RUTCOR
RESEARCH
REPORT

A Pumping Algorithm for Ergodic Stochastic Mean Payoff Games with perfect information

Endre Boros[a] Khaled Elbassioni[b]
Vladimir Gurvich[c] Kazuhisa Makino[d]

RRR 19-2009, September 2009

RUTCOR
Rutgers Center for
Operations Research
Rutgers University
640 Bartholomew Road
Piscataway, New Jersey
08854-8003
Telephone: 732-445-3804
Telefax: 732-445-5472
Email: rrr@rutcor.rutgers.edu
http://rutcor.rutgers.edu/∼rrr

[a]RUTCOR, Rutgers University, 640 Bartholomew Road, Piscataway NJ 08854-8003; (boros@rutcor.rutgers.edu)

[b]Max-Planck-Institute for Informatics; Stuhlsatzenhausweg 85, 66123, Saarbruecken, Germany (elbassio@mpi-sb.mpg.de).

[c]RUTCOR, Rutgers University, 640 Bartholomew Road, Piscataway NJ 08854-8003; (gurvich@rutcor.rutgers.edu).
The third author also gratefully acknowledges the partial support of Research Foundation and Center for Algorithmic Game Theory at University of Aarhus and Laboratory of Combinatorics and Optimization of Dep. of Mathematics at University of Pierre and Marie Curie, Paris VI.

[d]Graduate School of Information Science and Technology, University of Tokyo, Tokyo, 113-8656, Japan; (makino@mist.i.u-tokyo.ac.jp).

Rutcor Research Report
RRR 19-2009, September 2009

A Pumping Algorithm for Ergodic Stochastic Mean Payoff Games with perfect information

Abstract. We consider two-person zero-sum stochastic mean payoff games with perfect information, or BWR-games, given by a digraph $G = (V = V_B \cup V_W \cup V_R, E)$, with local rewards $r : E \to \mathbb{R}$, and three types of vertices: black V_B, white V_W, and random V_R. The game is played by two players, White and Black.
When the play is at a white (black) vertex v, White (Black) selects an outgoing arc (v, u). When the play is at a random vertex v, a vertex u is picked with a given probability $p(v, u)$. In all cases, Black pays White the value $r(v, u)$.
The play continues forever, and White aims to maximize (Black aims to minimize) the limiting mean (that is, average) payoff. We prove that this class of games is polynomially equivalent with the classical Gillette games and it includes many well-known subclasses, such as cyclic games, simple stochastic games, stochastic parity games, and Markov decision processes.
It was shown in [BGEM09] that every BWR-game can be reduced by a potential transformation to a canonical form in which the value and optimal strategies of both players are obvious, for every initial position, since a locally optimal move in it is optimal in the whole game. In particular, the optimal strategies are uniformly optimal (or ergodic), while the values might depend on the initial position.
Like in the Gillette paper, the proof in [BGEM09] goes through the discounted version and then taking the discount factor to the limit 1; hence it does not result directly in an efficient algorithm. In this paper, we give such an algorithm in the *ergodic case*, namely, when the optimal values do not depend on the initial position. We show that this algorithm is pseudo-polynomial when the number of random nodes is a constant. and give an almost matching lower bound on its running time. The corresponding example shows that this lower bound holds in a wide class of the considered "pumping" algorithms.
An important application of the BWR-games was recently found. It is known that so-called "parlor" games with perfect information, like Chess and Back Gammon, can be solved in pure strategies. Yet, it was an open question, whether these pure strategies can be chosen **positional** (that is, the move in a position v depends only on v, not on previous positions and moves) and **uniformly optimal** (that is, independent on the initial position). In [BG09] this question was answered in the positive by a reduction of Chess and Back Gammon to a BW- and a BWR-game, respectively.

Keywords: Back Gammon, Chess, parlor games, local reward, Gillette model, perfect information, potential, stochastic game

---

Acknowledgements: This research was supported by DIMACS, Center for Discrete Mathematics and Theoretical Computer Science

1 Introduction

1.1 Sketch of the results

We consider two-person zero-sum stochastic games with perfect information and mean payoff. Let $G = (V, E)$ a digraph (digraph) whose vertex-set V is partitioned in three subsets $V = V_B \cup V_W \cup V_R$ that correspond to black, white, and random positions, controlled respectively, by two players, *Black* - the *minimizer* and *White* - the *maximizer*, and by nature. Furthermore, let us fix a *local reward* function $r : E \to \mathbb{R}$, Vertices $v \in V$ and arcs $e \in E$ are called *positions* and *moves*, respectively. In a personal position $v \in V_W$ (respectively, $v \in V_B$) White (Black) selects an outgoing arc (v, u), while in a random position $v \in V_R$ a move (v, u) is picked with a given probability $p(v, u)$, in accodance with a given probabilistic distribution. In all cases Black pays White the reward $r(v, u)$.

We shall assume that graph G contains no dead-ends, or in other words, at least one move (v, u) exists in every position $v \in V$. This assumption can be made without any loss of generality, since we can always add a loop $v, v)$ to each dead-end v and set $r(, v, v) = u(v)$, where $u(v)$ was the terminal payoff at v.

Finally, let us fix an initial position $v_0 \in V$. The interpretation is standard. The play starts in v_0 and continues forever; White's objective is to maximize the *limiting mean payoff*

$$c = \lim\inf_{n\to\infty} \frac{\sum_{i=0}^{n} b_i}{n+1}, \tag{1}$$

where b_i is the reward incurred at step i of the play, while the objective of Black is opposite, that is, to minimize $\lim\sup_{n\to\infty} \frac{\sum_{i=0}^{n} b_i}{n+1}$.

We shall prove that a *saddle point* exists in *pure positional uniformly optimal (ergodic)* strategies. Here "pure" means that the choice of a move (v, u) in a personal position $v \in V_B \cup V_R$ is deterministic, not random; "positional" means that this choice depends solely on v, not on previous positions or moves; finally, "ergoduic" means that it does not depend on the initial position v_0, either.

Our results and methods are similar to those of Gillette [Gil57]; see also Liggett and Lippman [LL69]. First, we analyze the so-called *discounted* version, in which the payoff is discounted by a factor β^i at step i, giving the effective payoff: $a_\beta = (1-\beta)\sum_{i=0}^{\infty} \beta^i b_i$, and then proceeding to the limit as the *discount factor* $\beta \in [0, 1)$ tends to 1.

The class of the *BWR-games* was introduced in [GKK88]; see also [CH08].

It was recently shown in [BGEM09]) that the BWR-games and classical Gillette games [Gil57] are polynomially equivalent; see also [GM08, AM09].

The special case when there are no random positions, $V_R = \emptyset$, is known as *cyclic, or mean payoff, or BW-games.* They were introduced for the complete bipartite digraphs in [Mou76b, Mou76a], for all (not necessarily complete) bipartite digraphs in [EM79], and for arbitrary digraphs in [GKK88].

Another special case of this was considered extensively in the literature under the name of *parity games* [BV01a, BV01b, CJH04, Hal07, Jur98, JPZ06], and later generalized also to include random nodes in [CH08]. The game is reduced to the *minimum mean cycle problem* in case $V_W = V_R = \emptyset$, see for example [Kar78]. On the other hand, if one of the sets V_B or V_W is empty, we obtain a *Markov decision process*; see, for example, [MO70], and if both are empty $V_B = V_W = \emptyset$, we get a *weighted Markov chain*.

In the special case of a BWR-game, when all rewards are zero except at a single node t called the terminal, at which there is a self-loop with reward 1, we obtain the so-called *simple stochastic games* (SSG), introduced by Condon [Con92]. In these games, the objective of White is to maximize the probability of reaching the terminal, while Black wants to minimize this probability. SSG's have also been considered in several papers [Con92, Con93, GH08, Hal07]. Recently, it was shown that Gillette games, and hence BWR-games, are equivalent to SSG's under polynomial-time reductions [AM09]. Thus, by recent results of Björklund, Vorobyov [BV05], and Halman [Hal07], all these games can be solved in randomized strongly subexponential time $2^{O(\sqrt{n_d \log n_d})}$, where $n_d = |V_B| + |V_W|$ is the number of *deterministic* vertices; see [BGEM09] for more details. Let us note that a number of pseudo-polynomial and subexponential algorithms already exists for the BW-games [GKK88, KL93, Pis99, BV05, BV07, HBV04, Hal07, ZP96]; see also [DG06], and for parity games [JPZ06]. These games are of interest to complexity theory; Karzanov and Lebedev [KL93] (see also [ZP96]) proved that decision problem "whether the value of a BW-game is positive" is in the intersection of NP and coNP, yet, no polynomial alorithm is known; see recent survey by Vorobyiov [Vor08].

Another challenging open question is whether a pseudo-polynomial algorithm exists for SSG, or more generally, for BWR-games. Besides their many applications, these classes also lie in NP$\cap$co-NP, and yet their complexity is still open.

Given a BWR-game, we consider potential transformations $x : V \to \mathbb{R}$, assigning a real-value $x(v)$ to each vertex $v \in V$, and transforming the local reward on each arc (v, u) into $r_x(v, u) = r(v, u) + x(v) - x(u)$. It is known that for cyclic games, there exists such a transformation such that, in the transformed game, the locally optimal strategies are globally optimal, and hence, the value and optimal strategies become obvious [GKK88]. This result was extended for the more general class of BWR-games in [BGEM09]: in the transformed game, the equilibrium value $\mu(v) = \mu_x(v)$ is given simply by the maximum local reward for $v \in V_W$, the minimum local reward for $v \in V_B$, and the average local reward for $v \in V_R$. In this case we say that the transformed game is in *canonical* form.

The special case when $V_R = \emptyset$ was shown in [GKK88]. However, it is not clear how the algorithm given in [GKK88] can be generalized to the case with random nodes. On the other hand, the proof in [BGEM09] follows by considering the discounted case and then taking the discount factor β to the limit. While such approach is enough to prove the existence of canonical form, it does not give immediately an algorithm.

In this paper, we give such an algorithm that does not go through the discounted case, but under the restriction that the given game is *ergodic*, i.e. the optimal values do not depend on the starting position. We show that this algorithm is pseudo-polynomial in the case when the number of random nodes is constant, and leave open the question whether such algorithm exists in the general non-ergodic case.

Theorem 1 *Consider a BWR-game with k random nodes, a total of n vertices, and integer rewards in the range $[-R, R]$, and assume that all probabilities are rational of maximum common denominator W. Then there is an algorithm that runs in time $O(n^{O(k)}W^{O(k^2)}R\log(nRW))$ and either brings any such game $\mathcal{G}$ by a potential transformation to canonical form, or proves that $\mathcal{G}$ is non-ergodic.*

Let us remark that the ergodic case is frequent in applications. For instance, it is the case when $G = (V_W \cup V_B \cup V_R, E)$ is a complete tripartite digraph (where $p(v, u) > 0$ for all $v \in V_R$ and $(v, u) \in E$); see Section 3 for better sufficient conditions.

1.2 Why Chess and Back Gammon can be solved in *positional* pure uniformly optimal strategies

An important application of the BWR-games was recently found in [BG09]. Let us consider two person zero-sum games with perfect information, so-called "parlor" games, like Chess or Back Gammon. Such a game is naturally modeled by a digraph G whose terminal vertices V_T are the outcomes of the game Yet, G might have directed cycles, since positions can be repeated. Hence, infinite plays are possible. Let us assume that *all infinite plays form one outcome* c, the AIPFOOT property. The set of outcomes $A = V_T \cup \{c\}$ is ranked arbitrarily (but oppositely) by the two players. For example, Chess and Back Gammon are AIPFOOT games, since any infinite play is a draw, by definition. It is well-known [Zer12] that positional games with perfect information can be solved in pure strategies. However, it was an open question, whether these pure strategies can be chosen **positional** (that is, the move in a position v depends only on v, not on previous positions and moves) and **uniformly optimal** (that is, independent on the in initial position). In [BG09] this question was answered in positive. Somewhat surprisingly, it appears convenient to reduce an AIPFOOT game $\mathcal{G}$ (for example, Back Gammon or even Chess) to a BWR-game. The reduction is very simple.

Let $u(a)$ denote the payoff (of player 1, White) in case of an outcome $a \in A = V_T(\mathcal{G}) \cup \{c\}$. For Chess and Back Gammon, $u(c) = 0$, since c is a draw. In general, let us subtract constant $u(c)$ from $u(a)$ for all $a \in A$. Obviously, the obtained zero-sum game $\mathcal{G}'$ is equivalent with $\mathcal{G}$ and $u'(c) = 0$. Thus, without any loss of generality, we may assume that each infinite play is a draw, like in Chess or Back Gammon. Then, let us set the local rewards to 0 for all moves of $\mathcal{G}'$; furthermore, let us add a loop to each terminal a in $\mathcal{G}'$ and assign the local reward

$u'(a)$ to this loop. It is clear that game G' and the obtained BWR-game $\mathcal{G}''$ are equivalent. See [BG09] for accurate definitions and more details.

Remark 1 *Since the BWR and Gillette models are equivalent, it might be shorter to derive our main result (ii) directly from the Gillette theorem on stochastic games with perfect information. Yet, Chess and Back Gammon ideally fit the BW and BWR models, respectively.*

1.3 Overview of the techniques

Our algorithm for proving Theorem 1 is quite simple. Starting form zero potentials, and depending on the current locally optimal rewards (maximum for White, minimum for Black, and average for Random), the algorithm keeps selecting a subset of nodes and reducing their potentials by the same value, until either the locally optimal rewards at different nodes become sufficiently close to each other, or a proof of non-ergodicity is obtained in the form of a certain partition of the nodes. The upper bound on the running time consists of three technical parts. The first one is to show that if the number of iterations becomes too large, then there is a large enough potential gap to ensure an ergodic partition. In the second part, we show that the range of potentials can be kept sufficiently small throughout the algorithm, namely $\|x^*\|_\infty \leq nRk(2W)^k$, and hence the range of the transformed rewards does not explode. The third part concerns the required accuracy. In Appendix C, we show that it is enough in our algorithm to get the value of the game within an accuracy of

$$\varepsilon = \frac{1}{n^{2(k+1)}k^{2k}(2W)^{4k+2k^2+2}}, \tag{2}$$

in order to guarantee that it is equal to the exact value.

One more desirable property of this algorithm is that it is of the *certifying* type (see e.g. [KMMS03]), in the sense that, given an optimal pair of strategies, the vector of potentials provided by the algorithm can be used to verify optimality in *linear* time (otherwise verifying optimality requires solving a system of equations).

One should also contrast the bound in Theorem 1 with the subexponential bounds in [Hal07]: roughly, the algorithm of Theorem 1 will be more efficient if $|V_R| < (|V_W| + |V_B|)^{\frac{1}{4}}$ (assuming that W and R are polynomials in n). However, our algorithm could be practically much faster since it can stop much earlier than its estimated worst-case running time (unlike the subexponential algorithms [Hal07], or those based on dynamic programming [ZP96], see Example 2 in Appendix E). We are not aware of any previous results bounding the running time of a class of BWR-games in terms of the number of random nodes, except for [GH08] which shows that simple stochastic games on k random nodes can be solved in time $O(k!(|V||E| + L))$, where L is the maximum bit length of a transition probability. It is worth remarking here that BWR-games are polynomially equivalent with simple stochastic games, but under the reduction from BWR-games to simple stochastic games, the number of random nodes K becomes a polynomial in n, even if original BWR-games have constantly many random nodes.

For the special case of Markov decision processes (when V_B or V_W is empty), the potentials mentioned in the theorem correspond to the *dual* variables in the standard linear programming formulation; see e.g. [MO70][1].

We also show the lower bound $W^{\Omega(k)}$ on the running time of the algorithm mentioned in Theorem 1 by providing an instance of the problem, with only random nodes.

The paper is organized as follows. In the next section, we formally define BWR-games, canonical forms, and state some useful propositions. In Section 3, we give a sufficient condition for the ergodicity of a BWR-game, which will be used as one possible stopping criterion in our algorithm. We give the algorithm in Section 4.1, and prove it converges in Section 4.2. In Section 5, we show that this convergence proof can, in fact, be turned into a quantitative statement giving the precise bounds stated in Theorem 1. The last section gives a lower bound example for the algorithm. Some of the proofs will be given in Appendix B. Two illustrative example are given in Appendix E. The first one illustrates the existence of a saddle point in pure strategies, while the second one shows, among other things, that local optimality does not imply global optimality, if the game is not in canonical form.

2 Preliminaries

2.1 BWR-games in positional form

Let $G = (V = V_W \cup V_B \cup V_R, E)$ be a digraph (digraph) that may have loops and multiple arcs. We assume without any loss of generality that G has no terminal vertices, i.e., vertices of out-degree 0, (otherwise, one can add a loop to each terminal vertex.) There are two players: White, the maximizer, and Black, the minimizer. The vertices of G are called *positions*. They are partitioned into three sets $V = V_B \cup V_W \cup V_R$ and called respectively *Black, White*, and *Random* positions.

An arc $(v, u) \in E$ is called a *move* from position v to u . This move is chosen by a player, White if $v \in V_W$ and Black if $v \in V_B$, or by chance if $v \in V_R$. In the latter case a probability $p(v, u)$ is assigned to each arc $(v, u) \in E$, i.e., $0 \le p(v, u) \le 1$ for all $v \in V_R, u \in V$ and $\sum_{u \mid (v,u)\in E} p(v, u) = 1 \quad \forall v \in V_R$. For convenience we will assume that $p(v, u) > 0$ whenever $(v, u) \in E$ and $v \in V_R$, and set $p(v, u) = 0$ for $(v, u) \notin E$. Let us denote by P the obtained set of probability distributions for all $v \in V_R$. Furthermore, to each arc $(v, u) \in E$ is assigned a real number $r(v, u)$ called the *local reward* (cost or payoff) on (v, u). It is assumed that Black pays and White gets $r(v, u)$ whenever the move (v, u) is made in the game.

[1]In fact, one can use Theorem 1 to derive the dual LP-formulation for Markov decision processes.

Finally, let us fix an initial position $v_0 \in V$. The game *in positional form* is then defined by the quadruple $\mathcal{G} = (G, P, p_0, r)$.

2.2 Effective limiting payoffs in weighted Markov chains

When $V_B = V_W = \emptyset$, and hence $V = V_R$ consists only of random nodes, we obtain a *weighted Markov chain*. In this case $P : V \times V \to [0, 1]$ is the *transition matrix* whose entry $p(v, u)$ is the probability of transition from v to u in one move, for every pair of positions $v, u \in V$. Then, it is obvious and well-known that for every integral $i \geq 0$ matrix $P^i : V \times V \to [0, 1]$ (the i-th power of P) is the i-move transition matrix, whose entry $p_i(v, u)$ is the probability of transition from v to u in exactly i moves, for every $v, u \in V$.

Let $q_i(v, u)$ be the probability that arc $(v, u) \in E$ will be the $(i + 1)$-st move, given the original distribution $p_0 = e_{v_0}$, where $i = 0, 1, 2, \ldots$ and e_{v_0} is the n-dimensional unit vector with 1 in position v_0, and denote by $q_i : E \to [0, 1]$ the corresponding probabilistic $|E|$-vector. For convenience, we introduce $|V| \times |E|$ vertex-arc transition matrix $Q : V \times E \to [0, 1]$ whose entry $q(\ell, (v, u))$ is equal to $p(v, u)$ if $\ell = v$ and 0 otherwise, for every $\ell \in V$ and $(v, u) \in E$. Then, it is clear that $q_i = p_0 P^i Q$.

Let r be the $|E|$-dimensional vector of local rewards, and b_i denote the expected reward for the $(i + 1)$-st move; $i = 0, 1, 2, \ldots$, i.e., $b_i = \sum_{(v,u) \in E} q_i(v, u) r(v, u) = p_0 P^i Q r$. Then the *effective payoff* of the weighted Markov chain is defined to be the average expected reward on the limit, i.e., $c = \lim_{n \to \infty} \frac{1}{n+1} \sum_{i=0}^{n} b_i$. It is well-known (see e.g. [MO70]) that this is equal to $c = p_0 P^* Q r$, where P^* is the *limit Markov matrix* (see Appendix A for more details).

2.3 BWR-games in normal form

Standardly, we define a strategy $s_W \in S_w$ (respectively, $s_B \in S_B$) as a mapping that assigns a move $(v, u) \in E$ to each position $v \in V_W$ (respectively, $v \in V_B$). A pair of strategies $s = (s_W, s_B)$ is called a *situation*. Given a BWR-game $\mathcal{G} = (G, P, v_0, r)$ and situation $s = (s_B, s_W)$, let us define a weighted Markov chain $\mathcal{G}_s = (G, P_s, v_0, r)$. To do so, we just extend probability distributions P_s as follows:

$$\begin{aligned} P_s(v, u) &= 1 \text{ whenever } v \in V_W \text{ and } u = s_W(v) \text{ or } v \in V_B \text{ and } u = s_B(v); \\ P_s(v, u) &= 0 \text{ whenever } v \in V_W \text{ and } u \neq s_W(v) \text{ or } v \in V_B \text{ and } u \neq s_B(v). \end{aligned} \tag{3}$$

In other words, in every position $v \in V_B \cup V_W$ we assign probability 1 to the (unique) move prescribed by the strategy s_B or s_W and probability 0 to every other move. For random position $v \in V_R$ we assign $P_s(v, u) = p(v, u) > 0$ if $(v, u) \in E$ and $P_s(v, u) = p(v, u) = 0$ otherwise, as before.

In the obtained weighted Markov chain $\mathcal{G}_s = (G, P_s, v_0, r)$, we define the limiting (mean) effective payoff c_s as above. Thus, we obtain a matrix game $C : S_W \times S_B \to \mathbb{R}$.

2.4 Solvability and ergodicity

It is known that every such game has a saddle point in pure strategies [Gil57, LL69].

Moreover, there are optimal strategies that do not depend on the starting position v_0, so-called *ergodic* optimal strategies. In contrast, the value of the game can depend on v_0.

The triplet $\mathcal{G} = (G, P, r)$ is called a *not initialized BWR-game*. Furthermore, $\mathcal{G}$ is called *ergodic* if the value $\mu(v_0)$ of each corresponding BWR-game (G, P, v_0, r) is the same for all initial positions $v_0 \in V$.

2.5 Potential transforms

Given a BWR-game $\mathcal{G} = (G, P, v_0, r)$, let us introduce a mapping $x : V \to \mathbb{R}$, whose values $x(v)$ will be called *potentials*, and define the transformed reward function $r_x : E \to \mathbb{R}$ as:

$$r_x(v, u) = r(v, u) + x(v) - x(u), \quad \text{where} \quad (v, u) \in E. \tag{4}$$

Potential transforms were first introduced in 1958 by Gallai [Gal58], then applied to stochastic games in 1966 by Hoffman and Karp [HK66] and to B-games in 1978 by Karp [Kar78].

It is not difficult to verify that the two normal form matrices C_x and C, of the obtained game $\mathcal{G}_x$ and the original game $\mathcal{G}$, are equal (see [BGEM09]). In particular, their optimal (pure positional) strategies coincide, and the values also coincide: $\mu_x(v_0) = \mu(v_0)$.

2.6 Ergodic canonical form

Given a BWR-game $\mathcal{G} = (G, P, r)$, let us define a mapping $m : V \to \mathbb{R}$ as follows:

$$m(v) = \begin{cases} \max(r(v,u) \mid u \,:\, (v,u) \in E) & \text{for } v \in V_W, \\ \min(r(v,u) \mid u \,:\, (v,u) \in E) & \text{for } v \in V_W, \\ mean(r(v,u) \mid u \,:\, (v,u) \in E) = \sum_{u \mid (v,u) \in E} r(v,u)\, p(v,u), & \text{for } v \in V_R. \end{cases} \tag{5}$$

A move $(v, u) \in E$ in a position $v \in V_W$ (respectively, $v \in V_B$) is called *locally optimal* if it realizes the maximum (respectively, minimum) in (5). A strategy s_W of White (respectively, s_B of Black) is called *locally optimal* if it chooses a locally optimal move $(v, u) \in E$ in every position $v \in V_W$ (respectively, $v \in V_B$).

Definition 1 *We say that a BWR-game $\mathcal{G}$ is in* ergodic canonical form *if function (5) is constant: $m(v) \equiv m$ for some number m.*

Canonical forms were defined for BW-games in [GKK88], and extended to BWR-games in [BGEM09].

Proposition 1 *If a BWR-game is in ergodic canonical form then*
(i) every locally optimal strategy is optimal and
(ii) the game is ergodic: m is its value for every initial position $v_0 \in V$.

3 Sufficient conditions for ergodicity of BWR-games

A digraph $G = (V_W \cup V_B \cup V_R, E)$ (whose vertices are partitioned in three subsets, $V = V_W \cup V_B \cup V_R$) is called *ergodic* if every corresponding not initialized BWR-game $\mathcal{G} = (G, P, r)$ is ergodic, that is, the values of the games $\mathcal{G} = (G, P, v_0, r)$ do not depend on v_0.

We will give a simple characterization of ergodic digraphs, which, obviously, provides sufficient conditions for ergodicity of the BWR-games. For BW-games (that is, in case $R = \emptyset$) such characterization was given in [GL89]; see Appendix 2 of [BGMW07] for more details. Here, we generalize this result to BWR-games.

In addition to partition $\Pi_p : V = V_W \cup V_B \cup V_R$, let us consider one more partition $\Pi_r : V = V^W \cup V_B \cup V^R$ with the following properties:

(i) Sets V^W and V^B are not empty (while V^R might be empty).

(ii) There is no arc $(v, u) \in E$ such that $[v \in (V_W \cup V_R) \cap V^B$ and $u \notin V^B]$ or, vice versa, $[v \in (V_B \cup V_R) \cap V^W$ and $u \notin V^W]$. In other words, White cannot leave V^B, Black cannot leave V^W, and there are no random moves from $V^W \cup V^B$.

(iii) For each $v \in V_W \cap V^W$ (respectively, $v \in V_B \cap V^B$) there is a move $(v, u) \in E$ such that $u \in V^W$ (respectively, $u \in V^B$). In other words, White (Black) cannot be forced to leave V^W (respectively, V^B). In particular, this implies that the induced subgraphs $G[V^W]$ and $G[V^B]$ have no dead-ends.

Partition $\Pi_r : V = V^W \cup V^B \cup V^R$ satisfying (i), (ii), and (iii) will be called a *contra-ergodic* partition for digraph $G = (V_W \cup V_B \cup V_R, E)$.

Theorem 2 *Digraph G is ergodic if and only if it has no contra-ergodic partitions.*

The "only if part" can be strengthen as follows:

Proposition 2 *Given a BWR-game $\mathcal{G}$ whose graph has a contra-ergodic partition, if $m(v) > m(u)$ for every $v \in V^W, u \in V^B$ then $\mu(v) > \mu(u)$ for every $v \in V^W, u \in V^B$.*

Definition 2 *A* contra-ergodic decomposition *of $\mathcal{G}$ is a contra-ergodic partition $\Pi_r : V = V^W \cup V^B \cup V^R$ such that $m(v) > m(u)$ for every $v \in V^W$ and $u \in V^B$.*

By Proposition 2, if $\mathcal{G}$ has a contra-ergodic decomposition, then $\mathcal{G}$ is not ergodic.

The above Theorem and Proposition are easily reduced to the BW-games [GL89]. It is sufficient to set $V_R = V^R = \emptyset$. Already in this case, verification of conditions (i), (ii), and (iii) is NP-hard [GL89]; see also Appendix 2 of [BGMW07] for the complete proof.

Yet, in several cases it is easy to show that these conditions cannot hold; see Corollaries 3-6 of [BGMW07] for the BW-games; they can be easily extended to the BWR-games.

For example, no contra-ergodic partition can exist if $G = (V_W \cup V_B \cup V_R, E)$ is a tripartite digraph such that the underlying non-directed tripartite graph is complete; see, for instance, the example in Appendix E, where each part consists of only one vertex.

4 Pumping algorithm for the ergodic BWR-games

4.1 Description of the algorithm

Given a BWR-game $\mathcal{G} = (G, P, r)$, let us compute $m(v)$ for all $v \in V$ using (5). Throughout, we will denote by $[m] \stackrel{\text{def}}{=} [m^-, m^+]$ and $[r] \stackrel{\text{def}}{=} [r^-, r^+]$ the range of $m(\cdot)$ and $r(\cdot, \cdot)$, respectively.

Given a subset $I \subseteq [m]$, let $V(I) = \{v \in V \mid m(v) \in I\} \subseteq V$. In the following algorithm, set I will always be a closed or semi-closed interval within $[m]$.

Let $m^- = t_0 < t_1 < t_2 < t_3 < t_4 = m^+$ be given thresholds. We will successively apply potential transforms $x : V \to \mathbb{R}$ such that no vertex ever leaves the interval $[t_0, t_4]$, nor the interval $[t_1, t_3]$; in other words, $V_x[t_0, t_4] = V[t_0, t_4]$ and $V_x[t_1, t_3] \supseteq V[t_1, t_3]$ for all considered transforms x, where $V_x(I) = \{v \in V \mid m_x(v) \in I\}$.

Let us initialize potentials $x(v) = 0$ for all $v \in V$. We will fix

$$t_0 := m_x^-,\ t_1 := m_x^- + \frac{1}{4}|[m_x]|,\ t_2 := m_x^- + \frac{1}{2}|[m_x]|,\ t_3 := m_x^- + \frac{3}{4}|[m_x]|,\ t_4 := m_x^+. \qquad (6)$$

Then, let us reduce all potentials of $V_x[t_2, t_4]$ by a maximum constant δ such that no vertex leaves the closed interval $[t_1, t_3]$; in other words, we stop the iteration whenever a vertex from this interval reaches its border. After this we compute potentials $x(v)$, new values $m_x(v)$, for $v \in V$, and start the next iteration.

It is clear that δ can be computed in linear time: it is the maximum value δ such that $m_x^\delta(v) \geq t_1$ for all $v \in V_x[t_2, t_4]$ and $m_x^\delta(v) \leq t_3$ for all $v \in V_x[t_0, t_2)$, where $m_x^\delta(v)$ is given by the following formula:

$$\begin{array}{lll}
\max\left\{\max\limits_{(v,u)\in E, u\in V_x[t_2,t_4]}\{r_x(v,u)\}, \max\limits_{(v,u)\in E, u\in V_x[t_0,t_2)}\{r_x(v,u)\}-\delta\right\} & \text{for} & v\in V_W\cap V_x[t_2,t_4],\\
\min\left\{\min\limits_{(v,u)\in E, u\in V_x[t_2,t_4]}\{r_x(v,u)\}, \min\limits_{(v,u)\in E, u\in V_x[t_0,t_2)}\{r_x(v,u)\}-\delta\right\} & \text{for} & v\in V_B\cap V_x[t_2,t_4],\\
\sum\limits_{(v,u)\in E, u\in V} p(v,u)r_x(v,u)-\delta\sum\limits_{(v,u)\in E, u\in V_x[t_0,t_2)} p(v,u) & \text{for} & v\in V_R\cap V_x[t_2,t_4],\\
\max\left\{\max\limits_{(v,u)\in E, u\in V_x[t_2,t_4]}\{r_x(v,u)\}+\delta, \max\limits_{(v,u)\in E, u\in V_x[t_0,t_2)}\{r_x(v,u)\}\right\} & \text{for} & v\in V_W\cap V_x[t_0,t_2),\\
\min\left\{\min\limits_{(v,u)\in E, u\in V_x[t_2,t_4]}\{r_x(v,u)\}+\delta, \min\limits_{(v,u)\in E, u\in V_x[t_0,t_2)}\{r_x(v,u)\}\right\} & \text{for} & v\in V_B\cap V_x[t_0,t_2),\\
\sum\limits_{(v,u)\in E, u\in V} p(v,u)r_x(v,u)+\delta\sum\limits_{(v,u)\in E, u\in V_x[t_2,t_4]} p(v,u) & \text{for} & v\in V_R\cap V_x[t_0,t_2).
\end{array} \tag{7}$$

It is also clear from (7) (and important) that $\delta \geq |[m_x]|/4$. Indeed, vertices from $[t_2, t_4]$ can only go down, while vertices from $[t_0, t_2)$ can only go up. Each of them must traverse a distance of at least $|[m_x]|/4$ before it can reach the border of the interval $[t_1, t_3]$. Moreover, if after some iteration one of the sets $V_x[t_0, t_1)$ or $V_x(t_3, t_4]$ becomes empty then range of m_x is reduced at least by 25%.

Procedure PUMP$(\mathcal{G}, \varepsilon)$ below tries to reduce any BWR-game $\mathcal{G}$ by a potential transformation x into one in which $|[m_x]| \leq \varepsilon$. Two subroutines are used in the procedure. REDUCE-POTENTIALS$(\mathcal{G}, x)$ replaces the current potential x with another potential with a sufficiently small norm (cf. Lemma 4 below). This reduction is needed since, without it, the potentials and hence the transformed local rewards can grow exponentially. The second routine FIND-PARTITION$(\mathcal{G}, x)$ uses the current potential vector x to construct a contra-ergodic decomposition of $\mathcal{G}$ (cf. line 17).

We will prove in Lemma 3 that if the number of pumping iterations performed is large enough, namely,

$$N = \frac{8n^2 R_x}{|[m_x]|\theta^{k-1}} + 1, \tag{8}$$

where $R_x = r_x^+ - r_x^-$, $\theta = \min\{p(v,u) : (v,u) \in E\}$, and k is the number of random nodes, and yet the range of m_x is not reduced, then we will be able to find a contra-ergodic decomposition.

In Section 4.2, we will first argue that the algorithm terminates in finite time if $\varepsilon = 0$ and the considered BWR-game is ergodic. In the following section, this will be turned into a quantitative argument with the precise bound on the running time. Yet, in Section 6, we will show that this time can be exponential already for R-games.

Algorithm 1 PUMP($\mathcal{G}, \varepsilon$)

Require: A BWR-game $\mathcal{G} = (G = (V, E), P, r)$ and a desired accuracy ε
Ensure: a potential $x : V \to \mathbb{R}$ s.t. $|m_x(v) - m_x(u)| \leq \varepsilon$ for all $u, v \in V$ if the game is ergodic, and a contra-ergodic decomposition otherwise

```
let x^0(v) := x(v) := 0 for all v ∈ V; i := 1
let t_0, t_1, ..., t_4, and N be as defined by (6) and (8)
while i ≤ N do
   if |[m_x]| ≤ ε then
      return x
   end if
   δ := max{δ' | m_x^{δ'}(v) ≥ t_1 for all v ∈ V_{x^0}[t_2, t_4] and m_x^{δ'}(v) ≤ t_3 for all v ∈ V_{x^0}[t_0, t_2)}
   if δ = ∞ then
      return the ergodic partition V_{x^0}[t_0, t_2) ∪ V_{x^0}[t_2, t_4]
   end if
   x(v) := x(v) − δ for all v ∈ V_{x^0}[t_2, t_4]
   if V_{x^0}[t_0, t_1) = ∅ or V_{x^0}(t_3, t_4] = ∅ then
      x := x^0 :=REDUCE-POTENTIALS(G, x); i := 1
      recompute the thresholds t_0, t_1, ..., t_4 and N using (6) and (8)
   end if
end while
V^W ∪ V^B ∪ V^R :=FIND-PARTITION(G, x)
return contra-ergodic partition V^W ∪ V^B ∪ V^R
```

4.2 Proof of finiteness for the ergodic case

Let us assume without loss of generality that range of m is $[0,1]$, and the initial potential $x^0 = 0$. Suppose that during N iterations no new vertex enters the interval $[1/4, 3/4]$. Then, $-x(v) \geq N/4$ for each $v \in V(3/4, 1]$, since these vertices "were pumped" N times, and $x(v) \equiv 0$ for each $v \in V[0, 1/4)$, since these vertices "were not pumped" at all. We will show that if N is sufficiently large then the considered game is not ergodic.

Consider infinitely many iterations $i = 0, 1, \ldots$, and denote by $V^B \subseteq V$ (respectively, by $V^W \subseteq V$) the set of vertices that were pumped just finitely many times (respectively, always but finitely many times); in other words, $m_{x^0}(v) \in [1/2, 1]$ if $v \in V^W$ (respectively, $m_{x^0}(v) \in [0, 1/2)$ if $v \in V^B$) for all but finitely many i's. It is not difficult to verify that the partition $\Pi_r : V = V^W \cup V^B \cup V^R$, where $V^R = V \setminus (V^W \cup V^B)$, is contra-ergodic. It is also clear that after sufficiently many iteration $m_{x^0}(v) > 1/2$ for all $v \in V^W$, while $m_{x^0}(v) \leq 1/2$ for all $v \in V^B$. Thus, by Proposition 2, the considered game $\mathcal{G}$ is not ergodic, or in other words, our algorithm is finite for the ergodic BWR-games.

But how many times a vertex can oscillate around $1/2$ until it finally settles itself down in $[1/4, 1/2)$ or in $(1/2, 3/4]$? We shall give an upper bound below.

5 Running time analysis

Consider the execution of the algorithm on a given BWR-game. We define a *phase* to be a set of iterations during which the range of m_x, defined with respect to the current potential x, is not reduced by a constant factor of what it was at the beginning of the phase, i.e., none of the sets $V_x[t_0, t_1)$ or $V_x(t_3, t_4]$ becomes empty (cf. line 12 of the procedure). Note that the number of iterations in each phase is at most N defined by (8). Lemma 3 states that if N iterations are performed in a phase then, the game is not ergodic. Lemma 5 bounds the total number of phases and estimates the overall running time.

5.1 Finding a contra-ergodic decomposition

We assume throughout this section that we are inside phase h of the algorithm, which started with a potential x^h, and that $|[m_x]| > \frac{3}{4}|[m_{x^h}]|$ in all N iterations of the phase, and hence we proceed to step 17. For convenience, we will write $(\cdot)_{x^h}$ as $(\cdot)_h$, where $(\cdot)$ could be m, r, r^+, etc, (e.g., $m_h^- = m_{x^h}^-$, $m_h^+ = m_{x^h}^+$). We assume that the phase starts with local reward function $r = r_h$ and hence [2] $x^h = 0$.

Given a potential vector x, we use the following notation:

$$\mathrm{EXT}_x = \{(v,u) \in E \;:\; v \in V_B \cup V_W \text{ and } r_x(v,u) = m_x(v)\}, \quad \Delta_x = \min\{x(v) : v \in V\}.$$

[2] in particular, note that $r_x(v,u)$ and $m_x(v)$ are used, for short, to actually mean $r_{x+x^h}(v,u)$ and $m_{x+x^h}(v)$, respectively

Lemma 1 *Consider any two nodes $u, v \in E$ and let x be the current potential. Then*

$$x(u) \geq \begin{cases} x(v) - (m_h^+ - r_h^-) & \text{if either } v \in V_W,\ (v,u) \in E \text{ or } v \in V_B,\ (v,u) \in \mathrm{EXT}_x \\ \frac{x(v)-(m_h^+ - r_h^-)}{\theta} & \text{if } v \in V_R \text{ and } (v,u) \in E, \end{cases}$$

and

$$x(u) \leq \begin{cases} x(v) + r_h^+ - m_h^- & \text{if either } v \in V_B,\ (v,u) \in E \text{ or } v \in V_W,\ (v,u) \in \mathrm{EXT}_x \\ \frac{x(v)+r_h^+ - m_h^- -(1-\theta)\Delta_x}{\theta} & \text{if } v \in V_R \text{ and } (v,u) \in E. \end{cases}$$

Let $t_l < 0$ be the largest value satisfying the following conditions:

(i) there are no arcs $(v, u) \in E$ with $v \in V_W \cup V_R$, $x(v) \geq t_l$ and $x(u) < t_l$;

(i) there are no arcs $(v, u) \in \mathrm{EXT}_x$ with $v \in V_B$, $x(v) \geq t_l$ and $x(u) < t_l$.

Let $X = \{v \in V \ : \ x(v) \geq t_l\}$. In other words, X is the set of nodes with potential as close to 0 as possible, such that no white or random node in X has an arc crossing to $V \setminus X$, and no black node has an extremal arc crossing to $V \setminus X$. Similarly, define $t_u > \Delta_x$ to be the smallest value satisfying the following conditions:

(i) there are no arcs $(v, u) \in E$ with $v \in V_B \cup V_R$, $x(v) \leq t_u$ and $x(u) > t_u$;

(i) there are no arcs $(v, u) \in \mathrm{EXT}_x$ with $v \in V_W$, $x(v) \leq t_u$ and $x(u) > t_u$,

and let $Y = \{v \in V \ : \ x(v) \leq t_u\}$. Note that the sets X and Y can be computed in $O(|E| \log |V|)$ time.

Lemma 2 *It holds that* $\max\{-t_l, t_u - \Delta_x\} \leq nR_h \left(\frac{1}{\theta}\right)^{k-1}$.

The correctness of the algorithm follows from the following lemma.

Lemma 3 *Suppose that pumping is performed for $N_h \geq 2nT_h + 1$ iterations, where $T = \frac{4nR_h}{|[m_h]|\theta^{k-1}}$, and neither the set $V_h[t_0, t_1)$ nor $V_h(t_3, t_4]$ becomes empty. Let $V^B = X$ and $V^W = Y$ be the sets constructed as above, and $V^R = V \setminus (X \cup Y))$. Then $V^W \cup V^B \cup V^R$ is a contra-ergodic decomposition.*

5.2 Potential reduction

One problem that arises during the pumping procedure is that the potentials can increase exponentially in the number of phases, making our bounds on the number of iteration per phase also exponential in n. For the BW-case Pisaruk [Pis99] solved this problem by giving a procedure that reduces the range of the potentials after each round, while keeping all its desired properties needed for the running time analysis. (For convenience, we give a variant of this procedure in the appendix.)

Pisaruk's potential reduction procedure can be thought of as a combinatorial procedure for finding an *extreme point* of a polyhedron, given a point in it. Indeed, given a BWR-game and a potential x, let us assume without loss of generality, by shifting the potentials if necessary, that $x \geq 0$, and let $E' = \{(v,u) \in E \ : \ r_x(v,u) \in [m_x^-, m_x^+], \ v \in V_B \cup V_W\}$, where r is the *original* local reward function. Then the following polyhedron is non-empty:

$$P_x = \left\{ x' \in \mathbb{R}^V \;\middle|\; \begin{array}{cl} m_x^- \leq r(v,u) + x'(v) - x'(u) \leq m_x^+, & \forall (v,u) \in E' \\ r(v,u) + x'(v) - x'(u) \leq m_x^+, & \forall v \in V_W, \ (v,u) \in E \setminus E' \\ m_x^- \leq r(v,u) + x'(v) - x'(u), & \forall v \in V_B, \ (v,u) \in E \setminus E' \\ m_x^- \leq \sum_{u \in V} p(v,u)(r(v,u) + x'(v) - x'(u)) \leq m_x^+, & \forall v \in V_R \\ x(v) \geq 0 & \forall \ \ v \in V \end{array} \right\}.$$

Moreover, P_x is pointed, and hence must have an extreme point.

Lemma 4 *Consider a BWR-game in which all rewards are integral with range $R = r^+ - r^-$, and probabilities $p(v,u)$ are rational with common denominator at most W, and let $k = |V_R|$. Then any extreme point x^* of P_x satisfies $\|x^*\|_\infty \leq nRk(2W)^k$.*

Note that any point $x' \in P_x$ satisfies $[m_{x'}] \subseteq [m_x]$, and hence replacing x by x^* does not increase the range of m_x.

5.3 Proof of Theorem 1

Consider a BWR-game $\mathcal{G} = (G = (V,E), P, r)$ with $|V| = n$ vertices and k random nodes. Assume r to be integral in the range $[-R, R]$ and all transition probabilities are rational with common denominator W.

Lemma 5 *Procedure PUMP$(\mathcal{G}, \varepsilon)$ terminates in $O(nk(2W)^k(\frac{1}{\varepsilon} + n^2|E|)R\log(\frac{R}{\varepsilon}))$ time.*

Theorem 1 follows by setting ε sufficiently small:

Corollary 1 *When procedure PUMP$(\mathcal{G}, \varepsilon)$ is run with ε as in (2), it either output a potential vector x such $m_x(v)$ is constant for all $v \in V$, or it finds a contra-ergodic partition. The total running time is $O(nk^3(2W)^k(n^{2(k+1)}(2W)^{4k+2k^2+2} + n^2|E|)R\log(RWn))$.*

6 Lower bound example

We show now that the execution time of the algorithm, in the worst case, can be exponential in the number of random nodes k, already for weighted Markov chains, that is, for R-games. Consider the following example.

Let $G = (V, E)$ be a digraph on $k = 2l+1$ vertices $u_l, \ldots, u_1, u_0 = v_0, v_1, \ldots, v_l$, and with the following set of arcs:

$$E = \{(u_l, u_l), (v_l, v_l)\} \cup \{(u_{i-1}, u_i), (u_i, u_{i-1}), (v_{i-1}, v_i), (v_i, v_{i-1}) : \ i = 1, \ldots, l\}.$$

Let $W \geq 1$ be an integer. All nodes are random with the following transition probabilities: $p(u_l, u_l) = p(v_l, v_l) = 1 - \frac{1}{W+1}$, $p(u_0, u_1) = p(u_0, v_1) = \frac{1}{2}$, $p(u_{i-1}, u_i) = p(v_{i-1}, v_i) = 1 - \frac{1}{W+1}$, for $i = 2, \ldots, l$, and $p(u_i, u_{i-1}) = p(v_i, v_{i-1}) = \frac{1}{W+1}$, for $i = 1, \ldots, l$. The local rewards are zero on every arc, except for $r(u_l, u_l) = -r(v_l, v_l) = 1$. Clearly this Markov chain consists of a single recurrent class, and it is easy to verify that the limiting distribution p^* is as follows:

$$\begin{aligned} p^*(u_0) &= \frac{W-1}{(W+1)W^l - 2} \\ p^*(u_i) &= p^*(v_i) = \frac{W^{i-1}(W^2-1)}{2((W+1)W^l - 2)}, \text{ for } i = 1, \ldots, l. \end{aligned}$$

The optimal expected reward at each vertex is

$$\mu(u_i) = \mu(v_i) = -1 \cdot (1 - \frac{1}{W+1})p^*(u_l) + 1 \cdot (1 - \frac{1}{W+1})p^*(u_l) = 0,$$

for $i = 0, \ldots, k$. Upto a shift, there is a unique set of potentials that transform the Markov chain into canonical form, and they satisfy the following system of equations:

$$\begin{aligned} 0 &= -\frac{1}{2}\Delta_1 - \frac{1}{2}\Delta_1', \\ 0 &= -(1 - \frac{1}{W+1})\Delta_{i+1} + \frac{1}{W+1}\Delta_i, \text{ for } i = 1, \ldots, k-1 \\ 0 &= -(1 - \frac{1}{W+1})\Delta_{i+1}' + \frac{1}{W+1}\Delta_i', \text{ for } i = 1, \ldots, k-1 \\ 0 &= -(1 - \frac{1}{W+1}) + \frac{1}{W+1}\Delta_l, \\ 0 &= 1 - \frac{1}{W+1} + \frac{1}{W+1}\Delta_l', \end{aligned}$$

where $\Delta_i = u_i - u_{i-1}$ and $\Delta_i' = v_i - v_{i-1}$; solving we get $\Delta_i = -\Delta_i' = W^{k-i+1}$, for $i = 1, \ldots, l$.

Lower bound on pumping algorithms. Any pumping algorithm that starts with 0 potentials and modifies the potentials in each iteration by at most γ will not have a number

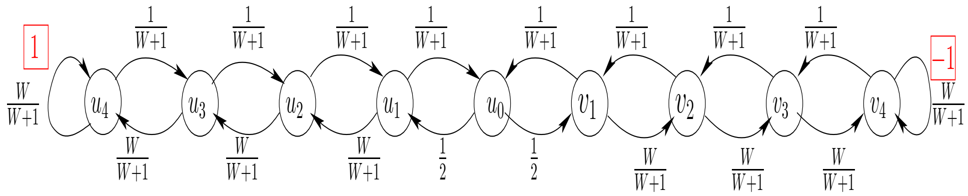

Figure 1: An exponential Example.

of iterations less than $\frac{W^{l-1}}{2\gamma}$ on the above example. By (7), the algorithm in Section 4 has $\gamma \leq 1/\min\{p(v,u): \ (v,u) \in E, \ p(v,u) \neq 0\}$, which is $\Omega(W)$ in our example. We conclude that the running time of the algorithm is $O(W^{l-2}) = \Omega(W^{\Omega(k)})$ on this example.

References

[AM09] D. Andersson and P.B. Miltersen. The complexity of solving stochastic games on graphs. In *Lecture Notes in Computer Science, ISAAC, to appear*, 2009.

[BG09] E. Boros and V. Gurvich. Why chess and back gammon can be solved in pure positional uniformly optimal strategies? RRR-21-2009, RUTCOR, Rutgers University, 2009.

[BGEM09] E. Boros, V. Gurvich, K. Elbassioni, and K. Makino. Every stochastic game with perfect information admits a canonical form. RRR-09-2009, RUTCOR, Rutgers University, 2009.

[BGMW07] E. Boros, V. Gurvich, K. Makino, and Shao Wei. Nash-solvable bidirected cyclic 2-person game forms. RRR-30-2007, revised in rrr-20-2009, RUTCOR, Rutgers University, 2007.

[BV01a] E. Beffara and S. Vorobyov. Adapting gurvich-karzanov-khachiyan's algorithm for parity games: Implementation and experimentation. Technical Report 2001-020, Department of Information Technology, Uppsala University, available at: https://www.it.uu.se/research/reports/#2001, 2001.

[BV01b] E. Beffara and S. Vorobyov. Is randomized gurvich-karzanov-khachiyan's algorithm for parity games polynomial? Technical Report 2001-025, Department of Information Technology, Uppsala University, available at: https://www.it.uu.se/research/reports/#2001, 2001.

[BV05] H. Björklund and S. Vorobyov. Combinatorial structure and randomized subexponential algorithms for infinite games. *Theoretical Computer Science*, 349(3):347–360, 2005.

[BV07] H. Björklund and S. Vorobyov. A combinatorial strongly sub-exponential strategy improvement algorithm for mean payoff games. *Discrete Applied Mathematics*, 155(2):210–229, 2007.

[CH08] K. Chatterjee and T. A. Henzinger. Reduction of stochastic parity to stochastic mean-payoff games. *Inf. Process. Lett.*, 106(1):1–7, 2008.

[CJH04] K. Chatterjee, M. Jurdziński, and T. A. Henzinger. Quantitative stochastic parity games. In *SODA '04: Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 121–130, Philadelphia, PA, USA, 2004. Society for Industrial and Applied Mathematics.

[Con92] A. Condon. The complexity of stochastic games. *Information and Computation*, 96:203–224, 1992.

[Con93] A. Condon. An algorithm for simple stochastic games. In *Advances in computational complexity theory, volume 13 of DIMACS series in discrete mathematics and theoretical computer science*, 1993.

[DG06] V. Dhingra and S. Gaubert. How to solve large scale deterministic games with mean payoff by policy iteration. In *valuetools '06: Proceedings of the 1st international conference on Performance evaluation methodolgies and tools*, page 12, New York, NY, USA, 2006. ACM.

[EM79] A. Eherenfeucht and J. Mycielski. Positional strategies for mean payoff games. *International Journal of Game Theory*, 8:109–113, 1979.

[Gal58] T. Gallai. Maximum-minimum Sätze über Graphen. *Acta Mathematica Academiae Scientiarum Hungaricae*, 9:395–434, 1958.

[GH08] H. Gimbert and F. Horn. Simple stochastic games with few random vertices are easy to solve. In *FoSSaCS*, pages 5–19, 2008.

[Gil57] D. Gillette. Stochastic games with zero stop probabilities. In A.W. Tucker M. Dresher and P. Wolfe, editors, *Contribution to the Theory of Games III, in Annals of Mathematics Studies,)*, volume 39, pages 179–187. Princeton University Press, 1957.

[GKK88] V. Gurvich, A. Karzanov, and L. Khachiyan. Cyclic games and an algorithm to find minimax cycle means in directed graphs. *USSR Computational Mathematics and Mathematical Physics*, 28:85–91, 1988.

[GL89] V. A. Gurvich and V. N. Lebedev. A criterion and verification of ergodicity of cyclic game forms. *Uspehi Mat. Nauk (in Russian)*, 265:193–194, English transl. in Russian Mathematical Surveys 44(1) (1989), 243–244, 1989.

[GM08] V. Gurvich and P. B. Miltersen. On the computational complexity of solving stochastic mean-payoff games. *CoRR*, abs/0812.0486, 2008.

[Hal07] N. Halman. Simple stochastic games, parity games, mean payoff games and discounted payoff games are all lp-type problems. *Algorithmica*, 49(1):37–50, 2007.

[HBV04] S. Sandberg H. Björklund and S. Vorobyov. A combinatorial strongly subexponential strategy improvement algorithm for mean payoff games. DIMACS Technical Report 2004-05, DIMACS, Rutgers University, 2004.

[HK66] A. J. Hoffman and R. M. Karp. On nonterminating stochastic games. *Management Science, Series A*, 12(5):359–370, 1966.

[JPZ06] M. Jurdziński, M. Paterson, and U. Zwick. A deterministic subexponential algorithm for solving parity games. In *SODA '06: Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 117–123, New York, NY, USA, 2006. ACM.

[Jur98] M. Jurdziński. Deciding the winner in parity games is in up ∩ co-up. *Inf. Process. Lett.*, 68(3):119–124, 1998.

[Kar78] R. M. Karp. A characterization of the minimum cycle mean in a digraph. *Discrete Math.*, 23:309–311, 1978.

[KL93] A. V. Karzanov and V. N. Lebedev. Cyclical games with prohibition. *Mathematical Programming*, 60:277–293, 1993.

[KMMS03] D. Kratsch, R. M. McConnell, K. Mehlhorn, and J. P. Spinrad. Certifying algorithms for recognizing interval graphs and permutation graphs. In *SODA '03: Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 158–167, Philadelphia, PA, USA, 2003. Society for Industrial and Applied Mathematics.

[KS63] J. G. Kemeny and J. L. Snell. *Finite Markov chains*. Springer-Verlag, New York, 1963.

[LL69] T. M. Liggett and S. A. Lippman. Stochastic games with perfect information and time-average payoff. *SIAM Review*, 4:604–607, 1969.

[MO70] H. Mine and S. Osaki. *Markovian decision process*. American Elsevier Publishing Co., New York, 1970.

[Mou76a] H. Moulin. Extension of two person zero sum games. *Journal of Mathematical Analysis and Application*, 5(2):490–507, 1976.

[Mou76b] H. Moulin. Prolongement des jeux à deux joueurs de somme nulle. *Bull. Soc. Math. France, Memoire*, 45, 1976.

[Pis99] N. N. Pisaruk. Mean cost cyclical games. *Mathematics of Operations Research*, 24(4):817–828, 1999.

[Vor08] S. Vorobyov. Cyclic games and linear programming. *Discrete Applied Mathematics*, 156(11):2195–2231, 2008.

[Zer12] E. Zermelo. über eine anwendung der mengenlehre auf die theorie des schachspiels. In *Proc. 5th Int. Cong. Math. Cambridge, vol. II; 501–504*, 1912.

[ZP96] U. Zwick and M. Paterson. The complexity of mean payoff games on graphs. *Theoretical Computer Science*, 158(1-2):343–359, 1996.

Appendix A: Limiting distribution of Markov chains

Let $(G = (V, E), P)$ be a Markov chain, and let $C_1, \ldots, C_k \subseteq V$ be the vertex sets of the strongly connected components (classes) of G. For $i \neq j$, let us (standardly) write $C_i \prec C_j$, if and only if there is an arc $(v, u) \in E$ such that $v \in C_i$ and $u \in C_j$. The components C_i, such that there is no C_j with $C_i \prec C_j$ are called the *absorbing* (or *recurrent*) classes, while the other components are called *transient* or *non-recurrent*. Let $J = \{i : \ C_i \text{ is absorbing}\}$, $A = \cup_{i\in J} C_i$, and $T = V \setminus A$. For $X, Y \subseteq V$, a matrix $H \subseteq \mathbb{R}^{V\times V}$, a vector $h \subseteq \mathbb{R}^V$, we denote by $H[X; Y]$ the submatrix of P induced by X as rows and Y as columns, and by $h[X]$ the subvector of h induced by X. Let $I = I[V; V]$ be the $|V| \times |V|$ identity matrix, $e = e[V]$ be the vector of all ones of dimension $|V|$. For simplicity, we drop the indices of $I[\cdot, \cdot]$ and $e[\cdot]$, when they are understood from the context. Then $P[C_i; C_j] = 0$ if $C_j \prec C_i$, and hence in particular, $P[C_i; C_i]e = e$ for all $i \in J$, while $P[T, T]e$ has at least one component of value strictly less than 1.

The following are well-known facts about P^i and the limiting distribution $p_w = e_w P^*$, when the initial distribution is the wth unit vector e_w of dimension $|V|$ (see, e.g., [KS63]):

(L1) $p_w[A] > 0$ and $p_w[T] = 0$;

(L2) $\lim_{i\to\infty} P^i[V; T] = 0$;

(L3) $\text{rank}(I - P[C_i; C_i]) = |C_i| - 1$ for all $i \in J$, $\text{rank}(I - P[T; T]) = |T|$, and $(I - P[T; T])^{-1} = \sum_{i=0}^{\infty} P[T; T]^i$;

(L4) the absorption probabilities $y_i \in [0, 1]^V$ into a class C_i, $i \in J$, are given by the unique solution of the linear system: $(I - P[T; T])y_i[T] = P[T; C_i]e$, $y_i[C_i] = e$ and $y_i[C_j] = 0$ for $j \neq i$;

(L5) the limiting distribution $p_w \in [0, 1]^V$ is given by the unique solution of the linear system: $p_w[C_i](I - P[C_i; C_i]) = 0$, $p_w[C_i]e = y_i(w)$, for all $i \in J$, and $p_w[T] = 0$.

Appendix B: Proofs

Proof of Proposition 1. Indeed, if White (Black) applies a locally optimal strategy then after every own move (s)he will get (pay) m, while for each move of the opponent the local reward will be at least (at most) m, and finally, for each random position the expected local reward is m. Thus, every locally optimal strategy of a player is optimal. Furthermore, if both players choose their optimal strategies then the expected local reward b_i equals m for every step i. Hence, the value of the game $\lim_{n\to\infty} \frac{1}{n+1} \sum_{i=0}^{n} b_i$ equals m, too. $\square$

Proof of Theorem 2. *"Only if part"*. Let $\Pi_r : V = V^W \cup V^B \cup V^R$ be a contra-ergodic partition of G. Let us assign arbitrary strictly positive probabilities to random moves such that $\sum_{u \mid (v,u)\in E} p(v, u) = 1$ for all $v \in V_R$. We still have to assign a local reward $r(v, u)$ to each move $(v, u) \in E$. Let us define $r(v, u) = 1$ whenever $v, u \in V^W$, $r(v, u) = -1$ whenever $v, u \in V^B$, and $r(v, u) = 0$ otherwise. Clearly, if the initial position is in V^W (respectively, in V^W) then the value of the obtained game is 1 (respectively, -1). Hence, the corresponding not initialized game is not ergodic.

"If part". Given a not initialized and not ergodic BWR-game $\mathcal{G} = (G, P, r)$, let $\mu(v)$ denote the value of the corresponding initialized game $\mathcal{G} = (G, P, v, r)$ for each initial position $v \in V$. The obtained function $\mu(v)$ is not constant, since $\mathcal{G}$ is not ergodic. Let μ_W and μ_B denote the maximum and minimum values, respectively. Then, let us set $V^W = \{v \in V \mid \mu(v) = \mu_W$, $V^B = \{v \in V \mid \mu(v) = \mu_B$, and $V^R = V \setminus (V^W \cup V^B)$. It is not difficult to verify that the obtained partition $\Pi_r : V = V^W \cup V^B \cup V^R$ is contra-ergodic. $\square$

Proof of Proposition 2. Let us choose a number μ such that $m(v) > \mu > m(u)$ for every $v \in V^W$ and $u \in V^B$; it exists, because set V of positions is finite. Obviously, properties (i,ii,iii) imply that White (Black) can guarantee at least (at most) μ for every initial position $v \in V^W$ (respectively, $v \in V^B$. Hence, $\mu(v) > \mu > \mu(u)$ for every $v \in V^W$ and $u \in V^B$. $\square$

Proof of Lemma 1. We only consider the case for $v \in V_R$, as the other claims are obvious from the definitions. For the first claim, assume that $x(v) \geq x(u)$, since otherwise there is nothing to prove. Then from $m_x(v) \leq m_0^+$, it follows that

$$\begin{aligned} m_h^+ - r_h^- &\geq m_h^+ - \sum_{u'} p(v,u')r_h(v,u') \\ &\geq p(v,u)(x(v)-x(u)) + \sum_{u' \neq u} p(v,u')(x(v)-x(u')) \\ &\geq \theta(x(v)-x(u)) + x(v)(1-\theta), \end{aligned}$$

from which our claim follows. The other claim can be proved by a similar argument (by replacing $x(\cdot)$ by $\Delta_x - x(\cdot)$ and $m_h^+ - r_h^-$ by $r_h^+ - m_h^-$). □

Proof of Lemma 2. By definition of X, for every node $v \in X$ there must exist (not necessarily distinct) nodes $v_0, v_1, \ldots, v_{2j} = v \in X$ such that $x(v_0) = 0$, and for $i = 1, 2, \ldots, j$, $x(v_{2i}) \geq x(v_{2i-1})$, and either $(v_{2i-2}, v_{2i-1}) \in E$ and $v_{2i-2} \in V_W \cup V_R$ or $(v_{2i-2}, v_{2i-1}) \in \mathrm{EXT}_x$ and $v_{2i-2} \in V_B$. Among the even-numbered nodes, let $v_{2i_1-2}, \ldots, v_{2i_l-2}$ be the ones belonging to V_R. Using Lemma 1, we obtain the following inequality by a telescoping sum:

$$x(v_{2i_{q+1}-2}) \geq x(v_{2i_q-1}) - (i_{q+1} - i_q - 1)(m_h^+ - r_h^-), \text{ for } q = 1, \ldots, l, \tag{9}$$

and $x(v_{2i_1-2}) \geq -i_1(m_h^+ - r_h^-)$.

Now applying Lemma 1 to the pair $v_{2i_q-2} \in V_R$ and v_{2i_q-1}, for $q = 1, \ldots, l$, and using (9) we obtain:

$$x_{q+1} \geq \frac{x_q}{\theta} - \left(\frac{1}{\theta} + i_{q+1} - i_q - 1\right)(m_h^+ - r_h^-), \tag{10}$$

where we write, for convenience, $x_q = x(v_{2i_q-2})$, for $q = 1, \ldots, l$. Iterating, we obtain:

$$x_l \geq -\left(\frac{1}{\theta^{l-1}} + \sum_{q=2}^{l} \frac{1}{\theta^{l-q}}\left(\frac{1}{\theta} + i_q - i_{q-1} - 1\right)\right)(m_h^+ - r_h^-).$$

Combining this with the inequality $x(v) \geq x_l - (j - i_q - 1)(m_h^+ - r_h^-)$ and using $\theta < 1$, we get

$$x(v) \geq -\frac{1}{\theta^{l-1}}(j - i_1 + 1)(m_h^+ - r_h^-) \geq -\frac{1}{\theta^{k-1}}|X|(m_h^+ - r_h^-).$$

Similarly, one can prove for any $v \in Y$ that $x(v) \leq \Delta_x + \frac{1}{\theta^{k-1}}|Y|(r_h^+ - m_h^-)$, and the lemma follows. □

Proof of Lemma 3. Note from (7) that we pump in each iteration by $\delta \geq \frac{\lfloor [m_h] \rfloor}{4}$. Furthermore, our formula for δ implies that once a vertex enters the region $V_h[t_1, t_3]$, it will never leave it. In particular, there are vertices $v_0 \in X \cap V_h[t_0, t_1)$ and $v_n \in Y \cap V_h(t_3, t_4]$ with $x(v_0) = 0$ and $x(v_n) = \Delta_x$.

For a vertex $v \in V$, let $N(v)$ denote the number of times the vertex was pumped. Then $N(v_0) = 0$ and $N(v_n) = N_h$.

We claim that $N(v) \leq T$ for any $v \in X$, and $N(v) \geq N_h - T$ for all $v \in Y$ (i.e., every vertex in X was *not* pumped in *all* steps but at most T, and every vertex in Y was pumped in *all* steps but at most T). Indeed, if $v \in X$ (respective;y, $v \in Y$) was pumped greater than (respectively, less than) T times then $x(v) - x(v_0) \leq -\frac{nR_h}{\theta^{k-1}}$ (respectively, $x(v_n) - x(v) \leq -\frac{nR_h}{\theta^{k-1}}$), in contradiction to Lemma 2.

Since $N_h > 2T$, it follows that $X \cap Y = \emptyset$. Furthermore, among the first $2nT + 1$ iterations, in at most nT iterations some vertex $v \in X$ was pumped, and in at most nT iterations some vertex in Y was not pumped. Thus, there must exist an iteration at which every vertex $v \in X$ was not pumped and every vertex $v \in Y$ was pumped. At that particular iteration, we must have $X \subseteq V_h[t_0, t_2)$ and $Y \subseteq V_h[t_2, t_4]$, and hence $m_x(v) < t_2$ for every $v \in X$ and $m_x(v) \geq t_2$ for every $v \in Y$. By the way the sets X and Y were constructed and from (7), we can easily see that X and Y will continue to have this property till the end of the N_h iterations, and hence they induce a contra-ergodic partition. The lemma follows. $\square$

Proof of Lemma 4. Consider such an extreme point x^*. Then x^* is uniquely determined by a system of n linearly independent equations chosen from the given inequalities. Thus there exist subsets $V' \subseteq V$, $V'_R \subseteq V_R$ and $E'' \subseteq E$ such that, $|V'| + |V'_R| + |E''| = n$, and x^* is the unique solution of the subsystem $x'(v) = 0$ for all $v \in V'$, $x'(v) - x'(u) = m^*_x - r(v, u)$ for $(v, u) \in E''$, and $x'(v) - \sum_{u \in V} p(v, u)x'(u) = m^*_x - \sum_{u \in V} p(v, u)r(v, u)$ for $v \in V'_R$, where m^*_x stands for either m^-_x or m^+_x.

Note that all variables $x'(v)$ must appear in this subsystem, and that the underlying undirected graph of the digraph $G' = (V, E'')$ must be a forest (otherwise the subsystem does not uniquely fix x^*, or it is not linearly independent).

Consider first the case $V_R = \emptyset$. For $i \geq 0$, let V_i be the set of vertices of V at (undirected) distance i from V' (observe that i is finite for every vertex). Then we claim by induction on i that $x^*(v) \leq i\gamma$ for all $v \in V_i$, where $\gamma = \max\{m^+_x - r^-, r^+ - m^-_x\}$. This is trivially true for $i = 0$. So let us assume that it is also true for some $i > 0$. For any $v \in V_{i+1}$, there must exist either an arc (v, u) or an arc (u, v) where $u \in V_{i-1}$. In the former case, we have $x^*(v) = x^*(u) + m^*_x - r(v, u) \leq i\gamma + U - r^- \leq (i + 1)\gamma$. In the latter case, we have $x^*(v) = x^*(u) - (m^*_x - r(u, v)) \leq i\gamma + r^+ - L \leq (i + 1)\gamma$.

Now suppose that $|V_R| > 0$. For each connected component C_l in the forest G', let us fix a node v_l as follows: if C_l has a node in V', v_l is chosen to be such a node, otherwise, v_l is chosen arbitrarily.

For every node $v \in C_l$ let $\mathcal{P}_v$ be an undirected path from v to v_l. Thus, we can write $x'(v)$ uniquely as

$$x'(v) = x'(v_l) + \ell_{v,1} m_x^+ + \ell_{v,2} m_x^- + \sum_{(u',u'') \in \mathcal{P}_v} \ell_{v,u',u''} r(u', u''), \tag{11}$$

for some $\ell_{v,1}, \ell_{v,2} \in \mathbb{Z}$, and $\ell_{v,u',u''} \in \{-1, 1\}$. Thus if $x^*(v_l) = 0$ for some component C_l, then by a similar argument as above, $x^*(v) \leq \gamma |C_l|$ for every $v \in C_l$.

Note that, upto this point, we have used all equations corresponding to arcs in G' and to vertices in V'. The remaining set of $|V'_R|$ equations should uniquely determine the values of the variables in any component which has no node in V'. Substituting the values of $x'(v)$ from (11), for the nodes in any such component, we end-up with a linearly independent system on $k' = |V'_R|$ variables $(I - P')x = b$, where P' is a $k' \times k'$ matrix in which, for $v, u \in V'_R$, $P'(v, u) = \sum_{u' \in U} p(v, u')$ for some $U \subseteq V$ (and hence each row sums up to at most 2 in absolute value), and $\|b\|_\infty \leq n(|[r]| + |[m_x]|) \leq 2nR$.

The rest of the argument follows (in a standard way) by Cramer's rule. Indeed, the value of each component in the solution is given by Δ'/Δ, where Δ is the determinant of $I - P'$ and Δ' is the determinant of a matrix A obtained by replacing one column of $I - P'$ by b. We upper bound Δ' by $k' \|b\|_\infty \Delta_{max}$, where Δ_{max} is the maximum absolute value of a subdeterminant of A of size $k' - 1$. To bound Δ_{max}, let us consider such a subdeterminant with rows $a_1, \ldots, a_{k'-1}$, and use Hadamard's inequality:

$$\Delta' \leq \prod_{i=1}^{k'-1} \|a_i\| \leq 2^{k'-1},$$

since $\|a_i\|_1 \leq 2$, for all i. To lower bound Δ, we note that $W^{k'} \Delta$ is a non-zero integer, and hence has absolute value at least 1. Combining the above inequalities, the lemma follows. □

Proof of Lemma 5. We note the following:

1. By (8), the number of iterations per phase h is at most $N_h = \frac{8n^2 R_h}{|[m_h]| \theta^{k-1}} + 1$.

2. Each iteration requires $O(|E|)$ time, and the end of a phase we need an additional $O(n^2|E|)$ (which is dominated by the time required for REDUCE-POTENTIALS).

3. By Lemma 4, for any $(v, u) \in E$, we have $r_x(v, u) = r(v, u) + x(v) - x(u) \leq (1 + 4nk(2W)^k)R$, and similarly, $r_x(v, u) \geq -(1 + 4nk(2W)^k)R$. In particular, $R_h \leq (1 + 4nk(2W)^k)R$ at the beginning of each phase h in the procedure.

Since $|[m_h]| \leq \frac{3}{4} |[m_{h-1}]|$ for $h = 1, 2, \ldots$, the maximum number of such phases until we reach the required accuracy is at most $H = \log_{4/3} \left(\frac{|[m_0]|}{\varepsilon} \right)$. Putting all the above together, we get that the total running time is

$$(1 + 4nk(2W)^k)R\sum_{h=0}^{H}\frac{1}{|[m_h]|} + O(n^2|E|)H.$$

Noting that $|[m_0]| \le 2R$ and $|[m_H]| \ge \varepsilon$, the lemma follows. □

Appendix C: An upper bound on the required accuracy

Consider a BWR-game in which all local rewards are integral in the range $[-R, R]$, and all probabilities $p(v, u)$, for $(v, u) \in E$ and $v \in V_R$, are rational numbers with least common denominator W. Fix an arbitrary situation s and a starting vertex w, and let respectively P_s^* and $p_s^* = e_w P_s^*$ be the limiting transition matrix and distribution corresponding to s. Using the notation in Appendix A, we let C_i, $i \in J$ be the absorbing classes and $T = V \setminus (\cup_{i\in J} C_i)$ be the set of transient nodes. For $i \in J$, let $y_i \in [0, 1]^T$ be the absorbing probability vector into class C_i.

Lemma 6 *The numbers* $\{y_i(v) \ : \ i \in J$ *and* $v \in T\}$ *are rational with common denominator at most* $(2W)^k$*.*

Proof Consider the system of equations defining y_i: $(I - P')y_i = p'_i$, for some $i \in J$, where $P' = P_s^*[T; T]$ and $p'_i = P_s[T; C_i]e$ (see Appendix A for details). As in the proof of Lemma 4, the idea is to eliminate the variables corresponding to black and white nodes and get a system only on random nodes. Let $E' = \{(v, u) \in E \ : \ v \in V_B \cup V_W$ and $P'(v, u) = 1\}$ and $G' = (V, E')$. Then by linear independence of the system (recall (L3)), G' is a forest. For a node $v \in V_B \cup V_W$, the equation is $y_i(v) = y_i(u) + p'_i(v)$. In particular, if for each connected component C_l in the forest G', we fix a node v_l arbitrarily, we can express $y_i(v)$, for every node $v \in C_l$, uniquely as $y_i(v) = y_i(v_l) + \sum_{(u,u')\in\mathcal{P}_v} \ell_{v,u,u'} p'_i(u)$, for some $\ell_{v,u,u'} \in \{-1, 1\}$, where $\mathcal{P}_v$ is the undirected path between v to v_l.

Substituting these values of $y_i(v)$ in the remaining $k' \le k$ equations $y_i(v) = \sum_u P'(v, u)y_i(u) + p'_i(v)$, for $v \in V_R$, we end-up with a linearly independent system on $k = |V'|$ variables: $(I - P'')x = b$, where P'' is a $k' \times k'$ matrix in which, for $v, u \in V'$, $P''(v, u) = \sum_{u'\in U} P'(v, u')$ for some $U \subseteq V$ (and hence each row sums up to at most 2 in absolute value), and for $v \in V'$, $b(v) = \sum_u \ell_u p'_i(u)$, where $\ell_u \in \mathbb{Z}$.

The rest of the argument follows as in Lemma 4. The value of each component in the solution is given by Δ'/Δ, where Δ is the determinant of $I - P''$ and Δ' is the determinant of a matrix A obtained by replacing one column of $I - P''$ by b. The lemma follows by observing that $\Delta \le 2^{k'}$ and that both $W^{k'}\Delta$, and $W^{k'}\Delta'$ are integral. □

For a situation s (that is a pair of strategies), let $\kappa(s)$ be the number of absorbing classes (and hence, directed cycles) with only black and white nodes.

Lemma 7 *The numbers $\{p_s^*(v) \;:\; v \in V\}$ are rational with with common denominator at most $n^{\kappa(s)+k}k^k(2W)^{k+k^2+k\kappa(s)}$.*

Proof Let $J' \subseteq J$, be the set of indices such that $C_i \cap V_R = \emptyset$. Note by definition that $\kappa(s) = |J'|$, and $|J \setminus J'| \leq k$. Consider any absorbing class C_i. Let $E' = \{(v,u) \in E \;:\; v, u \in C_i$ and $P_s^*(v,u) > 0\}$ and $G' = (V, E')$. Assume first that $i \in J'$. Then $p_s^*(v) = y_i(w)/|C_i|$ for all $v \in C_i$, where w is the starting vertex and $y_i(w)$ is the absorbing probability of w into C_i (so $y_i(w) = 1$ if $w \in C_i$). It follows from Lemma 6 that $\{p_s^*(v) \;:\; v \in C_i\}$ are rational with common denominator $M_i \leq |C_i|(2W)^k$.

Now consider the case $i \in J \setminus J'$. For $v \in C_i$, let $\mathcal{R}(v)$ be the set of vertices that can reach v by a directed path in G', all whose internal nodes (if any) are in $V_B \cup V_R$. Note that $\mathcal{R}(v) \cap V_R \neq \emptyset$ for each $v \in (V_B \cup V_W) \cap C_i$. Otherwise, there is a node $v \in (V_B \cup V_W) \cap C_i$ such that no random node in C_i can reach v, and hence the whole component C_i must be a cycle of black and white nodes only.

Consider the system of equations in (L5) defining $p_s^*(v)$:

$$\pi(v) = \sum_{u \in (V_B \cup V_W) \cap C_i} \pi(u) + \sum_{u \in V_R \cap C_i} p(u,v)\pi(u),$$

for $v \in C_i$, and $\sum_{v \in C_i} \pi(v) = y_i(w)$. Eliminating the variables $\pi(v)$ for $v \in (V_B \cup V_W) \cap C_i$, we end-up with a system on only random nodes $v \in V_R \cap C_i$: $\pi(v) = \sum_{u \in \mathcal{R}(v) \cap V_R} p'(u,v)\pi(u)$, where $p'(u,v) = p(u,v) + \sum_{u' \in \mathcal{R}(v) \cap (V_B \cup V_W)} p(u,u')$ (note that $\sum_{u \in V_R \cap C_i} p'(v,u) = 1$ for all $v \in V_R \cap C_i$). Similarly, we can reduce the normalization equation to $\sum_{v \in V_R \cap C_i}(1 + \sum_{u \in V_B \cup V_W,\; v \in \mathcal{R}(u)} p'(v,u))\pi(v) = y_i(w)$.

This gives a system on $k_i = |C_i \cap V_R|$ variables of the form $(I - P')x = 0,\; bx = y_i(w)$, where the matrix P' and the vector b have rational entries with common denominator at most W, each column of P' sums up to at most 1, and $\|b\|_1 \leq k_i|C_i|$. Any non-zero component $p_s^*(v)$ in the solution of this system takes the form $y_i(w)\frac{\Delta}{\sum_{i=1}^{k_i} b_i \Delta_i}$, where $\Delta, \Delta_1, \ldots, \Delta_{k_i}$ are subdeterminants of $I - P'$ of rank $k_i - 1$. Using $\Delta_i \leq 2^{k_i - 1}$ and Lemma 6, we get that all $p_s^*(v)$, for $v \in V_R \cap C_i$ are rational with common denominator $M_i \leq |C_i|k_i(2W)^{k_i-1+k}$.

After solving this system, we can get the value of $p_s^*(v)$, for $v \in (V_W \cap V_B) \cap C_i$ from the equation: $\pi(v) = \sum_{u \in \mathcal{R}(v) \cap V_R} p'(u,v)\pi(u)$. Again, we get rational numbers with common denominator at most $M_i \leq |C_i|k_i(2W)^{k_i+k}$.

It follows that all components of p_s^* are rational with common denominator at most:

$$\prod_{i=1}^{h} M_i \leq \prod_{i \in J'} \left[|C_i|(2W)^k\right] \prod_{i \in J \setminus J'} \left[|C_i|k_i(2W)^{k_i+k}\right] \leq [n(2W)^k]^{\kappa(s)}(nk)^k(2W)^{k+k^2}. \tag{12}$$

□

The following example shows that the bound in Lemma 7 cannot be made, in general, to depend exponentially only on the number of random nodes, if we consider an *arbitrary* situation s. Consider a graph with one random node w and κ disjoint directed cycles $C_1, \ldots, C_\kappa$, of sizes $n_i = |C_i|$ which are *relatively prime*. There is an arc from w to one arbitrary node in each cycle with transition probability $1/\kappa$, and the total reward of each cycle is 1. Then the limiting distribution starting from w is $p_s^*(w) = 0$ and $p_s^*(v) = 1/n_i$ and the value at node w is given by $\frac{1}{\kappa}\sum_{i=1}^{\kappa}\frac{1}{n_i}$. This has a denominator exponential in $n = \sum_{i=1}^{\kappa} n_i + 1$.

In view of this example, in order to get stronger bounds, one needs to consider special situations, namely, those that can be returned by our pumping algorithm. Let s be such a situation. Then for every $v \in V_B \cup V_W$, the arc $(v, s(v))$ selected by s is extremal, i.e., $m_x(v) = r_x(v, s(v))$ given the current potential x. We call such a situation *extremal* with respect to x.

Lemma 8 *Suppose that s is an extremal situation with respect to a potential vector x, such that C and C' are absorbing classes (and hence cycles) with only black and white nodes $C, C' \subseteq V_B \cup V_W$. If $|[m_x]| < \frac{1}{n^2}$, then $\frac{r(C)}{|C|} = \frac{r(C')}{|C'|}$, where $r(C), r(C')$ are the local rewards of the cycles C and C', respectively (that is, $r(C) = \sum_{v \in C} r(v, s(v))$).*

Proof Since s is extremal, we have $r(C) = r_x(C) = \sum_{v\in C} m_x(v) \in [m_x^-|C|, m_x^+|C|]$, and hence $r(C)/|C| \in [m_x^-, m_x^+]$. Similarly, $r(C')/|C'| \in [m_x^-, m_x^+]$. Thus, if $r(C)/|C| \neq r(C')/|C'|$, we must have

$$\frac{1}{n^2} \leq \left|\frac{|C'|r(C) - |C|r(C')}{|C||C'|}\right| = \left|\frac{r(C)}{|C|} - \frac{r(C')}{|C'|}\right| \leq m_x^+ - m_x^- < \frac{1}{n^2}.$$

□

Lemma 9 *Let s be an extremal situation with respect to some potential x such that $|[m_x]| < \frac{1}{n^2}$. Then c_s (the expected limiting payoff corresponding to situation s) is rational with denominator at most $n^{k+1}k^k(2W)^{2k+k^2+1}$.*

Proof Let $C_1, \ldots, C_{\kappa(s)}$ be the absorbing classes with no random nodes, in the Markov chain defined by s, and let $V' \subseteq V \setminus \cup_{i=1}^{\kappa(s)} C_i$ be the set of nodes in the remaining absorbing classes. Since s is extremal, Lemma 8 implies that either $\kappa(s) \leq 1$ or $r(C_i)/|C_i|$ is some constant γ for all $i = 1, \ldots, \kappa(s)$. In the former case, $c(s) = p_s^* Q_s r = \sum_{v \in V} p_s^*(v) \sum_u p(v,u) r(v,u)$ is rational with denominator at most $n^{k+1}k^k(2W)^{2k+k^2+1}$ by Lemma 7. In the latter case, we get

$$c(s) = \sum_{i=1}^{\kappa(s)} y_i(w)\frac{r(C_i)}{|C_i|} + \sum_{v\in V'} p_s^*(v)\sum_u p(v,u)r(v,u) = \gamma\sum_{i=1}^{\kappa(s)} y_i(w) + \sum_{v\in V'} p_s^*(v)\sum_u p(v,u)r(v,u),$$

which by (12) is again rational with denominator as stated in the lemma. □

The next lemma shows that it is enough in our pumping algorithm to take

$$\varepsilon = \frac{1}{n^{2(k+1)} k^{2k} (2W)^{4k+2k^2+2}}. \tag{13}$$

Lemma 10 *Assume that the given BWR-game is ergodic, and let* ε *be as defined in (13). Consider two situations* s *and* s' *such that* s *is optimal,* s' *is extremal with respect to some potential* x *for which* $|[m_x]| \leq \varepsilon$. *Then* $c_s = c_{s'}$.

Proof Since both s an s' are extremal with respect to some potentials, and $\varepsilon < \frac{1}{n^2}$, Lemma 9 implies that both $c(s)$ and $c(s')$ are rational with denominator at most $M = n^{k+1}k^k(2W)^{2k+k^2+1}$. It follows that $|c(s) - c(s')|$ is rational with denominator at most M^2, and hence if $c(s) \neq c(s')$, then $|c(s) - s(s')| \geq \frac{1}{M^2}$. □

Appendix D: Potential reduction for BW-games

For completeness, we give here a version of the potential reduction procedure of Pisaruk [Pis99] for BW-games.

Algorithm 2 REDUCE-BW($\mathcal{G}, x$)

Require: A BW game $\mathcal{G} = (G = (V, E), P, r)$ and a current set of potentials $x : V \to \mathbb{R}$
Ensure: A reduced potential x'

$X := emptyset$; $x' := x$
while $X \neq V$ **do**
  $\epsilon_1 := \min\{m_x^+ - r_{x'}(v,u) \ : \ v \in X,\ u \in V \setminus X,\ (v,u) \in E,\ r_{x'}(v,u) \leq m_x^+\}$
  $\epsilon_2 := \min\{r_{x'}(v,u) - m_x^- \ : \ v \in V \setminus X,\ u \in X,\ (v,u) \in E,\ r_{x'}(v,u) \geq m_x^-\}$
  $\epsilon_3 := \min\{x'(v) \ : \ v \in V \setminus X\}$
  $x'(v) := x'(v) - \min\{\epsilon_1, \epsilon_2, \epsilon_3\}$ for all $v \in V \setminus X$
  while $\exists v \in V \setminus X$ such that $x'(v) \leq \frac{m_x^+ - m_x^-}{2}|X|$ **do**
    $X := X \cup \{v\}$
  end while
end while
return x'

Lemma 11 *Let* $\mathcal{G} = (G, r)$ *be a BW-game and* x *be a given potential vector. Then in* $O(n|E|)$ *time, procedure REDUCE(*$\mathcal{G}, x$*) returns another potential vector* x' *such that*

(a) $0 \leq x'(v) \leq \frac{m^+ - m^-}{2}(n-1)$ *for all* $v \in V$;

(b) $[m_{x'}] \subseteq [m_x]$.

Proof After the first iteration of the outer while loop, we are guaranteed that the new potential x' satisfies $x'(v) \geq 0$, and it will remain so for the rest of the iterations. In particular, condition (a) will be satisfied when the procedure terminates (provided it does terminate) as follows from the way the set X is updated. Condition (b) is maintained throughout the procedure by our choice of $\epsilon = \min\{\epsilon_1, \epsilon_2, \epsilon_3\}$. So it remains to show that the procedure terminates in at most n steps. Consider any iteration of the outer while loop, where the potential is x' at the beginning of the iteration, and note that any $v \in X$ and $u \in V \setminus X$ have a potential difference of $x'(v) - x'(u) < -\frac{m_x^+ - m_x^-}{2}$. Thus, if $v \in X$, $u \in V \setminus X$ (respectively, $v \in V \setminus X$, $u \in X$), $(v, u) \in E$, and $r_{x'}(v, u) \leq m_x^+$ (respectively, $r_{x'}(v, u) \geq m_x^-$) then $r_{x'}(v, u) = r_x(v, u) + x'(v) - x'(u) < \frac{m_x^+ + m_x^-}{2} \leq m_x^+$ (respectively, $r_{x'}(v, u) > \frac{m_x^+ + m_x^-}{2} \geq m_x^-$). This implies that $\epsilon_1 > \frac{m_x^+ + m_x^-}{2}$ and $\epsilon_2 > \frac{m_x^+ + m_x^-}{2}$. Also $\epsilon_3 > 0$ (except possibly for the first iteration). Thus $\epsilon > 0$. When $\epsilon = \epsilon_3$, then with respect to the new potentials x'' (obtained by updating x': $x''(v) = x'(v) - \epsilon$ for all $v \in V \setminus X$), there is a vertex v having $x''(v) = 0$ which will be added to X. On the other hand, when $\epsilon = \epsilon_1$ (respectively, $\epsilon = \epsilon_2$), then there is an edge $(v, u) \in E$ such that $v \in X$, $u \in V \setminus X$ and $\epsilon_1 = m_x^+ - r_{x'}(v, u)$ (respectively, $v \in V \setminus X$, $u \in X$ and $\epsilon_2 = r_{x'}(v, u) - m_x^-$), i.e., $m_x^+ = r_{x''}(v, u) = r_x(v, u) + x''(v) - x''(u) \leq m_x^+ + x''(v) - x''(u)$ (respectively, $m_x^- = r_{x''}(v, u) = r_x(v, u) + x''(v) - x''(u) \geq m_x^- + x''(v) - x''(u)$), from which follows $x''(u) \leq x''(v) \leq \frac{m_x^+ - m_x^-}{2}(|X| - 1)$ (respectively, $x''(v) \leq x''(u) \leq \frac{m_x^+ - m_x^-}{2}(|X| - 1)$). (Note that we used the fact that $r_x(v, u) \leq r_{x'}(v, u)$ for all $(v, u) \in E$ such that $v \in X$ and $u \in V \setminus X$, and $r_x(v, u) \geq r_{x'}(v, u)$ for all $(v, u) \in E$ such that $v \in V \setminus X$ and $u \in X$.) Thus u (respectively, v) will be added to X. □

Appendix E: Examples

Example 1: Solvability in pure positional strategies

Let us consider a graph in which all three classes, White, Black, and Random positions, are not empty. A minimal such graph $G = (V, E)$ contains three vertices $V = \{0, 1, 2\}$, where $V_R = \{0\}$, $V_W = \{1\}$, and $V_B = \{2\}$. Furthermore, let E consists of six arcs $[i, j)$ for all $i, j \in \{0, 1, 2\}$ such that $i \neq j$. Thus, G is a complete tripartite graph without loops and multiple edges. For all local costs and probabilities, the obtained game has a value (in pure positional strategies).

Yet, even for this simple example the reduction to canonical form would demand a long case analysis. Instead, let us consider the normal form. Each player, White and Black, has only two strategies: $s_W^1 = (1, 2)$, $s_W^2 = (1, 0)$ for White and $s_B^1 = (2, 1)$, $s_B^2 = (2, 0)$ for Black.

We will consider the corresponding 2×2 matrix, with entries $a_{1,1}$, $a_{1,2}$, $a_{2,1}$, $a_{2,2}$, and show that it has a saddle point in pure strategies for all possible parameters of the game.

As usual, we denote by $r(i,j)$ the cost of the move (i,j) and by $p_1 = p_{0,1}$ and $p_2 = p_{0,2}$ the probabilities to move from 0 to 1 and to 2, respectively; assuming that $p_1 > 0, p_2 > 0$ and $p_1 + p_2 = 1$.

Let us consider all four situations. It is easy to see that (s_W^1, s_B^1) results in a simple cycle on vertices 1 and 2 and, hence, $a_{1,1} = (1/2)(r(1,2) + r(2,1))$.

Situation (s_W^1, s_B^2) results in a Markov chain with the limit distribution:

$P_1 = p_1/(2+p_1), \quad P_0 = P_2 = 1/(2+p_1) \quad$ and, hence,

$a_{1,2} = (p_1/(2+p_1))(r(0,1) + r(1,2)) + (p_2/(2+p_1))r(0,2) + (1/(2+p_1))r(2,0)$.

By symmetry, (s_W^2, s_B^1) results in a Markov chain with the limit distribution:

$P_2 = p_2/(2+p_2), \quad P_0 = P_1 = 1/(2+p_2) \quad$ and, hence,

$a_{2,1} = (p_2/(2+p_2))(r(0,2) + r(2,1)) + (p_1/(2+p_2))r(0,1) + (1/(2+p_2))r(1,0)$.

Finally, (s_W^2, s_B^2) results in

$a_{2,2} = (1/2)[p_1(r(0,1) + r(1,0))) + p_2(r(0,2) + r(2,0))]$.

Let us remark that all four limit distributions do not depend on the initial position, in agreement with the ergodicity of G.

Multiplying all entries by $2(2+p_1)(2+p_2)$ we obtain:

$a'_{1,1} = (2+p_1)(2+p_2)(r(1,2) + r(2,1))$;

$a'_{1,2} = 2(2+p_2)[p_1(r(0,1) + r(1,2)) + p_2 r(0,2) + r(2,0)]$;

$a'_{2,1} = 2(2+p_1)[p_2(r(0,2) + r(2,1)) + p_1 r(0,1) + r(1,0)]$;

$a'_{2,2} = (2+p_1)(2+p_2)[p_1(r(0,1) + r(1,0)) + p_2(r(0,2) + r(2,0))]$.

We claim that this matrix game has a saddle point in pure strategies for all values $r(i,j)$ and p_1, p_2 such that $p_1 > 0, p_2 > 0$, and $p_1 + p_2 = 1$. It is well-known that a 2×2 matrix game has no saddle point in pure strategies if and only if each entry of one diagonal is strictly larger than each entry of the other diagonal. Let us assume indirectly that $\max\{a_{1,1}, a_{2,2}\} < \min\{a_{1,2}, a_{2,1}\}$, that is,

$a_{1,1} < a_{1,2}$, $a_{1,1} < a_{2,1}$, $a_{2,2} < a_{1,2}$, and $a_{2,2} < a_{2,1}$.

Substituting four entries $a_{i,j}$ we can rewrite this system as follows:

$$2p_1 r(0,1) + 2p_2 r(0,2) + 2r(2,0) - (2-p_1)r(1,2) - (2+p_1)r(2,1) > 0,$$
$$2p_2 r(0,2) + 2p_1 r(0,1) + 2r(1,0) - (2-p_2)r(2,1) - (2+p_2)r(1,2) > 0,$$
$$-p_1^2 r(0,1) - p_1 p_2 r(0,2) + (2 - (2+p_1)p_2)r(2,0) + 2p_1 r(1,2) - (2+p_1)p_1 r(1,0) > 0$$
$$-p_2^2 r(0,2) - p_2 p_1 r(0,1) + (2 - (2+p_2)p_1)r(1,0) + 2p_2 r(2,1) - (2+p_2)p_2 r(2,0) > 0$$

If we assume that $\min\{a_{1,1}, a_{2,2}\} > \max\{a_{1,2}, a_{2,1}\}$ then we obtain the same four inequalities multiplied by -1. In both cases we get a contradiction, since their linear combination with the strictly positive coefficients: $p_1p_2^2$, $p_2p_1^2$, p_2, and p_1 results in 0 in the left hand side.

Example 2: Local and global optimality

Let us consider a graph with 2 nodes: a white node u and a black node v. There are two self loops (u, u) and (v, v) with rewards -1 and 1 respectively. There are two additional arcs making a cycle $(u, v), (v, u)$ with rewards, $-R$ and R, respectively, where R is a very large integer. Even though this is a very small example with no random nodes, it is enough to illustrate many useful things. First, the locally attractive moves (u, u) and (v, v) are not globally optimal, since the optimal solution is to follow the cycle of average reward 0. Second, the range of m (which $[-1, 1]$ in this case) could be much smaller than the range of R, and yet arcs with very large local rewards are not eligible for deletion. Third, our pumping algorithm will finish in a constant number of iterations on this example, while a dynamic programming approach might take $O(R)$ iterations.

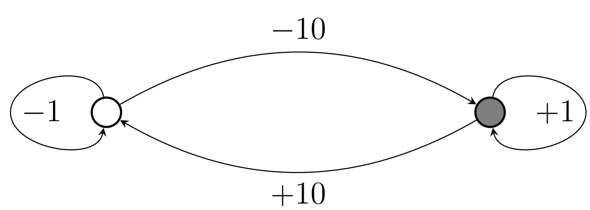

Figure 2: Example 2: $R = 10$