, a_1, a_2) = \{s\}$. In the present paper we will also require that $|\Gamma_1(s)| = |\Gamma_2(s)| = 1$ if s is absorbing.

Objectives. A quantitative objective $\Phi : \Omega \to \mathbb{R}$ is a measurable function. In this work we will consider *limit-average* (or mean-payoff) objectives. We will consider concurrent games with a reward function $\mathrm{r} : S \times A \times A \to [0, 1]$ that assigns a reward value $\mathrm{r}(s, a_1, a_2)$ for all $s \in S$, $a_1 \in \Gamma_1(s)$ and $a_2 \in \Gamma_2(s)$. For a path $\omega = \big((s_0, a_1^0, a_2^0), (s_1, a_1^1, a_2^1), \dots\big)$, the limit-inferior average (resp. limit-superior average) is defined as follows: $\mathsf{LimInfAvg}(\omega) = \liminf_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} \mathrm{r}(s_i, a_1^i, a_2^i)$ (resp. $\mathsf{LimSupAvg}(\omega) = \limsup_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} \mathrm{r}(s_i, a_1^i, a_2^i)$). For the analysis of concurrent games with Boolean limit-average objectives (with rewards 0 and 1 only) we will also need *reachability* and *safety* objectives. Given a target set $U \subseteq S$, the reachability objective $\mathsf{Reach}(U)$ requires some state in U be visited at least once, i.e., defines the set

$$\mathsf{Reach}(U) = \{\omega = \big((s_0, a_1^0, a_2^0), (s_1, a_1^1, a_2^1), \dots\big) \mid \exists i \geq 0.\ s_i \in U\}$$

of paths. The dual safety objective for a set $F \subseteq S$ of safe states requires that the set F is never left, i.e.,

$$\mathsf{Safe}(F) = \{\omega = \big((s_0, a_1^0, a_2^0), (s_1, a_1^1, a_2^1), \dots\big) \mid \forall i \geq 0.\ s_i \in F\}.$$

We also consider the eventual safety objective, namely *coBüchi* objective, that requires for a given set F that ultimately only states in F are visited, i.e.,

$$\mathsf{coBuchi}(F) = \{\omega = \big((s_0, a_1^0, a_2^0), (s_1, a_1^1, a_2^1), \dots\big) \mid \exists j \geq 0. \forall i \geq j.\ s_i \in F\}.$$

Observe that reachability objectives are a very special case of Boolean reward limit-average objectives where states in U are absorbing and are exactly the states with reward 1, and similarly for safety objectives.

Markov chains. A game structure G is a *Markov chain* if $m = 1$. We will in that case write $\delta(s)$ for the distribution $\delta(s, a_1, a_2)$, where a_1 is the unique action in $\Gamma_1(s)$ and a_2 is the unique action in $\Gamma_2(s)$. Markov chains defines a weighted graph (S, E, w), where $(s, s') \in E$ iff $\delta(s)(s') > 0$ and for all $(s, s') \in E$ we have that $w((s, s')) = \delta(s)(s')$. For an event $\mathcal{A} \subseteq \Omega$, we denote by $\Pr_s(\mathcal{A})$ the probability $\Pr_s^{\sigma_1,\sigma_2}(\mathcal{A})$, where σ_1 and σ_2 are the unique strategies for player 1 and player 2, respectively. A state s is reachable from another state s' iff s' is reachable from s in (S, E, w). A set of states Z is reachable from a state s iff a state in Z is reachable from s. For any set of states Z in a Markov chain, let $R_S(Z)$, be the set of states from which Z is not reachable. Clearly, $R_S(Z) \subseteq (S \setminus Z)$. A set of states L is called a *recurrent class* if

for each pair of states $s, s' \in L$ we have that s' is reachable from s and for each pair of states $s \in L$ and $s'' \in (S \setminus L)$ we have that s'' is not reachable from s. A *recurrent class* in a Markov chain is a bottom scc (strongly connected component) in the graph of the Markov chain, where a bottom scc L is an scc with no edges leaving the scc.

Properties of Markov chains to be explicitly used in proofs. We will use several basic properties of Markov chains in our proof and we explicitly state them here. Let us fix a Markov chain with state space S.

1. Given a set $Z \subseteq S$, for all $s \in S$, with probability 1 either Z is visited infinitely often or $R_S(Z)$ is reached.
2. Given $Z \subseteq S$, for all $s \in S$, with probability 1 $R_S(Z)$ or Z is reached, i.e., $\Pr_s(\mathsf{Reach}(R_S(Z) \cup Z)) = 1$.
3. Given sets $Z \subseteq S$ and $Z' \subseteq S$, such that Z can only be left from $(Z' \cap Z)$, then for all $s \in Z$ with probability 1 $(R_S(Z') \cap Z)$ or $(Z' \cap Z)$ is reached, i.e., $\Pr_s(\mathsf{Reach}((R_S(Z') \cap Z) \cup (Z' \cap Z))) = 1$. Note the similarity with the previous property, only intersection with Z is taken.
4. Given sets $Z \subseteq S$ and $Z' \subseteq S$, such that Z can only be left from $(Z' \cap Z)$ and from each state in $(Z' \cap Z)$ there is a positive probability to leave Z, then for all $s \in Z$ with probability 1 $(R_S(Z') \cap Z)$ or $(S \setminus Z)$ is reached, i.e., $\Pr_s(\mathsf{Reach}((R_S(Z') \cap Z) \cup (S \setminus Z))) = 1$.
5. From every state $s \in S$, with probability 1 some recurrent class L is reached; and given a recurrent class L is reached, with probability 1 every state in L is reached.
6. Consider $Z \subseteq S$ and $Z' \subseteq S$ such that for all $z \in Z$ the set Z' is reachable. Then for all $s \in S$ with probability 1 either $R_S(Z)$ or Z' is reached, i.e., $\Pr_s(\mathsf{Reach}(R_S(Z) \cup Z')) = 1$.
7. Consider $Z \subseteq S$ and $Z' \subseteq S$ such that for all $s \in (S \setminus (Z \cup Z'))$, we have that $\delta(s)(Z) \cdot \epsilon \geq \delta(s)(Z')$, for $\epsilon > 0$. Then, for all $s \in (S \setminus (Z \cup Z'))$ the probability to reach Z or $R_S(Z \cup Z')$ is at least $1 - \epsilon$, i.e., $\Pr_s(\mathsf{Reach}(Z \cup R_S(Z \cup Z'))) \geq 1 - \epsilon$.
8. Consider $Z \subseteq S$ and $Z' \subseteq S$ such that for all $s \in Z$ the set Z' is reachable. Then for all $s \in Z$ with probability 1 $(S \setminus Z)$ or Z' is reached, i.e., $\Pr_s(\mathsf{Reach}((S \setminus Z) \cup Z')) = 1$.

We will refer to these properties as Markov property 1 to Markov property 8, respectively.

μ-calculus. Consider a μ-calculus expression $\Psi = \mu X.\psi(X)$ over a finite set S, where $\psi : 2^S \mapsto 2^S$ is monotonic. The least fixpoint $\Psi = \mu X.\psi(X)$ is equal to the limit $\lim_{k \to \infty} X_k$, where $X_0 = \emptyset$, and $X_{k+1} = \psi(X_k)$. For every state $s \in \Psi$, we define the *level* $k \geq 0$ of s to be the integer such that $s \notin X_k$ and $s \in X_{k+1}$. The greatest fixpoint $\Psi = \nu X.\psi(X)$ is equal to the limit $\lim_{k \to \infty} X_k$, where $X_0 = S$, and $X_{k+1} = \psi(X_k)$. For every state $s \notin \Psi$, we define the *level* $k \geq 0$ of s to be the integer such that $s \in X_k$ and $s \notin X_{k+1}$. The *height* of a μ-calculus expression $\gamma X.\psi(X)$, where $\gamma \in \{\mu, \nu\}$, is the least integer h such that $X_h = \lim_{k \to \infty} X_k$. An expression of height h can be computed in $h + 1$ iterations. A μ-calculus formula with nested μ and ν operators is a very succinct description of a nested iterative algorithm.

Interpretation of μ-calculus formula. Consider a μ-calculus formula

$$\nu Y.\mu X.[f(Y, X)],$$

where f is pointwise monotonic. The intuitive way to read the formula is as $\nu Y.(\mu X.[f(Y, X)])$, i.e., given a value of Y (say Y_i) we compute the inner least fixpoint with function $f(Y_i, X)$ which has only one free variable X. Thus for every Y_i, $\mu X.[f(Y_i, X)]$ assigns a value for Y_i. In other words, the function $\mu X.[f(Y, X)]$

can be interpreted as a function $g(Y)$ on Y, and the outer fixpoint computes the greatest fixpoint of g. The interpretation for computation of $\mu Y.\nu X.[f(Y,X)]$ is similar, and is extended straightforwardly to more nested μ-calculus formula.

The value problem. Given an objective Φ, and a class $\mathcal{C}$ of strategies for player 1, the value for player 1 under the class $\mathcal{C}$ of strategies is the maximal payoff that player 1 can guarantee with a strategy in class $\mathcal{C}$. Formally, $\mathrm{val}(\Phi,\mathcal{C})(s) = \sup_{\sigma_1\in\mathcal{C}} \inf_{\sigma_2\in\Sigma_2} \mathbb{E}_s^{\sigma_1,\sigma_2}[\Phi]$. In this work we will consider the computation of the *value 1 set* under finite-memory strategies, i.e., the computation of the set $\{s \in S \mid \mathrm{val}(\mathsf{LimInfAvg}(\mathrm{r}),\Sigma_1^F)(s) = 1\}$. Observe that to ensure value 1, player 1 must ensure that for all $\varepsilon > 0$, the probability to visit reward 1 is at least $1-\varepsilon$, and hence it follows if all rewards less than 1 are decreased to 0 the value 1 set still remains the same, and hence for simplicity for the value 1 set computation we will consider Boolean reward functions.

3 The Value 1 Set Computation

In this section we will present a polynomial-time algorithm to compute the value 1 set, $\mathrm{val}_1(\Phi,\Sigma_1^F)$, for mean-payoff objectives Φ. We start with a very basic and informal overview of the algorithm.

Basic overview of the algorithm. The algorithm will compute the value 1 set W by iteratively adding chunks of states that are guaranteed to be in the value 1 set, and the iteration will finally converge to W. Let $U \subseteq W$ be the set of states that are already guaranteed to be in the value 1 set (already identified as subset of W in some previous iteration). Then a new chunk X of states are added such that $U \subseteq X \subseteq W$, and the new chunk of states are also added iteratively (the algorithm is a nested iterative algorithm). For the set X, let $U \subseteq Y \subseteq X$ be the subset that is already added, and then a new chunk $Y \subseteq Z \subseteq X$ is added such that player 1 can ensure that one of the following three conditions hold: (1) the probability to reach U in one step can be made arbitrarily large as compared to the probability to leave W in one step (then U can be reached with probability arbitrarily close to 1); or (2) the probability to stay in X in one step is 1 and the probability to reach Y in one step is positive (then Y can be reached with probability 1); or (3) the probability to stay in X in one step is 1, the one step expected reward and the probability to stay in Z in one step can be made arbitrarily close to 1. Figure 2, Figure 3, and Figure 4 illustrate the above three conditions, respectively, pictorially. Very informally, if always one of the the last two conditions is satisfied, then then the mean-payoff can be made arbitrarily close to 1; and the first condition ensures that the already computed value 1 set can be reached with probability arbitrarily close to 1. The initialization of the sets are as follows: U and Y are initialized to the empty set, and W, X, and Z are initialized to the set of all states. Note that the above three conditions are *local* (one-step) conditions and we will first define an one-step predecessor operator to capture the above conditions. We will then show how to compute the one-step predecessor operator in polynomial time, and finally show how to use the one-step predecessor operator in a nested iterative algorithm to compute the value 1 set in polynomial time.

3.1 One-step predecessor operator

We first formally define the one-step predecessor operator that was described informally in the basic overview of the algorithm. Given a state s and two distributions $\xi_1 \in \mathcal{D}(\Gamma_1(s))$ and $\xi_2 \in \mathcal{D}(\Gamma_2(s))$, the expected one-step reward $\mathsf{ExpRew}(s,\xi_1,\xi_2)$ is defined as follows: $\sum_{a_1\in\Gamma_1(s),a_2\in\Gamma_2(s)} \xi_1(a_1)\cdot\xi_2(a_2)\cdot \mathrm{r}(s,a_1,a_2)$. We often use distributions for player 2 that plays a single action a_2 with probability 1, and use a_2 to denote such a distribution. For sets $U \subseteq Y \subseteq Z \subseteq X \subseteq W$, the one-step predecessor operator for limit-average (mean-payoff) objectives, denoted as $\mathsf{LimAvgPre}(W,U,X,Y,Z)$, is the set of states s such

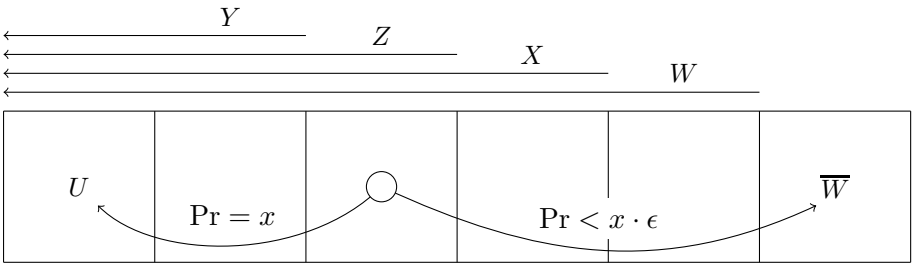

Figure 2: Pictorial illustration of Equation 1.

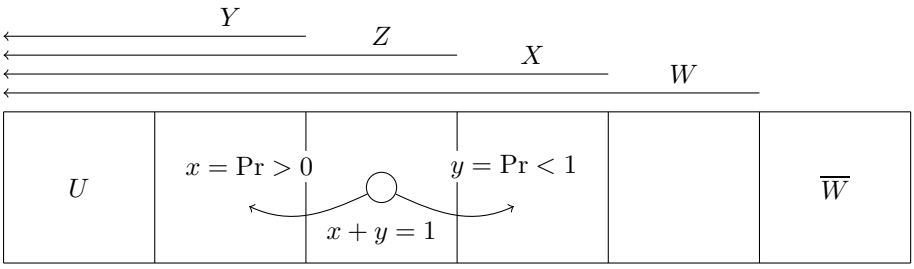

Figure 3: Pictorial illustration of Equation 2.

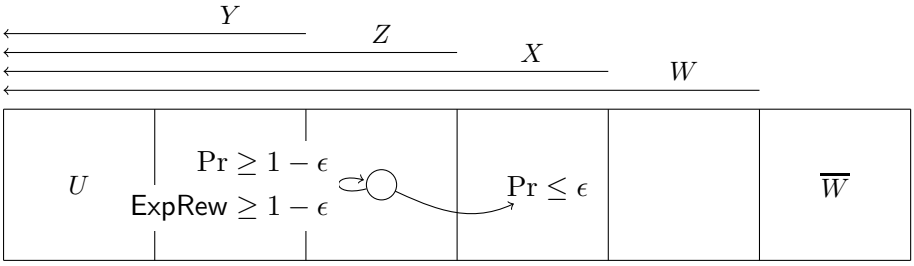

Figure 4: Pictorial illustration of Equation 3.

that for all $0 < \epsilon < \frac{1}{2}$, there exists a distribution ξ_1^ϵ over $\Gamma_1(s)$ such that for all actions a_2 in $\Gamma_2(s)$, we have that

$$\big(\epsilon \cdot \delta(s, \xi_1^\epsilon, a_2)(U) > \delta(s, \xi_1^\epsilon, a_2)(\overline{W})\big) \tag{1}$$

$$\vee \big(\delta(s, \xi_1^\epsilon, a_2)(X) = 1 \wedge \delta(s, \xi_1^\epsilon, a_2)(Y) > 0\big) \tag{2}$$

$$\vee \big(\delta(s, \xi_1^\epsilon, a_2)(X) = 1 \wedge \mathsf{ExpRew}(s, \xi_1^\epsilon, a_2) \geq 1 - \epsilon \wedge \delta(s, \xi_1^\epsilon, a_2)(Z) \geq 1 - \epsilon\big) \ . \tag{3}$$

We denote the above conditions as Equation 1, Equation 2, and Equation 3, respectively. Also our nested iterative algorithm (as informally described) that uses the $\mathsf{LimAvgPre}(W, U, X, Y, Z)$ operator will ensure the required inclusion $U \subseteq Y \subseteq Z \subseteq X \subseteq W$. Before presenting the algorithm for the computation of the LimAvgPre set, we first discuss the special case when we only have the first condition Equation 1, then describe some key properties of witness distributions, and finally present an iterative algorithm to compute LimAvgPre.

The LPre operator and witness parametrized distribution. An algorithm for the computation of the predecessor operator (called the LPre operator) for reachability games was presented in [13] where only Equation 1 is required to be satisfied. We extend the results of [13, 9] to obtain the following properties (details presented in technical appendix):

- *(Input and output).* The algorithm takes as input a state s, two sets $U \subseteq W$ of states, two sets of action sets $A_1 \subseteq \Gamma_1(s)$ and $A_2 \subseteq \Gamma_2(s)$, and either rejects the input or returns the largest set $A_3 \subseteq A_2$ such that the following conditions hold: for every $0 < \epsilon < \frac{1}{2}$ there exists a witness distribution $\xi_1^\epsilon \in \mathcal{D}(A_1)$, with patience at most $\left(\frac{\epsilon \cdot \delta_{\min}}{2}\right)^{-(|A_1|-1)}$, such that (i) for all actions $a_2 \in A_3$ Equation 1 is satisfied; and (ii) for all actions $a_2' \in (A_2 \setminus A_3)$ we have $\mathrm{Succ}(s, \xi_1^\epsilon, a_2') \subseteq W$. The set A_3 is largest in the sense that if $A_4 \subseteq A_2$ and A_4 satisfies the above conditions, then $A_4 \subseteq A_3$. Notice that this indicates that for all $a_2 \in (A_2 \setminus A_3)$ we have $\mathrm{Succ}(s, \xi_1^\epsilon, a_2) \cap U = \emptyset$, because otherwise a_2 would be in A_3. Moreover, the distribution ξ_1^ϵ has the largest possible support, i.e., for all actions $a_1 \in (A_1 \setminus \mathrm{Supp}(\xi_1^\epsilon))$, there exists an action a_2 in $(A_2 \setminus A_3)$ such that $\mathrm{Succ}(s, a_1, a_2) \cap \overline{W} \neq \emptyset$. An input would only be rejected if for each action $a_1 \in A_1$ there exists an action $a_2 \in A_2$ such that $\mathrm{Succ}(s, a_1, a_2) \cap \overline{W} \neq \emptyset$.

- *(Parametrized distribution).* Finally, the witness family of distributions ξ_1^ϵ, for $0 < \epsilon < \frac{1}{2}$, is presented in a parametrized fashion as follows: the support $\mathrm{Supp}(\xi_1^\epsilon)$ for all $0 < \epsilon < \frac{1}{2}$ is the same (denoted as A^*), and the algorithm gives the support set A^*, and a ranking function that assigns a number from 0 to at most $|A^*|$ to every action in A^*, and for any $0 < \epsilon < \frac{1}{2}$, the witness distribution ξ_1^ϵ plays actions with rank i with probability proportional to ϵ^i. In other words, the support set A^* and the ranking number of the actions in A^* is a polynomial witness for the parametrized family of witness distributions ξ_1^ϵ, for all $0 < \epsilon < \frac{1}{2}$.

We summarize the important properties which we explicitly use later: $\mathsf{LPre}(s, W, U, A_1, A_2)$ for $U \subseteq W$ returns the following (see Technical Appendix for correctness proof):

1. *(Reject property of LPre).* Reject and then for all $a_1 \in A_1$ there exists $a_2 \in A_2$ such that $\mathrm{Succ}(s, a_1, a_2) \cap \overline{W} \neq \emptyset$

2. *(Accept properties of LPre).* Accepts and returns the set $A_3 \subseteq A_2$ and a parametrized distribution ξ_1^ϵ, for $0 < \epsilon < \frac{1}{2}$, with support $\mathrm{Supp}(\xi_1^\epsilon) \subseteq A_1$, such that the following properties hold:

- (Accept property a). For all $a_2 \in A_3$, the distribution ξ_1^ϵ satisfies Equation 1 for a_2.
- (Accept property b). For all $a_2 \in (A_2 \setminus A_3)$, we have $\mathrm{Succ}(s, \xi_1^\epsilon, a_2) \cap \overline{W} = \emptyset$ and $\mathrm{Succ}(s, \xi_1^\epsilon, a_2) \cap U = \emptyset$.
- (Accept property c). For all $a_1 \in (A_1 \setminus \mathrm{Supp}(\xi_1^\epsilon))$, there exists an action a_2 in $(A_2 \setminus A_3)$ such that $\mathrm{Succ}(s, a_1, a_2) \cap \overline{W} \neq \emptyset$.
- (Accept property d). The set A_3 is largest in the sense that for all $a_2 \in (A_2 \setminus A_3)$ and for all parametrized distributions ξ_1^ϵ over A_1, the Equation 1 cannot be satisfied, while satisfying actions in A_2 using Equation 1, or Equation 2, or Equation 3, for any X, Y, Z such that $U \subseteq Y \subseteq Z \subseteq X \subseteq W$.

One action with large probability property. We will now show that if a state belongs to LimAvgPre, then there is a family of witness distributions where one action a is played with very large probability.

Lemma 1. *Given $U \subseteq Y \subseteq Z \subseteq X \subseteq W$, if $s \in \mathsf{LimAvgPre}(W, U, X, Y, Z)$, then for all $0 < \epsilon \leq \frac{\delta_{\min}}{m}$ there is a witness distribution to satisfy at least one of the three conditions (Equation 1, Equation 2, or Equation 3) of* LimAvgPre *where an action $a \in \Gamma_1(s)$ is played with probability at least $1 - \epsilon \cdot \delta_{\min}$.*

Proof. Given $0 < \epsilon \leq \frac{\delta_{\min}}{m}$, let ξ_1^ϵ be a witness distribution such that for all actions in $\Gamma_2(s)$ at least one of the three conditions for LimAvgPre is satisfied. Let C_1 be the set of actions a_2 in $\Gamma_2(s)$ such that ξ_1^ϵ and a_2 satisfy Equation 1; respectively, C_2 for Equation 2, and C_3 for Equation 3. Let a be some action such that $\xi_1^\epsilon(a) \geq \frac{1}{m}$ (note that such an action must exist). If $\xi_1^\epsilon(a) \geq 1 - \epsilon \cdot \delta_{\min}$, then we already have the desired action a; and we are done. Otherwise, we consider the distribution ξ_1' defined as follows:

$$\xi_1'(a_1) = \begin{cases} 1 - \epsilon \cdot \delta_{\min} & \text{if } a = a_1 \\ \epsilon \cdot \delta_{\min} \cdot \frac{\xi_1^\epsilon(a_1)}{1 - \xi_1^\epsilon(a)} & \text{otherwise} \end{cases}.$$

We now consider three cases to show ξ_1' is also a witness distribution to satisfy at least one of the three conditions of LimAvgPre for ϵ.

1. Consider an action a_2 in C_1. Since a_2 in C_1 and $\epsilon < \frac{\delta_{\min}}{m}$, we must have that $\mathrm{Succ}(s, a, a_2) \cap \overline{W} \neq \emptyset$, because otherwise given ξ_1^ϵ and a_2 the set $\overline{W}$ is reached with probability at least $\frac{\delta_{\min}}{m}$ (as a is played with probability at least $\frac{1}{m}$ by ξ_1^ϵ), i.e., $\delta(s, \xi_1^\epsilon, a_2)(\overline{W}) \geq \frac{\delta_{\min}}{m} > \epsilon$. This contradicts that a_2 satisfies Equation 1 for ξ_1^ϵ for the given $\epsilon < \frac{\delta_{\min}}{m}$. Hence given a and a_2, the probability to leave the set W is 0; and since all the other actions are only scaled in ξ_1' as compared to ξ_1^ϵ we have

$$\frac{\delta(s, \xi_1^\epsilon, a_2)(U)}{\delta(s, \xi_1^\epsilon, a_2)(\overline{W})} \leq \frac{\delta(s, \xi_1', a_2)(U)}{\delta(s, \xi_1', a_2)(\overline{W})}$$

   Hence, given ξ_1' the action a_2 must also satisfy Equation 1 for ϵ.

2. Consider an action a_2 in C_2. Since a_2 in C_2 (i.e., satisfies Equation 2) we must have $\mathrm{Succ}(s, \xi_1^\epsilon, a_2) \subseteq X$ (stay in X with probability 1) and $\mathrm{Succ}(s, \xi_1^\epsilon, a_2) \cap Y \neq \emptyset$ (next state in Y with positive probability). Since ξ_1' assigns positive probability to precisely the same set of actions as ξ_1^ϵ, i.e., $\mathrm{Supp}(\xi_1') = \mathrm{Supp}(\xi_1^\epsilon)$, we have that $\mathrm{Succ}(s, \xi_1', a_2) = \mathrm{Succ}(s, \xi_1^\epsilon, a_2) \subseteq X$ (stay in X with probability 1) and $\mathrm{Succ}(s, \xi_1', a_2) \cap Y = \mathrm{Succ}(s, \xi_1^\epsilon, a_2) \cap Y \neq \emptyset$ (next state in Y with positive probability). Hence we have that ξ_1' and a_2 must also satisfy Equation 2.

3. Finally consider an action a_2 in C_3. We must have that (i) $\mathrm{Succ}(s, a, a_2) \subseteq Z$ and (ii) $\mathrm{r}(s, a, a_2) = 1$; because otherwise we would either not end up in Z or not get reward 1 with probability at least $\frac{\delta_{\min}}{m}$ when a_2 is played against ξ_1^ϵ (contradicting that a_2 satisfies Equation 3). Since ξ_1' plays a with larger probability than ξ_1^ϵ, and all other actions are scaled with probabilities of ξ_1^ϵ, it follows that for every a_2 in C_3 we must have that ξ_1' and a_2 satisfy Equation 3.

The desired result follows. □

The action with large probability. In Lemma 1 we showed that some action is played with large probability. In the lemma the action was chosen depending on ϵ, but since there are only finitely many actions and if an action satisfies for some $0 < \epsilon < \frac{1}{2}$, then it also satisfies for all ϵ' such that $\epsilon \leq \epsilon' < \frac{1}{2}$, and thus it follows that there is an action that is played with large probability. We will call a parametrized distribution ξ_1^ϵ, for $0 < \epsilon < \frac{1}{2}$, an a-*large* distribution if the distribution plays action a with probability at least $1 - \epsilon \cdot \delta_{\min}$. Thus the existence of witness a-large distributions, if such distributions exist, follows from Lemma 1. The main crux of the algorithm would be to find an action a and a parametrized distribution that is a-large as a witness distribution for LimAvgPre. Our algorithm will use the LPre operator iteratively. The key information we need is encoded as a matrix as follows.

The matrix for action sets. Given a state s, and the sets $U \subseteq Y \subseteq Z \subseteq X \subseteq W$, we define an $|\Gamma_1(s)| \times |\Gamma_2(s)|$-matrix M, such that $M_{a_1,a_2} \in \{\overline{W}, W, U, X, Y, Z^0, Z^1\}$, that corresponds to the type of successor encountered if player 1 plays action a_1 and player 2 plays action a_2. Let

$$M_{a_1,a_2} = \begin{cases} \overline{W} & \text{if } \mathrm{Succ}(s,a_1,a_2) \cap \overline{W} \neq \emptyset \\ U & \text{if } \mathrm{Succ}(s,a_1,a_2) \cap U \neq \emptyset \text{ and } \mathrm{Succ}(s,a_1,a_2) \cap \overline{W} = \emptyset \\ W & \text{if } \mathrm{Succ}(s,a_1,a_2) \cap (W \setminus X) \neq \emptyset \text{ and } \mathrm{Succ}(s,a_1,a_2) \cap (\overline{W} \cup U) = \emptyset \\ Y & \text{if } \mathrm{Succ}(s,a_1,a_2) \cap (Y \setminus U) \neq \emptyset \text{ and } \mathrm{Succ}(s,a_1,a_2) \cap (\overline{W} \cup U \cup (W \setminus X)) = \emptyset \\ X & \text{if } \mathrm{Succ}(s,a_1,a_2) \cap (X \setminus Z) \neq \emptyset \\ & \quad \text{and } \mathrm{Succ}(s,a_1,a_2) \cap (\overline{W} \cup U \cup (W \setminus X) \cup (Y \setminus U)) = \emptyset \\ Z^\ell & \text{if } \mathrm{Succ}(s,a_1,a_2) \cap (Z \setminus Y) \neq \emptyset \\ & \quad \text{and } \mathrm{Succ}(s,a_1,a_2) \cap (\overline{W} \cup U \cup (W \setminus X) \cup (Y \setminus U) \cup (X \setminus Z)) = \emptyset \\ & \quad \text{and } \mathrm{r}(s,a_1,a_2) = \ell, \text{ for } \ell \in \{0,1\} \ . \end{cases}$$

The matrix uses that $U \subseteq Y \subseteq Z \subseteq X \subseteq W$, to ensure that the matrix is well-defined. Notice that M encodes all the information needed by LPre (the entries equal to W, Y, X, Z^1, Z^0 all ensures both $\overline{W}$ and U are not reached, U ensures that U is reached with probability at least $\delta_{\min}$ and $\overline{W}$ is not reached. The entries $\overline{W}$ ensures that $\overline{W}$ is reached with probability between $\delta_{\min}$ and 1). Hence, we could alternatively give M as input to LPre.

Intuitive description of the algorithm. We first present an intuitive description of our algorithm and then present it formally. The basic idea of the algorithm is to use LPre iteratively and the existence of a-large witness distributions. Given a candidate action a, we reject a or accept a using the following procedure. First, given the action a, if there is an action a_2 such that W is left with positive probability given a and a_2 (i.e., $M_{a,a_2} = \overline{W}$), then we reject a. Second, we check if playing a with probability 1 satisfies all actions (by either of the three conditions), and if so we accept. If neither of the first two conditions hold, then we use an iterative procedure. Let C be the set of actions which are guaranteed to be satisfied (by Equation 1) by playing an a-large distribution (C consists of each action a_2 such that $M_{a,a_2} = U$). We run LPre, and start with $(\Gamma_1(s) \setminus \{a\})$ as *available actions* for player 1 (we are only interested in a-large distributions and

we do not consider a for LPre) and $(\Gamma_2(s) \setminus C)$ as available actions for player 2. If LPre rejects, we also reject: this is because no matter which action $a_1 \neq a$ is played with the largest probability (and we could not play a alone) there is an action a_2, such that $M_{a_1,a_2} = \overline{W}$ and $M_{a,a_2} \neq U$, which ensures that all three equations are violated. If LPre accepts, then we obtain a witness distribution ξ_1 and a set A_3 of actions of player 2 such that ξ_1 satisfies Equation 1 for all actions in A_3. We then create ξ_1', which is ξ_1 scaled so that it plays an a-large distribution (note that ξ_1 plays a with probability 0). Afterwards we check if all actions for player 2 are satisfied by ξ_1'. If so, we accept. Otherwise, we check that whether for each action a_2 outside $(A_3 \cup C)$ we can satisfy either Equation 2 or Equation 3: for a_2 to be satisfied using Equation 3, we must have that $M_{a,a_2} = Z^1$; and for a_2 to be satisfied using Equation 2, the distribution ξ_1' must play some action a_1 with positive probability such that $M_{a_1,a_2} = Y$. If for some a_2 outside $(A_3 \cup C)$, neither $M_{a,a_2} = Z^1$, nor $M_{a_1,a_2} = Y$, for some a_1 played with positive probability, we reject. Otherwise, if we did not reject, we remove each action a_1 for player 1 from available actions, for which there exists an $a_2 \in (A_3 \cup C)$, such that $M_{a_1,a_2} = W$. Note that if $M_{a_1,a_2} = W$, then we cannot satisfy a_2 using either Equation 2 or Equation 3, if we play a_1 with positive probability. If the set of available actions does not contain a, then we cannot play a with positive probability in an a-large distribution, which clearly means that no a-large distribution exists and thus we reject. If this new, smaller set of actions for player 1 contains a, we iterate on with the new set as the set of available actions for player 1, and the available set for player 2 always remains as $(\Gamma_2(s) \setminus C)$. Since, in every iteration, we get a smaller set of actions for player 1, we terminate at some point.

The algorithm ALGOPRED**.** We now describe the steps of the algorithm which we refer as ALGOPRED (algorithm for predecessor computation). For a state s, we consider every action $a \in \Gamma_1(s)$ as a candidate for the existence of an a-large witness distribution. For each action a we execute the following steps:

1. *(Reject 1).* Reject the choice of a if there exists $a_2 \in \Gamma_2(s)$ such that $M_{a,a_2} = \overline{W}$.

2. *(Accept 1).* Accept a if for all $a_2 \in \Gamma_2(s)$ we have $M_{a,a_2} \in \{U, Y, Z^1\}$, and then return the distribution that plays a with probability 1, and return "Accept" for state s.

3. Let C be the set of actions a_2 in $\Gamma_2(s)$ such that $M_{a,a_2} \neq U$. Initialize B_1^0 and A_1^0 as $(\Gamma_1(s) \setminus \{a\})$. The remainder of the algorithm will be done in iterations.

4. *(Iteration).* In iteration $i \geq 1$, run $\mathsf{LPre}(s, W, U, ((A_1^{i-1} \cap B_1^{i-1}) \setminus \{a\}), C)$.
(Reject 2): if $\mathsf{LPre}(s, W, U, ((A_1^{i-1} \cap B_1^{i-1}) \setminus \{a\}), C)$ rejects the input, then reject this choice of a. Otherwise let A_2^i be the returned set; and let $\xi_1^{\epsilon,i}$ be a witness parametrized distribution (parametrized by $0 < \epsilon < \frac{1}{2}$ which is obtained by the support of $\xi_1^{\epsilon,i}$ and the ranking of the actions in the support). We will now define some sets of actions.

   (a) Let $A_1^i = \mathrm{Supp}(\xi_1^{\epsilon,i}) \cup \{a\}$.

   (b) Let B_1^i be all actions a_1 in $\Gamma_1(s)$ such that for all $a_2 \in (C \setminus A_2^i)$ we have $M_{a_1,a_2} \neq W$.

   (c) Let B_2^i be all actions a_2 in $(C \setminus A_2^i)$ such that either (i) $M_{a,a_2} = Z^1$; or (ii) there exists an action $a_1 \in A_1^i$ with $M_{a_1,a_2} = Y$.

5. We reject in the following cases:

   - *(Reject 3).* If $((A_1^i \cap B_1^i) \setminus \{a\}) = \emptyset$, then reject this choice of a.
   - *(Reject 4).* If $(C \setminus A_2^i) \neq B_2^i$, then reject this choice of a.
   - *(Reject 5).* If $a \notin B_1^i$, then reject this choice of a.

6. *(Accept 2).* Otherwise if $A_1^i \subseteq B_1^i$, then return accept a, and return the parametrized distribution ξ_1^ϵ, for $0 < \epsilon < \frac{1}{2}$, that plays a with probability $1 - \epsilon \cdot \delta_{\min}$ and with probability $\epsilon \cdot \delta_{\min}$ follows $\xi_1^{\epsilon,i}$, and also "Accept" state s.

7. If the action is neither accepted nor rejected, then go to iteration $i + 1$ in step 4.

If all choices of action $a \in \Gamma_1(s)$ get rejected, then "Reject" state s.

The parametrized distribution for Accept 2 is returned as the special action a (to be played with probability $1 - \epsilon \cdot \delta_{\min}$, for $0 < \epsilon < \frac{1}{2}$), the support set of $\xi_1^{\epsilon,i}$ and the ranking function of the support as given by the LPre operator (which gives the parametrized distribution for $\xi_1^{\epsilon,i}$ which is multiplied by $\epsilon \cdot \delta_{\min}$ to get the parametrized a-large witness distribution ξ_1^ϵ and a is played with the remaining probability).

Illustrations with examples. We illustrate our algorithm on four M-matrices shown in Figure 5. First observe that the only feasible candidate for an a-large distribution is the first row, because each other row contains an $\overline{W}$ entry, and thus will be rejected at the start. The first matrix shown in Figure 5a will be accepted by the algorithm and the other three will be rejected by the algorithm.

1. Consider first the matrix in Figure 5a. Then the algorithm is run with the first row as a, it will call LPre with the all rows but the first row for player 1 and all columns but the first column for player 2 (since given the first row, the first column satisfies Equation 1). The LPre algorithm will then return the distribution d of playing the second row with probability $1 - \frac{\epsilon}{2}$ and the third row with probability $\frac{\epsilon}{2}$. It also returns the set A_3 containing the second and third column (they satisfy Equation 1). We then get accept in that iteration, because column 4 and column 5 can be satisfied by Equation 2 and column 6 can be satisfied by Equation 3.

2. Consider now the second matrix, the one in Figure 5b. It will get rejected at start, because in this case each row contains an $\overline{W}$ entry.

3. The third matrix, the one in Figure 5c, will get rejected in the second iteration. In the first iteration, LPre will return the same distribution d as for the first matrix along with the same A_3. This time, we cannot accept directly, because d no longer satisfies any of the three equations, for column 5. At that point, the algorithm considers that each column $a_2 \in \{4, 5, 6\}$ such that $M_{a_1,a_2} = Y$ for some $a_1 \in \{1, 2, 3\}$ or $M_{a,a_2} = Z^1$ (where $a = 1$). Thus, the algorithm removes row 2, from the set of possible rows, because column 5 is such that $M_{2,5} = W$, and $5 \notin A_3$ and iterate. Then the algorithm calls LPre and gets back reject, because each of the rows left contains at least one instance of $\overline{W}$. Hence the algorithm rejects.

4. For the last matrix, the one in Figure 5d, the algorithm calls LPre and gets d and A_3, but this time the algorithm rejects at that point, because row 6 (which is not in A_3) does not contain an action a_1 played with positive probability such that $M_{a_1,6} = Y$ or is such that $M_{a,6} = Z^1$.

Lemma 2. *Given $U \subseteq Y \subseteq Z \subseteq X \subseteq W$ and a state s, if algorithm* ALGOPRED *accepts s, then $s \in \mathsf{LimAvgPre}(W, U, X, Y, Z)$. Furthermore, for every $0 < \epsilon < \frac{1}{2}$ there exists a witness distribution ξ_1^ϵ with patience at most $\left(\frac{\epsilon \cdot \delta_{\min}}{2}\right)^{-(|\Gamma_1(s)|-1)}$ to satisfy at least one of the three required conditions (Equation 1, Equation 2, or Equation 3) for* LimAvgPre *for every action $a_2 \in \Gamma_2(s)$.*

Proof. We will next show that if ALGOPRED returns a parametrized distribution ξ_1^ϵ, then for all $0 < \epsilon < \frac{1}{2}$ and for all actions $a_2 \in \Gamma_2(s)$, at least one of the three conditions of LimAvgPre is satisfied. This will show

$$M = \begin{pmatrix} U & W & W & X & Y & Z^1 \\ \overline{W} & U & W & Y & X & X \\ \overline{W} & \overline{W} & U & X & X & X \\ \overline{W} & \overline{W} & \overline{W} & \overline{W} & \overline{W} & \overline{W} \end{pmatrix}$$

(a) This illustrates a M-matrix, which has an a-large distribution, where a corresponds to the first row.

$$M = \begin{pmatrix} U & \textcircled{\overline{W}} & W & X & Y & Z^1 \\ \overline{W} & U & W & Y & W & X \\ \overline{W} & \overline{W} & U & X & X & X \\ \overline{W} & \overline{W} & \overline{W} & \overline{W} & \overline{W} & \overline{W} \end{pmatrix}$$

(b) This illustrates a M-matrix, which has no a-large distribution. The cicled entry is the only entry changed as compared to Figure 5a.

$$M = \begin{pmatrix} U & W & W & X & Y & Z^1 \\ \overline{W} & U & W & Y & \textcircled{W} & X \\ \overline{W} & \overline{W} & U & X & X & X \\ \overline{W} & \overline{W} & \overline{W} & \overline{W} & \overline{W} & \overline{W} \end{pmatrix}$$

(c) This illustrates a M-matrix, which has no a-large distribution. The cicled entry is the only entry changed as compared to Figure 5a.

$$M = \begin{pmatrix} U & W & W & X & Y & \textcircled{X} \\ \overline{W} & U & W & Y & X & \textcircled{Z^1} \\ \overline{W} & \overline{W} & U & X & X & X \\ \overline{W} & \overline{W} & \overline{W} & \overline{W} & \overline{W} & \overline{W} \end{pmatrix}$$

(d) This illustrates a M-matrix, which has no a-large distribution. The circled entries are the only entries changed as compared to Figure 5a.

Figure 5

that $s \in \mathsf{LimAvgPre}(W, U, X, Y, Z)$. The algorithm accepts state s and returns a distribution at two places, namely, (Accept 1) and (Accept 2). For the case of Accept 1: the algorithms returns a distribution that plays some action a with probability 1; and for the case of Accept 2 it returns a distribution that plays some subset of actions (at least 2) with positive probability. We analyze both the cases below.

1. *Case Accept 1.* In the first case for all actions a_2 we have that $M_{a,a_2} \in \{U, Y, Z^1\}$. We analyze the three sub-cases.
   (a) If $M_{a,a_2} = U$, then $\mathrm{Succ}(s, a, s_2) \cap U \neq \emptyset$ (i.e., the next state is in U with positive probability) and $\mathrm{Succ}(s, a, a_2) \cap \overline{W} = \emptyset$ (i.e., the next state is in $\overline{W}$ with probability 0) and hence Equation 1 is satisfied.
   (b) If $M_{a,a_2} = Y$, then (i) $\mathrm{Succ}(s, a, a_2) \cap (Y \setminus U) \neq \emptyset$ which implies that $\mathrm{Succ}(s, a, a_2) \cap Y \neq \emptyset$, since $(Y \setminus U) \subseteq Y$; and (ii) $\mathrm{Succ}(s, a, a_2) \cap (\overline{W} \cup U \cup (W \setminus X)) = \emptyset$ which implies that $\mathrm{Succ}(s, a, a_2) \cap (\overline{X} \cup U) = \emptyset$ because as $X \subseteq W$ we have $(\overline{W} \cup U \cup (W \setminus X)) = \overline{X} \cup U$; and hence $\mathrm{Succ}(s, a, a_2) \subseteq X$. The first condition ensures that the next state is in Y with positive probability and the second condition ensures the next state is in X with probability 1, and thus Equation 2 is satisfied.
   (c) If $M_{a,a_2} = Z^1$, then (i) $\mathrm{Succ}(s, a, a_2) \cap (Z \setminus Y) \neq \emptyset$ which implies that $\mathrm{Succ}(s, a, a_2) \cap Z \neq \emptyset$; and (ii) $\mathrm{Succ}(s, a, a_2) \cap (\overline{W} \cup U \cup (W \setminus X) \cup (Y \setminus U) \cup (X \setminus Z)) = \emptyset$ which implies that $\mathrm{Succ}(s, a, a_2) \cap (\overline{Z} \cup U \cup Y) = \emptyset$, because as $Z \subseteq X \subseteq W$ we have $(\overline{W} \cup U \cup (W \setminus X) \cup Y \cup (X \setminus Z)) = (\overline{Z} \cup U \cup Y)$, and hence $\mathrm{Succ}(s, a, a_2) \subseteq Z$ (i.e., next state in Z with probability 1); and (iii) $\mathrm{r}(s, a, a_2) = 1$ (i.e., expected reward is 1). It follows that Equation 3 is satisfied.
2. *Case Accept 2.* In the second case, we consider the case when the algorithm returns a parameterized

distribution ξ_1^ϵ, for $0 < \epsilon < \frac{1}{2}$, in iteration i. Let the action played with probability $1 - \epsilon \cdot \delta_{\min}$ be a. Such an action clearly exists, by construction. For any $a_2 \in \Gamma_2(s)$ such that $M_{a,a_2} = U$, then the next state is in U with probability at least $(1 - \epsilon \cdot \delta_{\min}) \cdot \delta_{\min}$ and the next state is in $\overline{W}$ with probability at most $\epsilon \cdot \delta_{\min}$ and the ratio is at least $2 \cdot \epsilon$; thus the distribution ξ_1^ϵ and a_2 satisfy Equation 1 for $2 \cdot \epsilon$. As $0 < \epsilon < \frac{1}{2}$ is arbitrary the result follows for all a_2 such that $M_{a,a_2} = U$. We consider the set C of remaining actions in $\Gamma_2(s)$, i.e., for all $a_2 \in C$ we have $M_{a,a_2} \neq U$.

Satisfying Equation 1 in A_2^i. We have that $M_{a,a_2} \neq \overline{W}$, for all $a_2 \in \Gamma_2(s)$, because otherwise the guess of action a would have been rejected, in (Reject 1). We also have that $\mathsf{LPre}(s, W, U, B', C)$, for $B' \subseteq (\Gamma_1(s) \setminus \{a\})$ must return an distribution ξ_1' over B' and a set $A' \subseteq C$, such that for all $a_2 \in A'$, the action a_2 and the distribution ξ_1' satisfies Equation 1 (by Accept property a of LPre). In the last iteration the set A_2^i is the set returned by $\mathsf{LPre}(s, W, U, ((A_1^{i-1} \cap B_1^{i-1}) \setminus \{a\}), C)$, and the distribution $\xi_1^{\epsilon,i}$ satisfies Equation 1 for all actions in A_2^i (again by Accept property a of LPre since A_2^i is the returned subset of C). Since ξ_1^ϵ only plays a with high probability and only scales the distribution $\xi_1^{\epsilon,i}$ it follows (similarly to Case 1 of Lemma 1) that ξ_1^ϵ satisfies Equation 1 for all actions in A_2^i.

Satisfying Equation 2 or Equation 3 in $(C \setminus A_2^i)$. By definition of B_1^i and A_1^i (Step 4 (a) and Step 4 (b) of the algorithm), and that $A_1^i \subseteq B_1^i$ (from Accept 2 of the algorithm), it follows that the distribution ξ_1^ϵ is such that for all $a_2 \in (C \setminus A_2^i)$ and $a_1 \in \mathrm{Supp}(\xi_1^\epsilon) \cup \{a\} = A_1^i$ we have $M_{a_1,a_2} \neq W$. Also for all $a_2 \in (C \setminus A_2^i)$ and all a_1 such that $\xi_1^\epsilon(a_1) > 0$, we have from Accept property b of LPre that $M_{a_1,a_2} \neq \overline{W}$ and $M_{a_1,a_2} \neq U$. Notice that therefore for all $a_1 \in \mathrm{Supp}(\xi_1^\epsilon)$ and $a_2 \in (C \setminus A_2^i)$ we have $M_{a_1,a_2} \in \{X, Y, Z^0, Z^1\}$, which implies that $\mathrm{Succ}(s, \xi_1^\epsilon, a_2)(X) = 1$. For all $a_2 \in (C \setminus A_2^i)$ we have that either (i) $M_{a,a_2} = Z^1$; or (ii) ξ_1^ϵ assigned positive probability to some a_1 such that $M_{a_1,a_2} = Y$, because otherwise $(C \setminus A_2^i) \neq B_2^i$ and we would have rejected this choice of a (by Reject 4 of the algorithm). Notice that $M_{a,a_2} = Z^1$ implies that $\mathrm{Succ}(s, a, a_2)(Z) = 1$ and that $\mathrm{r}(s, a, a_2) = 1$, thus, since the distribution the algorithm returned was a-large, we get that we reach Z in one step with probability at least $1 - \epsilon \cdot \delta_{\min}$ and get reward 1 with probability at least $1 - \epsilon \cdot \delta_{\min}$, hence Equation 3 is satisfied. If the second case holds (i.e., $M_{a_1,a_2} = Y$), we have $\mathrm{Succ}(s, \xi_1^\epsilon, a_2) \cap (Y \setminus U) \neq \emptyset$ (i.e., Y is reached with positive probability in one step), thus implying that Equation 2 is satisfied.

Therefore the distribution ξ_1^ϵ is a witness distribution to satisfy the required conditions for $0 < \epsilon < \frac{1}{2}$ for $\mathsf{LimAvgPre}$. It follows that $s \in \mathsf{LimAvgPre}(W, U, X, Y, Z)$.

Patience. The distribution returned by LPre over $|\Gamma_1(s)| - 1$ actions has patience at most $\left(\frac{\epsilon \cdot \delta_{\min}}{2}\right)^{-(|\Gamma_1(s)|-2)}$. Hence it is clear from the algorithm that the distribution returned by the algorithm has patience at most $\left(\frac{\epsilon \cdot \delta_{\min}}{2}\right)^{-(|\Gamma_1(s)|-1)}$. □

Our next goal is to present a lemma that complements the previous lemma. In other words, we would show that if ALGOPRED rejects an action a, then there would be no a-large distributions as witnesses for $\mathsf{LimAvgPre}$. The algorithm rejects an action a at four places, and we will show that all the rejections are *sound* (i.e., if a is rejected, then there is no a-large witness distribution). We first show that the first rejection is sound.

Soundness of Reject 1. We consider the case of Reject 1. In this case, there exists an action a_2 such that $M_{a,a_2} = \overline{W}$. Given an a-large distribution ξ_1^ϵ, the one step probability to reach $\overline{W}$ (i.e., $\delta(s, \xi_1^\epsilon, a_2)(\overline{W})$) is at least $x = (1 - \epsilon \cdot \delta_{\min}) \cdot \delta_{\min} > \epsilon$, since $\epsilon < \frac{1}{2}$ and $\delta_{\min} \leq 1$, and even if U is reached with the remaining probability (i.e., even if $\delta(s, \xi_1^\epsilon, a_2)(U) = 1 - x$), it follows that Equation 1 is violated, for all $0 < \epsilon < \frac{1}{2}$.

The remaining two expressions cannot be satisfied because $X \subseteq W$ and since we leave W with positive probability we as well leave X with positive probability. It follows that the rejection of action a is sound for Reject 1.

Rejects in iteration. The other places the algorithm can reject action a, i.e., (Reject 2), (Reject 3), (Reject 4), and (Reject 5), are part of the iterative procedure. To prove soundness of these rejects we will define a loop invariant and prove the loop invariant inductively. We will also show that with the loop invariant we can establish soundness of the rejects in the iterative procedure as well as the termination of the algorithm.

The loop invariant. The *loop invariant* is as follows:

- Any a-large witness distribution ξ_1^ϵ for LimAvgPre only plays actions in $(A_1^i \cap B_1^i) \cup \{a\}$ with positive probabilities, for all $i \geq 0$, i.e., $\mathrm{Supp}(\xi_1^\epsilon) \subseteq (A_1^i \cap B_1^i) \cup \{a\}$.

We will also establish the *monotonicity* (strictly decreasing till a fixpoint is reached) property that $(A_1^i \cap B_1^i) \cup \{a\} \subseteq (A_1^{i-1} \cap B_1^{i-1}) \cup \{a\}$, for all $i > 0$; and equality implies termination in iteration i.

Inductive proof of loop invariant. We present the basic inductive argument for the loop invariant:

- **The base case, $i = 0$.** The base case, for $i = 0$ is trivial, since $A_1^0 = B_1^0 = (\Gamma_1(s) \setminus \{a\})$, thus implying that $(A_1^i \cap B_1^i) \cup \{a\} = \Gamma_1(s)$.
- **The induction case, $i > 0$.** By inductive hypothesis, any a-large witness distribution ξ_1^ϵ only plays actions in $(A_1^{i-1} \cap B_1^{i-1}) \cup \{a\}$ with positive probabilities, and we need to establish for i. We will show that any a-large witness distribution can only play actions in $A_1^i \cup \{a\} = A_1^i$, (see the following description of A_1^i which uses the inductive hypothesis). We refer to this as required property 1 for loop invariant. Similarly, we establish the same for B_1^i (see the following description of B_1^i which uses the inductive hypothesis). We refer to this as required property 2 for loop invariant. Hence any witness a-large distribution can only play actions in $(A_1^i \cap B_1^i) \cup \{a\}$.

The above proof requires to establish the key properties of A_1^i and B_1^i. Before establishing them we first show the monotonicity property.

Monotoncity property. We will show that we have $(A_1^i \cap B_1^i) \cup \{a\} \subseteq (A_1^{i-1} \cap B_1^{i-1}) \cup \{a\}$, for all $i > 0$, and equality implies termination of the inner loop in iteration i. Notice that this implies that for any choice of a the inner loop rejects a or finds a distribution after at most $|\Gamma_1(s)|$ iterations. We have that $A_1^i = \mathrm{Supp}(\xi_1^\epsilon) \cup \{a\}$ (by Step 4 (a) of ALGOPRED), where ξ_1^ϵ is a witness distribution returned by $\mathsf{LPre}(s, W, U, ((A_1^{i-1} \cap B_1^{i-1}) \setminus \{a\}), C)$. Since $\mathrm{Supp}(\xi_1^\epsilon) \subseteq ((A_1^{i-1} \cap B_1^{i-1}) \setminus \{a\})$, if LPre accepts, we have that $A_1^i \subseteq (A_1^{i-1} \cap B_1^{i-1}) \cup \{a\}$. Thus we get that $(A_1^i \cap B_1^i) \cup \{a\} \subseteq A_1^i \cup \{a\} \subseteq (A_1^{i-1} \cap B_1^{i-1}) \cup \{a\}$. This establish monotonicity and now we show the termination. Assume that $(A_1^i \cap B_1^i) \cup \{a\} = (A_1^{i-1} \cap B_1^{i-1}) \cup \{a\}$. Therefore we have that ξ_1^ϵ can only use actions in $((A_1^{i-1} \cap B_1^{i-1}) \setminus \{a\})$, which is thus also $((A_1^i \cap B_1^i) \setminus \{a\})$. But then either (i) $a \notin B_1^i$ or (ii) $\mathrm{Supp}(\xi_1^\epsilon) \cup \{a\} = A_1^i \subseteq (A_1^i \cap B_1^i) \cup \{a\}$; which implies that $A_1^i \subseteq B_1^i$. But in the first case we reject (in (Reject 5)) and in the second case we accept (in (Accept 2)). This establishes the termination property.

The properties of the sets for loop invariant. We now present the associated properties of the sets A_1^i, A_2^i, B_1^i, and B_2^i to complete the inductive proof of the loop invariant.

1. *The property of the set A_2^i.* We first argue that A_2^i has certain properties which will imply the key properties for A_2^i.

(a) Since $\mathsf{LPre}(s, W, U, ((A_1^{i-1} \cap B_1^{i-1}) \setminus \{a\}), C)$ accepts, we have that A_2^i is a subset of C. There exists a witness parametrized distribution ξ_1^ϵ, over $((A_1^{i-1} \cap B_1^{i-1}) \setminus \{a\})$ such that for all $a_2 \in A_2^i$ we have that ξ_1^ϵ and a_2 satisfies Equation 1 (by Accept property a of LPre).

(b) Also for all $a_2 \in (C \setminus A_2^i)$ we have that $M_{a_1,a_2} \neq \overline{W}$ for all $a_1 \in \mathrm{Supp}(\xi_1^\epsilon)$ (Accept property b of LPre).

(c) Notice also that for any action $a_2 \in C$, if a distribution over $A_1^{i-1} \cap B_1^{i-1}$ cannot satisfy a_2 using Equation 1, then no distribution over $(A_1^{i-1} \cap B_1^{i-1}) \cup \{a\}$ can either, since $M_{a,a_2} \neq U$ (from the definition of the set C) and hence U cannot be reached as long as the distribution plays a. For an distribution ξ_1' to be a witness distribution, all actions in $\Gamma_2(s)$ must satisfy either (i) Equation 1; or (ii) Equation 2; or (iii) Equation 3. But if an action a_2 must satisfy either Equation 2 or Equation 3, we must have that ξ_1' ensures that $\overline{X}$ is reached with probability 0 (i.e., $\mathrm{Succ}(s, \xi_1', a_2) \subseteq X$). Hence, since $X \subseteq W$ we also must have that $\overline{W}$ is reached with probability 0.

By Accept property d of LPre we have that, since A_2^i is returned by LPre, no a-large witness distribution ξ_1' can satisfy any action a_2 in $(C \setminus A_2^i)$ using Equation 1, while satisfying all actions in C using Equation 1, or Equation 2, or Equation 3. Also, for all a_2 in $(C \setminus A_2^i)$ and all $a_1 \in \mathrm{Supp}(\xi_1^\epsilon)$ we have that $M_{a_1,a_2} \neq U$ (by Accept property b of LPre). Furthermore, by definition of C for all $a_2 \in C$ we have that $M_{a,a_2} \neq U$. Therefore we have established the following key properties for A_2^i:

- Any a-large witness distribution ξ_1' must satisfy all actions a_2 in $(C \setminus A_2^i)$ using either Equation 2 or Equation 3.
- For all $a_2 \in (C \setminus A_2^i)$ and $a_1 \in \mathrm{Supp}(\xi_1^\epsilon) \cup \{a\} = A_1^i$ we have that $M_{a_1,a_2} \neq U$.

2. *The property of the set A_1^i.* By accept property c of LPre and since we did not reject in Reject 1, the set A_1^i is the largest set, such that for all $a_1 \in A_1^i$ there exists no a_2 in $(C \setminus A_2^i)$ with $M_{a_1,a_2} = \overline{W}$. But this means that any distribution that satisfies for all actions in $(C \setminus A_2^i)$ either Equation 2 or Equation 3, must play only actions in A_1^i. But from our description of A_2^i we obtain that all a-large witness distributions must ensure that all actions in $(C \setminus A_2^i)$ are satisfied using either Equation 2 or Equation 3. Therefore we have established the following key property for A_1^i: All a-large witness distributions must play only actions in A_1^i with positive probability. This proves the required property 1 of the loop invariant.

3. *The property of the set B_2^i.* From the first key property of A_2^i we have that any a-large witness distribution must ensure that all actions in $(C \setminus A_2^i)$ satisfy either Equation 2 or Equation 3. From the second key property of A_2^i, for all $a_1 \in A_1^i$ and all $a_2 \in (C \setminus A_2^i)$, we have that $M_{a_1,a_2} \neq U$. The key property of A_1^i implies that any a-large witness distribution must play only actions in A_1^i.

   Hence, for an a-large witness distribution ξ_1', for all a_2 in $(C \setminus A_2^i)$ we must have that either (i) $M_{a,a_2} = Z^1$ (to satisfy Equation 3); or (ii) there is an action a_1 in A_1^i such that $M_{a_1,a_2} = Y$ (to satisfy Equation 2 — it would also be satisfied if $M_{a_1,a_2} = U$ but we know that $M_{a_1,a_2} \neq U$ by Accept property b of LPre). But that is precisely the definition of B_2^i (Step 4 (c) of ALGOPRED). Therefore, we have the following key property for B_2^i: Actions a_2 in $(C \setminus (A_2^i \cup B_2^i))$ cannot be satisfied by Equation 1 or Equation 2 or Equation 3 by any a-large witness distribution.

4. *The property of the set B_1^i.* We know from the first key property of A_2^i that all actions in $(C \setminus A_2^i)$ must satisfy Equation 2 or Equation 3. But to do so we must leave X with probability 0. But B_1^i is the

largest set of actions such that for all actions a_1 in B_1^i and for all actions a_2 in $(C \setminus A_2^i)$, we have that $M_{a_1,a_2} \neq W$ (Step 4 (b) of ALGOPRED). Hence we have that an a-large distribution that plays an action in $(\Gamma_1(s) \setminus B_1^i)$ with positive probability violates both Equation 2 and Equation 3 for some a_2 in $(C \setminus A_2^i)$. Therefore, we have the following key property for B_1^i: All a-large witness distributions only plays actions in B_1^i. This also proves the required property 2 of the loop invariant.

This establishes the inductive proof of the loop invariant.

Lemma 3. *For a given $U \subseteq Y \subseteq Z \subseteq X \subseteq W$, if Algorithm* ALGOPRED *rejects state s, then $s \notin \mathsf{LimAvgPre}(W, U, X, Y, Z)$. Also, algorithm* ALGOPRED *accepts or rejects a choice of action a as a candidate for the existence of a-large witness distributions at most $\min(|\Gamma_1(s)|, |\Gamma_2(s)|)$ iterations of the inner loop.*

Proof. In the algorithm there are five places where a choice of a might get rejected. We have already argued the soundness of Reject 1. We prove the soundness of the other rejects below.

1. *(Reject 2).* If $\mathsf{LPre}(s, W, U, ((A_1^{i-1} \cap B_1^{i-1}) \setminus \{a\}), C)$ is rejected, then for all actions a_1 in $((A_1^{i-1} \cap B_1^{i-1}) \setminus \{a\})$, there exists an action a_2 in C such that $M_{a_1,a_2} = \overline{W}$, by the reject property of LPre. But then consider any distribution ξ_1 over $((A_1^{i-1} \cap B_1^{i-1}) \setminus \{a\})$, some action a_1 is played with probability at least $\frac{1}{m}$. Hence the action a_2 such that $M_{a_1,a_2} = \overline{W}$, cannot be satisfied using neither (i) Equation 1; nor (ii) Equation 2; nor (iii) Equation 3. The latter two because $\overline{W}$ is entered with positive probability in one step and hence X is left with positive probability in one step. The first is because we reach $\overline{W}$ with probability at least $x = \frac{\delta_{\min}}{m}$ and even if we reach U with probability $1 - x$, we still do not satisfy Equation 1. Now consider some distribution ξ_1' over $(A_1^{i-1} \cap B_1^{i-1}) \cup \{a\}$. Either it plays a with probability 1 or not. If it does, then it cannot be a witness distribution, since it otherwise would have been accepted in Accept 1. If it does not then the argument is similar to the previous argument (in the case of Equation 1, the argument also uses that $M_{a,a_2} \neq U$ from the definition of C). Hence no witness distribution exists that only uses actions in $(A_1^{i-1} \cap B_1^{i-1}) \cup \{a\}$. Thus Reject 2 is a sound reject, by the loop invariant.

2. *(Reject 3).* If a is not accepted by Accept 1, then a could not be played with probability 1. For Reject 3, the condition $((A_1^i \cap B_1^i) \setminus \{a\}) = \emptyset$ is satisfied. Thus no a-large witness distribution can play anything but a by the loop invariant. Therefore no a-large witness distribution can exist in this case. Thus, Reject 3 is a sound reject.

3. *(Reject 4).* Consider an a-large witness distribution ξ_1^ϵ. The key property of B_2^i implies that any action $a_2 \in (C \setminus (A_2^i \cup B_2^i))$ cannot be satisfied using either of the equations. But since $B_2^i \subseteq (C \setminus A_2^i)$ we must have that $B_2^i = (C \setminus A_2^i)$ for any a-large witness distribution to exists. Therefore we can reject the choice of a if $(C \setminus A_2^i) \neq B_2^i$. Hence Reject 4 is a sound reject.

4. *(Reject 5).* From the key property of the set B_1^i, we have that if $a \notin B_1^i$, then no a-large witness distribution can play a with positive probability, which implies that no a-large witness distribution can exist. Hence Reject 5 is also a sound reject.

Termination. We have already established (in "monotonicity and termination for loop invariant") that $(A_1^i \cap B_1^i) \cup \{a\} \subseteq (A_1^{i-1} \cap B_1^{i-1}) \cup \{a\}$, for all $i > 0$ and equality implies termination of the inner loop in iteration i. Notice that this implies that for any choice of a the inner loop rejects a or finds a distribution after at most $|\Gamma_1(s)|$ iterations. We will now show that $A_2^i \subseteq A_2^{i-1}$, for all $i > 0$ and equality

implies termination in iteration i. Notice that this implies that for any choice of a the inner loop rejects a or finds a distribution after at most $|\Gamma_2(s)|$ iterations. We have that $A_2^i \subseteq A_2^{i-1}$, because $\xi_1^{\epsilon,i}$ could also be returned in iteration $i-1$ and LPre maximizes the number of a_1's for which $\xi_1^{\epsilon,i}(a_1) > 0$ (Accept property c). Assume that $A_2^i = A_2^{i-1}$. Then $(C \setminus A_2^i) = (C \setminus A_2^{i-1})$ and thus $B_1^i = B_1^{i-1}$. We also have that $A_1^i \subseteq (A_1^{i-1} \cap B_1^{i-1}) \cup \{a\}$, thus implying that $A_1^i \subseteq (A_1^{i-1} \cap B_1^i) \cup \{a\}$. Therefore $A_1^i \subseteq B_1^i$, since if B_1^i does not contain a, neither does B_1^{i-1} and thus we would have rejected the choice of a in iteration $i-1$, because of (Reject 5). The desired result follows. □

Lemma 4. *Given $U \subseteq Y \subseteq Z \subseteq X \subseteq W$ and a state s,* AlgoPred *terminates in time $O(|\Gamma_1(s)|^2 \cdot |\Gamma_2(s)|^2 + \sum_{a_1 \in \Gamma_1(s), a_2 \in \Gamma_2(s)} |\mathrm{Supp}(s, a_1, a_2)|)$. Alternatively, if M is given as input, the running time is $O(|\Gamma_1(s)|^2 \cdot |\Gamma_2(s)|^2)$.*

Proof. The calculation of M can be done in time $\sum_{a_1 \in \Gamma_1(s), a_2 \in \Gamma_2(s)} |\mathrm{Supp}(s, a_1, a_2)|$. As mentioned in the definition of M, we could alternatively use M as input to LPre since it encodes all information needed. There are $|\Gamma_1(s)|$ different choices for which action a to play with high probability. Given a, there are at most $\min(|\Gamma_1(s)|, |\Gamma_2(s)|)$ iterations of the inner loop, see Lemma 3. Each iteration of the inner loop can be done in $O(|\Gamma_1(s)| \cdot |\Gamma_2(s)|)$ time, and is dominated by the running time of LPre, which runs in time $O(\Gamma_1(s)| \cdot |\Gamma_2(s)|)$ on M, see [13]. Hence, if M is given as input we get a running time of $O(|\Gamma_1(s)| \cdot \min(|\Gamma_1(s)|, |\Gamma_2(s)|) \cdot |\Gamma_1(s)| \cdot |\Gamma_2(s)|)$, which is less than $O(|\Gamma_1(s)|^2 \cdot |\Gamma_2(s)|^2)$. □

Combining Lemma 2, Lemma 3 and Lemma 4 we get the following lemma.

Lemma 5. *The algorithm* AlgoPred*, for a given state s and sets $U \subseteq Y \subseteq Z \subseteq X \subseteq W$, correctly computes if $s \in \mathsf{LimAvgPre}(W, U, X, Y, Z)$ and runs in time $O(|\Gamma_1(s)|^2 \cdot |\Gamma_2(s)|^2 + \sum_{a_1 \in \Gamma_1(s), a_2 \in \Gamma_2(s)} |\mathrm{Supp}(s, a_1, a_2)|)$.*

3.2 Iterative algorithm for value 1 set computation

In this section we will present the nested iterative algorithm for the value 1 set computation. The nested iterative algorithm is succinctly represented as the following nested fixpoint formula (μ-calculus formula) that uses the LimAvgPre one-step predecessor operator. Let

$$W^* = \nu W.\mu U.\nu X.\mu Y.\nu Z.\mathsf{LimAvgPre}(W, U, X, Y, Z) \ .$$

We will show that $W^* = \mathrm{val}_1(\mathsf{LimInfAvg}, \Sigma_1^F)$ (also see the appendix, Section 6, for an algorithmic description of computation of the μ-calculus formula). First in the next subsection we show that $W^* \subseteq \mathrm{val}_1(\mathsf{LimInfAvg}, \Sigma_1^S) \subseteq \mathrm{val}_1(\mathsf{LimInfAvg}, \Sigma_1^F)$; and in the following subsection will establish the other inclusion.

3.2.1 First inclusion: $W^* \subseteq \mathrm{val}_1(\mathsf{LimInfAvg}, \Sigma_1^S)$

Let Θ_i denote the random variable for the reward at the i-th step of the game. We will show that for all states s in W^* for all $\epsilon > 0$, there exists a stationary (hence finite-memory) strategy σ_1^ϵ for player 1 such that for all positional strategies σ_2 for player 2 we have that

$$\lim_{t \to \infty} \frac{\sum_{i=0}^{t} \mathbb{E}_s^{\sigma_1^\epsilon, \sigma_2}[\Theta_i]}{t} \geq 1 - \epsilon \ .$$

This will show that $W^* \subseteq \mathsf{val}_1(\mathsf{LimInfAvg}, \Sigma_1^S) \subseteq \mathsf{val}_1(\mathsf{LimInfAvg}, \Sigma_1^F)$. Notice that the statement is trivially satisfied if $W^* = \emptyset$, and hence we will assume that this is not so.

Computation of W^*. We first analyze the computation of W^*. Since W^* is a fixpoint, we can replace W by W^* and get rid of the outer most ν operator, and the rest of the μ-calculus formula also computes W^*. In other words, we have

$$W^* = \mu U.\nu X.\mu Y.\nu Z.\mathsf{LimAvgPre}(W^*, U, X, Y, Z) \ ,$$

Thus the computation of W^* is achieved as follows: U_0 is the empty set; and $U_i = \nu X.\mu Y.\nu Z.\mathsf{LimAvgPre}(W^*, U_{i-1}, X, Y, Z)$, for $i \geq 1$. Let ℓ be the least index such that $U_\ell = W^*$. For any $i \geq 0$, we also have that $Y_{i,0}$ is the empty set and that $Y_{i,j} = \nu Z.\mathsf{LimAvgPre}(W^*, U_{i-1}, U_i, Y_{i,j-1}, Z)$, for $j \geq 1$. For a state $s \in W^*$, let the rank of state s (denoted $\mathsf{rk}(s) = (i,j)$) be the tuple of (i,j) such that i is the least index with $s \in U_i$ (i.e., $s \in U_i \setminus U_{i-1}$); and j is the least index with $s \in Y_{i,j}$ (i.e., $s \in Y_{i,j} \setminus Y_{i,j-1}$). For $1 \leq i \leq \ell$, let $\mathsf{rk}(i) = j$ be the least index when the fix point converges for U_i, i.e., the least j such that $Y_{i,j} = Y_{i,j+1}$. By definition of W^*, for all states $s \in W^*$, if $\mathsf{rk}(s) = (i,j)$, then we must have that for all $\epsilon > 0$ there is a distribution ξ_1^ϵ over $\Gamma_1(s)$ such that for all actions $a_2 \in \Gamma_2(s)$ for player 2 we have that

$$\big(\epsilon \cdot \delta(s, \xi_1^\epsilon, a_2)(U_{i-1}) > \delta(s, \xi_1^\epsilon, a_2)(\overline{W}^*)\big) \tag{4}$$

$$\vee \ \big(\delta(s, \xi_1^\epsilon, a_2)(U_i) = 1 \wedge \delta(s, \xi_1^\epsilon, a_2)(Y_{i,j-1}) > 0\big) \tag{5}$$

$$\vee \ \big(\delta(s, \xi_1^\epsilon, a_2)(U_i) = 1 \wedge \mathsf{ExpRew}(s, \xi_1^\epsilon, a_2) \geq 1 - \epsilon \wedge \delta(s, \xi_1^\epsilon, a_2)(Y_{i,j}) \geq 1 - \epsilon\big) \ ; \tag{6}$$

where $\overline{W}^* = S \setminus W^*$ is the complement of W^*. We refer to the above as Equation 4, Equation 5, and Equation 6, respectively.

The construction of stationary witness strategy σ_1^ϵ. Fix $0 < \epsilon < \frac{1}{2}$. The desired witness stationary strategy σ_1^ϵ will be constructed from a finite sequence of stationary strategies,

$$\sigma_1^{\epsilon,1,0}, \sigma_1^{\epsilon,1,1}, \ldots, \sigma_1^{\epsilon,1,\mathsf{rk}(1)}, \sigma_1^{\epsilon,2,0}, \ldots, \sigma_1^{\epsilon,2,\mathsf{rk}(2)}, \ldots, \sigma_1^{\epsilon,\ell,0}, \ldots, \sigma_1^{\epsilon,\ell,\mathsf{rk}(\ell)}.$$

The strategies will be constructed inductively. First we will construct it for states in U_1 and $(U_\ell \setminus U_{\ell-1})$, and then we will present the inductive construction for $(U_i \setminus U_{i-1})$, for $2 \leq i \leq \ell - 1$.

- *(Base case).* We will first describe the construction of the strategy $\sigma_1^{\epsilon,1,0}$ (resp. $\sigma_1^{\epsilon,\ell,0}$).
  1. The stationary strategy $\sigma_1^{\epsilon,1,0}$ (resp. $\sigma_1^{\epsilon,\ell,0}$) is arbitrary except for states in $(Y_{1,\mathsf{rk}(1)} \setminus Y_{1,\mathsf{rk}(1)-1})$ (resp. $(Y_{\ell,\mathsf{rk}(\ell)} \setminus Y_{\ell,\mathsf{rk}(\ell)-1})$).
  2. For states s in $(Y_{1,\mathsf{rk}(1)} \setminus Y_{1,\mathsf{rk}(1)-1})$ (resp. $(Y_{\ell,\mathsf{rk}(\ell)} \setminus Y_{\ell,\mathsf{rk}(\ell)-1})$) the strategy plays the distribution ξ_1^η over $\Gamma_1(s)$, for $\eta = \frac{\epsilon}{2}$.
  3. We next describe the construction of the strategy $\sigma_1^{\epsilon,1,j}$ (resp. $\sigma_1^{\epsilon,\ell,j}$), for $j \geq 1$, using induction in j.
     (a) The strategy $\sigma_1^{\epsilon,1,j}$ (resp. $\sigma_1^{\epsilon,\ell,j}$) plays as $\sigma_1^{\epsilon,1,j-1}$ (resp. $\sigma_1^{\epsilon,\ell,j-1}$) except for states in $(Y_{1,\mathsf{rk}(1)-j} \setminus Y_{1,\mathsf{rk}(1)-(j+1)})$ (resp. $(Y_{\ell,\mathsf{rk}(\ell)-j} \setminus Y_{\ell,\mathsf{rk}(\ell)-(j+1)})$).
     (b) For states s in $(Y_{1,\mathsf{rk}(1)-j} \setminus Y_{1,\mathsf{rk}(1)-(j+1)})$ (resp. $(Y_{\ell,\mathsf{rk}(\ell)-j} \setminus Y_{\ell,\mathsf{rk}(\ell)-(j+1)})$) the strategy plays the distribution ξ_1^η over $\Gamma_1(s)$, for $\eta = \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{(2m)^j}$.

- *(Inductive case).* We will next construct the strategy for the remaining states, in two steps, first for $\sigma_1^{\epsilon,i,0}$ and then for $\sigma_1^{\epsilon,i,j}$, for $2 \leq i \leq \ell - 1$ and $j \geq 1$. We will do so using induction backwards in i. That is the base case is $i = \ell$ and we then proceed downward.

  1. The strategy $\sigma_1^{\epsilon,i,0}$ plays as the strategy $\sigma_1^{\eta,i+1,\mathsf{rk}(i+1)}$, for $\eta = \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{(2m)^{\mathsf{rk}(i)}}$, except for states in $(Y_{i,\mathsf{rk}(i)} \setminus Y_{i,\mathsf{rk}(i)-1})$.
  2. For states s in $Y_{i,\mathsf{rk}(i)} \setminus Y_{i,\mathsf{rk}(i)-1}$ the strategy plays ξ_1^{η} over $\Gamma_1(s)$, for $\eta = \frac{\epsilon}{2}$.
  3. We now finally construct $\sigma_1^{\epsilon,i,j}$, for $2 \leq i \leq \ell - 1$, using induction in j.
     (a) The strategy $\sigma_1^{\epsilon,i,j}$ plays as $\sigma_1^{\epsilon,i,j-1}$ except for states in $(Y_{i,\mathsf{rk}(i)-j} \setminus Y_{i,\mathsf{rk}(i)-(j+1)})$.
     (b) For states s in $(Y_{i,\mathsf{rk}(i)-j} \setminus Y_{i,\mathsf{rk}(i)-(j+1)})$ the strategy plays ξ_1^{η} over $\Gamma_1(s)$, for $\eta = \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{(2m)^j}$.

- *(The entire strategy).* Let $\sigma_1^{\epsilon,i} = \sigma_1^{\epsilon,i,\mathsf{rk}(i)}$ for all i. Let σ_1^{ϵ} play as $\sigma_1^{\beta,1}$ in U_1 and $\sigma_1^{\beta,2}$, for $\beta = \frac{\epsilon}{2}$, in the remaining states.

Lemma 6. *The patience of $\sigma_1^{\epsilon,i}(s)$ for states s of rank $(i, \mathsf{rk}(i) - j)$ is at most $\left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(\frac{(2m)^{j+1}}{2} - 1)}$.*

Proof. By construction, the patience $\sigma_1^{\epsilon,i}(s)$ of states s of rank $(i, \mathsf{rk}(i))$ is $\left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(m-1)}$ (by Lemma 2). Also for $j \geq 1$, the patience $\sigma_1^{\epsilon,i}(s)$ of states s of rank $(i, \mathsf{rk}(i) - j)$ is at most

$$\begin{aligned}
\left(\frac{\left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{(2m)^j} \cdot \delta_{\min}}{2}\right)^{-(m-1)} &= \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(2m)^j \cdot (m-1)} \cdot \left(\frac{\delta_{\min}}{2}\right)^{-(m-1)} \\
&= \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(2m)^j \cdot (m-1)} \cdot \left(\frac{\delta_{\min}}{2}\right)^{-m} \cdot \left(\frac{\delta_{\min}}{2}\right) \\
&= \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(2m)^j \cdot m} \cdot \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{(2m)^j} \cdot \left(\frac{\delta_{\min}}{2}\right)^{-m} \cdot \left(\frac{\delta_{\min}}{2}\right) \\
&\leq \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(2m)^j \cdot m} \cdot \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right) \\
&= \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(\frac{(2m)^{j+1}}{2} - 1)},
\end{aligned}$$

where the inequality is as follows: $\left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{(2m)^j} \cdot \left(\frac{\delta_{\min}}{2}\right)^{-m} = \left(\frac{\epsilon}{2}\right)^{(2m)^j} \cdot \left(\frac{\delta_{\min}}{2}\right)^{(2m)^j} \cdot \left(\frac{\delta_{\min}}{2}\right)^{-m} \leq \frac{\epsilon}{2}$ since $(2m)^j \geq m \geq 1$ and $\epsilon < 1$. The desired result follows. □

Lemma 7. *Let $0 < \epsilon < \frac{1}{2}$ be given. The patience of the witness stationary strategy σ_1^{ϵ} is less than $\left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(2m)^n}$.*

Proof. We first present the bound for U_1 (also U_2) and then for other states.

The patience of $\sigma_1^{\epsilon,1}$ for states in U_1 (also similar for U_2). For each state s in U_1, the corresponding distribution $\sigma_1^{\epsilon,1}(s)$ has patience at most $\left(\frac{\epsilon\cdot\delta_{\min}}{4}\right)^{-(\frac{(2m)^{\mathsf{rk}(1)}}{2}-1)}$, since no states are in $Y_{1,0}$. Similarly for s in U_2 and the corresponding distribution $\sigma_1^{\epsilon,1}(s)$.

The η for which the strategy $\sigma_1^{\epsilon,2}$ follows $\sigma_1^{\eta,i}$: Inductive statement. We will argue using induction that for each state $S \in (W^* \setminus U_{i-1})$, for $i \geq 3$, we have that the strategy $\sigma_1^{\epsilon,2}$ follows the strategy $\sigma_1^{\eta,i}$, for

$$\eta \geq \left(\frac{\epsilon\cdot\delta_{\min}}{4}\right)^{\sum_{k=2}^{i-1}\prod_{k'=k}^{i-1}(2m)^{\mathsf{rk}(k')}}.$$

Base case. For each state $s \in (S \setminus U_2)$, the strategy $\sigma_1^{\epsilon,2}$ follows the strategy $\sigma_1^{\eta,3}$, for $\eta \geq \left(\frac{\epsilon\cdot\delta_{\min}}{4}\right)^{(2m)^{\mathsf{rk}(2)}}$, by construction, which is the wanted expression.

Induction case $i+1$. For $i \geq 4$, for each state $s \in (S \setminus U_{i-1})$, the strategy $\sigma_1^{\epsilon,2}$ follows the strategy $\sigma_1^{\eta,i}$, for $\eta \geq \left(\frac{\epsilon\cdot\delta_{\min}}{4}\right)^{\sum_{k=2}^{i-1}\prod_{k'=k}^{i-1}(2m)^{\mathsf{rk}(k')}}$, by induction. In each state $s \in (S \setminus U_i)$, the strategy $\sigma_1^{\eta,i}$ follows the strategy $\sigma_1^{\eta',i+1}$, for $\eta' \geq \left(\frac{\eta\cdot\delta_{\min}}{4}\right)^{(2m)^{\mathsf{rk}(i)}}$, by construction. Thus, the strategy $\sigma_1^{\epsilon,2}$ follows $\sigma_1^{\eta',i+1}$ for

$$\begin{aligned}
\eta' &\geq \left(\frac{\eta\cdot\delta_{\min}}{4}\right)^{(2m)^{\mathsf{rk}(i)}} \\
&\geq \left(\frac{\left(\frac{\epsilon\cdot\delta_{\min}}{4}\right)^{\sum_{k=2}^{i-1}\prod_{k'=k}^{i-1}(2m)^{\mathsf{rk}(k')}}\cdot\delta_{\min}}{4}\right)^{(2m)^{\mathsf{rk}(i)}} \\
&\geq \left(\left(\frac{\epsilon\cdot\delta_{\min}}{4}\right)^{1+\sum_{k=2}^{i-1}\prod_{k'=k}^{i-1}(2m)^{\mathsf{rk}(k')}}\right)^{(2m)^{\mathsf{rk}(i)}} \\
&= \left(\frac{\epsilon\cdot\delta_{\min}}{4}\right)^{\sum_{k=2}^{i}\prod_{k'=k}^{i}(2m)^{\mathsf{rk}(k')}}.
\end{aligned}$$

The first inequality comes from our preceding explanation. The second inequality uses the inductive hypothesis. The third uses that $\frac{\delta_{\min}}{4} > \frac{\epsilon\cdot\delta_{\min}}{4}$. The last equality is the inductive hypothesis for $i+1$ and follows from

$$\begin{aligned}
(2m)^{\mathsf{rk}(i)} + (2m)^{\mathsf{rk}(i)}\cdot\sum_{k=2}^{i-1}\prod_{k'=k}^{i-1}(2m)^{\mathsf{rk}(k')} &= (2m)^{\mathsf{rk}(i)} + \sum_{k=2}^{i-1}\prod_{k'=k}^{i}(2m)^{\mathsf{rk}(k')} \\
&= \sum_{k=2}^{i}\prod_{k'=k}^{i}(2m)^{\mathsf{rk}(k')}.
\end{aligned}$$

Patience of $\sigma_1^{\epsilon,2}(s)$ for states in U_i, for $i \geq 3$. We see that for $i \geq 3$ and for each s in U_i we have that $\sigma_1^{\eta,i}(s)$ follows $\xi_1^{\eta'}$ for $\eta' \geq \left(\frac{\eta\cdot\delta_{\min}}{4}\right)^{(2m)^{\mathsf{rk}(i)-1}}$ (since $Y_{i,0}$ is empty), by construction. Hence, we get that

$\sigma_1^{\epsilon,2}(s) = \xi_1^{\eta'}$ for $\eta' \geq \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{\frac{\sum_{k=2}^{i} \prod_{k'=k}^{i} (2m)^{\mathsf{rk}(k')}}{2m}}$, using a similar argument as the one used in the inductive case. Since $\mathsf{rk}(i) \geq 1$ and $m \geq 1$, we see that each term in the sum $\sum_{k=2}^{i} \prod_{k'=k}^{i} (2m)^{\mathsf{rk}(k')}$ is at least twice as large as the following. Thus, we have that

$$\sum_{k=2}^{i} \prod_{k'=k}^{i} (2m)^{\mathsf{rk}(k')} < 2 \cdot \prod_{k'=2}^{i} (2m)^{\mathsf{rk}(k')} = 2 \cdot (2m)^{\sum_{k'=2}^{i} \mathsf{rk}(k')} \leq 2 \cdot (2m)^{n-1} \leq (2m)^{n} \ .$$

The first inequality is because U_1 must contain at least 1 state. The second comes from $m \geq 1$. Hence, $\eta' \geq \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{(2m)^{n-1}}$. Using an argument similar to the one used to prove Lemma 6, we get that the patience for $\xi_1^{\eta'}$ is then at most $\left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(\frac{(2m)^n}{2}-1)}$.

Patience of σ_1^{ϵ}. We now need to consider the strategy σ_1^{ϵ}. It follows $\sigma_1^{\beta,1}$ in U_1 and $\sigma_1^{\beta,2}$ elsewhere, for $\beta = \frac{\epsilon}{2}$, We see that

$$\begin{aligned}\left(\frac{\beta \cdot \delta_{\min}}{4}\right)^{-(\frac{(2m)^n}{2}-1)} &= \left(\frac{\epsilon \cdot \delta_{\min}}{8}\right)^{-(\frac{(2m)^n}{2}-1)} \\ &< \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(2m)^n}\end{aligned}$$

The inequality is because $4^2 = 16 > 8$ (and the last expression more than squares the preceding). This completes the proof. □

Basic overview of the proof. We first present the basic overview of the proof. Let σ_1 be a stationary strategy that follows distribution ξ_1^{η} over $\Gamma_1(s)$ in state $s \in W^*$ for some $\eta > 0$ and let σ_2 be a positional counter-strategy for player 2. For state s in W^*, $\sigma_1(s)$ and $\sigma_2(s)$ satisfies at least one of Equation 4, Equation 5, or Equation 6 in s. Let $C_1^{\sigma_1,\sigma_2} \subseteq W^*$ (resp. $C_2^{\sigma_1,\sigma_2} \subseteq W^*$ and $C_3^{\sigma_1,\sigma_2} \subseteq W^*$) be the set of states in W^* that satisfies Equation 4 (resp. Equation 5 and Equation 6). We will prove that σ_1^{ϵ} ensures value at least $1-\epsilon$ for each states s in W^*. We will split the proof into four parts, first we will show some properties for states in U_1, then for states in $U_\ell \setminus U_{\ell-1}$, and finally for states in $U_i \setminus U_{i-1}$ for $2 \leq i \leq \ell - 1$. In the fourth part, we will then combine the three properties to establish the desired result. The three properties are as follows

- *(Property 1).* For all states s in U_1 we will show that $\sigma_1^{\epsilon,1}$ ensures $\mathsf{Safe}(U_1)$ with probability 1 and mean-payoff at least $1-\epsilon$ (i.e., for all positional strategies σ_2 we have $\lim_{t\to\infty} \frac{\sum_{i=0}^{t} \mathbb{E}_s^{\sigma_1^{\epsilon,1},\sigma_2}[\Theta_i]}{t} \geq 1-\epsilon$).
- *(Property 2).* For all states s in $(U_\ell \setminus U_{\ell-1})$ we will show that $\sigma_1^{\epsilon,\ell}$ ensures that against all positional strategies σ_2 we have that
  1. given the event $\mathsf{Safe}(U_\ell \setminus U_{\ell-1})$, the mean-payoff is at least $1-\epsilon$;
  2. $\Pr_s^{\sigma_1^{\epsilon,\ell},\sigma_2}(\mathsf{Safe}(U_\ell \setminus U_{\ell-1}) \cup \mathsf{Reach}(U_{\ell-1} \cup \overline{W}^*)) = 1$; and
  3. $\Pr_s^{\sigma_1^{\epsilon,\ell},\sigma_2}(\mathsf{Safe}(U_\ell \setminus U_{\ell-1}) \cup \mathsf{Reach}(U_{\ell-1})) \geq 1-\epsilon$.

- *(Property 3).* For all states s in $(U_\ell \setminus U_{\ell-(i+1)})$, for $1 \leq i \leq \ell-2$, we will show that $\sigma_1^{\epsilon,i}$ ensures that against all positional strategies σ_2 we have that

  1. given the event $\bigcup_{j\leq i} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)})$, the mean-payoff is at least $1-\epsilon$;
  2. $\Pr_s^{\sigma_1^{\epsilon,\ell-i},\sigma_2}(\bigcup_{j\leq i} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)}) \cup \mathsf{Reach}(U_{\ell-(i+1)} \cup \overline{W}^*)) = 1$; and
  3. $\Pr_s^{\sigma_1^{\epsilon,\ell-i},\sigma_2}(\bigcup_{j\leq i} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)}) \cup \mathsf{Reach}(U_{\ell-(i+1)})) \geq 1-\epsilon$.

In Lemma 8, Lemma 9, and Lemma 12 we establish Properties 1, 2, and 3, respectively. We first present the basic intuition of the proof of Lemma 8.

The basic intuition of Lemma 8. The key idea of the proof is as follows. Once we fix the strategies for both the players we have a Markov chain. Let C_2 and C_3 denote the set of states in U_1 that satisfy Equation 5 and Equation 6, respectively. Since U_0 is empty, no state in U_1 can satisfy Equation 4. For states s in C_2 of rank $(1,j)$, the fact that Equation 5 is satisfied ensures that a state of rank $(1,j')$, for $j' < j$, is visited from s with positive probability. Let $\mathsf{pat}(j)$ denote the patience of the strategy $\sigma_1^{\epsilon,1}$ for states of rank $(1, \mathsf{rk}(1)-j)$. We now consider the following case analysis.

1. First we consider the set of states in $(Y_{1,\mathsf{rk}(1)} \setminus Y_{1,\mathsf{rk}(1)-1})$ and show that if we stay in the set $(Y_{1,\mathsf{rk}(1)} \setminus Y_{1,\mathsf{rk}(1)-1})$, then the mean-payoff is at least $1-\epsilon$. The argument is as follows: By Markov property 5, we must reach a recurrent class with probability 1. A recurrent class contained in $(Y_{1,\mathsf{rk}(1)} \setminus Y_{1,\mathsf{rk}(1)-1})$ must consist of only states in C_3 (since from states in C_2 we reach lower rank states with positive probability), and since Equation 6 is satisfied for states in C_3 it follows that the mean-payoff value is at least $1-\epsilon$. Hence, if we have a recurrent class of the Markov chain contained in $(U_1 \setminus Y_{1,\mathsf{rk}(1)-1}) = (Y_{1,\mathsf{rk}(1)} \setminus Y_{1,\mathsf{rk}(1)-1})$, then the mean-payoff of the recurrent class is at least $1-\epsilon$. This completes the argument. Also, if the set $(Y_{1,\mathsf{rk}(1)} \setminus Y_{1,\mathsf{rk}(1)-1})$ is left, then we can *bound* the number of visits to states in C_2 (and in the worst case each such visit gives reward 0) in expectation encountered before leaving the set $(Y_{1,\mathsf{rk}(1)} \setminus Y_{1,\mathsf{rk}(1)-1})$. This bound on the number of visits in expectation to C_2 (which we say has not been accounted for by visits to C_3) is $\kappa(0) = (\delta_{\min})^{-1} \cdot \mathsf{pat}(0)$. There is an illustration of this base case in Figure 6.

2. Now we consider that we are at some intermediate part of the computation, i.e., in some state in $(Y_{1,\mathsf{rk}(1)-j} \setminus Y_{1,\mathsf{rk}(1)-(j+1)})$, for $j \geq 1$. Inductively we have an upper bound $\kappa(j)$ on the number of times that states in C_2 were visited (in the worst case each such visit gives reward 0) in expectation that has not been accounted for by visits to states in C_3 till we reach the set $(Y_{1,\mathsf{rk}(1)-j} \setminus Y_{1,\mathsf{rk}(1)-(j+1)})$ from any state in $Y_{1,\mathsf{rk}(1)-j+1}$. The one-step probability distribution ξ_1^η is chosen such that $\eta \cdot \kappa(j) \leq \epsilon$. In other words, η decreases rapidly as i increases, and the small η ensures that if the play stays in $(U_1 \setminus Y_{1,\mathsf{rk}(1)-(j+1)})$, then the mean-payoff is at least $1-\epsilon$, i.e., if we have a recurrent class L contained in $(U_1 \setminus Y_{1,\mathsf{rk}(1)-(j+1)})$ and $(L \cap Y_{1,\mathsf{rk}(1)-j})$ is non-empty, then all states in $(L \cap Y_{1,\mathsf{rk}(1)-j})$ belong to C_3, and the mean-payoff of the recurrent class is at least $1-\epsilon$. Moreover, we can also upper bound the number of visits to states in C_2 in expectation that has not been accounted for by visits to states in C_3 before reaching the set $Y_{1,\mathsf{rk}(1)-(j+1)}$ if we leave $(U_1 \setminus Y_{1,\mathsf{rk}(1)-(j+1)})$ by $\kappa(j+1) = (\kappa(j)+1) \cdot (\delta_{\min})^{-1} \cdot \mathsf{pat}(j)$, and then proceed inductively. There is an illustration of this inductive case in Figure 7.

Lemma 8. (Property 1). *Let $0 < \epsilon < \frac{1}{2}$. The strategy $\sigma_1^{\epsilon,1}$ ensures that for all $s \in U_1$ and all positional strategies σ_2 for player 2 we have $\Pr_s^{\sigma_1^{\epsilon,1},\sigma_2}(\mathsf{Safe}(U_1)) = 1$ and $\lim_{t\to\infty} \frac{\sum_{i=0}^{t} \mathbb{E}_s^{\sigma_1^{\epsilon,1},\sigma_2}[\Theta_i]}{t} \geq 1-\epsilon$.*

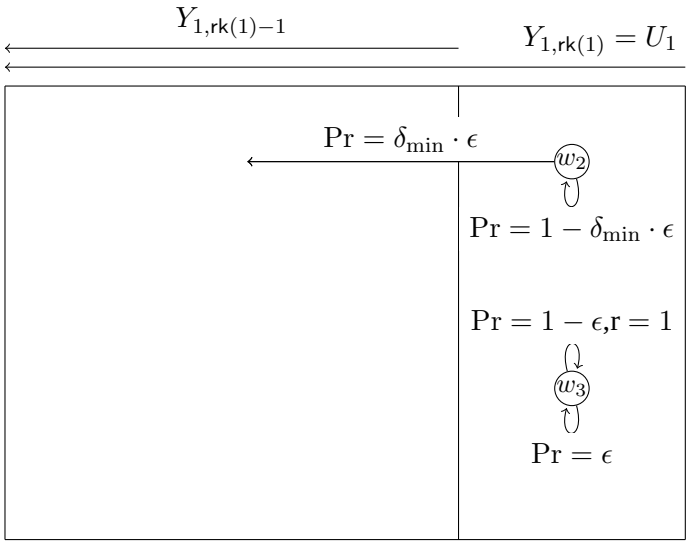

Figure 6: Pictorial illustration of the intuitive explanation of the base case of Lemma 8.

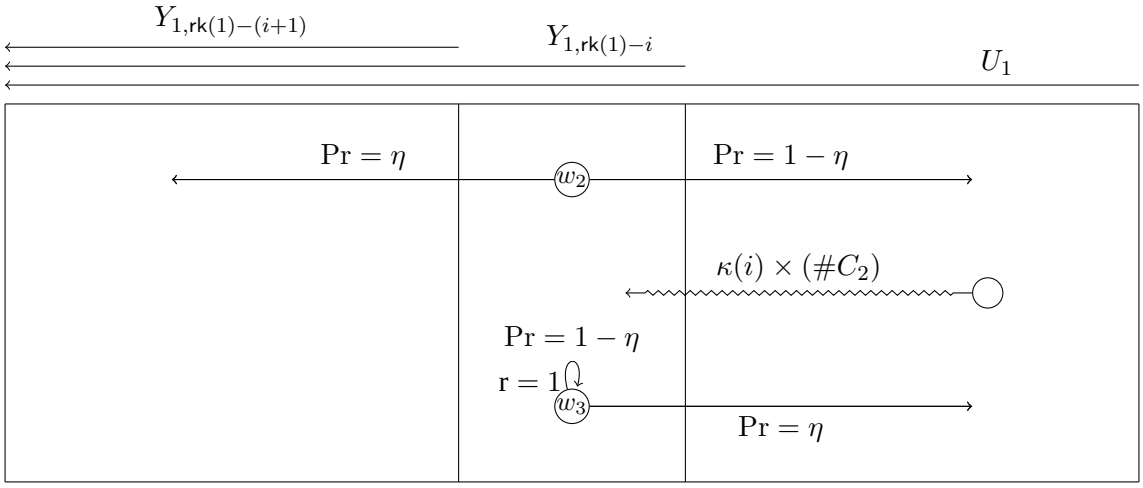

Figure 7: Pictorial illustration of the intuitive explanation of the inductive case of Lemma 8.

Proof. Given $\sigma_1^{\epsilon,1}$, let σ_2 be an arbitrary positional counter-strategy for player 2. Let $C_i^{\sigma_1^{\epsilon,1},\sigma_2} \cap U_1 = C_i$, i.e., given $\sigma_1^{\epsilon,1}$ and σ_2, we have that C_1, C_2, C_3 are the set of states of U_1 that satisfy Equation 4, Equation 5, Equation 6, respectively. Notice that since U_0 is the empty set we have that C_1 is also empty. Therefore we cannot leave U_1 if player 1 follows $\sigma_1^{\epsilon,1}$ (because both Equation 5 and Equation 6 require that we stay in U_1). This ensures that $\mathsf{Safe}(U_1)$ is satisfied with probability 1. We now focus on the mean-payoff.

Basic notations. Let us consider the Markov chain obtained given $\sigma_1^{\epsilon,1}$ and σ_2. For a state $s \in U_1$, let the rank of s be $\mathsf{rk}(s) = (1, j)$, and then we denote j by $\mathsf{rk}_2(s)$ (the second component of the rank). Given a play P in the Markov chain, and a number $t \in \mathbb{N}$, let $\widetilde{r}(P, t)$ be the expected number of times we get reward 0 in the first t steps of P. This implies that $\widetilde{r}(P, 0) = 0$. For each state $s \in U_1$, let P_s^j be (a prefix of) a play in the Markov chain, which ends if a state in $Y_{1,j}$ is reached after the starting point s (i.e., the play does not end at s if $s \in Y_{1,j}$), and if $Y_{1,j}$ is not reached, then the walk does not end. We will also use the following notations: for $0 \leq j \leq \mathsf{rk}(1) - 1$, let us denote by $\kappa(j+1) = \frac{\epsilon}{2} \cdot \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(2m)^{j+1}}$; and let $\mathsf{pat}(j) = \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(\frac{(2m)^{j+1}}{2}-1)}$, the patience of $\sigma_1^{\epsilon,1}$ for states in U_1 of rank $(1, \mathsf{rk}(1) - j)$ (by Lemma 6).

Using recurrent class property. First, observe that since $Y_{1,0}$ is the empty set, the set $Y_{1,0}$ can never be reached, and hence P_s^0 represents the entire play from the start state s, for $s \in U_1$. By Markov property 5 in the Markov chain, the recurrent classes are reached in a finite number of steps with probability 1, and given a recurrent class L is reached, every state in L is reached with probability 1 in a finite number of steps. Given a recurrent class L in U_1, and consider a state s^* in L that has the maximum rank among states in L (i.e., $\mathsf{rk}_2(s^*) = \max_{s' \in L} \mathsf{rk}_2(s')$). Then all states visited after s^* has rank at most the rank of s^*. Hence every play P_s^0 with probability 1, after finitely many steps reaches a state s^* such that all states s' visited after s^* satisfy that $\mathsf{rk}_2(s') \geq \mathsf{rk}_2(s^*)$. Since the mean-payoff is invariant under finite prefixes, we only need to obtain bounds for the mean-payoff of $P_{s^*}^{\mathsf{rk}(s^*)-1}$ (and this play has infinite length by definition as no state with smaller rank is reached in the Markov chain after s^*).

Inductive proof statement. We will show, inductively, that for all $0 \leq j \leq \mathsf{rk}(1)$, all $t \geq 1$, and all states $s \in U_1$, if $\mathsf{rk}_2(s) = \mathsf{rk}(1) - j$, then

$$\widetilde{r}(P_s^{\mathsf{rk}_2(s)-1}, t) \leq t \cdot \epsilon + \frac{\kappa(j+1)}{2} = t \cdot \epsilon + \frac{\epsilon}{4} \cdot \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(2m)^{j+1}}$$

This will imply the desired result, since then the mean-payoff of $P_{s^*}^{\mathsf{rk}_2(s^*)-1}$ is at least $1 - \epsilon$: the play $P_{s^*}^{\mathsf{rk}_2(s^*)-1}$ has infinite length and therefore the expected number of reward 1's must be $t - \widetilde{r}(P_{s^*}^{\mathsf{rk}_2(s^*)-1}, t)$ in the first t steps for all t, because all rewards are either 0 or 1, and hence the mean-payoff of $P_{s^*}^{\mathsf{rk}_2(s^*)-1}$ is $\inf_{t\to\infty} \frac{t - \widetilde{r}(P_{s^*}^{\mathsf{rk}_2(s^*)-1}, t)}{t} \geq 1 - \epsilon$.

Splitting the play. Consider a play $P_s^{\mathsf{rk}_2(s)-1}$ for $s \in U_1$. We will split up the play $P_s^{\mathsf{rk}_2(s)-1}$ into a (possible infinite) sequence of *rank preserving* plays $(P_{s_i}^{\mathsf{rk}_2(s_i)})_{i \geq 0}$, such that $s_0 = s$, and for $i \geq 0$, the play $P_{s_i}^{\mathsf{rk}_2(s_i)}$ ends in state s_{i+1} (which is formally a random variable and must be such that $\mathsf{rk}_2(s_i) = \mathsf{rk}_2(s_{i+1})$ by definition of $P_{s_i}^{\mathsf{rk}_2(s_i)}$ and since if a state of lower rank than $\mathsf{rk}_2(s)$ is reached, then the play $P_s^{\mathsf{rk}_2(s)-1}$ ends). In other words, the next play begins where the previous play ends, and all the starting points of the play has the same rank. Similarly, we will split up plays P_s^j, for $0 \leq j < \mathsf{rk}_2(s)$, into a finite sequence of *rank decreasing* plays $(P_{s_i}^{\mathsf{rk}_2(s_i)-1})_{i \geq 0}$, such that $s_0 = s$, and for $i \geq 0$, the play $P_{s_i}^{\mathsf{rk}_2(s_i)-1}$ ends in state s_{i+1} (which must be such that $\mathsf{rk}_2(s_i) > \mathsf{rk}_2(s_{i+1}) > j$). Note that since the play sequence is decreasing, the

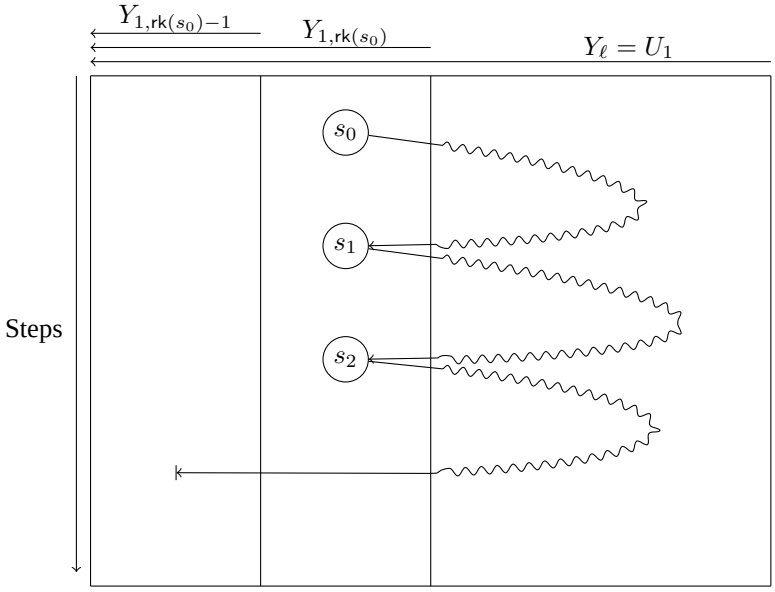

Figure 8: Pictorial illustration of a play $P_{s_0}^{\mathsf{rk}(s_0)-1}$ split into a finite sequence $\left(P_{s_i}^{\mathsf{rk}(s_i)}\right)_{i\geq 0}$ of rank preserving plays. Straight line segments indicate that all states are shown on them, while non-straight segements indicate that there might be states which are not shown.

sequence of plays is finite and the length of the sequence is at most $\mathsf{rk}_2(s) - j$. Pictorial illustrations of rank preserving (both when the sequence is finite and infinite) and rank decreasing plays are given in Figure 8, Figure 9, and Figure 10, respectively.

(Base case). We first consider the base case, where $j = 0$, i.e., we consider s such that $\mathsf{rk}_2(s) = \mathsf{rk}(1)$. Consider the rank preserving split up of the play $P_s^{\mathsf{rk}_2(s)-1}$ into the sequence of plays $(P_{s_i}^{\mathsf{rk}_2(s_i)})_{i\geq 0}$, mentioned above. As already mentioned, safety in $U_1 = Y_{1,\mathsf{rk}(1)}$ is guaranteed, and hence each play $P_{s_i}^{\mathsf{rk}_2(s_i)}$ has length 1. We will consider $\widetilde{r}(P_{s'}^{\mathsf{rk}_2(s')}, t)$, for all s' such that $\mathsf{rk}(s') = \mathsf{rk}(s)$. We will now split the proof into the following two cases: (1) $s' \in C_2$; and (2) $s' \in C_3$; (as already argued at the start of the proof of this lemma, the set C_1 is empty).

1. In each state s' in $(C_2 \cap (Y_{1,\mathsf{rk}(1)} \setminus Y_{1,\mathsf{rk}(1)-1}))$ we reach a state s'' of rank $\mathsf{rk}_2(s'') = \mathsf{rk}_2(s) - 1$ in the next step with probability at least $\left(\frac{\epsilon\cdot\delta_{\min}}{4}\right)^{m-1} \cdot \delta_{\min} = \frac{4}{\epsilon} \cdot \left(\frac{\epsilon\cdot\delta_{\min}}{4}\right)^{m}$ (since $\left(\frac{\epsilon\cdot\delta_{\min}}{4}\right)^{-(m-1)}$ is an upper bound on the patience of states of rank $(1, \mathsf{rk}(1))$ in $\sigma_1^{\epsilon,1}$ by Lemma 6), otherwise we reach a state of rank $\mathsf{rk}(s)$. Hence the expected number of visits to states in C_2 is at most $\frac{\epsilon}{4} \cdot \left(\frac{\epsilon\cdot\delta_{\min}}{4}\right)^{-m}$ before we reach $Y_{1,\mathsf{rk}(1)-1}$. In the worst case we get a reward of 0 in each such step.
2. In each step we are in state s' in $(C_3 \cap (Y_{1,\mathsf{rk}(1)} \setminus Y_{1,\mathsf{rk}(1)-1}))$ we get reward 1 with probability at least $1 - \epsilon$ (by Equation 6).

For the play $P_s^{\mathsf{rk}_2(s)-1} = (P_{s_i}^{\mathsf{rk}_2(s_i)})_{i\geq 0}$, the expected number of indices i such that $s_i \in C_2$ is at most $\frac{\epsilon}{4} \cdot \left(\frac{\epsilon\cdot\delta_{\min}}{4}\right)^{-m}$ (by the first item above). The remaining (in the worst case, at least $t - \frac{\epsilon}{4} \cdot \left(\frac{\epsilon\cdot\delta_{\min}}{4}\right)^{-m}$ in

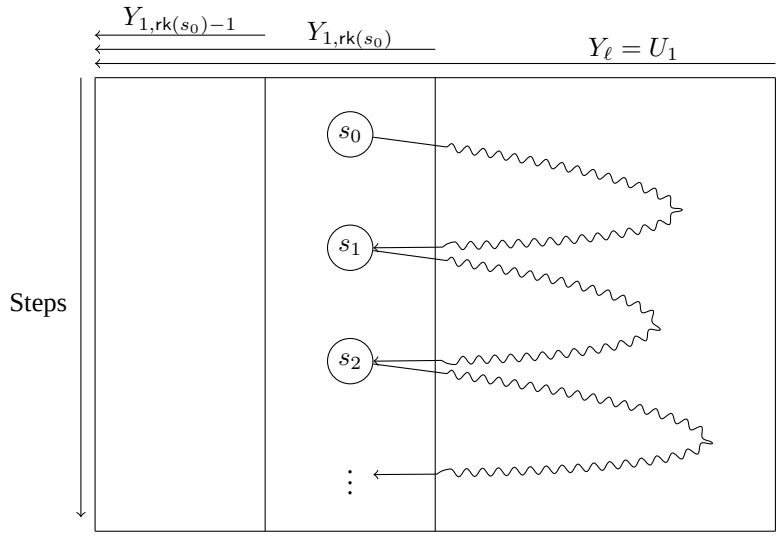

Figure 9: Pictorial illustration of a play $P_{s_0}^{\mathsf{rk}(s)-1}$ split into an infinite sequence $\left(P_{s_i}^{\mathsf{rk}(s_i)}\right)_{i\geq 0}$ of rank preserving plays. Note that the last play could be infinite (which is not pictorially illustrated). Straight line segments indicate that all states are shown on them, while non-straight segements indicate that there might be states which are not shown.

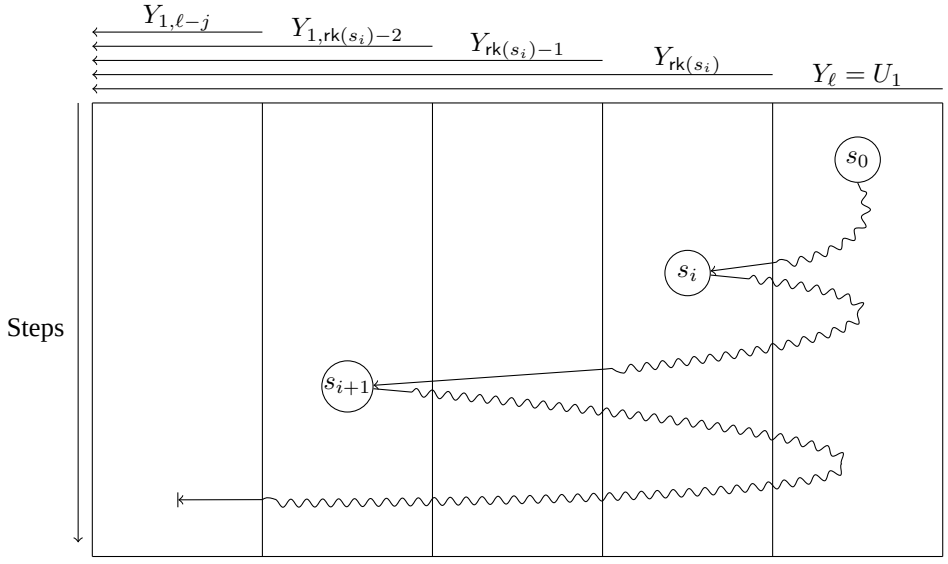

Figure 10: Pictorial illustration of a play $P_{s_0}^{\ell-j}$ split into a (always finite) sequence $\left(P_{s_i}^{\mathsf{rk}(s_i)-1}\right)_{i\geq 0}$ of rank decreasing plays. Note that the last play could be infinite (which is not pictorially illustrated). Straight line segments indicate that all states are shown on them, while non-straight segements indicate that there might be states which are not shown.

expectation) indices i' are such that $s_{i'} \in C_3$, for which the expected reward is at least $1-\epsilon$ (by the second item above). Thus we have

$$\widetilde{r}(P_s^{\mathsf{rk}(s)-1}, t) \leq t \cdot \epsilon + \frac{\epsilon}{4} \cdot \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-m} \leq t \cdot \epsilon + \frac{\epsilon}{4} \cdot \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-2m} = t \cdot \epsilon + \frac{\kappa(1)}{2} \ ,$$

as desired.

(Inductive case). We now consider the inductive case for $j \geq 1$, i.e., we now consider s such that $\mathsf{rk}_2(s) = \mathsf{rk}(1) - j$. Consider the rank preserving split of the play $P_s^{\mathsf{rk}_2(s)-1}$ as $(P_{s_i}^{\mathsf{rk}_2(s_i)})_{i \geq 0}$ as explained before the base case. We will consider $\widetilde{r}(P_{s'}^{\mathsf{rk}_2(s')}, t)$, for all s' with $\mathsf{rk}(s') = \mathsf{rk}(s)$. As in the base case, we will split the proof into the two cases: (1) $s' \in C_2$; and (2) $s' \in C_3$; (and recall C_1 is empty). Before we consider the case analysis, we first present the use of the inductive hypothesis.

Use of inductive hypothesis. The inductive hypothesis will be used in the same way for both cases in the case analysis. Let $t \in \mathbb{N}$ be given. For all states $s'' \in U_1$ such that $\mathsf{rk}_2(s'') > \mathsf{rk}_2(s) = \mathsf{rk}(1) - j$, we will use the inductive hypothesis to upper bound $\widetilde{r}(P_{s''}^{\mathsf{rk}(1)-j}, t)$. Consider the rank decreasing split of $P_{s''}^{\mathsf{rk}(1)-j}$ as $(P_{s'_i}^{\mathsf{rk}_2(s'_i)-1})_{i \geq 0}$. There are most j such plays in the sequence, one for each rank strictly higher than $\mathsf{rk}(1) - j$. We only argue about the worst case, and in the worst case, s'_i is such that $\mathsf{rk}_2(s'_i) = \mathsf{rk}(1) - i$. Let t_i be the random variable indicating the number of steps among the first t steps such that $P_{s''}^{\mathsf{rk}(1)-j}$ is exactly $P_{s'_i}^{\mathsf{rk}_2(s'_i)-1}$. We see that $\widetilde{r}(P_{s''}^{\mathsf{rk}(1)-j}, t) = \sum_{i=0}^{j-1} \widetilde{r}(P_{s'_i}^{\mathsf{rk}_2(s'_i)-1}, t_i)$. By the inductive hypothesis we have that $\widetilde{r}(P_{s'_i}^{\mathsf{rk}(s'_i)-1}, t') \leq t' \cdot \epsilon + \frac{\kappa(i+1)}{2}$ for each $t' \geq 1$. Thus, we get that

$$\widetilde{r}(P_{s''}^{\mathsf{rk}(1)-j}, t) = \sum_{i=0}^{j-1} \widetilde{r}(P_{s'_i}^{\mathsf{rk}(s'_i)-1}, t_i) \leq \sum_{i=0}^{j-1} \left(t_i \cdot \epsilon + \frac{\kappa(i+1)}{2}\right) \leq t \cdot \epsilon + \kappa(j)$$

The first inequality is the inductive hypothesis, and we now argue that $\sum_{i=0}^{j-1} \frac{\kappa(i+1)}{2} \leq \kappa(j)$. We have

$$\sum_{i=0}^{j-1} \frac{\kappa(i+1)}{2} = \frac{\epsilon}{4} \cdot \sum_{i=0}^{j-1} \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(2m)^{i+1}} \leq \frac{\epsilon}{2} \cdot \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(2m)^j} = \kappa(j) \ ,$$

because each term of the sum is over 4 times as large as the preceding (because $(2m)^{i+1} \geq 1 + (2m)^i$, for $m \geq 2$ and $i \geq 0$ and the factor of 4) and thus, the last term is over 2 times larger than the sum of all the other terms (we just use that it is larger). We now consider the case analysis.

- *(States in C_2).* In this case we consider $\widetilde{r}(P_{s'}^{\mathsf{rk}_2(s')}, t)$, for $s' \in C_2$, such that $\mathsf{rk}(s') = \mathsf{rk}(s)$. We know that $\sigma_1^{\epsilon,1}$, has patience $\mathsf{pat}(j)$ for states $s'' \in U_1$ such that $\mathsf{rk}_2(s'') = \mathsf{rk}_2(s) = \mathsf{rk}(1) - j$ (from Lemma 6). In expectation the play $P_s^{\mathsf{rk}_2(s)-1}$ is therefore in a state s'' in C_2 such that $\mathsf{rk}(s'') = \mathsf{rk}(s)$ at most $\mathsf{pat}(j) \cdot (\delta_{\min})^{-1}$ times before reaching a state with lower rank (i.e., before the play ends). If the play does not end, whenever we have been in C_2, we reach some state s'' in U_1 (as safety to U_1 is guaranteed). Also, in the worst case we get a reward of 0 in the every step we are in a state of rank $\mathsf{rk}_2(s)$ in C_2. There are two sub-cases. Either $\mathsf{rk}_2(s'') = \mathsf{rk}_2(s)$ or $\mathsf{rk}_2(s'') > \mathsf{rk}_2(s)$ (because if the rank is lower the walk ends). In the first sub-case the play $P_{s'}^{\mathsf{rk}_2(s')}$ has length 1. In the other case, we have already given an upper bound on $\widetilde{r}(P_{s''}^{\mathsf{rk}(1)-j}, t')$, for all $t' \geq 1$, using the inductive hypothesis. We therefore have that

  $$\widetilde{r}(P_{s'}^{\mathsf{rk}(s')}, t) \leq 1 + \widetilde{r}(P_{s''}^{\mathsf{rk}(1)-j}, t-1) \leq 1 + (t-1) \cdot \epsilon + \kappa(j) = t \cdot \epsilon + (1-\epsilon) + \kappa(j) \leq t \cdot \epsilon + 2 \cdot \kappa(j)$$

where we have just explained the first inequality. The second inequality is our use of the inductive hypothesis as previously explained. The last inequality uses that $\kappa(j) = \frac{\epsilon}{2} \cdot \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(2m)^j} > 8 > 1$ (since $4^{(2m)^j} \geq 16$ and hence $\left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(2m)^j} \geq \frac{16}{\epsilon}$ for $i, m \geq 1$) and $1 - \epsilon < 1$.

- *(States in C_3).* In this case we consider $\widetilde{r}(P_{s'}^{\mathsf{rk}_2(s')}, t)$, for $s' \in C_3$, such that $\mathsf{rk}(s') = \mathsf{rk}(s)$. By construction, the strategy $\sigma_1^{\epsilon,1}$ plays the distribution ξ_1^{η} over $\Gamma_1(s')$, for $\eta = \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{(2m)^j} = \frac{\epsilon}{2} \cdot \frac{1}{\kappa(j)}$. For the play $P_{s'}^{\mathsf{rk}_2(s')}$, the next state s_1 after the start state s' is in U_1 with probability 1; the reward is 1 with probability at least $1 - \eta$, and as well $s' \in Y_{1,\mathsf{rk}(1)-i}$ with probability at least $1 - \eta$ (since Equation 6 is ensured). With the remaining probability of at most η, the play $P_{s'}^{\mathsf{rk}_2(s')}$ goes to a state s'' in U_1. As before the worst case (for the proof) is that with the remaining probability of at most η the state s'' is such that $\mathsf{rk}_2(s'') > \mathsf{rk}_2(s)$, for which we have a upper bound by inductive hypothesis on $\widetilde{r}(P_{s''}^{\mathsf{rk}(1)-i}, t')$, for all $t' \geq 1$. Thus we have that

$$\widetilde{r}(P_{s'}^{\mathsf{rk}_2(s')}, t) \leq \eta + \eta \cdot \widetilde{r}(P_{s''}^{\mathsf{rk}(1)-i}, t-1) \leq \eta + \eta \cdot ((t-1) \cdot \epsilon + \kappa(j))$$

$$= \eta + (t-1) \cdot \eta \cdot \epsilon + \frac{\epsilon}{2} \leq \eta + (t-1) \cdot \epsilon + \frac{\epsilon}{2} \leq t \cdot \epsilon \ .$$

The first inequality is by the preceding explanation. The second inequality uses the inductive hypothesis as previously described. In the first equality, we use that by definition we have $\eta \cdot \kappa(j) = \frac{\epsilon}{2}$. In the third inequality we use that $\eta \cdot \epsilon \leq \epsilon$ since $\eta \leq 1$ and $t \geq 1$; and the final inequality uses that since $\eta \leq \frac{\epsilon}{4}$ we have $\eta + \frac{\epsilon}{2} < \epsilon$ and $\eta \cdot \epsilon < \epsilon$, for $\epsilon < 1$; for $i, m \geq 1$ which ensures $\eta \leq \frac{\epsilon}{4}$.

We now combine the above case analysis to establish the inductive proof. We will now consider $\widetilde{r}(P_s^{\mathsf{rk}_2(s)-1}, t)$ and our rank preserving split $(P_{s_i}^{\mathsf{rk}_2(s_i)})_{i \geq 0}$ of $P_s^{\mathsf{rk}_2(s)-1}$. For all $i \geq 0$, let t_i be the random variable indicating the number of steps $P_s^{\mathsf{rk}_2(s)-1}$ is exactly $P_{s_i}^{\mathsf{rk}_2(s_i)}$ among the first t steps of $P_{s_i}^{\mathsf{rk}_2(s_i)}$. We see that $\widetilde{r}(P_s^{\mathsf{rk}_2(s)-1}, t) = \sum_{i=0}^{k} \widetilde{r}(P_{s_i}^{\mathsf{rk}_2(s_i)}, t_i)$ (the random variable k indicates the highest index such

that $t_k \geq 1$, implying that $t_i \geq 1$ for $0 \leq i \leq k$). Hence, we have that

$$\begin{aligned}
\widetilde{r}(P_s^{\mathsf{rk}_2(s)-1}, t) &= \sum_{i=0}^{k} \widetilde{r}(P_{s_i}^{\mathsf{rk}_2(s_i)}, t_i) \\
&= \sum_{s_i \in C_2,\, i \leq k} \widetilde{r}(P_{s_i}^{\mathsf{rk}_2(s_i)}, t_i) + \sum_{s_i \in C_3,\, i \leq k} \widetilde{r}(P_{s_i}^{\mathsf{rk}_2(s_i)}, t_i) \\
&\leq \sum_{s_i \in C_2,\, i \leq k} (t_i \cdot \epsilon + 2 \cdot \kappa(j)) + \sum_{s_i \in C_3,\, i \leq k} (t_i \cdot \epsilon) \\
&= \sum_{i=0}^{k} (t_i \cdot \epsilon) + \sum_{s_i \in C_2,\, i \leq k} (2 \cdot \kappa(j)) \\
&\leq t \cdot \epsilon + \mathsf{pat}(j) \cdot (\delta_{\min})^{-1} \cdot 2 \cdot \kappa(j) \\
&= t \cdot \epsilon + (\delta_{\min})^{-1} \cdot \epsilon \cdot \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(\frac{(2m)^{j+1}}{2} - 1 + (2m)^j)} \\
&\leq t \cdot \epsilon + (\delta_{\min})^{-1} \cdot \epsilon \cdot \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-((2m)^{j+1}-1)} \\
&\leq t \cdot \epsilon + \frac{\epsilon}{4} \cdot \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(2m)^{j+1}} \\
&= t \cdot \epsilon + \frac{\kappa(j+1)}{2} \ .
\end{aligned}$$

The first equality follows from our preceding explanation. The first inequality uses our bound on $\widetilde{r}(P_{s_i}^{\mathsf{rk}_2(s_i)}, t_i)$ from the respective items above, depending on whether $s_i \in C_2$ or $s_i \in C_3$. The second inequality uses that there are at most $\mathsf{pat}(j) \cdot (\delta_{\min})^{-1}$ indices i such that $s_i \in C_2$, from the first item above, and that $t = \sum_{i=0}^{k} t_i$. The third inequality uses that $(2m)^j \leq \frac{(2m)^{j+1}}{2}$ for $m \geq 2$ and $j \geq 1$. The last follows from $\frac{\epsilon \cdot \delta_{\min}}{4} < \frac{\delta_{\min}}{4}$ and gives the expression we required to establish our inductive claim for j.

This completes the inductive proof and gives us the desired result. □

The combinatorial property established in Lemma 8. The proof of Lemma 8 shows that the strategy $\sigma_1^{\epsilon,1}$ against all positional counter-strategies of the opponent ensures that in the resulting Markov chain all recurrent classes that intersect with U_1 are contained in U_1, all states in U_1 have successors only in U_1; (i.e., the recurrent classes in U_1 are reached with probability 1 from all states in U_1); and in every recurrent class in U_1 the mean-payoff value is at least $1 - \epsilon$.

Lemma 9. (Property 2). *Let $0 < \epsilon < \frac{1}{2}$. The strategy $\sigma_1^{\epsilon,\ell}$ ensures that against all positional strategies σ_2 for all states $s \in (U_\ell \setminus U_{\ell-1})$ we have that*

1. *given the event $\mathsf{Safe}(U_\ell \setminus U_{\ell-1})$, the mean-payoff is at least $1 - \epsilon$;*
2. $\Pr_s^{\sigma_1^{\epsilon,\ell},\sigma_2}(\mathsf{Safe}(U_\ell \setminus U_{\ell-1}) \cup \mathsf{Reach}(U_{\ell-1} \cup \overline{W}^*)) = 1$*; and*
3. $\Pr_s^{\sigma_1^{\epsilon,\ell},\sigma_2}(\mathsf{Safe}(U_\ell \setminus U_{\ell-1}) \cup \mathsf{Reach}(U_{\ell-1})) \geq 1 - \epsilon$.

Proof. Given $\sigma_1^{\epsilon,\ell}$, let σ_2 be an arbitrary positional counter-strategy for player 2. We see that $\sigma_1^{\epsilon,\ell}$ is stationary and follows the distribution ξ^η over $\Gamma_1(s)$ for some $0 < \eta < \epsilon$ in state $s \in (W^* \setminus U_{\ell-1})$. Let $C_i^{\sigma_1^{\epsilon,\ell},\sigma_2} = C_i$, i.e., given $\sigma_1^{\epsilon,\ell}$ and σ_2, we have that C_1, C_2, C_3 are the set of states of $(U_\ell \setminus U_{\ell-1})$ that satisfy Equation 4, Equation 5, Equation 6, respectively. Let R_S be the set of states in $(U_\ell \setminus U_{\ell-1})$, from which $(C_1 \cap (U_\ell \setminus U_{\ell-1}))$ is not reachable in the Markov chain (i.e., in the graph of the Markov chain given $\sigma_1^{\epsilon,\ell}$ and σ_2, the set R_S is the set of states in $(U_\ell \setminus U_{\ell-1})$ from which no state in $(C_1 \cap (U_\ell \setminus U_{\ell-1}))$ is reachable). Equivalently, R_S is the set from which $(U_{\ell-1} \cup \overline{W}^*)$ cannot be reached (the definitions are equivalent, because, from each state s in $(U_\ell \setminus U_{\ell-1}) = (S \setminus (U_{\ell-1} \cup \overline{W}^*))$, the set $(U_{\ell-1} \cup \overline{W}^*)$ can be reached in one-step iff $s \in C_1$). Consider now the segment of the play from state s in $(U_\ell \setminus U_{\ell-1})$ till the play leaves $(U_\ell \setminus U_{\ell-1})$.

1. First we consider the case when $s \in R_S$. This corresponds to the proof of correctness for states in U_1 (note that in the correctness proof of U_1 the set C_1 was empty; and if C_1 is not reached, then the proof is identical to Lemma 8, by construction of the strategy). Hence we have that $\mathsf{Safe}(U_\ell \setminus U_{\ell-1})$ is ensured with probability 1 (because $(U_\ell \setminus U_{\ell-1})$ can only be left from states in $C_1 \cap (U_\ell \setminus U_{\ell-1})$) and $\lim_{t\to\infty} \frac{\sum_{i=0}^{t} \mathbb{E}_s^{\sigma_1^{\epsilon,\ell},\sigma_2}[\Theta_i]}{t} \geq 1 - \epsilon$ (as in Lemma 8). This establishes all the required conditions of the lemma.

2. By Markov property 2, we have that $\mathsf{Reach}(U_{\ell-1} \cup \overline{W}^* \cup R_S)$ happens with probability 1 (since R_S is the set from which $(U_{\ell-1} \cup \overline{W}^*)$ cannot be reached). Note that since $(S \setminus (U_{\ell-1} \cup \overline{W}^*)) = (U_\ell \setminus U_{\ell-1})$, it follows that $\mathsf{Reach}(U_{\ell-1} \cup \overline{W}^* \cup R_S)$ with probability 1 implies $\mathsf{Reach}(U_{\ell-1} \cup \overline{W}^*) \cup \mathsf{Safe}(U_\ell \setminus U_{\ell-1})$ is also ensured with probability 1, since $(U_\ell \setminus U_{\ell-1})$ cannot be left once R_S is reached. This also shows that every recurrent class contained in $(U_\ell \setminus U_{\ell-1})$ must be contained in R_S (and by the first item has mean-payoff value at least $1 - \epsilon$). This shows that given the event $\mathsf{Safe}(U_\ell \setminus U_{\ell-1})$, the mean-payoff is at least $1 - \epsilon$. From every state in $(U_\ell \setminus U_{\ell-1})$, in the Markov chain, we have that $\delta(s)(U_{\ell-1}) \cdot \epsilon \geq \delta(s)(\overline{W}^*)$ (from states which are not in C_1, both probabilities are 0 and C_1 by Equation 4). Hence, Markov property 7 implies that event $\mathsf{Reach}(U_{\ell-1} \cup R_S)$ happens with probability $1-\epsilon$ (since R_S is the set from which $(U_{\ell-1} \cup \overline{W}^*)$ cannot be reached), i.e., we have $\Pr_s^{\sigma_1^{\epsilon,\ell},\sigma_2}(\mathsf{Safe}(U_\ell \setminus U_{\ell-1}) \cup \mathsf{Reach}(U_{\ell-1})) \geq 1 - \epsilon$.

The desired result follows. □

Remark 10. *Lemma 9 proves the desired result only for states in $(U_\ell \setminus U_{\ell-1})$ and can be considered as the base case of Lemma 12 which proves a similar result for states in $(U_{\ell-i} \setminus U_{\ell-(i+1)})$, for $1 \leq i \leq \ell - 2$. The case for states $(U_1 \setminus U_0) = U_1$ is handled by Lemma 8. Note that $\mathsf{Safe}(U_\ell \setminus U_{\ell-1}) \subseteq \mathsf{coBuchi}(U_\ell \setminus U_{\ell-1})$ and since mean-payoff objectives are independent of finite prefixes, it also follows from Lemma 9 that given the event $\mathsf{coBuchi}(U_\ell \setminus U_{\ell-1})$, we have that the mean-payoff is at least $1 - \epsilon$.*

Before presenting the proof for Property 3 we first present a lemma that we will use to prove the property.

Lemma 11. *Given $0 \leq x \leq \frac{1}{2}$ and $0 \leq \epsilon, \eta \leq 1$, consider the four-state Markov chain $G_4^{x,\epsilon,\eta}$ shown in Figure 11. The probability to eventually reach s_1 from s_2 and s_3 is $\frac{x}{\eta+(1+\frac{\epsilon}{2})\cdot x\cdot(1-\eta)}$ and $\frac{x\cdot(1-\eta)}{\eta+(1+\frac{\epsilon}{2})\cdot x\cdot(1-\eta)}$, respectively.*

Proof. Let y_2 and y_3 denote the probability to reach s_1 from s_2 and s_3, respectively. Then we have

$$y_2 = x + (1 - (1 + \frac{\epsilon}{2}) \cdot x) \cdot y_3; \qquad y_3 = (1 - \eta) \cdot y_2 \ .$$

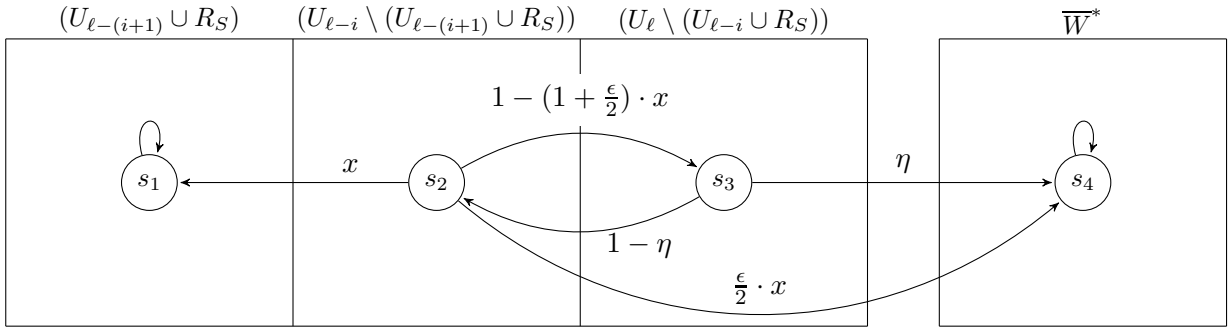

Figure 11: Pictorial illustration of the Markov chain $G_4^{x,\epsilon,\eta}$.

Hence we have

$$y_2 = x + (1-(1+\frac{\epsilon}{2})\cdot x)\cdot(1-\eta)\cdot y_2 \ .$$

Solving for y_2, and then inserting into $y_3 = (1-\eta)\cdot y_2$, we obtain the desired result. □

Lemma 12. (Property 3). *Let $0 < \epsilon < \frac{1}{2}$ and $1 \leq i \leq \ell - 2$. The strategy $\sigma_1^{\epsilon,\ell-i}$ ensures that against all positional strategies σ_2 for all states $s \in (U_\ell \setminus U_{\ell-(i+1)})$ we have that*

1. *given the event $\bigcup_{j\leq i} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)})$, the mean-payoff is at least $1-\epsilon$;*
2. $\Pr_s^{\sigma_1^{\epsilon,\ell-i},\sigma_2}(\bigcup_{j\leq i} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)}) \cup \mathsf{Reach}(U_{\ell-(i+1)} \cup \overline{W}^*)) = 1$*; and*
3. $\Pr_s^{\sigma_1^{\epsilon,\ell-i},\sigma_2}(\bigcup_{j\leq i} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)}) \cup \mathsf{Reach}(U_{\ell-(i+1)})) \geq 1-\epsilon$.

Proof. Given $\sigma_1^{\epsilon,\ell-i}$, let σ_2 be an arbitrary positional counter-strategy for player 2. Let $C_i^{\sigma_1^{\epsilon,\ell-i},\sigma_2} = C_i$, i.e., given $\sigma_1^{\epsilon,\ell-i}$ and σ_2, we have that C_1, C_2, C_3 are the set of states of $(U_\ell \setminus U_{\ell-(i+1)})$ that satisfy Equation 4, Equation 5, Equation 6, respectively. This proof is similar to the proof of Lemma 9. The proof will be by induction in i, where $i = 0$ is the base case. Hence, the base case is settled by Lemma 9. We see that $\sigma_1^{\epsilon,\ell-i}$ is stationary and follows the distribution ξ_1^η over $\Gamma_1(s)$ for some $\eta > 0$ in state $s \in (W^* \setminus U_{\ell-(i+1)})$. We consider the Markov chain obtained by fixing the two strategies. In the worst case, states in $\overline{W}^*$ are absorbing with reward 0; and since the target is to reach $U_{\ell-(i+1)}$ we consider that the plays end if they leave $T = (W^* \setminus U_{\ell-(i+1)})$, i.e., we are interested in the segment of the play in $(W^* \setminus U_{\ell-(i+1)})$. The play can only end from a state in $C_1 \cap T$ because $T = \bigcup_{j\leq i}(U_{\ell-j} \setminus U_{\ell-(j+1)})$ and if a state s in $(U_{\ell-j} \setminus U_{\ell-(j+1)})$ satisfies either Equation 5 (in C_2) or Equation 6 (in C_3), then the set $(U_{\ell-j} \setminus U_{\ell-(j+1)})$ is not left from s in one-step. Now consider a play P in the Markov chain. Let R_S be the subset of T, from which $C_1 \cap T$ is not reachable in the Markov chain. There are two cases

1. **(P starts in $s \in R_S$).** Let $(\ell - i', j') = \mathsf{rk}(s)$. Note that $i' \leq i$, by definition of R_S. Precisely, like in the proof of Lemma 9, we have that $\mathsf{Safe}(U_{\ell-i'} \setminus U_{\ell-(i'+1)})$ is ensured with probability 1,

because the set $(U_{\ell-i'} \setminus U_{\ell-(i'+1)})$ cannot be left from states in C_2 or C_3. Hence, if $i' < i$, then we are done, by induction, since $\sigma_1^{\epsilon,\ell-i}$ follows $\sigma_1^{\eta,\ell-i+1}$ in such states, by construction of $\sigma_1^{\epsilon,\ell-i}$, for $\eta = \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{(2m)^{\mathsf{rk}(\ell-i)}}$ and we have that $\eta < \epsilon$, for $m \geq 2$ and $\mathsf{rk}(\ell-i) \geq 1$. If $i' = i$, then, precisely like in the proof of Lemma 9, the set $(U_{\ell-i} \setminus U_{\ell-(i+1)})$ cannot be left in C_2 or C_3 and hence, using an argument like Lemma 8, we have that $\lim_{t\to\infty} \frac{\sum_{i=0}^{t} \mathbb{E}_s^{\sigma_1^{\epsilon,\ell-i},\sigma_2}[\Theta_i]}{t} \geq 1-\epsilon$, because of the similarities between the construction of the strategy $\sigma_1^{\epsilon,i}$ and $\sigma_1^{\epsilon,1}$ for states in $(U_{\ell-i} \setminus U_{\ell-(i+1)})$ and states in U_1, respectively. Observe that this case is the same as the corresponding case in Lemma 9 and ensures all the required items of the lemma.

2. **(P starts outside R_S: Item (1) of the lemma statement)**. First observe that we can only ensure $\mathsf{Safe}(U_{\ell-j} \setminus U_{\ell-(j+1)})$, for some $j \leq i$, from states in R_S, since from all other states C_1 is reachable and for every j, states in $(C_1 \cap (U_{\ell-j} \setminus U_{\ell-(j+1)}))$, can reach $U_{\ell-(j+1)}$ in one-step with positive probability, by Equation 4. Hence, if $\bigcup_{j\leq i} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)})$ is ensured, then given the event $\bigcup_{j\leq i} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)})$ a recurrent class that is reached must be contained in R_S. Hence given the event $\bigcup_{j\leq i} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)})$, the set R_S is reached in a finite number of steps with probability 1. Since mean-payoffs are independent of finite-prefixes, the finite prefix to reach R_S does not change the mean-payoff. Moreover, since if we start in R_S the mean-payoff is at least $1-\epsilon$, it follows that given the event $\bigcup_{j\leq i} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)})$ we have that the mean-payoff is at least $1-\epsilon$.

3. **(P starts outside R_S: Item (2) of the lemma statement)**. For $0 \leq i' \leq i$, let $\mathcal{E}_{i'}$ denote the following event,
$$\mathcal{E}_{i'} = \bigcup_{j\leq i'} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)}) \cup \mathsf{Reach}(U_{\ell-(i'+1)} \cup \overline{W}^*).$$
Let $\mathrm{SP}(s, \ell-i') = \Pr_s^{\sigma_1^{\epsilon,\ell-i'},\sigma_2}(\mathcal{E}_{i'})$, for all $0 \leq i' \leq i$, denote the *success probability* of the event $\mathcal{E}_{i'}$. We need to argue that $\mathrm{SP}(s, \ell-i) = 1$, for all states in $(U_\ell \setminus U_{\ell-(i+1)})$. By induction we have that $\mathrm{SP}(s, \ell-(i-1)) = 1$, from states in $(U_\ell \setminus U_{\ell-i})$. Since $\sigma_1^{\epsilon,\ell-i}$ has the same support as $\sigma_1^{\epsilon,\ell-(i-1)}$ for all states in $(U_\ell \setminus U_{\ell-i})$, it follows that for each state s in $(U_\ell \setminus U_{\ell-i})$ we have $\mathrm{SP}(s, \ell-i) = 1$. If the event $\bigcup_{j\leq \ell-(i+1)} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)}) \cup \mathsf{Reach}(\overline{W}^*)$ happens, then we are done. Thus, in the worst case we have that $\Pr_s^{\sigma_1^{\epsilon,\ell-i},\sigma_2}(\mathsf{Reach}(U_{\ell-i})) = 1$ from state s in $(U_\ell \setminus U_{\ell-i})$ (clearly, from such states $U_{\ell-i}$ is reachable in the Markov chain since they are reached with probability 1). We only need to argue about the worst case. Let R'_S be the subset of $(U_{\ell-i} \setminus U_{\ell-(i+1)})$, from which $(C_1 \cap (U_{\ell-i} \setminus U_{\ell-(i+1)}))$ cannot be reached in the Markov chain. Hence, for each state s in $(U_{\ell-i} \setminus U_{\ell-(i+1)})$, the state s must either be in R'_S (in which case R'_S is reachable) or the set $(C_1 \cap (U_{\ell-i} \setminus U_{\ell-(i+1)}))$ must be reachable from s. From the set $(C_1 \cap (U_{\ell-i} \setminus U_{\ell-(i+1)}))$, the set $U_{\ell-(i+1)}$ is reached in one-step with positive probability. We therefore get that from any state in $T = ((U_\ell \setminus U_{\ell-i}) \cup (U_{\ell-i} \setminus U_{\ell-(i+1)}))$, the set $(U_{\ell-(i+1)} \cup R'_S)$ is reachable, by transitivity of reachabillity. Hence, by Markov property 8 we have that $\Pr_s^{\sigma_1^{\epsilon,\ell-i},\sigma_2} \mathsf{Reach}((S \setminus T) \cup U_{\ell-(i+1)} \cup R'_S) = 1$, from any state $s \in T$. Note that from states in R'_S no state in $C_1 \cap (U_{\ell-i} \setminus U_{\ell-(i+1)})$ is reachable, and the set $(U_{\ell-i} \setminus U_{\ell-(i+1)})$ can be left only from states in $C_1 \cap (U_{\ell-i} \setminus U_{\ell-(i+1)})$. Hence reachability to R'_S

ensures $\mathsf{coBuchi}((U_{\ell-i} \setminus U_{\ell-(i+1)}))$. Thus we have that

$$\begin{aligned}\mathsf{Reach}((S \setminus T) \cup U_{\ell-(i+1)} \cup R'_S) &= \mathsf{Reach}(U_{\ell-(i+1)} \cup \overline{W}^* \cup U_{\ell-(i+1)} \cup R'_S) \\ &= \mathsf{Reach}(U_{\ell-(i+1)} \cup \overline{W}^* \cup R'_S) \\ &\subseteq \mathsf{Reach}(U_{\ell-(i+1)} \cup \overline{W}^*) \cup \mathsf{coBuchi}(U_{\ell-i} \setminus U_{\ell-(i+1)}) \subseteq \mathcal{E}_i \ .\end{aligned}$$

The first equality uses that $(S \setminus T) = (U_{\ell-(i+1)} \cup \overline{W}^*)$. The first inclusion uses that $\mathsf{Reach}(R'_S)$ ensures $\mathsf{coBuchi}(U_{\ell-i} \setminus U_{\ell-(i+1)})$. Hence, from each state $s \in T$ we have that $\mathrm{SP}(s, \ell - i) = 1$ as desired.

4. **(P starts outside $(R_S \cap T)$: Item (3) of the lemma statement.)**. We will now show that the probability of the event $(\bigcup_{j \leq i} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)}) \cup \mathsf{Reach}(U_{\ell-i}))$ is at least $1 - \epsilon$. We will do so by modeling the worst case using the Markov chain $G_4^{x,\epsilon,\eta}$ of Lemma 11. There is an illustration of the Markov chain $G_4^{x,\epsilon,\eta}$ in Figure 11. We have one state representing each of the following sets

   (1) $(U_{\ell-(i+1)} \cup R_S)$
   (2) $(U_{\ell-i} \setminus (U_{\ell-(i+1)} \cup R_S))$
   (3) $(U_\ell \setminus (U_{\ell-i} \cup R_S))$
   (4) $\overline{W}^*$

   We will refer to the states as s_1, s_2, s_3 and s_4, respectively. We will now argue about the transition probabilities, and first consider the absorbing states.

   The state s_1. We are interested in the probability that $(U_{\ell-(i+1)} \cup R_S)$ is eventually reached. This probability does not depend on what happens after $(U_{\ell-(i+1)} \cup R_S)$ is reached. Hence, we consider s_1 as absorbing, like in $G_4^{x,\epsilon,\eta}$.

   The state s_4. In the worst case $\overline{W}^*$ cannot be left, once reached. Thus s_4 is an absorbing state, like in $G_4^{x,\epsilon,\eta}$.

   The state s_2. For each state $s \in (U_{\ell-i} \setminus (U_{\ell-(i+1)} \cup R_S)) \subseteq (U_{\ell-i} \setminus U_{\ell-(i+1)})$, we must eventually reach a state in either $(C_1 \cap (U_{\ell-i} \setminus U_{\ell-(i+1)})) = ((C_1 \cap T) \cap (U_{\ell-i} \setminus U_{\ell-(i+1)}))$ or $(R_S \cap (U_{\ell-i} \setminus U_{\ell-(i+1)}))$, with probability 1, by Markov property 3 (recall that we cannot reach states outside $(U_{\ell-i} \setminus U_{\ell-(i+1)})$, except from states in $(C_1 \cap (U_{\ell-i} \setminus U_{\ell-(i+1)}))$ by Equation 4, Equation 5 and Equation 6. Also, $(R_S \cap (U_{\ell-i} \setminus U_{\ell-(i+1)}))$ is the subset of $(U_{\ell-i} \setminus U_{\ell-(i+1)})$ from which $(C_1 \cap T)$ cannot be reached). If we reach R_S, an argument similar to the first item in the proof of this lemma shows that we satisfy the desired statement. Thus, in the worst case we always reach $(C_1 \cap (U_{\ell-i} \setminus U_{\ell-(i+1)}))$. For each state s in $(C_1 \cap (U_{\ell-i} \setminus U_{\ell-(i+1)}))$, let $x_s = \delta(s, \sigma_1^{\epsilon,\ell-i}, \sigma_2)(U_{\ell-(i+1)})$ be the one-step transition probability to $U_{\ell-(i+1)}$. By Equation 4, and the construction of the strategy, we have that $\frac{\epsilon}{2} \cdot x_s > \delta(s, \sigma_1^{\epsilon,\ell-i}, \sigma_2)(\overline{W}^*)$. Clearly, in the worst case we have that $\frac{\epsilon}{2} \cdot x_s = \delta(s, \sigma_1^{\epsilon,\ell-i}, \sigma_2)(\overline{W}^*)$ (recall that $\overline{W}^*$ is absorbing). Also, the fact $x_s > \delta(s, \sigma_1^{\epsilon,\ell-i}, \sigma_2)(\overline{W}^*)$ implies that $x_s > 0$ and therefore we have that $x_s \geq \frac{\delta_{\min}}{\mathsf{pat}(\ell-i)}$, where $\mathsf{pat}(\ell - i) = \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(\frac{(2m)^{\mathsf{rk}(\ell-i)}}{2}-1)}$, is an upper bound on the patience of the distribution $\sigma_1^{\epsilon,\ell-i}(s)$, by Lemma 6. Thus with probability x_s we go to $U_{\ell-(i+1)}$, with probability $\frac{\epsilon}{2} \cdot x_s$ we go to $\overline{W}^*$, and with the remaining probability of $(1 - (1 + \frac{\epsilon}{2}) \cdot x_s)$ we go to a state in T, which in the worst case is a state in $(U_\ell \setminus (U_{\ell-i} \cup R_S))$. This is so, because, in

the worst case, to reach $(U_{\ell-(i+1)} \cup R_S)$ from $(U_\ell \setminus (U_{\ell-i} \cup R_S))$ we must go through a state in $(U_{\ell-i} \setminus (U_{\ell-(i+1)} \cup R_S))$, and hence the probability to reach $U_{\ell-(i+1)}$ is minimized when x_s is as small as possible, for all s. That is, $x_s = \frac{\delta_{\min}}{\mathsf{pat}(\ell-i)}$, for all $s \in (C_1 \cap (U_{\ell-i} \setminus U_\ell))$. Let $x = \frac{\delta_{\min}}{\mathsf{pat}(\ell-i)}$. Thus, the transition probabilities are as follows: (i) from s_2 to s_4 is $\frac{\epsilon}{2} \cdot x$; (ii) from s_2 to s_1 is x; and (iii) from s_2 to s_3 is $1 - (1 + \frac{\epsilon}{2}) \cdot x$. Thus, s_2 is like in $G_4^{x,\epsilon,\eta}$.

The state s_3. For each state $s \in (U_\ell \setminus (U_{\ell-i} \cup R_S)) \subseteq (U_\ell \setminus U_{\ell-i})$, by induction and since $\sigma_1^{\epsilon,\ell-i}$ follows $\sigma_1^{\eta,\ell-i}$, we satisfy that $\Pr_s^{\sigma_1^{\epsilon,\ell-i},\sigma_2}(\bigcup_{j\le i-1} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)}) \cup \mathsf{Reach}(U_{\ell-i})) \ge 1-\eta$, where η is $\left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{(2m)^{\mathsf{rk}(\ell-i)}}$. By item (2) of the lemma statement, we enter $\overline{W}^*$ with the remaining probability (which is absorbing). Hence, the worst case must be where $\Pr_s^{\sigma_1^{\epsilon,\ell-i},\sigma_2}(\bigcup_{j\le i-1} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)}) \cup \mathsf{Reach}(U_{\ell-i})) = 1-\eta$ (and thus $\Pr_s^{\sigma_1^{\epsilon,\ell-i},\sigma_2}(\mathsf{Reach}(\overline{W}^*)) = \eta$). As previously argued, in the first item and second item of this lemma, the event $\bigcup_{j\le i-1} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)})$ ensures reachability to R_S (i.e., ensures $\mathsf{Reach}(R_S)$). In the worst case for the proof the probability to reach $(R_S \cup U_{\ell-i-1})$ is minimized, and thus in the worst case we have $\Pr_s^{\sigma_1^{\epsilon,\ell-i},\sigma_2}(\mathsf{Reach}((U_{\ell-i} \setminus (U_{\ell-(i+1)} \cup R_S))) = 1-\eta$ and $\Pr_s^{\sigma_1^{\epsilon,\ell-i},\sigma_2}(\mathsf{Reach}(\overline{W}^*)) = \eta$. Thus, from s_3 the transition probability to s_2 and s_4 are $1-\eta$ and η, respectively. Thus, s_3 is like in $G_4^{x,\epsilon,\eta}$.

The probability to eventually reach s_1 from s_2 or s_3. We have that $x \le \frac{1}{2}$ (since $\mathsf{pat}(\ell-i) \le \frac{1}{2}$, for $m \ge 2$ and $\mathsf{rk}(\ell-i) \ge 1$). Also, $0 < \eta, \epsilon < 1$ (in the case of η, because $m \ge 2$ and $\mathsf{rk}(\ell-i) \ge 1$). Hence we can apply Lemma 11 and get that the probability to eventually reach s_1 from s_2 and s_3 is $\frac{x}{\eta + (1+\frac{\epsilon}{2}) \cdot x \cdot (1-\eta)}$ and $\frac{x \cdot (1-\eta)}{\eta + (1+\frac{\epsilon}{2}) \cdot x \cdot (1-\eta)}$, respectively. Cleary, the probability from s_3 is the smallest. We will show that it is greater than $1-\epsilon$. We have that

$$\frac{x \cdot (1-\eta)}{\eta + (1+\frac{\epsilon}{2}) \cdot x \cdot (1-\eta)} = \frac{1}{\frac{\eta}{x \cdot (1-\eta)} + 1 + \frac{\epsilon}{2}} \ge \frac{1}{1+\epsilon} \ge 1 - \epsilon \ .$$

We will argue about the first inequality last. The second inequality follows from $1 > 1 - \epsilon^2 = (1+\epsilon) \cdot (1-\epsilon) \Rightarrow \frac{1}{1+\epsilon} > 1-\epsilon$. To show the first inequality we will argue that $\frac{\eta}{x \cdot (1-\eta)} \le \frac{\epsilon}{2}$ or, equivalently, that $\frac{2 \cdot \eta}{x \cdot (1-\eta) \cdot \epsilon} \le 1$, since $\epsilon > 0$. We have that

$$\frac{2 \cdot \eta}{x \cdot (1-\eta) \cdot \epsilon} < \frac{4 \cdot \eta}{x \cdot \epsilon} = \frac{4 \cdot \eta \cdot \mathsf{pat}(\ell-i)}{\delta_{\min} \cdot \epsilon} = \eta \cdot \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-\frac{(2m)^{\mathsf{rk}(\ell-i)}}{2}} = \eta^{\frac{1}{2}} < 1 \ .$$

The inequalities comes from $\eta < \frac{1}{2}$ (which is the case because $m \ge 2$ and $\mathsf{rk}(\ell-i) \ge 1$). The first equality is because $x = \frac{\delta_{\min}}{\mathsf{pat}(\ell-i)}$, by definition. The second equality is because $\mathsf{pat}(\ell-i) = \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(\frac{(2m)^{\mathsf{rk}(\ell-i)}}{2} - 1)}$, by definition. The third equality uses that $\eta = \left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{(2m)^{\mathsf{rk}(\ell-i)}}$, by definition.

Ensuring item (3) of the lemma statement. We see that the probability to reach $(U_{\ell-(i+1)} \cup R_S)$ from T is more than $1-\epsilon$ (by recalling the definition of s_1, s_2 and s_3) and thus item (3) of the lemma statement is ensured, because from states in R_S the event $\bigcup_{j\le i} \mathsf{Safe}(U_{\ell-j} \setminus U_{\ell-(j+1)})$ is ensured (as argued in the beginning of the lemma) and hence reaching R_S ensures $\bigcup_{j\le i} \mathsf{coBuchi}(U_{\ell-j} \setminus U_{\ell-(j+1)})$.

The desired result follows. □

Lemma 13. *Let $0 < \epsilon < \frac{1}{2}$. The stationary strategy σ_1^ϵ ensures that for all states $s \in W^*$ and all strategies σ_2 we have $\mathbb{E}_s^{\sigma_1^\epsilon,\sigma_2}[\mathsf{LimSupAvg}] \geq \mathbb{E}_s^{\sigma_1^\epsilon,\sigma_2}[\mathsf{LimInfAvg}] \geq 1-\epsilon$.*

Proof. By construction σ_1^ϵ plays as $\sigma_1^{\beta,1}$ in U_1 and $\sigma_1^{\beta,2}$, for $\beta = \frac{\epsilon}{2}$, in the remaining states. Therefore σ_1^ϵ ensures that the mean-payoff of any play that starts in U_1 is at least $1-\beta$, by Lemma 8. Since σ_1^ϵ is stationary, once σ_1^ϵ is fixed we obtain an MDP for player 2, and in MDPs positional strategies always suffice to minimize mean-payoff objectives [18]. Hence, Lemma 12 shows that if the play starts in $s \in (U_\ell \setminus U_1)$, then with probability $1-\beta$ the play either stays in $(U_j \setminus U_{j-1})$ for some $j \geq 2$ and ensures mean-payoff of at least $1-\beta$ or reaches U_1, from which we will get mean-payoff $1-\beta$. By simple multiplication (using that rewards are at least 0) we therefore see that we get mean-payoff at least

$$(1-\beta)^2 = 1 + \beta^2 - 2\beta \geq 1 - \epsilon.$$

The desired result follows. □

Lemma 13 implies the following inclusion.

Lemma 14. *We have $W^* \subseteq \mathrm{val}_1(\mathsf{LimInfAvg}(\mathrm{r}), \Sigma_1^S) \subseteq \mathrm{val}_1(\mathsf{LimSupAvg}(\mathrm{r}), \Sigma_1^S)$.*

3.2.2 Second inclusion: $\overline{W}^* \subseteq S \setminus \mathrm{val}_1(\mathsf{LimInfAvg}, \Sigma_1^F)$

We will now show that for all states $s \in \overline{W}^*$ that there exists a constant $c > 0$ such that no finite-memory strategy σ_1 for player 1 can ensure value more than $1 - \frac{c^n}{n}$. Again the statement is trivially true if $\overline{W}^*$ is empty, and hence we assume that this is not the case.

Computation of $\overline{W}^*$. We first analyze the computation of $\overline{W}^*$. To analyze the computation of $\overline{W}^*$ we consider the iterative computation W^*

- Let W_0 be S and W_i be $\mu U.\nu X.\mu Y.\nu Z.\mathsf{LimAvgPre}(W_{i-1}, U, X, Y, Z)$.
- Let $X_{i,0}$ be S and $X_{i,j}$ be $\nu X.\mu Y.\nu Z.\mathsf{LimAvgPre}(W_{i-1}, W_i, X_{i,j-1}, Y, Z)$.
- Also let $Z_{i,j,0}$ be S and $Z_{i,j,k}$ be $\mathsf{LimAvgPre}(W_{i-1}, W_i, X_{i,j-1}, X_{i,j}, Z_{i,j,k-1})$.

Let $\overline{\ell} \geq 0$ be the smallest number such that $W_{\overline{\ell}} = W_{\overline{\ell}+1} = W^*$. Let $\overline{\mathsf{rk}}(i)$, be the smallest number j such that $X_{i,j} = X_{i,j+1}$. Also, let $\overline{\mathsf{rk}}(i,j)$, be the smallest number k such that $Z_{i,j,k} = Z_{i,j,k+1}$. We have that for any state s in $\overline{W}^*$, there must be some smallest number i such that s is not in W_i (since W_0 is S, we have that $i > 0$). Also, there must be some smallest j such that s is not in $X_{i,j}$ and similar for k and $Z_{i,j,k}$. We define the rank of a state $s \in \overline{W}^*$ as $\overline{\mathsf{rk}}(s) = (i,j,k)$, where i (resp. j, and k) is the smallest number such that s not in W_i (resp. $X_{i,j}$ and $Z_{i,j,k}$). By definition of $\overline{W}^*$, there exists a constant $c > 0$, such that for a state s, with $\overline{\mathsf{rk}}(s) = (i,j,k)$, for all distributions ξ_1 over $\Gamma_1(s)$ there must exist an counter-action $a_2^{s,\xi_1} \in \Gamma_2(s)$ for player 2 such that all the following conditions hold (i.e., the negation of the conditions of $\mathsf{LimAvgPre}$ hold):

$$(c \cdot \delta(s,\xi_1,a_2^s)(W_i) \leq \delta(s,\xi_1,a_2^s)(\overline{W}_{i-1}))$$

$$\wedge(\delta(s,\xi_1,a_2^s)(X_{i,j-1}) < 1 \vee \delta(s,\xi_1,a_2^s)(X_{i,j}) = 0)$$

$$\wedge(\delta(s,\xi_1,a_2^s)(X_{i,j-1}) < 1 \vee \mathsf{ExpRew}(s,\xi_1,a_2^s) < 1-c \vee \delta(s,\xi_1,a_2^s)(Z_{i,j,k-1}) < 1-c) \ .$$

If the above conditions hold, then one of the following three conditions hold as well. We first explain the following cases: (i) if $\delta(s,\xi_1,a_2^{s,\xi_1})(W_i) > 0$, then $c \cdot \delta(s,\xi_1,a_2^{s,\xi_1})(W_i) \leq \delta(s,\xi_1,a_2^{s,\xi_1})(\overline{W}_{i-1})$ must hold to ensure the first condition above (this corresponds to Case (3) below); (ii) if $\delta(s,\xi_1,a_2^{s,\xi_1})(W_i) = 0$, then the first condition above is satisfied; then we have two sub-cases: (a) if $\delta(s,\xi_1,a_2^{s,\xi_1})(X_{i,j-1}) < 1$, then both the second and third condition is satisfied (this corresponds to Case (2) below); (b) otherwise we must have $\delta(s,\xi_1,a_2^{s,\xi_1})(X_{i,j}) = 0$ to satisfy the second condition above and $(\mathsf{ExpRew}(s,\xi_1,a_2^{s,\xi_1}) < 1-c \quad \vee \quad \delta(s,\xi_1,a_2^{s,\xi_1})(Z_{1,j,i-1}) < 1-c)$ to satisfy the third condition above (this corresponds to Case (1) below). Thus we have that either

- **Case** (1)**.** There is a a_2^{s,ξ_1} such that

$$\begin{aligned} & \delta(s,\xi_1,a_2^{s,\xi_1})(W_i) = 0 \\ \wedge \quad & \delta(s,\xi_1,a_2^{s,\xi_1})(X_{i,j}) = 0 \\ \wedge \quad & \big(\mathsf{ExpRew}(s,\xi_1,a_2^{s,\xi_1}) < 1-c \quad \vee \quad \delta(s,\xi_1,a_2^{s,\xi_1})(Z_{1,j,i-1}) < 1-c\big) \end{aligned}$$

  or;

- **Case** (2)**.** There is a a_2^{s,ξ_1} such that

$$\big(\delta(s,\xi_1,a_2^{s,\xi_1})(W_i) = 0\big) \quad \wedge \quad \big(\delta(s,\xi_1,a_2^{s,\xi_1})(X_{i,j-1}) < 1\big)$$

  or;

- **Case** (3)**.** There is a a_2^{s,ξ_1} such that

$$\big(c \cdot \delta(s,\xi_1,a_2^{s,\xi_1})(W_i) \leq \delta(s,\xi_1,a_2^{s,\xi_1})(\overline{W}_{i-1})\big) \quad \wedge \quad \big(\delta(s,\xi_1,a_2^{s,\xi_1})(W_i) > 0\big) \ .$$

We will use the above three cases explicitly in our proof.

The counter-strategy σ_2 given σ_1. Fix an arbitrary finite-memory strategy σ_1 for player 1. Let the finite set of memories used by σ_1 be Mem. A counter-strategy σ_2 given σ_1 is defined as follows: given the current state s of the game, and current memory state $m \in \mathsf{Mem}$, let ξ_1 be the distribution played by σ_1. The strategy σ_2 for player 2 plays an action a_2^{s,ξ_1} (if there are more than one option for a_2^{s,ξ_1}, pick one arbitrarily) with probability one. If σ_1 uses memory set Mem, then σ_2 also uses the memory set Mem and has the same memory update function.

Upper bound on value ensured by σ_1. We will show that given σ_1 and the counter-strategy σ_2 the mean-payoff value is at most $1 - \frac{c^n}{n}$ for all starting states in $\overline{W}^*$. Also note that the upper bound on the value is independent of the size of the memory, and this shows that in the complement of W^* the values achievable by finite-memory strategies is strictly bounded below 1.

The game $G \times \mathsf{Mem}$. Consider the game G and a product with any deterministic automaton A with state space Q. Every state in $\overline{W}^* \times Q$ in the synchronous product game belongs to the set $\overline{W}^*$ computed in the product game and the ranks also coincide (by the properties of μ-calculus formulae). Consider the synchronous product game $G \times \mathsf{Mem}$ of G and the memories of σ_1 and σ_2, where states corresponds to pairs in (S, Mem) and where $\delta((t,m),a,b)((t',m')) = \delta(s,a,b)(t)$ where $\sigma_1^u(t,a,b,m) = m'$ and hence also $\sigma_2^u(t,a,b,m) = m'$. In this game the strategy corresponding to σ_1 can be interpreted as a stationary strategy σ_1'. Also the strategy corresponding to σ_2 can be interpreted as a positional strategy σ_2' in $G \times \mathsf{Mem}$. Hence given the strategies σ_1 and σ_2 we can obtain a Markov chain on $G \times \mathsf{Mem}$, considering the stationary

strategies σ_1' and σ_2' on the product game. Also for all states $t \in \overline{W}^*$ in G, all the corresponding states (t, m) in $G \times \mathsf{Mem}$ belong to $\overline{W}^*$ computed in the product game and has the same rank as t in G.

Upper bound on value ensured by σ_1. We show that given σ_1 and the counter-strategy σ_2 the mean-payoff value is at most $1 - \frac{c^n}{n}$ for all starting states in $\overline{W}^*$. The proof is split in the following cases, and the basic intuitive arguments are as follows:

1. Consider a play that starts in $\overline{X}_{1,1}$. We show that the play always stays in $\overline{X}_{1,1}$ and Case (1) is satisfied always. Thus we show that from every state there is a path of length at most n where reward 0 occurs at least once.
2. For a play that starts in $\overline{W}_1 \setminus \overline{X}_{1,1}$, we always satisfy either Case (1) or Case (2). First we establish that the event of Case (2) being satisfied infinitely often has probability 0. Hence from some point on Case (1) is always satisfied, and then the argument is similar to the previous case.
3. Finally we consider a play that starts in $\overline{W}^* \setminus \overline{W}_1$. Whenever Case (3) is satisfied, and if the current state is $\overline{W}_j$, for $j > 1$, then $\overline{W}_{j-1}$ is reached with positive probability in one-step. We establish that either (i) we are similar to the previous case or (ii) reach W or $\overline{W}_1$ and the probability to reach $\overline{W}_1$ is at least c^n.

Intuitively, in the first two cases above, we reach a recurrent class that consists of states satisfying Case (1) only, and in such recurrent classes the mean-payoff value is at most $1 - c^n$. In the last case, either we reach a recurrent class of the above type, or whenever we satisfy Case (3) with positive probability $c > 0$ we make progress to a recurrent class of the above type. The above case analysis establish the proof. We now present the formal proof.

Lemma 15. *Fix an arbitrary finite-memory strategy σ_1 and consider the counter-strategy σ_2 given σ_1. For all states in $\overline{W}^*$ we have that $\mathbb{E}_s^{\sigma_1,\sigma_2}[\mathsf{LimSupAvg}] \leq 1 - \frac{c^n}{n}$.*

Proof. In game $G \times \mathsf{Mem}$, let $\overline{C}_i$ be the set of states where Case (i) is satisfied[3]. That is $\overline{C}_1, \overline{C}_2$, and $\overline{C}_3$ satisfy Case (1), Case (2), and Case (3), respectively. We consider the Markov chain given σ_1 and σ_2, and consider a play P^s starting from state s. We will consider three cases to establish the result.

1. **Plays starting in $s \in \overline{X}_{1,1}$.** Recall that $\overline{X}_{1,1}$ is the complement of $X_{1,1}$. Consider state s in $\overline{Z}_{1,1,k}$, for some $k \geq 1$ (that is: states in $\overline{X}_{1,1}$). Since $W_0 = X_{1,0} = S$, we have that the play corresponding to P^s in $G \times \mathsf{Mem}$ is always in $\overline{C}_1$ (note that only in Case (1) do we have probability 0 to go to $\overline{W}_0$ and $\overline{X}_{1,0}$). Hence the play P^s always stays in $\overline{X}_{1,1}$. Hence, from states in $\overline{Z}_{1,1,k}$, if player 1 plays according to σ_1 and player 2 plays σ_2, with probability c we either (i) reach a state in $\overline{Z}_{1,1,k-1}$, or (ii) get a reward of 0. Since $Z_{1,1,0} = S$ we must get a reward of 0 with at least probability c when in $\overline{Z}_{1,1,1}$. Hence, for all states in $\overline{X}_{1,1}$, given player 1 follows σ_1 and player 2 follows σ_2, there is a path of play of length at most $\overline{\mathsf{rk}}(1,1) > \overline{\mathsf{rk}}(1)$ where each step happens with probability at least c and the reward 0 happens at least once. Thus, for any state s in $\overline{X}_{1,1}$, the play P^s stays in $\overline{X}_{1,1}$ and gives a expected average reward of at most $1 - \frac{c^j}{j}$, with probability 1, where $j = \overline{\mathsf{rk}}(1)$. In other words, we have established the following property: in the Markov chain all recurrent classes that intersect with $(\overline{X}_{1,1} \times \mathsf{Mem})$ are contained in $(\overline{X}_{1,1} \times \mathsf{Mem})$ and have mean-payoff at most $1 - \frac{c^n}{n}$.

2. **Plays starting in $s \in (\overline{W}_1 \setminus \overline{X}_{1,1})$.** Consider now state s in $(\overline{W}_1 \setminus \overline{X}_{1,1})$. Since $W_0 = S$, we have that the play P^s_{Mem}, corresponding to P^s in $G \times \mathsf{Mem}$, is always in $(\overline{C}_1 \cup \overline{C}_2)$ (note that in Case (3) we have positive probability to goto W_0). This is the only property of $(\overline{W}_1 \setminus \overline{X}_{1,1})$ we will use. Notice

[3]Note that $\overline{C}_i \neq (S \setminus C_i)$, for $i \in \{1, 2, 3\}$, in general, where C_i is the set defined in Subsection 3.2.1, but this notation is used because $\overline{C}_1, \overline{C}_2, \overline{C}_3$ serve similar roles for properties of $\overline{W}^*$ as C_1, C_2, C_3 did for properties of W^*

that this ensures that P^s always stays in $\overline{W}_1$. Let R_S be the set of states from which no state in $\overline{C}_2$ can be reached. There are now two cases, either P^s_{Mem} reaches a state in R_S or it does not.

- **The play P^s_{Mem} reaches a state in R_S.** Let $j = \overline{\mathsf{rk}}(1)$. Then the mean-payoff is at most $1 - \frac{c^j}{j}$ after reaching R_S, by a argument similar to the one for states in $\overline{X}_{1,1}$. Therefore, in this case, the mean-payoff of P^s is at most $1 - \frac{c^j}{j}$, since the mean-payoff is independent of the finite-prefix.
- **The play P^s_{Mem} does not reach a state in R_S.** In this case, we must visit states in $\overline{C}_2$ infinitely often with probability 1, by Markov property 1. Whenever we are in a state s' in $\overline{C}_2 \cap ((\overline{X}_{1,j} \times \mathsf{Mem}) \setminus (\overline{X}_{1,j-1} \times \mathsf{Mem}))$, we have probability at least $p \cdot \delta_{\min}$ to reach $(\overline{X}_{1,j-1} \times \mathsf{Mem})$ in one-step where $\frac{1}{p}$ is the maximum patience of any distribution played by σ_1. Whenever we are in a state s' in $\overline{C}_1 \cap ((\overline{X}_{1,j} \times \mathsf{Mem}) \setminus (\overline{X}_{1,j-1} \times \mathsf{Mem}))$, we have probability 0 to leave $((\overline{X}_{1,j} \times \mathsf{Mem}) \setminus (\overline{X}_{1,j-1} \times \mathsf{Mem}))$ in one-step. Therefore we must reach $(\overline{X}_{1,1} \times \mathsf{Mem})$ in a finite number of steps with probability 1 and from $(\overline{X}_{1,1} \times \mathsf{Mem})$ we get a mean-payoff of at most $1 - \frac{c^j}{j}$, where $j = \overline{\mathsf{rk}}(1)$, as we have already established in the first item[4].

Therefore, in both cases we get a mean-payoff of at most $1 - \frac{c^j}{j}$ with probability 1, where $j = \overline{\mathsf{rk}}(1)$, i.e., all recurrent classes have mean-payoff of at most $1 - \frac{c^j}{j}$.

3. **Plays starting in $s \in (\overline{W}^* \setminus \overline{W}_1)$.** Consider now state s in $(\overline{W}^* \setminus \overline{W}_1)$. Consider the play P^s in G and the corresponding play P^s_{Mem} in $G \times \mathsf{Mem}$. For $i \geq 1$, let $L_i = W_i \cup \overline{W}_{i-1}$ and note that $\overline{L}_i = \overline{W}_i \setminus \overline{W}_{i-1}$. Let $\overline{R}_i$ be the set of states in $\overline{L}_i$ from which no state in $\overline{C}_3 \cap \overline{L}_i$ is reachable; (note that $\overline{R}_i \subseteq \overline{L}_i \cap (\overline{C}_1 \cup \overline{C}_2)$). Note that from $\overline{L}_i$, the set $\overline{L}_i$ can be left only from states in $\overline{C}_3 \cap \overline{L}_i$. We now consider two sub-cases.

   - We first consider the case where we reach $\overline{R}_i$. Let $j = \overline{\mathsf{rk}}(i)$. In this case, the mean-payoff is at most $1 - \frac{c^j}{j}$ by an argument similar to the argument for s in $\overline{W}_1 \setminus \overline{X}_{1,1}$. The argument for s in $\overline{W}_1 \setminus \overline{X}_{1,1}$ only uses that states in $\overline{C}_1 \cup \overline{C}_2$ are visited. Once $\overline{R}_i$ is reached we are guaranteed that only states in $\overline{R}_i$ are visited, and hence the recurrent classes in $\overline{R}_i$ has mean-payoff of at most $1 - \frac{c^n}{n}$.
   - If $\overline{R}_i$ is not reached, then since from every state $\overline{C}_3 \cap \overline{L}_i$ we have positive transition probability to L_i, it follows that L_i is reached with probability 1, by Markov property 4. But if we reach either W_i or $\overline{W}_{i-1}$, we have a probability of at least c that it will be $\overline{W}_{i-1}$ (since it can only be done whenever P^s_{Mem} is in $\overline{C}_3 \cap L_i$, which ensures so).

   Each time we repeat the second case, all states in $\overline{L}_i$, will never be visited again, in the worst case. Since each set $\overline{L}_i$ must contain atleast one state, we see that, if we repeat the second case k times and thereafter enter $\overline{R}_{i'}$ (and are thus in the first case), then $n - k \geq \overline{\mathsf{rk}}(i')$. We have a probability of c^k to follow such a play and we then get value at most $1 - \frac{c^{n-k}}{n-k}$. Even if we got mean-payoff 1 with the remaining probability of $1 - c^k$, we still have a expected mean-payoff of at most $1 - \frac{c^n}{n-k}$. Thus, we see that in the worst case $k = 0$ with probability 1, in which case we get mean-payoff at most $1 - \frac{c^n}{n}$.

The desired result follows. □

[4] In fact, alternatively we can prove this case using contradiction, since $(\overline{X}_{1,1} \times \mathsf{Mem}) \subseteq \overline{C}_1$ and therefore $(\overline{X}_{1,1} \times \mathsf{Mem}) \subseteq R_S$, since $(\overline{X}_{1,1} \times \mathsf{Mem})$ cannot be left in the Markov chain

Lemma 15 implies the following inclusion.

Lemma 16. *We have* $\mathrm{val}_1(\mathsf{LimSupAvg(r)}, \Sigma_1^F) \subseteq W^*$.

4 Improved Rank-Based Algorithm

In this section we present an improved rank-based algorithm, which is based on the same principle as the small-progress measure algorithm [24] (for parity games). While the naive computation of the μ-calculus formula for the value 1 set requires $O(n^4)$ iterations, the improved algorithm will require $O(n^2)$ iterations.

Basic idea. The basic idea of the algorithm is to consider the ranking function rk from Section 3.2.1 and use that to obtain an algorithm. Notice that $\mathsf{rk}(s)$ for $s \in W^*$ is always a pair (i, j) such that $2 \leq i + j \leq n + 1$ and where $1 \leq i, j \leq n$. We see that for any number k there are $k - 1$ pairs (i, j) such that $i + j = k$ and such that $1 \leq i, j \leq k - 1$. Hence, there are $\sum_{k=1}^{n} k = \frac{n(n+1)}{2}$ such pairs (i, j) such that $2 \leq i + j \leq n + 1$ and where $1 \leq i, j \leq n$. Furthermore we also have a special rank $\top$ for not being in W^*. The ranks are lexicographically ordered as follows

$$(1, 1) < (1, 2) < \cdots < (1, n) < (2, 1) < \cdots < (n, 1) < \top \ .$$

We will thus say that $(i, j) < \top$ for all i, j and $(i, j) < (i', j')$ if $i < i'$ or $i = i'$ and $j < j'$; (and for $(i, j) \leq (i', j')$ we change $j \leq j'$). To distinguish with the ranking function in Section 3.2.1, we denote the ranking function of the improved algorithm as $\mathsf{rk}'(s)$.

Definition of matrix. Consider a given assignment of ranks to states. Let s be some state of rank $\mathsf{rk}'(s) \neq \top$ and therefore of rank (i, j) for some i and j; and also consider a state s' of rank (i', j'). We define some sets, U_s, Y_s, Z_s, X_s, W_s as follows:

1. The state s' is in U_s, if $i > i'$.
2. The state s' is in Y_s, if $i > i'$ or $i' = i$ and $j > j'$.
3. The state s' is in Z_s, if $i > i'$ or $i' = i$ and $j \geq j'$.
4. The state s' is in X_s, if $i \geq i'$.
5. The state s' is in W_s independent of s.

Also if a state s'' has rank $\top$, then it is in the set $\overline{W}_s$. This set also does not depend on s. Let $M^s_{a_1,a_2} \in \{\overline{W}_s, U_s, W_s, Y_s, X_s, Z^1_s, Z^0_s\}$, for $a_1 \in \Gamma_1(s)$ and $a_2 \in \Gamma_2(s)$, be the matrix similar to the matrix M from Section 3.1, except that instead of set $\overline{W}$ use $\overline{W}_s$ and similar for U, Y, Z, X and W.

The RANKALGO **algorithm.** We will refer to our algorithm as RANKALGO and the description is as follows:

1. For each state s set $\mathsf{rk}'(s) \leftarrow (1, 1)$
2. Let $i \leftarrow 0$ and $S^0 \leftarrow S$.
3. (Iteration) While S^i is not the empty set:
   (a) Let $Q^i = S^i \cup \{s \mid \exists a_1 \in \Gamma_1(s), \exists a_2 \in \Gamma_2(s).\ \mathrm{Succ}(s, a_1, a_2) \cap S^i \neq \emptyset\}$ be the set of states in S^i and their predecessors.

(b) For each state $s \in Q^i$ such that $\mathsf{rk}'(s) \neq \top$, run ALGOPRED on M^s (if M^s has not changed since the last time ALGOPRED was run on M^s, then use the result from the last time instead of rerunning ALGOPRED). Let S^{i+1} be the set of states which ALGOPRED rejected.

(c) Increment the rank (according to the lexicographic ordering) of all states in S^{i+1}.

(d) Let $i \leftarrow i + 1$.

4. Return the set of states which does not have rank $\top$.

4.1 Running time of algorithm RANKALGO

We now analyze the running time of the algorithm. We first analyze the work done for updating matrices M^s and then analyze the work done for ALGOPRED computation.

- *Work to update matrix.* For a state s of rank (i, j), notice that we do not need to recalculate the entire M^s whenever some successor s' of s changes rank, but only the entries (a_1, a_2) such that $s' \in \mathrm{Succ}(s, a_1, a_2)$. Also notice that we do not need to change M^s at all whenever s' changes rank to ranks other than in $\{(i, 1), (i, j), (i, j + 1), (i + 1, 1), \top\}$. Hence, as long as s has some rank (i, j), we can do all updates of M^s in time $O(\sum_{a\in\Gamma_1(s),b\in\Gamma_2(s)} |\mathrm{Supp}(s, a, b)|)$. We also recalculate M^s whenever s changes rank, and since each state has at most $O(n^2)$ different ranks therefore we use $O(n^2 \cdot \sum_{s\in S} \sum_{a\in\Gamma_1(s),b\in\Gamma_2(s)} |\mathrm{Supp}(s, a, b)|)$ time to do all updates of M^s for all states s.
- *Work of* ALGOPRED. Note that each entry of M^s can take at most 7 different values, and as long as s has a fixed rank each update makes some entry worse than before. Hence as long as s has some fixed rank (i, j) we can do no more than $6 \cdot |\Gamma_1(s)| \cdot |\Gamma_2(s)|$ updates of M^s. Hence we run ALGOPRED at most $\frac{n(n+1)}{2} \cdot 6 \cdot |\Gamma_1(s)| \cdot |\Gamma_2(s)|$ times for a fixed s.

Therefore, we get a total running time of $O(n^2 \cdot \sum_{s\in S}(|\Gamma_1(s)|^3 \cdot |\Gamma_2(s)|^3 + \sum_{a_1\in\Gamma_1(s),a_2\in\Gamma_2(s)} |\mathrm{Supp}(s, a_1, a_2)|))$, using Lemma 4.

4.2 Proof of correctness of algorithm RANKALGO

The correctness proof is similar to the results of [24]. The proof of [24] shows the equivalence of μ-calculus formula and a rank-based algorithm (called small-progress measure algorithm) for parity games; and the crucial argument of the correctness was based on the fact that the predecessor operator is monotonic. Our correctness proof is similar and uses that $\mathsf{LimAvgPre}$ is monotonic. We just present the proof of one inclusion and the other inclusion is similar. For simplicity we will say that the rank of s is $\mathsf{rk}(s) = \top$ if $s \in \overline{W}^*$. Let $\widetilde{W}^*$ be the output of the algorithm. We show that $\widetilde{W}^* = W^*$.

$\widetilde{W}^* \subseteq W^*$: $\mathsf{rk}'(s) \leq \mathsf{rk}(s)$. We only need to show the statement for $\mathsf{rk}(s) \neq \top$ since otherwise the statement follows by definition. Hence, assume towards contradiction that $\mathsf{rk}'(s) > \mathsf{rk}(s)$ and let $\mathsf{rk}(s) = (i, j)$. Also, we can WLOG assume that s gets assigned a rank higher than $\mathsf{rk}(s)$ in the first iteration for which any state s' gets assigned rank higher than $\mathsf{rk}(s')$ by the algorithm. Therefore in that iteration all states s' are such that the rank assigned by the algorithm is at most $\mathsf{rk}(s')$ and s has rank $\mathsf{rk}(s)$ assigned. Therefore $W^* \subseteq W_s, U_{i-1} \subseteq U_s, U_i \subseteq X_s, Y_{i,j-1} \subseteq Y_s, Y_{i,j} \subseteq Z_s$. But s is in $\mathsf{LimAvgPre}(W^*, U_{i-1}, U_i, Y_{i,j-1}, Y_{i,j})$ by definition since s is such that $\mathsf{rk}(s) = (i, j)$. By monotonicity of $\mathsf{LimAvgPre}$ we have that s is also in $\mathsf{LimAvgPre}(W_s, U_s, X_s, Y_s, Z_s)$, contradicting that s changes rank.

Lemma 17. *The algorithm* RANKALGO *correctly computes the set* $\mathrm{val}_1(\mathsf{LimInfAvg}(\mathrm{r}), \Sigma_1^F)$ *of states in time* $O(n^2 \cdot \sum_{s\in S}(|\Gamma_1(s)|^3 \cdot |\Gamma_2(s)|^3 + \sum_{a_1\in\Gamma_1(s),a_2\in\Gamma_2(s)} |\mathrm{Supp}(s, a_1, a_2)|))$.

5 Main result and Concluding Remarks

We now summarize the main result, and conclude with an open question.

Theorem 18. *The following assertions hold for concurrent mean-payoff games.*

1. (Value 1 set characterization). *Let* $W^* = \nu W.\mu U.\nu X.\mu Y.\nu Z.\mathsf{LimAvgPre}(W, U, X, Y, Z)$*, then we have*

$$W^* = \mathrm{val}_1(\mathsf{LimSupAvg}(\mathrm{r}), \Sigma_1^S) = \mathrm{val}_1(\mathsf{LimSupAvg}(\mathrm{r}), \Sigma_1^F)$$
$$= \mathrm{val}_1(\mathsf{LimInfAvg}(\mathrm{r}), \Sigma_1^S) = \mathrm{val}_1(\mathsf{LimInfAvg}(\mathrm{r}), \Sigma_1^F)$$

2. (Running time). *The value 1 sets* $\mathrm{val}_1(\mathsf{LimSupAvg}(\mathrm{r}), \Sigma_1^S) = \mathrm{val}_1(\mathsf{LimSupAvg}(\mathrm{r}), \Sigma_1^F)$ *can be computed in time* $O(n^2 \cdot \sum_{s \in S}(|\Gamma_1(s)|^3 \cdot |\Gamma_2(s)|^3 + \sum_{a_1 \in \Gamma_1(s), a_2 \in \Gamma_2(s)} |\mathrm{Supp}(s, a_1, a_2)|))$.

3. (Optimal patience). *For all* $\epsilon > 0$*, there exist stationary* ϵ*-optimal strategies in the set* $\mathrm{val}_1(\mathsf{LimSupAvg}(\mathrm{r}), \Sigma_1^S)$ *with patience at most* $\left(\frac{\epsilon \cdot \delta_{\min}}{4}\right)^{-(2m)^n}$.

Proof. The first item follows from Lemma 16 together with Lemma 14. The second item comes from Lemma 17. The third item follows from Lemma 7. □

Notice that the patience closely matches the patience obtained for the concurrent reachability game Purgatory, by Hansen, Ibsen-Jensen and Miltersen [20, Theorem 10] (the bound for $m = 2$ is also in [22]). Concurrent reachability games is a subclass of concurrent mean-payoff games and always have ϵ-optimal stationary strategies, for all $\epsilon > 0$, and all states in Purgatory have value 1. Thus the example provides a closely matching lower bound for patience.

Robustness. Our results show that the value 1 set computation can be achieved by an iterative algorithm with the LimAvgPre operator. Our algorithm for the LimAvgPre operator computation is based on the matrix construction M, and observe that the entries in the matrix depends only on the support set, but not the precise probabilities. It follows that given two concurrent games where the support sets of the transition functions match, but the precise transition probabilities may differ, the value 1 set remains unchanged.

Concluding remarks. In this work we considered concurrent mean-payoff games and presented a polynomial-time algorithm to compute the value 1 set for finite-memory strategies for player 1. An interesting open question is whether the value 1 set with infinite-memory strategies can also be computed in polynomial time.

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

References

[1] T. Bewley and E. Kohlberg. The asymptotic behavior of stochastic games. *Math. Op. Res.*, (1), 1976.

[2] D. Blackwell and T.S. Ferguson. The big match. *AMS*, 39:159–163, 1968.

[3] T. Brázdil, V. Brozek, Kousha Etessami, A. Kucera, and D. Wojtczak. One-counter markov decision processes. In *SODA*, pages 863–874, 2010.

[4] K. Chatterjee. Concurrent games with tail objectives. *Theor. Comput. Sci.*, 388(1-3):181–198, 2007.

[5] K. Chatterjee, L. de Alfaro, and T.A. Henzinger. Qualitative concurrent parity games. *ACM ToCL*, 2011.

[6] K. Chatterjee, R. Majumdar, and T. A. Henzinger. Stochastic limit-average games are in exptime. *Int. J. Game Theory*, 37(2):219–234, 2008.

[7] K. Chatterjee and M. Tracol. Decidable problems for probabilistic automata on infinite words. In *LICS*, pages 185–194, 2012.

[8] Krishnendu Chatterjee. Qualitative concurrent parity games: Bounded rationality. In *CONCUR 2014 - Concurrency Theory - 25th International Conference, CONCUR 2014, Rome, Italy, September 2-5, 2014. Proceedings*, pages 544–559, 2014.

[9] Krishnendu Chatterjee, Luca de Alfaro, and Thomas A. Henzinger. Qualitative concurrent parity games, 2008.

[10] Krishnendu Chatterjee, Arkadeb Ghosal, Thomas A. Henzinger, Daniel T. Iercan, Christoph M. Kirsch, Claudio Pinello, and Alberto L. Sangiovanni-Vincentelli. Logical reliability of interacting real-time tasks. In *Design, Automation and Test in Europe, DATE 2008, Munich, Germany, March 10-14, 2008*, pages 909–914, 2008.

[11] Krishnendu Chatterjee and Rasmus Ibsen-Jensen. Qualitative analysis of concurrent mean-payoff games, arxiv:1409.5306, 2014.

[12] A. Condon. The complexity of stochastic games. *I&C*, 96(2):203–224, 1992.

[13] L. de Alfaro, T.A. Henzinger, and O. Kupferman. Concurrent reachability games. In *FOCS'98*, pages 564–575. IEEE, 1998.

[14] A. Ehrenfeucht and J. Mycielski. Positional strategies for mean payoff games. *Int. Journal of Game Theory*, 8(2):109–113, 1979.

[15] K. Etessami and M. Yannakakis. Recursive concurrent stochastic games. In *ICALP'06 (2)*, LNCS 4052, Springer, pages 324–335, 2006.

[16] H. Everett. Recursive games. In *CTG*, volume 39 of *AMS*, pages 47–78, 1957.

[17] N. Fijalkow, H. Gimbert, and Y. Oualhadj. Deciding the value 1 problem for probabilistic leaktight automata. In *LICS*, pages 295–304, 2012.

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

[19] D. Gillette. Stochastic games with zero stop probabilitites. In *CTG*, pages 179–188. Princeton University Press, 1957.

[20] K. A. Hansen, R. Ibsen-Jensen, and P. B. Miltersen. The complexity of solving reachability games using value and strategy iteration. In *CSR*, pages 77–90, 2011.

[21] K. A. Hansen, M. Koucký, N. Lauritzen, P. B. Miltersen, and E. P. Tsigaridas. Exact algorithms for solving stochastic games: extended abstract. In *STOC*, pages 205–214, 2011.

[22] K. A. Hansen, M. Koucký, and P. B. Miltersen. Winning concurrent reachability games requires doubly-exponential patience. In *LICS*, pages 332–341, 2009.

[23] R. Ibsen-Jensen. *Strategy complexity of two-player, zero-sum games*. PhD thesis, Aarhus University, 2013.

[24] M. Jurdzinski. Small progress measures for solving parity games. In *STACS'00*, pages 290–301. LNCS 1770, Springer, 2000.

[25] J.F. Mertens and A. Neyman. Stochastic games. *Int. J. Game Theory*, 10:53–66, 1981.

[26] L.S. Shapley. Stochastic games. *PNAS*, 39:1095–1100, 1953.

[27] M.Y. Vardi. Automatic verification of probabilistic concurrent finite-state systems. In *FOCS'85*, pages 327–338. IEEE Computer Society Press, 1985.

[28] U. Zwick and M. Paterson. The complexity of mean payoff games on graphs. *Theoretical Computer Science*, 158:343–359, 1996.

6 Appendix — Expanded mu-calculus formula

Description of algorithm. Note that we established that if

$$W^* = \nu W.\mu U.\nu X.\mu Y.\nu Z.\mathsf{LimAvgPre}(W, U, X, Y, Z);$$

then $W^* = \{s \in S \mid \text{val}(\mathsf{LimInfAvg}(\mathrm{r}), \Sigma_1^F)(s) = 1\}$. The μ-calculus formula is a very succinct description of an algorithm. The expanded iterative algorithm is presented as Algorithm 1.

Algorithm 1: Naive μ-calculus Algorithm

Input: A concurrent mean-payoff game G over the set of states S
Output: The set of states W^*

$W \leftarrow S$
repeat
  $W' \leftarrow W$
  $U \leftarrow \emptyset$
  repeat
    $U' \leftarrow U$
    $X \leftarrow W$
    repeat
      $X' \leftarrow X$
      $Y \leftarrow U$
      repeat
        $Y' \leftarrow Y$
        $Z \leftarrow X$
        repeat
          $Z' \leftarrow Z$
          $Z \leftarrow \textsc{AlgoPred}(W, U, X, Y, Z)$
        until $Z = Z'$;
        $Y \leftarrow Z$
      until $Y = Y'$;
      $X \leftarrow Y$
    until $X = X'$;
    $U \leftarrow X$
  until $U = U'$;
  $W \leftarrow U$
until $W = W'$;
return W

7 Technical appendix — Computation of LPre

We now present the details of the computation of $\mathsf{LPre}(s, W, U, A_1, A_2)$. We will establish the Reject property and Accept properties a—d of LPre. We first recall the properties:

(Accept properties of LPre*).* Accepts and returns the set $A_3 \subseteq A_2$ and a parametrized distribution ξ_1^ϵ, for $0 < \epsilon < \frac{1}{2}$, with support $\mathrm{Supp}(\xi_1^\epsilon) \subseteq A_1$, such that the following properties hold:

- (Accept property a). For all $a_2 \in A_3$, the distribution ξ_1^ϵ satisfies Equation 1 for a_2.
- (Accept property b). For all $a_2 \in (A_2 \setminus A_3)$, we have $\mathrm{Succ}(s, \xi_1^\epsilon, a_2) \cap \overline{W} = \emptyset$ and $\mathrm{Succ}(s, \xi_1^\epsilon, a_2) \cap U = \emptyset$.
- (Accept property c). For all $a_1 \in (A_1 \setminus \mathrm{Supp}(\xi_1^\epsilon))$, there exists an action a_2 in $(A_2 \setminus A_3)$ such that $\mathrm{Succ}(s, a_1, a_2) \cap \overline{W} \neq \emptyset$.
- (Accept property d). The set A_3 is largest in the sense that for all $a_2 \in (A_2 \setminus A_3)$ and for all parametrized distributions ξ_1^ϵ over A_1, the Equation 1 cannot be satisfied, while satisfying actions in A_2 using Equation 1, or Equation 2, or Equation 3, for any X, Y, Z such that $U \subseteq Y \subseteq Z \subseteq X \subseteq W$.

The computation of $\mathsf{LPre}(s, W, U, A_1, A_2)$ will be done similar to the computation of the similar named $\mathsf{LPre}(s, W, U)$ in [13, 9], and we will follow notations from [9]. We will use the two methods Stay and Cover, defined as follows:

$$\text{Stay}(s, W, A_1, A_2, A) = \{a_1 \in A_1 \mid \forall a_2 \in (A_2 \setminus A).\big[(\mathrm{Succ}(s, a_1, a_2) \cap \overline{W}) = \emptyset\big]\}$$

$$\text{Cover}(s, U, A_1, A_2, A) = \{a_2 \in A_2 \mid \exists a_1 \in (A_1 \cap A).\big[(\mathrm{Succ}(s, a_1, a_2) \cap U) \neq \emptyset\big]\}$$

The algorithm $\mathsf{LPre}(s, W, U, A_1, A_2)$ is then as follows:

1. Let $A^* \leftarrow \mu A.\big[\text{Stay}(s, W, A_1, A_2, A) \cup \text{Cover}(s, U, A_1, A_2, A)\big]$ and for all $a_1 \in (A^* \cap A_1)$ let $\ell(a_1)$ be the level of a_1 in the formula.
2. If $(A^* \cap A_1)$ is empty, return reject. Otherwise, return accept and $(A^* \cap A_2, \xi_1^\epsilon)$, where ξ_1^ϵ is the parametrized distribution, with support $(A^* \cap A_1)$, and the ranking function of $a_1 \in (A^* \cap A_1)$ is $\frac{\ell(a_1)-1}{2}$.

The algorithm for $\mathsf{LPre}(s, W, U)$ of [13, 9] can be obtained as a special case of our description above as follows:

1. Let $(A_3, \xi_1^\epsilon) \leftarrow \mathsf{LPre}(s, W, U, \Gamma_1(s), \Gamma_2(s))$. If either (i) $\mathsf{LPre}(s, W, U, \Gamma_1(s), \Gamma_2(s))$ rejects; or (ii) $A_3 \neq \Gamma_2(s)$, then return reject, otherwise return accept and ξ_1^ϵ.

We will now show that $\mathsf{LPre}(s, W, U, A_1, A_2)$ satisfies the desired properties.

Lemma 19. *The algorithm* $\mathsf{LPre}(s, W, U, A_1, A_2)$ *satisfies the Reject property of* LPre *and Accept properties a—d. Also, the patience of* ξ_1^ϵ *is at most* $\left(\frac{\epsilon \cdot \delta_{\min}}{2}\right)^{|A_1|-1}$.

Proof. We establish the desired properties.

The reject property of LPre**.** We see that $\mathsf{LPre}(s, W, U, A_1, A_2)$ only rejects if $(A^* \cap A_1)$ is empty. By definition of $\text{Stay}(s, W, A_1, A_2, A)$ we have $(A^* \cap A_1)$ is empty iff for all $a_1 \in A_1$ there exists $a_2 \in (A_2 \setminus A^*)$

such that $(\mathrm{Succ}(s, a_1, a_2) \cap \overline{W}) \neq \emptyset$. We also see the reverse, since we see that also $(A_2 \cap A^*)$ is empty if $(A^* \cap A_1)$ is empty by definition of $\mathrm{Cover}(s, U, A_1, A_2, A)$. This implies that the empty set is a fixpoint of $\mu A.\big[\mathrm{Stay}(s, W, A_1, A_2, A) \cup \mathrm{Cover}(s, U, A_1, A_2, A)\big]$ and thus must be A^*. Since A^* is empty, it follows that for all $a_1 \in A_1$ there exists $a_2 \in (A_2 \setminus A^*) = A_2$ such that $(\mathrm{Succ}(s, a_1, a_2) \cap \overline{W}) \neq \emptyset$. Hence, if $\mathsf{LPre}(s, W, U, A_1, A_2)$ rejects, then the reject property of LPre is satisfied.

Properties of the set A^*. We have that if $\mathsf{LPre}(s, W, U, A_1, A_2)$ returns (A_3, ξ_1^ϵ), then $A^* = (\mathrm{Supp}(\xi_1^\epsilon) \cup A_3)$ and A^* is a fixpoint of $\mu A.\big[\mathrm{Stay}(s, W, A_1, A_2, A) \cup \mathrm{Cover}(s, U, A_1, A_2, A)\big]$.

Accept property a. We note that if we restrict the set of actions of player 1 to $A^* \cap A_1$ and actions of player 2 to A_3, then $\mathsf{LPre}(s, W, U)$ would return accept and the same parametrized distribution, and then the proof of [9, Lemma 4] ensures Accept property a and the desired patience.

Accept property b. We see that for an action a_1 to be in $(A^* \cap A_1) = \mathrm{Supp}(\xi_1^\epsilon)$, by definition of $\mathrm{Stay}(s, W, A_1, A_2, A^*)$, for all a_2 in $(A^* \cap A_2) = A_3$ we have that $(\mathrm{Succ}(s, a_1, a_2) \cap \overline{W}) = \emptyset$ (or equivalently that $(\mathrm{Succ}(s, \xi_1^\epsilon, a_2) \cap \overline{W}) = \emptyset$). This establishes the first half of Accept property b. Also, we see that if an an action a_2 is in $(A_2 \setminus A^*) = (A_2 \setminus A_3)$, then by definition of $\mathrm{Cover}(s, U, A_1, A_2, A^*)$ for all a_1 in $(A^* \cap A_1) = \mathrm{Supp}(\xi_1^\epsilon)$ we have that $(\mathrm{Succ}(s, a_1, a_2) \cap U) = \emptyset$ (or equivalently that $(\mathrm{Succ}(s, \xi_1^\epsilon, a_2) \cap U) = \emptyset$). This establishes the second half of Accept property b.

Accept property c. For A^* to be a fixpoint we must have, by definition of $\mathrm{Stay}(s, W, A_1, A_2, A^*)$, that for each action $a_1 \in (A_1 \setminus A^*) = (A_1 \setminus \mathrm{Supp}(\xi_1^\epsilon))$ that the condition to be in $\mathrm{Stay}(s, W, A_1, A_2, A^*)$ must be violated and thus, there exists $a_2 \in (A_2 \setminus A^*) = (A_2 \setminus A_3)$ such that $(\mathrm{Succ}(s, a_1, a_2) \cap \overline{W}) \neq \emptyset$. This establishes Accept property c.

Accept property d. Along with U and W consider any X, Y, Z such that $U \subseteq Y \subseteq Z \subseteq X \subseteq W$. Consider a real number $0 < \epsilon < \frac{\delta_{\min}}{|A_1|}$ and a distribution ξ_1 over A_1. We will show that if Equation 1 is satisfied by ξ_1 for some action $a_2 \in (A_2 \setminus A_3)$, then there is some action $a_2' \in A_2$ which is not satisfied by either (i) Equation 1; or (ii) Equation 2; or (iii) Equation 3. The proof will be by contradiction and assume towards contradiction that such an action a_2 exists. Let $A_4 \subseteq A_2$ be the set of actions which does satisfy Equation 1 by ξ_1 and let the remaining actions be satisfied by either Equation 2 or Equation 3. Notice that $A_4 \not\subseteq A_3$, since $a_2 \in A_4$ and $a_2 \notin A_3$.

We consider two cases depending on whether or not $\mathrm{Supp}(\xi_1) \subseteq \mathrm{Supp}(\xi_1^\epsilon)$ to establish the result.

- We first consider the case, where $\mathrm{Supp}(\xi_1) \subseteq \mathrm{Supp}(\xi_1^\epsilon)$. Then Equation 1 is violated for all $a_2' \in (A_2 \setminus A_3)$, since U cannot be reached by Accept property b. In particular, it must be violated for a_2. That is a contradiction.

- We next consider the case, where $\mathrm{Supp}(\xi_1) \not\subseteq \mathrm{Supp}(\xi_1^\epsilon)$. Let $a_1 \in (\mathrm{Supp}(\xi_1) \setminus \mathrm{Supp}(\xi_1^\epsilon))$ be an action, such that $a_1 \in \arg\max_{a_1' \in (\mathrm{Supp}(\xi_1) \setminus \mathrm{Supp}(\xi_1^\epsilon))} \xi_1(a_1')$. By Accept property c, there exists an action $a_2' \in (A_2 \setminus A_3)$ such that $\mathrm{Succ}(s, a_1, a_2') \cap \overline{W} \neq \emptyset$, since $a_1 \in (\mathrm{Supp}(\xi_1) \setminus \mathrm{Supp}(\xi_1^\epsilon)) \subseteq (A_1 \setminus \mathrm{Supp}(\xi_1^\epsilon))$. We again split into two cases. Either a_2' is in A_4 or not.

  - We first consider the case then $a_2' \in A_4$. We will show that we go to $\overline{W}$ with too high probability, compared to the probability with which we go to U. We see that $\delta(s, \xi_1, a_2')(\overline{W}) \geq \delta_{\min} \cdot \xi_1(a_1)$, by definition of a_2'. Each action a_1' in $\mathrm{Supp}(\xi_1^\epsilon)$ ensures that $\mathrm{Succ}(s, a_1', a_2') \cap U = \emptyset$ by Accept property b, since $a_2' \notin A_3$. It follows that $\delta(s, \xi_1, a_2')(U) \leq \xi_1(a_1) \cdot (|A_1| - 1)$. This is because each action a_1' such that $\xi(a_1') > \xi(a_1)$ are in $\mathrm{Supp}(\xi_1^\epsilon)$ by definition of a_1 and there are at most $|A_1| - 1$ actions in $(\mathrm{Supp}(\xi_1) \setminus \mathrm{Supp}(\xi_1^\epsilon))$ (since ξ_1 and ξ_1^ϵ are distributions over A_1 and $|\mathrm{Supp}(\xi_1^\epsilon)| \geq 1$). But then $\delta(s, \xi_1, a_2')(U) \cdot \epsilon < \delta(s, \xi_1, a_2')(\overline{W})$ and thus Equation 1 is violated by ξ_1 and a_2'. This contradicts either that $a_2' \in A_4$ or the definition of A_4.

– We next consider the case then $a_2' \in (A_2 \setminus A_4)$. Recall that $\mathrm{Succ}(s, \xi_1, a_2') \cap \overline{W} \neq \emptyset$. Hence, Equation 2 and Equation 3 are violated, since $\mathrm{Succ}(s, \xi_1, a_2') \cap \overline{X} \neq \emptyset$ (because $X \subseteq W$ and if $\overline{W}$ is reached with positive probability, then $\overline{X}$ is reached with positive probability). Moreover, Equation 1 cannot be satisfied either, since $a_2' \notin A_4$. Thus we have a contradiction.

Thus, in all cases we reach contradiction and, hence Accept property d is satisfied.

The desired result follows. □