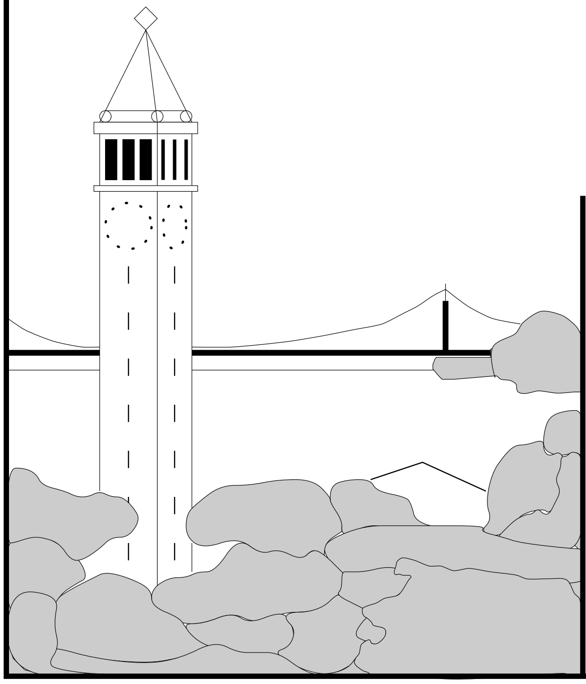
Algorithms for Stochastic Parity Games

Krishnendu Chatterjee and Thomas A. Henzinger



Report No. UCB/CSD-5-1391

May 2005

Computer Science Division (EECS)
University of California
Berkeley, California 94720

Algorithms for Stochastic Parity Games *

Krishnendu Chatterjee[†] Thomas A. Henzinger[†,‡]

[†] EECS, University of California, Berkeley, USA

[‡] EPFL, Switzerland

{c_krish,tah}@eecs.berkeley.edu

May 2005

Abstract

A stochastic graph game is played by two-players on a game graph with probabilistic transitions. We present a strategy improvement algorithm for stochastic graph games with ω-regular conditions specified as parity objectives. From the strategy improvement algorithm we obtain a randomized sub-exponential time algorithm to solve stochastic parity games.

1 Introduction

Graph games. A stochastic graph game [5] is played on a directed graph with three kinds of states: player-1, player-2, and probabilistic states. At player-1 states, player 1 chooses a successor state; at player-2 states, player 2 chooses a successor state; and at probabilistic states, a successor state is chosen according to a given probability distribution. The result of playing the game forever is an infinite path through the graph. If there are no probabilistic states, we refer to the game as a *2-player graph game*; otherwise, as a *$2^1\!/\!_2$-player graph game*.

Games with parity objectives. The theory of graph games with ω-regular winning conditions is the foundation for modeling and synthesizing reactive processes. In the case of stochastic reactive processes, the corresponding stochastic graph games have three players, two of them (System and Environment) behaving adversarially (represented by player 1 and

*This research was supported in part by the ONR grant N00014-02-1-0671, the AFOSR MURI grant F49620-00-1-0327, and the NSF ITR grant CCR-0225610.

player 2), and the third (Uncertainty) behaving probabilistically. The class of $2^1\!/_2$-player graph games with *parity* objectives provide an adequate model for the problem, since every ω-regular objective can be specified as a parity objective. The *quantitative* problem for $2^1\!/_2$-player games with parity objectives Φ asks for the maximal probability with which player 1 with objective Φ, can ensure the satisfaction of Φ from each state (this probability is called the *value* of the game at a state). An *optimal strategy* for player 1 is a strategy, which enable player 1 to win with maximal probability. The existence of *pure memoryless* optimal strategies for $2^1\!/_2$-player games with reachability objectives and 2-player games with parity objectives was extended to $2^1\!/_2$-player games with parity objectives in [14, 4, 18], (a pure memoryless strategy is a deterministic strategy that does not depend on the history of the game). The existence of pure memoryless optimal strategies establishes that the quantitative problem for $2^1\!/_2$-player games with parity objectives can be decided in NP $\cap$ coNP.

Algorithmic analysis. The results of Condon [5] and Emerson-Jutla [9] establish that $2^1\!/_2$-player games with reachability objectives and 2-player games with parity objectives can be decided in NP $\cap$ coNP. For both $2^1\!/_2$-player games with reachability objectives and 2-player games with parity objectives, no polynomial time algorithm is known to solve these games. However, "strategy improvement" algorithms [11] are known for both the above classes of games: Condon [6] presents a strategy improvement algorithm for $2^1\!/_2$-player games with reachability objectives and Vöge-Jurdziński [17] presents a strategy improvement algorithm for 2-player parity games. Although the best known bounds for the worst case running time of these algorithms are exponential, these algorithms work much faster in practice. Using the strategy improvement algorithm analysis, Ludwig [13] presents a randomized sub-exponential time algorithm for $2^1\!/_2$-player reachability games with *binary* game graphs (game graphs with maximum out-degree of at most 2). Björklund et.al. [1] uses a strategy improvement algorithm to present a randomized sub-exponential time algorithm for 2-player parity games. The technique of [1] also yields randomized sub-exponential time algorithm for the general class of $2^1\!/_2$-player reachability games. To solve $2^1\!/_2$-player games with parity objectives, no better algorithm is known other than enumerating over the set of all possible pure memoryless strategies, and choosing the best one as an optimal strategy.

Our results. In this work we present a strategy improvement algorithm for $2^1\!/_2$-player parity games. Our algorithm combines the techniques of 2-player parity games, $2^1\!/_2$-player reachability games and reduction techniques of $2^1\!/\!_2$-player games with parity objectives to 2-player games with parity objectives with some *qualitative* criteria. We then show how to combine the techniques of [1] and our strategy improvement algorithm to obtain a randomized sub-exponential algorithm for $2^1\!/\!_2$-player parity games. Given a game graph G and a parity objective with d-parities, the expected running time of our algorithm is $2^{O\left(\sqrt{d \cdot n \cdot \log(n)}\right)}$, where n is the number of states in G. The algorithm is sub-exponential if $d = O\big(\frac{n^{1-\varepsilon}}{\log(n)}\big)$, for some $\varepsilon > 0$, and for all constants d, the expected running time matches the bound for the best known (expected sub-exponential time) algorithm of $2^1\!/\!_2$-player reachability games.

2 Definitions

We consider several classes of turn-based games, namely, two-player turn-based probabilistic games ($2^1\!/\!_2$-player games), two-player turn-based deterministic games (2-player games), and Markov decision processes ($1^1\!/\!_2$-player games).

Game graphs. A *turn-based probabilistic game graph* (*$2^1\!/\!_2$-player game graph*) $G = ((S, E), (S_1, S_2, S_\bigcirc), \delta)$ consists of a directed graph (S, E), a partition (S_1, S_2, $S_\bigcirc$) of the finite set S of states, and a probabilistic transition function δ: $S_\bigcirc \to \mathcal{D}(S)$, where $\mathcal{D}(S)$ denotes the set of probability distributions over the state space S. The states in S_1 are the *player*-1 states, where player 1 decides the successor state; the states in S_2 are the *player*-2 states, where player 2 decides the successor state; and the states in $S_\bigcirc$ are the *probabilistic* states, where the successor state is chosen according to the probabilistic transition function δ. We assume that for $s \in S_\bigcirc$ and $t \in S$, we have $(s, t) \in E$ iff $\delta(s)(t) > 0$, and we often write $\delta(s, t)$ for $\delta(s)(t)$. For technical convenience we assume that every state in the graph (S, E) has at least one outgoing edge. For a state $s \in S$, we write $E(s)$ to denote the set $\{ t \in S \mid (s, t) \in E \}$ of possible successors.

A set $U \subseteq S$ of states is called *δ-closed* if for every probabilistic state $u \in U \cap S_\bigcirc$, if $(u, t) \in E$, then $t \in U$. The set U is called *δ-live* if for every nonprobabilistic state $s \in U \cap (S_1 \cup S_2)$, there is a state $t \in U$ such that $(s, t) \in E$. A δ-closed and δ-live subset U of S induces a *subgame graph* of G, indicated by $G \restriction U$.

The *turn-based deterministic game graphs* (*2-player game graphs*) are the special case of the $2^1\!/\!_2$-player game graphs with $S_\bigcirc = \emptyset$. The *Markov decision processes* (*$1^1\!/\!_2$-player game graphs*) are the special case of the $2^1\!/\!_2$-player game graphs with $S_1 = \emptyset$ or $S_2 = \emptyset$. We refer to the MDPs with

$S_2 = \emptyset$ as *player*-1 *MDPs*, and to the MDPs with $S_1 = \emptyset$ as *player*-2 *MDPs*.

Plays and strategies. An infinite path, or *play*, of the game graph G is an infinite sequence $\omega = \langle s_0, s_1, s_2, \ldots \rangle$ of states such that $(s_k, s_{k+1}) \in E$ for all $k \in \mathbb{N}$. We write Ω for the set of all plays, and for a state $s \in S$, we write $\Omega_s \subseteq \Omega$ for the set of plays that start from the state s.

A *strategy* for player 1 is a function σ: $S^* \cdot S_1 \to \mathcal{D}(S)$ that assigns a probability distribution to all finite sequences $\vec{w} \in S^* \cdot S_1$ of states ending in a player-1 state (the sequence represents a prefix of a play). Player 1 follows the strategy σ if in each player-1 move, given that the current history of the game is $\vec{w} \in S^* \cdot S_1$, she chooses the next state according to the probability distribution $\sigma(\vec{w})$. A strategy must prescribe only available moves, i.e., for all $\vec{w} \in S^*$, $s \in S_1$, and $t \in S$, if $\sigma(\vec{w} \cdot s)(t) > 0$, then $(s, t) \in E$. The strategies for player 2 are defined analogously. We denote by Σ and Π the set of all strategies for player 1 and player 2, respectively.

Once a starting state $s \in S$ and strategies $\sigma \in \Sigma$ and $\pi \in \Pi$ for the two players are fixed, the outcome of the game is a random walk $\omega_s^{\sigma,\pi}$ for which the probabilities of events are uniquely defined, where an *event* $\mathcal{A} \subseteq \Omega$ is a measurable set of paths. Given strategies σ for player 1 and π for player 2, a play $\omega = \langle s_0, s_1, s_2, \ldots \rangle$ is *feasible* if for every $k \in \mathbb{N}$ the following three conditions hold: (1) if $s_k \in S_{\bigcirc}$, then $(s_k, s_{k+1}) \in E$; (2) if $s_k \in S_1$, then $\sigma(s_0, s_1, \ldots, s_k)(s_{k+1}) > 0$; and (3) if $s_k \in S_2$ then $\pi(s_0, s_1, \ldots, s_k)(s_{k+1}) > 0$. Given two strategies $\sigma \in \Sigma$ and $\pi \in \Pi$, and a state $s \in S$, we denote by $\mathrm{Outcome}(s, \sigma, \pi) \subseteq \Omega_s$ the set of feasible plays that start from s given strategies σ and π. For a state $s \in S$ and an event $\mathcal{A} \subseteq \Omega$, we write $\mathrm{Pr}_s^{\sigma,\pi}(\mathcal{A})$ for the probability that a path belongs to $\mathcal{A}$ if the game starts from the state s and the players follow the strategies σ and π, respectively. In the context of player-1 MDPs we often omit the argument π, because Π is a singleton set.

The strategies that do not use randomization are called pure. A player-1 strategy σ is *pure* if for all $\vec{w} \in S^*$ and $s \in S_1$, there is a state $t \in S$ such that $\sigma(\vec{w} \cdot s)(t) = 1$. We denote by $\Sigma^P \subseteq \Sigma$ the set of pure strategies for player 1. A strategy that is not necessarily pure is called *randomized*. A memoryless player-1 strategy does not depend on the history of the play but only on the current state and hence can be represented as a function σ: $S_1 \to \mathcal{D}(S)$. A *pure memoryless strategy* is a pure strategy that is memoryless. A pure memoryless strategy for player 1 can be represented as a function σ: $S_1 \to S$. We denote by Σ^{PM} the set of pure memoryless strategies; that is, $\Sigma^{PM} = \Sigma^P \cap \Sigma^M$. Analogously we define the family Π^{PM} of pure memoryless strategies for player 2.

Given a pure memoryless strategy $\sigma \in \Sigma^{PM}$, let G_σ be the game graph obtained from G under the constraint that player 1 follows the strategy σ. The corresponding definition G_π for a player-2 strategy $\pi \in \Pi^{PM}$ is analogous, and we write $G_{\sigma,\pi}$ for the game graph obtained from G if both players follow the pure memoryless strategies σ and π, respectively. Observe that given a $2^1\!/\!_2$-player game graph G and a pure memoryless player-1 strategy σ, the result G_σ is a player-2 MDP. Similarly, for a player-1 MDP G and a pure memoryless player-1 strategy σ, the result G_σ is a Markov chain. Hence, if G is a $2^1\!/\!_2$-player game graph and the two players follow pure memoryless strategies σ and π, the result $G_{\sigma,\pi}$ is a Markov chain. These observations will be useful in the analysis of $2^1\!/\!_2$-player games.

Objectives. We specify objectives for the players by providing the set of *winning plays* $\Phi \subseteq \Omega$ for each player. In this paper we study only zero-sum games [15, 10], where the objectives of the two players are strictly competitive. In other words, it is implicit that if the objective of one player is Φ, then the objective of the other player is $\Omega \setminus \Phi$. Given a game graph G and an objective $\Phi \subseteq \Omega$, we write (G, Φ) for the game played on the graph G with the objective Φ for player 1. In this paper we consider *ω-regular objectives* [16] specified as parity objectives. The ω-regular objectives, and subclasses thereof, can be specified in the following forms. For a play $\omega = \langle s_0, s_1, s_2, \ldots \rangle \in \Omega$, we define $\mathrm{Inf}(\omega) = \{ s \in S \mid s_k = s \text{ for infinitely many } k \geq 0 \}$ to be the set of states that occur infinitely often in ω.

- *Reachability and safety objectives.* Given a set $T \subseteq S$ of "target" states, the reachability objective requires that some state of T be visited. The set of winning plays is thus $\mathrm{Reach}(T) = \{ \omega = \langle s_0, s_1, s_2, \ldots \rangle \in \Omega \mid s_k \in T \text{ for some } k \geq 0 \}$. Given a set $F \subseteq S$, the safety objective requires that only states of F be visited. Thus, the set of winning plays is $\mathrm{Safe}(F) = \{ \omega = \langle s_0, s_1, s_2, \ldots \rangle \in \Omega \mid s_k \in F \text{ for all } k \geq 0 \}$.

- *Büchi and coBüchi objectives.* Given a set $B \subseteq S$ of "Büchi" states, the Büchi objective requires that B is visited infinitely often. Formally, the set of winning plays is $\text{Büchi}(B) = \{ \omega \in \Omega \mid \mathrm{Inf}(\omega) \cap B \neq \emptyset \}$. Given $C \subseteq S$, the coBüchi objective requires that all states visited infinitely often are in C. Formally, the set of winning plays is $\text{coBüchi}(C) = \{ \omega \in \Omega \mid \mathrm{Inf}(\omega) \subseteq C \}$.

- *Parity objectives.* For $c, d \in \mathbb{N}$, we let $[c..d] = \{ c, c+1, \ldots, d \}$. Let $p : S \mapsto [0..d]$ be a function that assigns a *priority* $p(s)$ to every state $s \in S$, where $d \in \mathbb{N}$. The *Even parity objective* is defined as $\text{Parity}(p) = \{ \omega \in \Omega \mid \min\big(\text{Inf}(\omega)\big) \text{ is even } \}$, and the *Odd parity objective* as $\text{coParity}(p) = \{ \omega \in \Omega \mid \min\big(\text{Inf}(\omega)\big) \text{ is odd } \}$. Informally we say that a path ω satisfies the parity objective, $\text{Parity}(p)$, if $\omega \in \text{Parity}(p)$. Note that for a priority function $p : V \to \{ 0, 1 \}$, an even parity objective $\text{Parity}(p)$ is equivalent to the Büchi objective $\text{Büchi}(p^{-1}(0))$, i.e., the Büchi set consists of the states with priority 0.

Sure winning, almost-sure winning, and optimality. Given a player-1 objective Φ, a strategy $\sigma \in \Sigma$ is *sure winning* for player 1 from a state $s \in S$ if for every strategy $\pi \in \Pi$ for player 2, we have $\text{Outcome}(s, \sigma, \pi) \subseteq \Phi$. The strategy σ is *almost-sure winning* for player 1 from the state s for the objective Φ if for every player-2 strategy π, we have $\Pr_s^{\sigma,\pi}(\Phi) = 1$. The sure and almost-sure winning strategies for player 2 are defined analogously. Given an objective Φ, the *sure winning set* $\langle\langle 1 \rangle\rangle_{sure}(\Phi)$ for player 1 is the set of states from which player 1 has a sure winning strategy. The *almost-sure winning set* $\langle\langle 1 \rangle\rangle_{almost}(\Phi)$ for player 1 is the set of states from which player 1 has an almost-sure winning strategy. The sure winning set $\langle\langle 2 \rangle\rangle_{sure}(\Omega \setminus \Phi)$ and the almost-sure winning set $\langle\langle 2 \rangle\rangle_{almost}(\Omega \setminus \Phi)$ for player 2 are defined analogously. It follows from the definitions that for all $2\frac{1}{2}$-player game graphs and all objectives Φ, we have $\langle\langle 1 \rangle\rangle_{sure}(\Phi) \subseteq \langle\langle 1 \rangle\rangle_{almost}(\Phi)$. A game is sure (resp. almost-sure) winning for player i, if player i wins surely (resp. almost-surely) from every state in the game. Computing sure and almost-sure winning sets and strategies is referred to as the *qualitative* analysis of $2\frac{1}{2}$-player games [8].

Given ω-regular objectives $\Phi \subseteq \Omega$ for player 1 and $\Omega \setminus \Phi$ for player 2, we define the *value* functions $\langle\langle 1 \rangle\rangle_{val}$ and $\langle\langle 2 \rangle\rangle_{val}$ for the players 1 and 2, respectively, as the following functions from the state space S to the interval $[0, 1]$ of reals: for all states $s \in S$, let $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s) = \sup_{\sigma \in \Sigma} \inf_{\pi \in \Pi} \Pr_s^{\sigma,\pi}(\Phi)$ and $\langle\langle 2 \rangle\rangle_{val}(\Omega \setminus \Phi)(s) = \sup_{\pi \in \Pi} \inf_{\sigma \in \Sigma} \Pr_s^{\sigma,\pi}(\Omega \setminus \Phi)$. In other words, the value $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s)$ gives the maximal probability with which player 1 can achieve her objective Φ from state s, and analogously for player 2. The strategies that achieve the value are called optimal: a strategy σ for player 1 is *optimal* from the state s for the objective Φ if $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s) = \inf_{\pi \in \Pi} \Pr_s^{\sigma,\pi}(\Phi)$. The optimal strategies for player 2 are defined analogously. Computing values is referred to as the *quantitative* analysis of $2\frac{1}{2}$-player games. The set of states with value 1 is called the *limit-sure winning set* [8]. For $2\frac{1}{2}$-player game graphs with ω-regular objectives the almost-sure and limit-sure winning sets coincide [3].

Consider a family $\Sigma^{\mathcal{C}} \subseteq \Sigma$ of special strategies for player 1. We say that the family $\Sigma^{\mathcal{C}}$ *suffices* with respect to a player-1 objective Φ on a class $\mathcal{G}$ of game graphs for *sure winning* if for every game graph $G \in \mathcal{G}$ and state $s \in \langle\langle 1 \rangle\rangle_{sure}(\Phi)$, there is a player-1 strategy $\sigma \in \Sigma^{\mathcal{C}}$ such that for every player-2 strategy $\pi \in \Pi$, we have $\mathrm{Outcome}(s, \sigma, \pi) \subseteq \Phi$. Similarly, the family $\Sigma^{\mathcal{C}}$ *suffices* with respect to the objective Φ on the class $\mathcal{G}$ of game graphs for *almost-sure winning* if for every game graph $G \in \mathcal{G}$ and state $s \in \langle\langle 1 \rangle\rangle_{almost}(\Phi)$, there is a player-1 strategy $\sigma \in \Sigma^{\mathcal{C}}$ such that for every player-2 strategy $\pi \in \Pi$, we have $\Pr_s^{\sigma,\pi}(\Phi) = 1$; and for *optimality*, if for every game graph $G \in \mathcal{G}$ and state $s \in S$, there is a player-1 strategy $\sigma \in \Sigma^{\mathcal{C}}$ such that $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s) = \inf_{\pi \in \Pi} \Pr_s^{\sigma,\pi}(\Phi)$.

For sure winning, the $1^1\!/\!_2$-player and $2^1\!/\!_2$-player games coincide with 2-player (deterministic) games where the random player (who chooses the successor at the probabilistic states) is interpreted as an adversary, i.e., as player 2. Theorem 1 and Theorem 2 state the classical determinacy results for 2-player and $2^1\!/\!_2$-player game graphs with parity objectives.

Theorem 1 (Qualitative determinacy [9]**)** *For all 2-player game graphs and parity objectives Φ, we have $\langle\langle 1 \rangle\rangle_{sure}(\Phi) \cap \langle\langle 2 \rangle\rangle_{sure}(\Omega \setminus \Phi) = \emptyset$ and $\langle\langle 1 \rangle\rangle_{sure}(\Phi) \cup \langle\langle 2 \rangle\rangle_{sure}(\Omega \setminus \Phi) = S$. Moreover, on 2-player game graphs, the family of pure memoryless strategies suffices for sure winning with respect to parity objectives.*

Theorem 2 (Quantitative determinacy [4, 14]**)** *For all $2^1\!/\!_2$-player game graphs, all parity objectives Φ, and all states s, we have $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s) + \langle\langle 2 \rangle\rangle_{val}(\Omega \setminus \Phi)(s) = 1$. Moreover, on $2^1\!/\!_2$-player game graphs, the family of pure memoryless strategies suffices for optimality with respect to parity objectives.*

Since in $2^1\!/\!_2$-player games with parity objectives, pure memoryless strategies suffices for optimality, in sequel we consider only pure memoryless strategies for both players. Moreover, since parity objectives are infinitary objectives the following proposition is immediate.

Proposition 1 (Optimality conditions) *For a parity objective Φ, for every $s \in S$ the following conditions hold.*

1. *If $s \in S_1$, then for all $t \in E(s)$ we have $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s) \geq \langle\langle 1 \rangle\rangle_{val}(\Phi)(t)$, and for some $t \in E(s)$ we have $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s) = \langle\langle 1 \rangle\rangle_{val}(\Phi)(t)$.*
2. *If $s \in S_2$, then for all $t \in E(s)$ we have $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s) \leq \langle\langle 1 \rangle\rangle_{val}(\Phi)(t)$, and for some $t \in E(s)$ we have $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s) = \langle\langle 1 \rangle\rangle_{val}(\Phi)(t)$.*

3. If $s \in S_{\bigcirc}$, then $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s) = \big(\sum_{t \in E(s)} \langle\langle 1 \rangle\rangle_{val}(\Phi)(t)\ \delta(s,t)\big)$.

Similar conditions hold for the value function $\langle\langle 2 \rangle\rangle_{val}(\Omega \setminus \Phi)$ of player 2.

3 Strategy Improvement Algorithm

The main result of this section is a strategy improvement algorithm for $2^1\!/\!_2$-player games with parity objectives. In section 3.1 we gather a few key properties of $2^1\!/\!_2$-player games with parity objectives that were proved in [3, 2]. We use the properties in section 3.2 to develop a strategy improvement algorithm for $2^1\!/\!_2$-player parity games.

3.1 Key properties

We present a reduction of $2^1\!/\!_2$-player parity games to 2-player parity games preserving the ability of player 1 to win almost-surely.

Reduction. Given a $2^1\!/\!_2$-player game graph $G = ((S, E), (S_1, S_2, S_{\bigcirc}), \delta)$, with a priority function $p : S \to [0..d]$ we construct a 2-player game graph $\overline{G} = ((\overline{S}, \overline{E}), (\overline{S}_1, \overline{S}_2), \overline{\delta})$ together with a priority function $\overline{p} : \overline{S} \to [0..d]$. The construction is specified as follows. For every nonprobabilistic state $s \in S_1 \cup S_2$, there is a corresponding state $\overline{s} \in \overline{S}$ such that (1) $\overline{s} \in \overline{S}_1$ iff $s \in S_1$, and (2) $\overline{p}(\overline{s}) = p(s)$, and (3) $(\overline{s}, \overline{t}) \in \overline{E}$ iff $(s,t) \in E$. Every probabilistic state $s \in S_{\bigcirc}$ is replaced by the gadget shown in Figure 1. In the figure, diamond-shaped states are player-2 states (in $\overline{S}_2$), and square-shaped states are player-1 states (in $\overline{S}_1$). From the state $\overline{s}$ with $\overline{p}(\overline{s}) = p(s)$, the players play the following 3-step game in $\overline{G}$. First, in state $\overline{s}$ player 2 chooses a successor $(\widetilde{s}, 2k)$, for $k \in \{0, 1, \ldots, j\}$, where $p(s) = 2j$ or $p(s) = 2j - 1$. For every state $(\widetilde{s}, 2k)$, we have $\overline{p}(\widetilde{s}, 2k) = p(s)$. For $k > 1$, in state $(\widetilde{s}, 2k)$ player 1 chooses from two successors: state $(\widehat{s}, 2k-1)$ with $\overline{p}(\widehat{s}, 2k-1) = 2k-1$, or state $(\widehat{s}, 2k)$ with $\overline{p}(\widehat{s}, 2k) = 2k$. The state $(\widetilde{s}, 0)$ has only one successor $(\widehat{s}, 0)$, with $\overline{p}(\widehat{s}, 0) = 0$. Finally, in each state $(\widehat{s}, k)$ the choice is between all states $\overline{t}$ such that $(s,t) \in E$, and it belongs to player 1 if k is odd, and to player 2 if k is even.

We consider $2^1\!/\!_2$-player games played on the graph G with the parity objective $\mathrm{Parity}(p)$ for player 1. We denote by $\overline{G} = \mathrm{Tr}_{\mathrm{as}}(G)$ the 2-player game, with parity objective $\mathrm{Parity}(\overline{p})$, as defined by the reduction above. Also given a strategy (pure memoryless) $\overline{\sigma}$ in the 2-player game $\overline{G}$, a strategy $\sigma = \mathrm{Tr}_{\mathrm{as}}(\overline{\sigma})$ in the $2^1\!/\!_2$-player game G is defined as follows:

$$\sigma(s) = t, \text{ if and only if } \overline{\sigma}(\overline{s}) = \overline{t}; \text{ for all } s \in S_1.$$

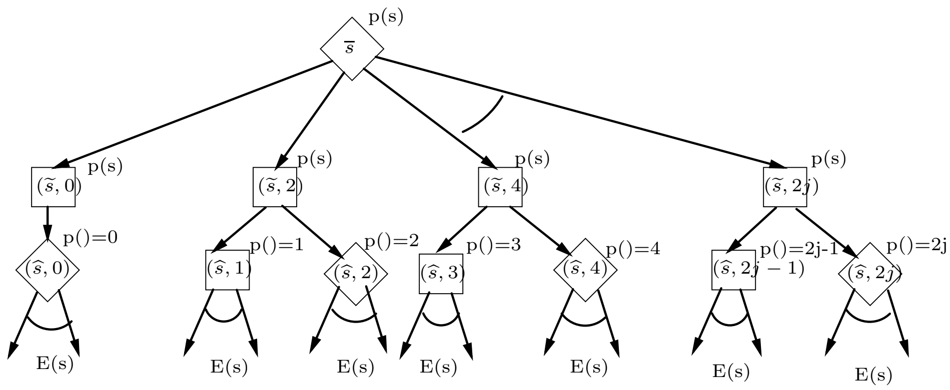


Figure 1: Gadget for the reduction of 2½-player parity games to 2-player parity games.

Similar definitions hold for player 2.

Lemma 1 ([3]) *Given a 2½-player game graph G with the parity objective Parity(p) for player 1, let $\overline{U}_1$ and $\overline{U}_2$ be the sure winning sets for players 1 and 2, respectively, in the 2-player game graph $\overline{G} = \mathrm{Tr}_{\mathrm{as}}(G)$ with the modified parity objective Parity($\overline{p}$). Define the sets U_1 and U_2 in the original 2½-player game graph G by $U_1 = \{ s \in S \mid \overline{s} \in \overline{U}_1 \}$ and $U_2 = \{ s \in S \mid \overline{s} \in \overline{U}_2 \}$. Then the following assertions hold:*

1. *$U_1 = \langle\langle 1 \rangle\rangle_{almost}(Parity(p)) = (S \setminus U_2)$.*
2. *If $\overline{\sigma}$ is a pure memoryless sure winning strategy for player 1 from $\overline{U}_1$ in $\overline{G}$, then $\sigma = \mathrm{Tr}_{\mathrm{as}}(\overline{\sigma})$ is an almost-sure winning strategy for player 1 from U_1 in G.*

Boundary probabilistic states. Given a set U of states, let $Bou(U) = \{ s \in U \cap S_{\bigcirc} \mid \exists t \in E(s), t \notin U \}$, be the set of *boundary* probabilistic states that have an edge out of U. Given a set U of states and a parity objective Parity(p) for player 1, we define a transformation $\mathrm{Tr}_{\mathrm{win}_1}(U)$ of U as follows: every state s in $Bou(U)$ is converted to an *absorbing* state (state with only a self-loop) and assigned an even priority $2\lfloor \frac{d}{2} \rfloor$, i.e., every state in $Bou(U)$ is converted to a sure winning state for player 1. Observe that if U is δ-live, then $\mathrm{Tr}_{\mathrm{win}_1}(G \upharpoonright U)$ is a gamegraph.

Value classes. Given a parity objective Φ, for every real $r \in \mathbb{R}$ the *value class* with value r, $\mathrm{VC}(r) = \{ s \in S \mid \langle\langle 1 \rangle\rangle_{val}(\Phi)(s) = r \}$, is the set of states with value r for player 1. It follows from Proposition 1 that for every $r > 0$, the value class $\mathrm{VC}(r)$ is δ-live. The following lemma establishes a connection between value classes, the transformation $\mathrm{Tr}_{\mathrm{win}_1}$ and the almost-sure winning states.

Lemma 2 ([2]) *For every value class* $\mathrm{VC}(r)$*, for* $r > 0$*, the game* $\mathrm{Tr}_{\mathrm{win}_1}(G \upharpoonright \mathrm{VC}(r))$ *is almost-sure winning for player 1.*

It follows from Lemma 1 and Lemma 2, that for every value class $\mathrm{VC}(r)$, with $r > 0$, the game $\mathrm{Tr}_{\mathrm{as}}(\mathrm{Tr}_{\mathrm{win}_1}(G \upharpoonright \mathrm{VC}(r)))$ is sure winning for player 1.

3.2 Strategy improvement algorithm

We now present a strategy improvement algorithm for $2^1\!/\!_2$-player games with parity objectives.

Notation. Given a strategy π and a set U of states, we denote by $(\pi \upharpoonright U)$ a strategy that for every state in U follows the strategy π.

Values and value class given strategies. Given a player-2 strategy π and a parity objective Φ, we denote the value of player 1 given the strategy π as follows: $\langle\!\langle 1 \rangle\!\rangle_{val}^{\pi}(\Phi)(s) = \sup_{\sigma \in \Sigma^{PM}} \mathrm{Pr}_s^{\sigma,\pi}(\Phi)$. Similarly we define the value classes given strategy π as $\mathrm{VC}^{\pi}(r) = \{ s \in S \mid \langle\!\langle 1 \rangle\!\rangle_{val}^{\pi}(\Phi)(s) = r \}$.

Witness for player 2. Given a $2^1\!/\!_2$-player gamegraph G, and a parity objective Φ for player 1, a witness $wit_2 = (\pi, \overline{\pi}_Q)$ for player 2 is described as follows:

- The strategy π is a strategy in the game G.
- For every value class $\mathrm{VC}^{\pi}(r)$, the strategy $(\overline{\pi}_Q \upharpoonright \mathrm{VC}^{\pi}(r))$ is a strategy in the 2-player game $\overline{G}_r = \mathrm{Tr}_{\mathrm{as}}(\mathrm{Tr}_{\mathrm{win}_1}(G \upharpoonright \mathrm{VC}^{\pi}(r)))$. Also we must have $\pi = \mathrm{Tr}_{\mathrm{as}}(\overline{\pi}_Q)$.

A witness $wit_2 = (\pi, \overline{\pi}_Q)$ for player 2 is an *optimal* witness if the strategy π is an optimal strategy for player 2.

Ordering of witnesses. We define an ordering relation $\prec$ on witnesses as follows: given two witnesses $wit_2 = (\pi, \overline{\pi}_Q)$ and $wit_2' = (\pi', \overline{\pi}_Q')$, we have $wit_2 \prec wit_2'$ if and only if the following conditions hold:

1. for all states s, we have $\langle\!\langle 1 \rangle\!\rangle_{val}^{\pi}(\Phi)(s) \geq \langle\!\langle 1 \rangle\!\rangle_{val}^{\pi'}(\Phi)(s)$ and for some state s we have $\langle\!\langle 1 \rangle\!\rangle_{val}^{\pi}(\Phi)(s) > \langle\!\langle 1 \rangle\!\rangle_{val}^{\pi'}(\Phi)(s)$; or

Algorithm 1 `ProfitableSwitch`

Input : A 2½-player game G with parity objective Φ for player 1
and a witness $wit_2 = (\pi, \overline{\pi}_Q)$ for player 2.
Output: A witness wit_2' for player 2 such that either $wit_2 = wit_2'$ or $wit_2 \prec wit_2'$.

1. (Step 1.) Compute $\langle\langle 1 \rangle\rangle_{val}^{\pi}(\Phi)(s)$ for all states s.
2. (Step 2.) Consider the set $I = \{ s \in S_2 \mid \exists t \in E(s).\ \langle\langle 1 \rangle\rangle_{val}^{\pi}(\Phi)(s) > \langle\langle 1 \rangle\rangle_{val}^{\pi}(\Phi)(t) \}$.
   2.1 (Value improvement.) **if** $I \neq \emptyset$, then set π' as follows:
   $\pi'(s) = \pi(s)$ for $s \in S_2 \setminus I$; and
   $\pi'(s) = t$ for $s \in I$, and $t \in E(s)$, such that $\langle\langle 1 \rangle\rangle_{val}^{\pi}(\Phi)(s) > \langle\langle 1 \rangle\rangle_{val}^{\pi}(\Phi)(t)$.
   and set $\overline{\pi}_Q'$ to be an arbitrary strategy such that $\pi' = \mathrm{Tr}_{\mathrm{as}}(\overline{\pi}_Q')$.
   2.2 (Qualitative improvement.) **else** for every value class $\mathrm{VC}^{\pi}(r)$,
   let $\overline{G}_r$ be the 2-player game $(\mathrm{Tr}_{\mathrm{as}}(\mathrm{Tr}_{\mathrm{win}_1}(G \upharpoonright \mathrm{VC}^{\pi}(r))))$
   set $(\overline{\pi}_Q' \upharpoonright \mathrm{VC}^{\pi}(r)) =$ `SwitchTwoPlParity`$(\overline{G}_r, (\overline{\pi}_Q \upharpoonright \mathrm{VC}^{\pi}(r)))$ and $\pi' = \mathrm{Tr}_{\mathrm{as}}(\overline{\pi}_Q')$,
   (where `SwitchTwoPlParity` is a strategy improvement step for 2-player parity games).
3. **return** $wit_2' = (\pi', \overline{\pi}_Q')$.

---

2. for all states s, we have $\langle\langle 1 \rangle\rangle_{val}^{\pi}(\Phi)(s) = \langle\langle 1 \rangle\rangle_{val}^{\pi'}(\Phi)(s)$, and in every value class $\mathrm{VC}^{\pi}(r) = \mathrm{VC}^{\pi'}(r)$, we have $(\overline{\pi}_Q \upharpoonright \mathrm{VC}^{\pi}(r)) \prec_Q (\overline{\pi}_Q' \upharpoonright \mathrm{VC}^{\pi}(r))$ in the 2-player parity game $\mathrm{Tr}_{\mathrm{as}}(\mathrm{Tr}_{\mathrm{win}_1}(G \upharpoonright \mathrm{VC}^{\pi}(r)))$, where $\prec_Q$ denotes the ordering of strategies for a strategy improvement algorithm for 2-player parity games (e.g., as defined in [17, 1]).

Profitable switch. Given a witness $wit_2 = (\pi, \overline{\pi}_Q)$ for player 2, we describe a procedure `ProfitableSwitch` to "improve" the witness according to the witness ordering $\prec$. The procedure is described in Algorithm 1. An informal description of the procedure is as follows: given a witness $wit_2 = (\pi, \overline{\pi}_Q)$, the algorithm computes the values $\langle\langle 1 \rangle\rangle_{val}^{\pi}(\Phi)(s)$ for all states. If there is a state $s \in S_2$, such that the strategy can be "value improved", i.e., there is a state $t \in E(s)$, with $\langle\langle 1 \rangle\rangle_{val}^{\pi}(\Phi)(t) < \langle\langle 1 \rangle\rangle_{val}^{\pi}(\Phi)(s)$, then the witness is modified setting $\pi(s)$ to t. This is achieved in Step 2.1 of `ProfitableSwitch`. Else in every value class $\mathrm{VC}^{\pi}(r)$, the strategy $\overline{\pi}_Q$ is "improved" for the game $(\mathrm{Tr}_{\mathrm{as}}(\mathrm{Tr}_{\mathrm{win}_1}(G \upharpoonright \mathrm{VC}^{\pi}(r))))$ w.r.t. the ordering $\prec_Q$ of strategies for 2-player parity games. This is achieved in Step 2.2 of `ProfitableSwitch`.

Lemma 3 *Consider a witness $wit_2 = (\pi, \overline{\pi}_Q)$ to be an input to Algorithm 1, and let $wit_2' = (\pi', \overline{\pi}_Q')$ be an output, i.e., $wit_2' =$*

$\texttt{ProfitableSwitch}(G, wit_2)$. *If the set I in Step 2 of Algorithm 1 is non-empty, then we have*

$$\langle\langle 1\rangle\rangle_{val}^{\pi}(\Phi)(s) \geq \langle\langle 1\rangle\rangle_{val}^{\pi'}(\Phi)(s) \ \ \forall s \in S; \qquad \langle\langle 1\rangle\rangle_{val}^{\pi}(\Phi)(s) > \langle\langle 1\rangle\rangle_{val}^{\pi'}(\Phi)(s) \ \ \forall s \in I.$$

The key argument to prove Lemma 3 is as follows. Let $wit_2 = (\pi, \overline{\pi}_Q)$ be an input to Algorithm 1 and $wit_2' = (\pi', \overline{\pi}_Q')$ be the output. Observe that given strategy π, for every state $s \in \mathrm{VC}^{\pi}(r) \cap S_1$, if $t \in E(s)$, then we have $\langle\langle 1\rangle\rangle_{val}^{\pi}(\Phi)(t) \leq r$, i.e., $t \in \bigcup_{0 \leq q \leq r} \mathrm{VC}^{\pi}(q)$. Hence player 1 can only choose edges with the target of the edge in equal or lower value classes. Using this fact, it can be shown that if player 2 switches to the strategy π', as constructed when Step 2.1 of Algorithm 1 is executed, then for all strategies σ for player 1 the following assertion hold: if there is a closed recurrent class $C \subseteq (S \setminus \mathrm{VC}^{\pi}(1))$ in the Markov chain $G_{\sigma,\pi'}$, then C is winning for player 2, i.e., $\min(p(C))$ is odd. It follows that given strategy π', a counter optimal strategy for player 1 maximizes the probability to reach $\mathrm{VC}^{\pi}(1)$. From arguments similar to $2^1\!/_2$-player games with reachability objectives [6], with $\mathrm{VC}^{\pi}(1)$ as the target for player 1, and the value improvement step (Step 2.1 of Algorithm 1) Lemma 3 follows.

Lemma 4 *Consider a witness $wit_2 = (\pi, \overline{\pi}_Q)$ to be an input to Algorithm 1, and let $wit_2' = (\pi', \overline{\pi}_Q')$ be an output, i.e., $wit_2' =$ $\texttt{ProfitableSwitch}(G, wit_2)$, such that $wit_2 \neq wit_2'$. If the set I in Step 2 of Algorithm 1 is empty, then we have*

1. *For all states s, $\langle\langle 1\rangle\rangle_{val}^{\pi}(\Phi)(s) \geq \langle\langle 1\rangle\rangle_{val}^{\pi'}(\Phi)(s)$.*
2. *If for all states $\langle\langle 1\rangle\rangle_{val}^{\pi}(\Phi)(s) = \langle\langle 1\rangle\rangle_{val}^{\pi'}(\Phi)(s)$, then for all value class $\mathrm{VC}^{\pi}(r)$, $(\overline{\pi}_Q \upharpoonright \mathrm{VC}^{\pi}(r)) \prec_Q (\overline{\pi}_Q' \upharpoonright \mathrm{VC}^{\pi}(r))$.*

A proof sketch for Lemma 4 is as follows: an argument similar to the argument for Lemma 3 shows that for a strategy π' constructed in Step 2.2 of Algorithm 1 the following assertion hold: for all strategies σ for player 1, if there is a closed recurrent class $C \subseteq (S \setminus \mathrm{VC}^{\pi}(1))$ in the Markov chain $G_{\sigma,\pi'}$, then C is winning for player 2, i.e., $\min(p(C))$ is odd. Since in strategy π' player 2 chooses every edge in the same value class as π, it follows that for all states s we have $\langle\langle 1\rangle\rangle_{val}^{\pi}(\Phi)(s) \geq \langle\langle 1\rangle\rangle_{val}^{\pi'}(\Phi)(s)$. If for all states s we have $\langle\langle 1\rangle\rangle_{val}^{\pi}(\Phi)(s) = \langle\langle 1\rangle\rangle_{val}^{\pi'}(\Phi)(s)$, then by properties of Procedure `SwitchTwoPlParity`, the condition 2 of Lemma 4 follows. This proves Lemma 4. Lemma 3 and Lemma 4 yields the following result.

Lemma 5 *For a witness $wit_2 = (\pi, \overline{\pi}_Q)$, we have if $wit_2 \neq$ $\texttt{ProfitableSwitch}(G, wit_2)$, then $wit_2 \prec \texttt{ProfitableSwitch}(G, wit_2)$.*

Algorithm 2 StrategyImprovementAlgorithm

Input : A $2^1/_2$-player game G with parity objective Φ for player 1.
Output: A witness $wit_2^* = (\pi^*, \overline{\pi}_Q^*)$ for player 2.
1. Pick an arbitrary witness $wit_2 = (\pi, \overline{\pi}_Q)$ for player 2.
2. **while** $wit_2 \neq \texttt{ProfitableSwitch}(G, wit_2)$
   do $wit_2 = \texttt{ProfitableSwitch}(G, wit_2)$.
3. **return** $wit_2^* = wit_2$.

The key argument to establish that if a witness wit_2 satisfy that $wit_2 = \texttt{ProfitableSwitch}(G, wit_2)$, then wit_2 is an optimal witness is as follows: let wit_2 be a witness such that $wit_2 = \texttt{ProfitableSwitch}(G, wit_2)$, and let $wit_1 = (\sigma, \overline{\sigma}_Q)$ be the counter optimal witness for player 1 against wit_2. Consider a value class $\mathrm{VC}^\pi(r)$, for $r > 0$, and the game $\overline{G}_r = \mathrm{Tr}_{\mathrm{as}}(\mathrm{Tr}_{\mathrm{win}_1}(G \upharpoonright \mathrm{VC}^\pi(r)))$. Since $\overline{\pi}_Q$ cannot be improve against $\overline{\sigma}_Q$ w.r.t. the ordering $\prec_Q$, it follows that $\overline{\sigma}_Q$ is a sure winning strategy in $\overline{G}_r$. Hence it follows from Lemma 1 that σ is an almost-sure winning strategy for player 1 in $\mathrm{Tr}_{\mathrm{win}_1}(G \upharpoonright \mathrm{VC}^\pi(r))$, since $\sigma = \mathrm{Tr}_{\mathrm{as}}(\overline{\sigma}_Q)$. Consider any strategy π' for player 2, against σ, and consider the Markov chain $G_{\sigma,\pi'}$. Since σ is almost-sure winning in $\mathrm{Tr}_{\mathrm{win}_1}(G \upharpoonright \mathrm{VC}^\pi(r))$, for all $r > 0$, it follows that for any closed recurrent class C of $G_{\sigma,\pi'}$, such that $C \subseteq \bigcup_{r>0} \mathrm{VC}^\pi(r)$, we have C is winning for player 1 (i.e., the minimum priority of C is even). Moreover, since the strategy π cannot be "value improved" it follows from arguments similar to [6] for $2^1/_2$-player reachability games that for all strategies π', for all states $s \in \mathrm{VC}^\pi(r)$, we have $\mathrm{Pr}_s^{\sigma,\pi'}(\Phi) \geq r$. Hence we have $\langle\langle 1 \rangle\rangle_{val}(\Phi)(s) \geq r$. Since σ is an optimal strategy against π, for all states $s \in \mathrm{VC}^\pi(r)$, we have $r = \langle\langle 1 \rangle\rangle_{val}^\pi(\Phi)(s) \geq \langle\langle 1 \rangle\rangle_{val}(\Phi)(s)$. This establishes optimality of π, and yields the following lemma.

Lemma 6 *For a witness $wit_2 = (\pi, \overline{\pi}_Q)$, we have if $wit_2 =$ $\texttt{ProfitableSwitch}(G, wit_2)$, then wit_2 is an optimal witness for player 2.*

A strategy improvement algorithm using the `ProfitableSwitch` procedure is described in Algorithm 2. Observe that it follows from Lemma 5 that if Algorithm 2 outputs a witness $wit_2^* = (\pi^*, \overline{\pi}_Q^*)$, then $wit_2^* = \texttt{ProfitableSwitch}(G, wit_2^*)$. The correctness of the algorithm follows from Lemma 6 and yields Theorem 2. An illustration of the working of the algorithm is presented in Example 1.

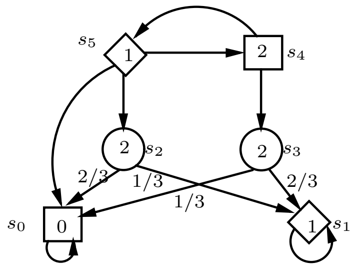


Figure 2: A 2½-player parity game.

Example 1 (Strategy improvement algorithm) *Consider the game shown in Fig. 2 where the set of states is $\{ s_0, s_1, s_2, s_3, s_4, s_5 \}$. The $\Box$-states are player 1 states, the $\Diamond$-states are player 2 states, and $\bigcirc$-states are the probabilistic states. The priorities of the states and the transition probabilities are indicated in Fig. 2. Consider the initial strategy π_0 for player 2 that chooses $s_5 \to s_0$ at state s_5. Given the strategy π_0, the counter optimal strategy σ_0 for player 1 is to choose $s_4 \to s_5$ at state s_4. Given the strategies σ_0 and π_0 the value vector $\vec{v}$ is $(1, 0, \frac{2}{3}, \frac{1}{3}, 1, 1)$, where $\vec{v}_i$ denotes the value for player 1 at state s_i. At this stage by "value improvement" step of procedure* `ProfitableSwitch`*, the strategy of player 2 switches to the strategy π_1 that chooses $s_5 \to s_2$ at state s_5. Given the strategy π_1, the counter optimal strategy σ_1 for player 1 is still to choose $s_4 \to s_5$ at state s_4. Given σ_1 and π_1, the value vector $\vec{v}$ is $(1, 0, \frac{2}{3}, \frac{1}{3}, \frac{2}{3}, \frac{2}{3})$. At this stage no value improvement is possible for player 2. Consider the value class $\mathrm{VC}^{\pi_1}(\frac{2}{3}) = \{s_2, s_4, s_5\}$, and assume the state s_2 to be an absorbing sure winning state for player 1. In the sub-game $\mathrm{Tr}_{\mathrm{win}_1}(\mathrm{VC}^{\pi}(\frac{2}{3}))$, player 2 switching to the strategy π_2 that chooses $s_5 \to s_4$ at state s_5, wins surely from s_5. Hence player 2 switches to the strategy π_2 by "qualitative improvement" step of* `ProfitableSwitch`*. Given the strategy π_2, the counter optimal strategy σ_2 for player 1 is to choose $s_4 \to s_3$ at state s_4. Given the strategies σ_2 and π_2, the value vector $\vec{v}$ is $(1, 0, \frac{2}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, and for all states s_i, $\vec{v}_i$ represents the value for player 1. The algorithm stops and the strategy π_2 is an optimal strategy for player 2. Also observe that if the game is slightly modified, by assigning priority 0 to state s_4 instead of 2, then after stage 1 of iteration, the sub-game $\mathrm{Tr}_{\mathrm{win}_1}(\mathrm{VC}^{\pi_1}(\frac{2}{3}))$ is surely winning for player 1. The algorithm would have stopped after iteration 1, by correctly discovering the value vector $\vec{v} = (1, 0, \frac{2}{3}, \frac{1}{3}, \frac{2}{3}, \frac{2}{3})$, as the values of the game.* ∎

Theorem 3 (Correctness of Algorithm 2) *Let $wit_2^* = (\pi^*, \overline{\pi}_Q^*)$ be an* *output of Algorithm 2. Then the strategy π^* is an optimal strategy for player 2.*

4 Randomized Sub-exponential Algorithm

In this section we combine the randomized sub-exponential time algorithm for 2-player parity games of Björklund et.al. [1] and the witness improvement procedure `ProfitableSwitch` to present a randomized sub-exponential time algorithm for $2^1\!/\!_2$-player games with parity objectives Parity(p). The algorithm works in sub-exponential time when the number of parities d of the function p satisfy that $d = O\big(\frac{n^{1-\epsilon}}{\log(n)}\big)$, for some $\epsilon > 0$. For all constants d, e.g., Büchi and coBüchi objectives, our algorithm works in comparable time with the best known algorithm for $2^1\!/\!_2$-player reachability games.

Games and improving sub-games. Let $\mathcal{G}(l,m)$ be the class of $2^1\!/\!_2$-player games with the set S_2 of player 2 states partitioned into two sets as follows:

- $O_1 = \{ s \in S_2 \mid |E(s)| = 1 \}$, i.e., the set of states with out-degree 1.
- $O_2 = S_2 \setminus O_1$, with $O_2 \leq l$, and $\sum_{s \in O_2} |E(s)| \leq m$.

There is no restriction for player 1. Given a game $G \in \mathcal{G}(l,m)$, a state $s \in O_2$, and an edge $e = (s,t)$, we consider the sub-game $\widetilde{G}_e$, by deleting all edges at s other than the edge e. Observe that $\widetilde{G}_e \in \mathcal{G}(l-1, m-|E(s)|)$, and hence also $\widetilde{G}_e \in \mathcal{G}(l,m)$. If $wit_2 = (\pi, \overline{\pi}_Q)$ is a witness for player 2 in $G \in \mathcal{G}(l,m)$, then a sub-game $\widetilde{G}$, is *wit_2-improving*, if some witness $wit_2' = (\pi', \overline{\pi}_Q')$ in $\widetilde{G}$, satisfies $wit_2 \prec wit_2'$. We now present an informal description of Algorithm 3.

Informal description of Algorithm 3. The algorithm takes a $2^1\!/\!_2$-player parity game and an initial witness wit_2^0, and proceeds in three steps as follows: in Step 1 it constructs r-pairs of wit_2^0-improving sub-games $\widetilde{G}$ and improving witness wit_2 in $\widetilde{G}$. This is achieved by procedure **ManyImprovingSubgames**. The parameter r depends on the algorithm and fixing r we would get different complexity analysis. In Step 2, the algorithm selects uniformly at random one of the improving sub-games $\widetilde{G}$ and the witness wit_2 and recursively computes an optimal witness wit_2^* in $\widetilde{G}$ with wit_2 as the initial witness. If the witness wit_2^*, is optimal in the original game G, then the algorithm terminates and returns wit_2^*. Else it improves wit_2^*, by a `ProfitableSwitch`, and continues by going to Step 1 with the improved witness `ProfitableSwitch`(G, wit_2^*) as the initial witness. The description of the procedure **ManyImprovingSubgames** is as follows: it constructs a sequence of games $(G^0, G^1, \ldots, G^{r-l})$ with $G^i \in \mathcal{G}(l, l+i)$ such that all the $(l+i)$-sub-games $\widetilde{G}^i_e$ of G^i are wit_2^0-improving. The procedure constructs G^{i+1} from G^i as follows: it computes an optimal witness wit_2^i in G^i, and if wit_2^i, is optimal in G, then we have discovered an optimal witness, otherwise construct G^{i+1} by adding a *target* edge e of $\texttt{ProfitableSwitch}(G, wit_2^i)$ in G^i.

Algorithm 3 Randomized Algorithm 2½-player Games

Input : A 2½-player parity game $G \in \mathcal{G}(l, m)$, and an initial witness wit_2^0 for player 2.
Output : An optimal witness $wit_2^* = (\pi^*, \overline{\pi}_Q^*)$ for player 2.

1. (Step 1.) Collect a set I of r pairs of $(\widetilde{G}, wit_2)$ of sub-games $\widetilde{G}$ of G, and witnesses wit_2 in $\widetilde{G}$, such that $wit_2^0 \prec wit_2$.
(This is achieved by Procedure **ManyImprovingSubgames**).

2. (Step 2.) Select a pair $(\widetilde{G}, wit_2)$ from I uniformly at random.
2.1 Find an optimal witness in $wit_2^* \in \widetilde{G}$ by applying the algorithm recursively, with wit_2 as the initial witness.

3. (Step 3.) **if** wit_2^* is an optimal witness in the original game G, then
return $wit_2^* = (\pi^*, \overline{\pi}_Q^*)$.
else let $wit_2 = \texttt{ProfitableSwitch}(G, wit_2^*)$, and
goto Step 1 with G and wit_2 as the initial witness.

Procedure **ManyImprovingSubgames**
1. Construct a sequence $(G^0, G^1, \ldots, G^{r-l})$ of sub-games with $G^i \in \mathcal{G}(l, l+i)$ as follows:
1.1 G^0 is the game where each edge is fixed according to wit_2^0.
1.2 Let wit_2^i be an optimal witness in G^i,
1.2.1 **if** wit_2^i is an optimal witness in the original game G,
terminate algorithm and return wit_2^i.
1.2.2 **else** let e be the target of $\texttt{ProfitableSwitch}(G, wit_2^i)$.
The sub-game G^{i+1} is the sub-game G^i with edge e added.

2. **return** r sub-games by fixing one of the r-edges in $G^{r-l} \in \mathcal{G}(l, r)$ and the corresponding witness.

Lemma 7 (Correctness and termination) *Algorithm 3 correctly computes an optimal witness wit_2^*.*

Proof. Observe that every time Step 1 of the algorithm is executed, the initial witness is improved w.r.t. the ordering $\prec$ of witness. Since the number of witnesses are bounded, the termination of the algorithm follows. Step 3 of Algorithm 3 and Step 1.2.1 of procedure **ManyImprovingSubgames** ensures that on termination of the algorithm, the witness returned is optimal. ∎

The following lemma bounds the expected number of iteration of Algorithm 3. The analysis is similar to the results of [1].

Lemma 8 (Expected iterations) *The expected number of iteration $T(\cdot,\cdot)$ of Algorithm 3 for a game $G \in \mathcal{G}(l,m)$ is bounded by the following recurrence*

$$T(l,m) \leq \sum_{i=l}^{r} T(l,i) + T(l-1,m-2) + \frac{1}{r}\sum_{i=1}^{r} T(l,m-i) + 1.$$

Proof. We justify every term of the right hand side of the recurrence. The first term represent the work by procedure **ManyImprovingSubgames** by recursive calls to Algorithm 3 to compute r pairs of wit_2^0-improving subgames and witnesses. The second term represents the work of the recursive call at Step 2 of Algorithm 3. The third term represents the work as the average of the r equally likely choices in Step 3 of Algorithm 3. All the subgames G^i can be partially ordered according to the values of the optimal witnesses in G^i. Since the algorithm only visits witnesses that are improving w.r.t. the $\prec$ ordering, it follows that sub-games that have equal, worse or incomparable optimal witness, to the witness wit_2^* will never be explored in the rest of the algorithm. In the worst case the algorithm selects the worst r sub-games and the Step 3 solves a game $G \in \mathcal{G}(l, m-i)$, for $i = 1, 2, \ldots, r$, each with probability $\frac{1}{r}$. This gives the bound for the recurrence. ∎

Using the analysis of Kalai for an algorithm for linear programming, Björklund et.al. in [1] proves that

$$m^{O\left(\sqrt{l/\log(l)}\right)} = 2^{O\left(\sqrt{l \log l}\right)}$$

is a solution to the recurrence of Lemma 8.

Lemma 9 *Given a $2^1\!/\!_2$-player parity game G, with a parity objective Parity(p), where $p : S \to [0..d]$, Algorithm 3 works in time*

$$2^{O\left(\sqrt{z \log(z)}\right)} \times \textit{ running time of } \texttt{ProfitableSwitch},$$

where $n_1 = |S_1|$, $n_2 = |S_2|$ and $n_0 = |S_\bigcirc|$, and $z = (n_0 \cdot d + n_2)$.

Proof. We first observe that the reduction of $2^1\!/\!_2$ player games to 2-player games by reduction $\mathrm{Tr}_{\mathrm{as}}(\cdot)$ causes a blow-up by a factor of d for states in $S_\bigcirc$. This fact, along with the bound of recurrence of Lemma 8, and plugging $l = d \cdot n_0 + n_2$ in the bound, yields that the expected number of iterations of Algorithm 3 is bounded by $2^{O\left(\sqrt{(d \cdot n_0 + n_2) \cdot \log(d \cdot n_0 + n_2)}\right)}$. Since each iteration of the algorithm requires to compute a `ProfitableSwitch`, the desired result follows. ∎

Lemma 10 *The procedure* `ProfitableSwitch` *can be computed in polynomial time.*

Proof. Computing a `ProfitableSwitch` is equivalent to solve a MDP with parity objectives quantitatively (Step 1 of `ProfitableSwitch`) and computing a switch of 2-player parity games (Step 2.2 of `ProfitableSwitch`). The quantitative solution of parity MDPs can be achieved in polynomial time [7, 4]. The result of [17, 1] describes procedure to compute in polynomial time a switch for 2-player parity games (i.e., a polynomial procedure for `SwitchTwoPlParity`). Hence the desired result follows. ∎

Using a symmetric version of Algorithm 3 for player 1 if $|S_1| \leq |S_2|$, and using Lemma 9 and Lemma 10 we obtain Theorem 4.

Theorem 4 *Given a $2^1\!/\!_2$-player parity game G, with a parity objective Parity(p), where $p : S \to [0..d]$, the value function $\langle\langle 1 \rangle\rangle_{val}(Parity(p))(s)$ can be computed for all s, in time*

$$2^{O\left(\sqrt{z \log(z)}\right)} \times O\big(poly(n)\big),$$

where $n_1 = |S_1|$, $n_2 = |S_2|$ and $n_0 = |S_\bigcirc|$, $z = (n_0 \cdot d + \min\{\, n_1, n_2 \,\})$, and poly represents a polynomial function.

References

[1] H. Bjorklund, S. Sandberg, and S. Vorobyov. A discrete subexponential algorithms for parity games. In *STACS'03*, pages 663–674. LNCS 2607, Springer, 2003.

[2] K. Chatterjee, L. de Alfaro, and T.A. Henzinger. The complexity of stochastic rabin and streett games. Technical Report UCB/CSD-3-1355, UC Berkeley, 2004. To appear ICALP 05.

[3] K. Chatterjee, M. Jurdziński, and T. A. Henzinger. Simple stochastic parity games. In *CSL'03*, volume 2803 of *LNCS*, pages 100–113. Springer, 2003.

[4] K. Chatterjee, M. Jurdziński, and T.A. Henzinger. Quantitative stochastic parity games. In *SODA'04*, pages 114–123. SIAM, 2004.

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

[6] A. Condon. On algorithms for simple stochastic games. In Jin-Yi Cai, editor, *Advances in Computational Complexity Theory*, volume 13 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 51–73. American Mathematical Society, 1993.

[7] L. de Alfaro. *Formal Verification of Probabilistic Systems*. PhD thesis, Stanford University, 1997.

[8] L. de Alfaro and T. A. Henzinger. Concurrent ω-regular games. In *LICS'00*, pages 141–154. IEEE Computer Society Press, 2000.

[9] E.A. Emerson and C. Jutla. The complexity of tree automata and logics of programs. In *FOCS'88*, pages 328–337. IEEE Computer Society Press, 1988.

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

[11] A. Hoffman and R. Karp. On nonterminating stochastic games. *Management Science*, 12:359–370, 1966.

[12] A. Kechris. *Classical Descriptive Set Theory*. Springer, 1995.

[13] W. Ludwig. A subexponential randomized algorithm for the simple stochastic game problem. *Information and Computation*, 117:151–155, 1995.

[14] A. K. McIver and C. C. Morgan. Games, probability, and the quantitative μ-calculus $qm\mu$. In *LPAR'02*, volume 2514 of *LNAI*, pages 292–310. Springer, 2002.

[15] T. E. S. Raghavan and J. A. Filar. Algorithms for stochastic games — a survey. *ZOR — Methods and Models of Operations Research*, 35:437–472, 1991.

[16] W. Thomas. Languages, automata, and logic. In *Handbook of Formal Languages*, volume 3, Beyond Words, chapter 7, pages 389–455. Springer, 1997.

[17] J. Vöge and M. Jurdziński. A discrete strategy improvement algorithm for solving parity games. In *CAV'00*, volume 1855 of *LNCS*, pages 202–215. Springer, 2000.

[18] W. Zielonka. Perfect-information stochastic parity games. In *FoSSaCS'04*, volume 2987 of *LNCS*, pages 499–513. Springer, 2004.