Extending Finite Memory Determinacy to Multiplayer Games[☆]

Stéphane Le Roux & Arno Pauly

Department d'Informatique
Université Libre de Bruxelles

Abstract

We show that under some general conditions the finite memory determinacy of a class of two-player win/lose games played on finite graphs implies the existence of a Nash equilibrium built from finite memory strategies for the corresponding class of multi-player multi-outcome games. This generalizes a previous result by Brihaye, De Pril and Schewe. We provide a number of example that separate the various criteria we explore.

Our proofs are generally constructive, that is, provide upper bounds for the memory required, as well as algorithms to compute the relevant Nash equilibria.

Keywords: Finite memory, Games played on finite graphs, finite-memory determinacy, Nash equilibrium, equilibrium transfer, energy parity games
2010 MSC: 91A06, 91A43, 68Q45

1. Introduction

The usual model employed for synthesis are sequential two-player win/lose games played on finite graphs. The vertices of the graph correspond to states of a system, and the two players jointly generate an infinite path through the graph (the *run*). One player, the protagonist, models the aspects of the system under the control of the designer. In particular, the protagonist will win the game iff the run satisfies the intended specification. The other player is assumed

[☆]The authors were supported by the ERC inVEST (279499) project.

to be fully antagonistic, thus wins iff the protagonist loses. One then would like to find winning strategies of the protagonist, that is, a strategy for her to play the game in such a way that she will win regardless of the antagonist's moves. Particularly desirable winning strategies are those which can be executed by a finite automaton.

Classes of games are distinguished by the way the winning conditions (or more generally, preferences of the players) are specified. Typical examples include:

- Muller conditions, where only the set of vertices visited infinitely many times matters;
- Parity conditions, where each vertex has a priority, and the winner is decided by the parity of the least priority visited infinitely many times;
- Energy conditions, where each vertex has an energy delta (positive or negative), and the protagonist loses if the cumulative energy values ever drop below 0;
- Discounted payoff conditions, where each vertex has a payoff value, and the outcome is determined by the discounted sum of all payoffs visited with some discount factor $0 < \lambda < 1$;
- Combinations of these, such as energy parity games, where the protagonist has to simultaneously ensure that the least parity visited infinitely many times is odd and that the cumulative energy value is never negative.

Our goal is to dispose of two restrictions of this setting: First, we would like to consider any number of players; and second allow them to have far more complicated preferences than just preferring winning over losing. The former generalization is crucial in a distributed setting (also e.g. [1, 2]): If different designers control different parts of the system, they may have different specifications they would like to enforce, which may be partially but not entirely overlapping. The latter seems desirable in a broad range of contexts. Indeed, rarely is the intention for the behaviour of a system formulated entirely in black and white: We prefer a programm just crashing to accidently erasing our hard-drive; we prefer a programm to complete its task in 1 minute to it taking 5 minutes, etc. We point to [3] for a recent survey on such notions of quality in synthesis.

Rather than achieving this goal by revisiting each individual type of game and proving the desired results directly (e.g. by generalizing the original proofs of the existence of winning strategies), we shall provide two transfer theorems: In both Theorem 8 and Theorem 12, we will show that (under some conditions), if the two-player win/lose version of a game is finite-memory determined, the corresponding multi-player multi-outcome games all have finite-memory Nash equilibria. The difference is that Theorem 8 refers to all games played on finite graphs using certain preferences, whereas Theorem 12 refers to one fixed graph only.

Theorem 12 is more general than a similar one obtained by Brihaye, De Pril and Schewe [1],[4, Theorem 4.4.14]. A particular class of games covered by our result but not the previous one are (a variant of) energy parity games as introduced by Chatterjee and Doyen [5]. The high-level proof idea follows earlier work by the authors on equilibria in infinite sequential games, using Borel determinacy as a blackbox [6][1] – unlike the constructions there (cf. [9]), the present ones however are constructive and thus give rise to algorithms computing the equilibria in the multi-player multi-outcome games given suitable winning strategies in the two-player win/lose versions.

The general theme of transferring determinacy results from antagonistic two-player games to the existence of Nash equilibria in multiplayer games is already present in [10] by Thuijsman and Raghavan, as well as [11] by Grädel and Ummels.

Echoing De Pril in [4], we would like to stress that our conditions ap-

[1]Precursor ideas are also present in [7] and [8] (the specific result in the latter was joint work with Neymann).

ply to the preferences of each player individually. For example, some players could pursue energy parity conditions, whereas others have preferences based on Muller conditions: Our results apply just as they would do if all players had preferences of the same type.

This article extends and supersedes the earlier [12] which appeared in the proceedings of Strategic Reasoning 2016.

Structure of the paper: After introducing notation and the basic concepts in Section 2, we state our two main theorems in Section 3. The proofs of our main theorems are given in the form of several lemmata in Section 4. The lemmata prove slightly more than required for the theorems, and might be of independent interest for some readers. In Section 5 we discuss how our results improve upon prior work, and explore several notions prominent in our main theorems in some more detail. Finally, in Section 6 we consider as applications two classes of games covered by our main theorems but not by previous work.

2. Background

Win/lose two-player games:. A win/lose two-player game played on a finite graph is specified by a directed graph (V, E) where every vertex has an outgoing edge, a starting vertex $v_0 \in V$, two sets $V_1 \subseteq V$ and $V_2 := V \setminus V_1$, a function $\Gamma : V \to C$ coloring the vertices, and a *winning condition* $W \subseteq C^\omega$. Starting from v_0, the players move a token along the graph, ω times, with player $a \in \{1, 2\}$ picking and following an outgoing edge whenever the current vertex lies in V_a. Player 1 wins iff the infinite sequence of the colors seen (at the visited vertices) is in W.

Winning strategies:. For $a \in \{1, 2\}$ let $\mathcal{H}_a$ be the set of finite paths in (V, E) starting at v_0 and ending in V_a. Let $\mathcal{H} := \mathcal{H}_1 \cup \mathcal{H}_2$ be the possible *finite histories* of the game, and let $[\mathcal{H}]$ be the infinite ones. For clarity we may write $[\mathcal{H}_g]$ instead of $[\mathcal{H}]$ for a game g. A *strategy* of player $a \in \{1, 2\}$ is a function of type $\mathcal{H}_a \to V$ such that $(v, s(hv)) \in E$ for all $hv \in \mathcal{H}_a$. A pair of strategies (s_1, s_2) for the two players induces a run $\rho \in V^\omega$: Let $s := s_1 \cup s_2$ and set

$\rho(0) := v_0$ and $\rho(n+1) := s(\rho(0)\rho(1)\dots\rho(n))$. For all strategies s_a of player a let $\mathcal{H}(s_a)$ be the finite histories in $\mathcal{H}$ that are compatible with s_a, and let $[\mathcal{H}(s_a)]$ be the infinite ones. Every sequence $v_0v_1v_2\dots$ of vertices naturally induces a color trace $\Gamma(v_0v_1v_2\dots) := \Gamma(v_0)\Gamma(v_1)\Gamma(v_2)\dots$. A strategy s_a is said to be winning if $\Gamma[[\mathcal{H}(s_a)]] \subseteq W$, *i.e.* a wins regardless of her opponent's moves. A game where either of the players has a winning strategy is called *determined*.

Finite-memory strategies:. A *strategic update* for player a using m bits of memory is a function $\sigma : V \times \{0,1\}^m \to V \times \{0,1\}^m$ that describes the two simultaneous updates of player a upon arrival at a vertex v if its memory content was M just before arrival: $(v, M) \mapsto \pi_2 \circ \sigma(v, M)$ describes the memory update and $(v, M) \mapsto \pi_1 \circ \sigma(v, M)$ the choice for the next vertex. This choice will be ultimately relevant only if $v \in V_a$, in which case we require that $(v, \pi_1 \circ \sigma(v, M)) \in E$.

A strategic update together with an initial memory content $M_\epsilon \in \{0,1\}^m$ is called a *strategic implementation*. The memory content after some history is defined by induction: $M_\sigma(M_\epsilon, \epsilon) := M_\epsilon$ and $M_\sigma(M_\epsilon, hv) := \pi_2 \circ \sigma(v, M_\sigma(M_\epsilon, h))$ for all $hv \in \mathcal{H}$. A strategic update σ together with initial memory content M_ϵ induce a *finite-memory strategy* s_a defined by $s_a(hv) := \pi_1 \circ \sigma(v, M_\sigma(M_\epsilon, h))$ for all $hv \in \mathcal{H}_a$. In a slight abuse we may call strategic implementation finite-memory strategies. If not stated otherwise, we will assume the initial memory to be 0^m.

Multi-outcome multi-player games and Nash equilibria:. A (general) game played on a finite graph is specified by a directed graph (V, E), a set of agents A, a cover $\{V_a\}_{a \in A}$ of V via pairwise disjoint sets, the starting vertex v_0, a function $\Gamma : V \to C$ coloring the vertices, and for each player a a preference relation[2] $\prec_a \subseteq \Gamma[[\mathcal{H}]] \times \Gamma[[\mathcal{H}]]$ (or more generously $\prec_a \subseteq C^\omega \times C^\omega$). We overload the

[2]Note that we do not understand preferences to automatically be total or satisfy other specific properties. From Definition 3 onwards we will restrict our attention to strict weak orders though.

notation by also writing $\rho \prec_a \rho'$ if $\Gamma(\rho) \prec_a \Gamma(\rho')$ for all $\rho, \rho' \in [\mathcal{H}]$, *i.e.* players compare runs by comparing their color traces. The two-player games with inverse preferences ($\prec_2 = \prec_1^{-1}$) are called antagonistic games, and they generalize win/lose two-player games.

The notions of strategies and induced runs generalize in the obvious way. In particular, instead of a pair of strategies (one per player), we consider families $(s_a)_{a\in A}$, which are called strategy profiles. The concept of a winning strategy no longer applies though. Instead, we use the more general notion of a Nash equilibrium: A family of strategies $(s_a)_{a\in A}$ is a Nash equilibrium, if there is no player $a_0 \in A$ and strategy s'_{a_0} such that a_0 would prefer the run induced by $(s_a)_{a\in A\setminus\{a_0\}} \cup (s'_a)_{a\in\{a_0\}}$ to the run induced by $(s_a)_{a\in A}$. Intuitively, no player can gain by unilaterally deviating from a Nash equilibrium. Note that the Nash equilibria in two-player win/lose games are precisely those pairs of strategy where one strategy is a winning strategy.

Threshold games and future games:. Our results, including transfer from the two-player win/lose case to the general case, rely on the idea that each general game induces a collection of two-player win/lose games, namely the threshold games of the future games, as below.

Definition 1 (Future game and one-vs-all threshold game)**.**

Let $g = \langle (V,E), v_0, A, \{V_a\}_{a\in A}, (\prec_a)_{a\in A}\rangle$ be a game played on a finite graph.

- For $a_0 \in A$ and $\rho \in [\mathcal{H}]$, the one-vs-all threshold game $g_{a_0,\rho}$ for a_0 and ρ is the win-lose two-player game played on (V,E), starting at v_0, with vertex subsets V_{a_0} and $\bigcup_{a\in A\setminus\{a_0\}} V_a$, and with winning set $\{\rho' \in [\mathcal{H}] \mid \rho \prec_{a_0} \rho'\}$ for Player 1.

- Let $v \in V$. For paths hv and vh' in (V,E) let $hv\hat{\ }vh' := hvh'$.

- For all $h \in \mathcal{H}$ with last vertex v let $g^h := \langle (V,E), v, A, \{V_a\}_{a\in A}, (\prec^h_a)_{a\in A}\rangle$ be called the future game of g after h, where for all $\rho, \rho' \in [\mathcal{H}_{g^h}]$ we set $\rho \prec^h_a \rho'$ iff $h\hat{\ }\rho \prec_a h\hat{\ }\rho'$. If s is a strategy in g, let s^h be the strategy in g^h such that $s^h(h') := s(h\hat{\ }h')$ for all $h' \in \mathcal{H}_{g^h}$.

Observation 2. Let $g = \langle (V, E), v_0, A, \{V_a\}_{a\in A}, (\prec_a)_{a\in A}\rangle$ be a game played on a finite graph.

1. g and its thresholds games have the same strategies.
2. for all $h, h' \in \mathcal{H}$ ending with the same vertex the games g^h and $g^{h'}$ have the same (finite-memory) strategies.
3. g, its future games, and their thresholds games have the same strategic implementations.
4. If a strategy s_a in g is finite-memory, for all $h \in \mathcal{H}$ the strategy s_a^h is also finite-memory.

Proof. We only prove the fourth claim. Since s_a is a finite-memory strategy, it comes from some strategic implementation (σ, M_ϵ). We argue that $(\sigma, M_\sigma(M_\epsilon, h)))$ is a strategic implementation for s_a^{hv}: First, $s_a^{hv}(v) = s_a(hv) = \pi_1\circ\sigma(v, M_\sigma(M_\epsilon, h)) = \pi_1\circ\sigma(v, M_\sigma(M_\sigma(M_\epsilon, h), \epsilon))$; second, for all $h'v' \in \mathcal{H}^{hv}$ we have $s_a^{hv}(vh'v') = s_a(hvh'v') = \pi_1\circ\sigma(v', M_\sigma(M_\epsilon, hvh')) = \pi_1\circ\sigma(v', M_\sigma(M_\sigma(M_\epsilon, h), vh'))$. □

Our (transfer) results rely on players having winning strategies that are implementable with *uniformly* finite memory, so that for every game they may be picked from a finite set of strategies. The following (shortenable) shorthands will be useful. Let g be a game, let a be a player.

- Let $m \in \mathbb{N}$ be such that in all threshold games for a in g, if player a has a winning strategy, she has one that is implementable using m bits of memory. Then we say that player a wins her winnable threshold games in g using uniformly finite memory m.
- Let $m \in \mathbb{N}$ be such that all (future) threshold games for a in g have finite-memory winning strategies that are implementable using m bits of memory. Then we say that the (future) threshold games for a in g are uniformly-finite-memory determined using m bits.

Note that speaking about future games above is the more general statement, as we prefix some finite history, and the sufficient memory depends on neither the threshold nor the history (cf Example **??**).

Strict weak orders:. The concepts so far are well-defined for preferences that are arbitrary binary relations. However, our results rely on the preferences being strict weak orders, as defined below, and all the preferences in the remainder of this article are assumed to be strict weak orders.

Definition 3 (Strict weak order)**.** Recall that a relation $\prec$ is called a *strict weak order* if it satisfies:

$$\begin{array}{ll} \forall x, & \neg(x \prec x) \\ \forall x, y, z, & x \prec y \wedge y \prec z \Rightarrow x \prec z \\ \forall x, y, z, & \neg(x \prec y) \wedge \neg(y \prec z) \Rightarrow \neg(x \prec z) \end{array}$$

Strict weak orders capture in particular the situation where each player cares only about a particular aspect of the run (e.g. her associated personal payoff), and is indifferent between runs that coincide in this aspect but not others (e.g. the runs with identical associated payoffs for her, but different payoffs for the other players). We will show in Subsection 6.2 that considering strict weak orders is strictly more general than working with payoff functions only.

Guarantees. Definition 4 below rephrases Definitions 2.3 and 2.5 from [7]: Given a strategy of a player a, the guarantee consists of the compatible runs plus the runs that are, according to $\prec_a$, not worse than all the compatible runs. The guarantee is thus upper-closed w.r.t. $\prec_a$. The best guarantee is the intersection of all the guarantees, and is thus also upper-closed.

Definition 4 (Player (best) future guarantee)**.** Let g be the game $\langle (V, E), v_0, A, \{V_a\}_{a\in A}, (\prec_a)_{a\in A}\rangle$ and let $a \in A$. For all $h \in \mathcal{H}$ and strategies s_a for a in g^h let $\gamma_a(h, s_a) := \{\rho \in [\mathcal{H}_{g^h}] \mid \exists \rho' \in [\mathcal{H}_{g^h}(s_a)], \neg(\rho \prec^h_a \rho')\}$ be the player future guarantee by s_a in g^h. Let $\Gamma_a(h) := \bigcap_{s_a} \gamma_a(h, s_a)$ be the best future guarantee of a in g^h. We write $\gamma_a(s_a)$ and Γ_a when h is the trivial history.

Note that in general the best guarantee may be empty, but our assumptions will (indirectly) rule this out: they will imply that each player has indeed a strategy realizing her best guarantee, i.e. a strategy s_a with $\Gamma_a = \gamma_a(s_a)$.

In Example 5 we construct a Nash equilibrium by starting with a strategy profile where everyone is realizing their guarantee, and then adding punishments against any deviators. The idea behind this construction is one ingredient of our results.

Example 5. Let the underlying graph be as in Figure 1, where circle vertices are controlled by Player 1 and diamond vertices are controlled by Player 2. The preference relation of Player 1 is $(ab)^\omega \succ_1 a(ba)^n x^\omega \succ_1 (ab)^n y^\omega$ and the preference relation of Player 2 is $(ab)^\omega \succ_2 (ab)^n y^\omega \succ_2 a(ba)^n x^\omega$ (in particular, both players care only about the tail of the run).

Then $\Gamma_1(a) = \{(ab)^\omega\} \cup \{a(ba)^n x^\omega \mid n \in \mathbb{N}\}$ and $\Gamma_2(a) = [\mathcal{H}]$. Player 1 realizing her guarantee means for her to move to x immediately, thus forgoing any chance of realizing the run $(ab)^\omega$. The Nash equilibrium constructed in the proof of both Theorem 8 and Theorem 12 will be Player 1 moving to x and Player 2 moving to y. Note that in this particular game, the preference of Player 2 has no impact at all on the Nash equilibrium that will be constructed.

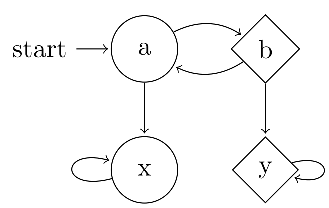


Figure 1: The game for Example 5

Note that the notion of best guarantee for a player does not at all depend on the preferences of the other players; and as such, it is a bit strenuous to consider a strategy realizing the guarantee (or the minimal runs therein) to be optimal in the game at hand (cf. Example 5). This strategy is rather optimal for a player that would play against a coalition of antagonistic opponents, and thus makes sense based on a worst-case assumption about the behaviour of the other players.

Optimality. Strategies may be optimal (in a weak sense as discussed above) either at the beginning of the game or at all histories. This useful game-theoretic concept is rephrased below in terms of best guarantee.

Definition 6. Let s_a be a strategy for some player a in some game g.

- s_a is *optimal*, if $\gamma_a(s_a) = \Gamma_a$.
- s_a is *optimal at history* $h \in \mathcal{H}$, if s_a^h is optimal in g^h.
- s_a is *subgame-optimal*, if it is optimal at all $h \in \mathcal{H}$.
- s_a is *consistent-optimal*, if it is optimal at all $h \in \mathcal{H}(s_a)$.

Note that the notion of optimality is orthogonal to that of determinacy. Especially, the players having optimal strategies does not imply determinacy of the derived threshold games (in an undetermined win/lose game, the guarantee of both players is the set of all runs – hence, any strategy is optimal).

The key Lemma 24 essentially consists in a quantifier inversion, like the one in [6]. Assuming that for all histories there is a finite-memory optimal strategy, we will use them to construct a finite-memory strategy that is subgame-optimal. To know when to use which strategy, we will use the assumption that optimality of a given strategy at an arbitrary history can be decided regularly.

Definition 7. A player a in a game g has the *optimality is regular* (OIR) property, if for all finite-memory strategies s_a for a in g, there exists a finite automaton that decides on input $h \in \mathcal{H}$ whether or not s_a is optimal at h.

A game g has the OIR property, if all the players have it in g.

A preference has the optimality is regular property, if players with this preference have the OIR in all games.

3. Main results

In the remainder of the paper, the games are always played on finite graphs has defined in Section 2, and they always involve colors in C, players in A, and preferences $(\prec_a)_{a\in A}$.

Theorem 8 presents implications, and absence of stated implication is discussed afterwards. For several implications we use the assumption that the set of preferences is closed under antagonism, *i.e.* for all $\prec_a$ there exists $b \in A$ such that $\prec_b = \prec_a^{-1}$.

Theorem 8. Let $(\prec_a)_{a\in A}$ be closed under antagonism. The statements below refer to all the games built with C, A, and $(\prec_a)_{a\in A}$, and the diagram displays implications between the statements.

- **OIR:** Optimality is regular.
- **fm-SOS:** There are finite-memory subgame-optimal strategies.
- **fm-NE:** There are finite-memory Nash equilibrium.
- **FTG-d:** The future threshold games are determined.
- **TG-ufmd** In every game, the threshold games are determined using uniformly finite memory.
- **FTG-ufmd:** In every game, the future threshold games are determined using uniformly finite memory.

$$
\begin{tikzcd}
 & \text{FTG-ufmd} \wedge \text{OIR} \arrow[d, Rightarrow, "1"'] & \\
 & \text{FTG-d} \wedge \text{fm-SOS} \arrow[dl, Rightarrow, "2"] \arrow[dr, Rightarrow, "3"'] & \\
\text{FTG-ufmd} \arrow[dr, Rightarrow, "4"] & & \text{fm-NE} \arrow[dl, Rightarrow, "5"'] \\
 & \text{TG-ufmd} &
\end{tikzcd}
$$

Proof. 1. By Lemma 24.

2. Subgame-optimal strategies in an antagonistic two-player win all future threshold games that are winnable by their player.

3. By Lemma 21.

4. Clear.

5. By Lemma 15.

□

We will show in Subsection 5.2 that neither Implication (1) (Example 32) nor Implication (2) (Example 31) are reversible. For the remaining implications, we do not have answers regarding their reversibility. Regarding Implication (4), note that if we were to fix a specific graph, it would be trivial to separate the two notions. The difficulty lies in the requirement to deal with all finite graphs simultaneously.

Open Question 9. Is there a preference $\prec$ such that all threshold games for $\prec$ are uniformly finite memory determined, but not all future threshold games?

Regarding Implications (3) and (5), the ultimate goal would be precise characterization of the properties of (future) threshold games sufficient and necessary to have finite-memory Nash equilibria, akin to the characterization of the existence of optimal positional strategies by Gimbert and Zielonka [14].

Open Question 10. How much more requirements are needed in addition to **TG-ufmd** to imply **fm-NE**?

Theorem 8 relies on many games having the assumed property, *i.e* the statements are universal quantifications over games. However, a given game might enjoy some property, *e.g.* existence of NE, not only due to the preferences on infinite sequences of colors, but also due to the specific structure of the underlying graph. Theorem 12 below captures this idea. It is a generalization of our main result in [12]. (The new Assumption 2 is indeed weaker.)

Definition 11. Given a game g, a preference $\prec$ is Mont (in g) if for every regular run $h_0\hat{}\rho \in [\mathcal{H}]$ and for every family $(h_n)_{n\in\mathbb{N}}$ of paths in (V, E) such that $h_0\hat{}\dots\hat{}h_n\hat{}\rho \in [\mathcal{H}]$ for all n, if $h_0\hat{}\dots\hat{}h_n\hat{}\rho \prec h_0\hat{}\dots\hat{}h_{n+1}\hat{}\rho$ for all n then $h_0\hat{}\rho \prec h_0\hat{}h_1\hat{}h_2\hat{}h_3\dots$.

Theorem 12. Let g be a game such that

1. For all $a \in A$ the future threshold games for a in g are determined, and each of Player 1 and 2 can win their winnable games via k fixed strategies using T bits each.
2. Optimality for such a strategy is regular using D bits.
3. The preferences in g are Mont.

Then g has a Nash equilibrium made of strategies using $|A|(kD+kT+\log k)+1$ bits each.

Proof. By Lemmata 24 and 22. □

The difference between Theorem 12 and Theorem 8 is similar to the difference between previous works by the authors for games on infinite trees (without memory concern): Namely, between [7] and [6]. The first result characterizes the preferences that guarantee existence of NE for all games; the second result relies on the specific structure of a given game to guarantee existence of NE for the given game.

4. Main proofs

This section organizes the lemmata for the two main results into several paragraphs, depending on the increasing strength of the assumptions.

Lemma without specific extra assumptions:. Lemma 13 below collects basic useful facts on how the guarantees behaves wrt strategies and future games.

Lemma 13. Let g be a game on a graph, let $\prec_a$ be a strict weak order preference for some player a, let $h \in \mathcal{H}$, let s_a be a strategy for a in g^h, let $h' \in \mathcal{H}(s_a)$, and let s'_a be a strategy for a in $g^{h\hat{}h'}$.

1. Then $h'\hat{}\gamma_a(h\hat{}h', s_a^{h'}) \subseteq \gamma_a(h, s_a)$ for all $h' \in \mathcal{H}(s_a)$.

2. If $\gamma_a(h\hat{}h', s'_a) \subsetneq \gamma_a(h\hat{}h', s_a^{h'})$, there exists $\rho \in \gamma_a(h\hat{}h', s_a^{h'})$ such that $\rho \prec_a^{h\hat{}h'} \rho'$ for all $\rho' \in \gamma_a(h\hat{}h', s'_a)$.

3. If $\gamma_a(h\hat{}h', s'_a) \subseteq \gamma_a(h\hat{}h', s_a^{h'})$ then $h'\hat{}\gamma_a(h\hat{}h', s'_a) \subseteq \gamma_a(h, s_a)$.

Proof.

1. Let $\rho \in \gamma_a(h\hat{}h', s_a^{h'})$, so by Definition 4 there exists $\rho' \in [\mathcal{H}(s_a^{h'})]$ such that $\neg(\rho \prec_a^{h\hat{}h'} \rho')$, *i.e.* $\neg(h'\hat{}\rho \prec_a^h h'\hat{}\rho')$. So $h'\hat{}\rho \in \gamma_a(h, s_a)$ since $h'\hat{}\rho' \in [\mathcal{H}(s_a)] \subseteq \gamma_a(h, s_a)$.
2. Let $\rho \in \gamma_a(h\hat{}h', s_a^{h'}) \setminus \gamma_a(h\hat{}h', s'_a)$, so $\rho \prec_a^{h\hat{}h'} \rho'$ for all $\rho' \in \gamma_a(h\hat{}h', s'_a)$ by Definition 4.
3. Let $\rho \in \gamma_a(h\hat{}h', s'_a)$, so $h'\hat{}\rho \in h'\hat{}\gamma_a(h\hat{}h', s_a^{h'})$ by assumption, so $h'\hat{}\rho \in \gamma_a(h, s_a)$ by Lemma 13.1.

□

Using uniformly finite memory in the threshold games of a given *game:.* Lemma 14 below establishes an equivalence between existence of simple optimal strategies and simple ways of winning threshold games, for a given player in a given game. Note that determinacy is not assumed.

Lemma 14. Let F_a be a finite set of strategies for some player a in some game g. The following are equivalent.

1. a has an optimal strategy in F_a for g;
2. a wins each of her winnable threshold games in g via strategies in F_a;
3. a wins each of her winnable non-strict threshold games in g via strategies in F_a.

Moreover replacing ”in F_a” above with ”using m bits of memory” is correct.

Proof. Let us assume 1, so let s_a be such that $\gamma_a(s_a) = \Gamma_a$. By definition of Γ_a there is no s'_a such that $\gamma_a(s_a) \subsetneq \Gamma_a$, so s_a wins all winnable (non-strict) threshold games in g, hence 2 and 3.

Conversely let us assume 2 or 3. For all $\rho \in [\mathcal{H}] \setminus \Gamma_a$, Player 1 has a winning strategy $s_a^\rho \in F_a$ in the (non-strict) threshold game for a and ρ in g. In this case let s_a^ρ be a winning strategy in F_a. By finiteness at least one of the s_a^ρ, which we name s_a, wins the (non-strict) threshold games for all $\rho \notin \Gamma_a$. This shows that $\gamma_a(s_a) \subseteq \Gamma_a$, so equality holds, hence optimality of s_a.

Moreover, a player in a game with n vertices has at most $(n2^m)^{(n2^m)}$ strategies using m bits of memory (by the σ representation). □

In Lemma 14 above, the implication 1. $\Rightarrow$ 2. $\wedge$ 3. holds in a more general context. However, the finiteness of F_a is key to the converse, as well as to the convenient remark that we can sometimes safely remain vague about whether we speak about strict or non-strict thresholds. This and Lemma 14 are used in Lemma 15 for antagonistic games below.

Lemma 15. Let F be a finite set of strategies for players a and/or b in some antagonistic game g. The following are equivalent.

1. g has an NE using strategies in F;
2. the threshold games for a in g are determined via strategies in F;
3. the threshold games for a in g are determined, and each player wins each of her winnable (non-strict) threshold games in g via strategies in F.

Moreover replacing "in F" above with "using m bits of memory" is correct.

Proof. By Lemma 14 the following are equivalent for all propositions D.

- $D \wedge$ each player has an optimal strategy for g in F;
- $D \wedge a$ wins each of her winnable threshold games in g, and b wins each of her winnable non-strict threshold games in g, all via strategies in F;
- $D \wedge$ each player wins each of her winnable (non-strict) threshold games in g via strategies in F.

Let D be "the threshold games for a in g are determined", and let us prove that the above assertions correspond to 1, 2, and 3 in the same order. It is a copy-paste for 3, it is a definition unfolding for 2, so let us focus on 1.

Let s be an NE that uses strategies in F. Let ρ be he run induced by s, and let ρ_t be a threshold. If $\rho_t \prec_a \rho$, Player 1 wins g_{a,ρ_t} by using s_a; otherwise Player 2 wins by using s_b. It shows that D holds. Since ρ is a $\prec_a$-minimum of $\gamma_a(s_a)$, the existence of some s'_a such that $\gamma_a(s'_a) \subsetneq \gamma_a(s_a)$ would contradict s being an NE, so $\gamma_a(s_a) = \Gamma_a$. And likewise $\gamma_b(s_b) = \Gamma_b$.

Conversely let us assume D and let $s_a \in F$ and $s_b \in F$ satisfy $\gamma_a(s_a) = \Gamma_a$ and $\gamma_b(s_b) = \Gamma_b$. First note that $\Gamma_a \cap \Gamma_b$ is non-empty, as witnessed by the run ρ induced by (s_a, s_b). Since $\rho \in \Gamma_a$, Player 1 cannot win $g_{a,\rho}$, so Player 2 wins it, which implies that runs $\prec_a$-greater than ρ are not in Γ_b. Likewise runs $\prec_b$-greater than ρ are not in Γ_a. Therefore elements of $\Gamma_a \cap \Gamma_b$ are $\prec_a$-equivalent, and (s_a, s_b) is an NE. $\square$

Note that Lemmata 14 and 15 can be extended to games in normal form since the two proofs do not use the sequentiality of the game at all.

Definition 16. If the assertions of Lemma 15 hold, we say that the game is finite-memory determined, and its value is the equivalence class $\Gamma_a \cap \Gamma_b$.

Corollary 17. If an antagonistic game is finite-memory determined, its value has a regular witness.

Proof. By Lemma 15 there is a finite-memory NE, whose induced run is regular. $\square$

Using uniformly finite memory in several *threshold games:.* Lemma 18.1 below relies on Lemma 14 to give a rather weak sufficient condition for a given player to have finite-memory best responses in all antagonistic games. Lemma 18.2 relies on Lemma 18.1 and will help us prove subgame-optimality in Lemma 24, by allowing us to restrict our attention to regular runs. For this, reg denotes the infinite sequences of the form $u_1 \dots u_n (w_1 \dots w_k)^\omega$.

Lemma 18. Fix a player a and her preference such that for all one-player games g_a, player a wins her winnable threshold games in g_a using uniformly finite memory. Then in all antagonistic games involving a and b,

1. all finite-memory strategies of b is met with a finite-memory best response by a;
2. for all finite-memory strategies s_b, we have $\mathrm{reg} \cap \gamma_b(s_b) = \mathrm{reg} \cap \Gamma_b$ implies $\gamma_b(s_b) = \Gamma_b$.

Proof. 1. Let g be such a game involving a and b, and let s_b be a finite-memory strategy of b, implemented by σ_b using m bits of memory. Let g' be a defined as follows: the vertices are in $V' := V \times \{0,1\}^m$, for the coloring let $\Lambda'(v,q) := \Lambda(v)$, the preferences are like in g, and $((v,q)(v',q')) \in E'$ iff $(v,v') \in E \wedge q' = \pi_2 \circ \sigma(v,q) \wedge (v \in V_b \Rightarrow v' = \pi_1 \circ \sigma(v,q)))$. Therefore only player a is playing in g', and she can induce exaclty the same color traces as in g when player b plays according to s_b. By uniformity assumption and Lemma 14 player a has a finite-memory optimal strategy s'_a in g'. Using s'_a and s_b one can construct a finite-memory best-response for a to s_b in g: player a feeds the history h in g to s_b and thus keeps track of the corresponding history h' in g'. Player a then uses s'_a to compute a move in g', which corresponds to a unique possible move to be played in g.

2. $\Gamma_b \subseteq \gamma_b(s_b)$ by definition. By Lemma 18.1 let s_a be a finite-memory best response to s_b. The run ρ induced by (s_a, s_b) is therefore a regular $\prec_b$-minimum of $\gamma_b(s_b)$. Since $\mathrm{reg} \cap \gamma_b(s_b) = \mathrm{reg} \cap \Gamma_b$, it is also in Γ_b.

□

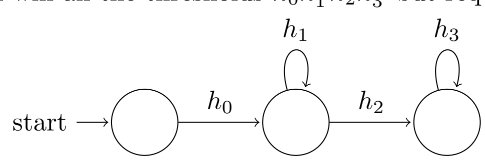
The regular-Mont condition below is a weakening of the Mont condition, by considering only some regular runs.

Definition 19. A preference $\prec$ is regular-Mont if the following holds: for all $h_0, h_1, h_2, h_3 \in C^*$, if $h_0 h_1^n h_2 h_3^\omega \prec h_0 h_1^{n+1} h_2 h_3^\omega$ for all $n \in \mathbb{N}$, then $h_0 h_2 h_3^\omega \prec h_0 h_1^\omega$.

The contraposition of Lemma 20 below will be used in the proof of Lemma 24 to say that suitable uniformity of a preference implies that it is regular-Mont.

Lemma 20. Fix a player a and her preference $\prec_a$. If $\prec_a$ is not regular-Mont, there is a one-player game where a does not win her winnable (non-strict) threshold games with uniformly finite memory.

Proof. Let us assume that $\prec_a$ is not regular-Mont, so let $h_0, h_1, h_2, h_3 \in C^*$ be such that $h_0 h_1^n h_2 h_3^\omega \prec h_0 h_1^{n+1} h_2 h_3^\omega$ for all $n \in \mathbb{N}$, but $\neg(h_0 h_2 h_3^\omega \prec h_0 h_1^\omega)$. So $h_0 h_1^\omega \prec h_0 h_1 h_2 h_3^\omega$, since $\prec_a$ is a strict weak order. In the game below, player a can win all the thresholds $h_0 h_1^n h_2 h_3^\omega$ but requires unbounded finite memory.



□

Lemmata 21 and 22 both show the existence of finite-memory NE. They both assume existence of finite-memory consistent-optimal strategies, but the other assumptions are slightly different. Lemma 21 will be weakened (for the sake of simplicity) to obtain Implication 3 of Theorem 8; whereas Lemma 22, making assumptions only about the given game and its derived games, will be combined to Lemma 24 to obtain Theorem 12. The proofs ot Lemmata 21 and 22 are similar: The beginning is the same and is writen only once; the ends are similar yet not enough to copy-paste it; and the middle parts are clearly different, thus making a factorization of the two lemmata difficult.

Lemma 21. Let g be a game, and let us assume the following.

1. All players have consistent-optimal strategies in g using S bits.
2. For all players a, for all games with n vertices, all (non-strict) threshold games for a are determined and Player 2 wins her winnable ones using $f(n)$ bits.

Then g has a Nash equilibrium made of strategies using $\max(|A|S, f(|V|(1 + 2^{|A|S}))) + 1$ bits each.

Proof. For all a let s_a be a consistent-optimal finite-memory strategy for a in g, and let ρ_{NE} be the run induced by the strategy profile $(s_a)_{a\in A}$. It will become the run induced by the claimed finite-memory NE, once we ensure via finite memory that no player has an incentive to deviate.

By finiteness of memory, $\rho_{NE} \in \text{reg}$, so let l be a lasso arena that corresponds to ρ_{NE}, *i.e.* where the copies of a vertex from g are labeled with the same player and color. Note that the lasso may be chosen with at most $|V|2^{|A|S}$ vertices, since when the ρ_{NE} comes back to a previously visited vertex with all the players having the same memory content, the lasso is cycling.

For every player a, we construct a game g^a as follows: the preferences are as in g, we take the disjoint union of l and the arena of g, and we let g^a start at the start of l. Finally, there is an edge from $v_l \in l$ to $v \in g$ if v_l is controlled by a and if there is one edge in g from v' to v, where v_l is a copy of v'. So the players other than a cannot deviate from l.

$g^a_{a,\rho_{NE}}$ is the threshold game derived from g^a for player a and threshold ρ_{NE}. Since s_a is consistent-subgame optimal, it is optimal at all finite prefixes of ρ_{NE}, *i.e.* $\rho \in \Gamma_a(h)$ for all decompositions $\rho_{NE} = h\hat{}\rho$, so Player 1 loses $g^a_{a,\rho_{NE}}$ (on behalf of player a). This game is finite-memory determined by assumption, so let s_{-a} be a finite-memory winning strategy for Player 2.

Now let s'_a be the following strategy for player a in g: Follow s_a until a player b deviates, in which case play anything positionally if $b = a$, and follow s_{-b} otherwise. This ensures that b cannot get a better outcome by deviating unilaterally, so $(s'_a)_{a\in A}$ is an NE.

Every player uses S bits of memory to follow ρ_{NE}, and she uses $(|A| - 1)S$ bits to know how the others are supposed to play and thus detect when someone has deviated and who. To be able to take part in the coalition "punishing" the deviator, she uses $f(|V|(1 + 2^{|A|S}))$ bits to remember the s_{-a}. Since the two phases of the play, *i.e.* following ρ_{NE} and "punishing" a deviator, are not simultaneous, the memory can be repurposed: altogether $\max(|A|S, f(|V|(1 + 2^{|A|S})))+1$ bits suffice, where the $+1$ is used to remember the current phase. □

Using uniformly finite memory for the future *threshold games of a* given *game:*.

Lemma 22. Let g be a game satisfying the following.

1. All players have consistent-optimal strategies in g using S bits.
2. For each player, her future threshold games in g are determined, and Player 2 wins her winnable ones using k fixed strategies using T bits each.
3. Optimality for each strategy from Assumption 2 is regular using D bits.

Then g has a Nash equilibrium made of strategies using $|A| \max(S, kD+kT)+1$ bits each.

Proof. Let s_a and ρ_{NE} be as in the proof of Lemma 21, and likewise let us build a strategy s_{-a} that prevents a from deviating. For all $a \in A$ let $t^1_{-a}, \dots, t^k_{-a}$ be the strategies from Assumption 2, and for each t^i_{-a} let A_i be an automaton using D bits of memory and telling for which histories t^i_{-a} is optimal. Let s'_a be the following strategy for player a in g: Follow s_a until a player b deviates, in which case play anything positionally if $b = a$, and otherwise take part in the optimal t^i_{-b} of smallest index i.

Consider the future games starting right after deviation of b. Their antagonisitic versions for a against the others have values at most ρ_{NE} wrt $\prec_b$, by optimality of s_b. So choosing the right t^i_{-b} as above ensures that b cannot get a better outcome by deviating unilaterally from ρ_{NE}, so $(s'_a)_{a\in A}$ is an NE.

Every player uses S bits of memory to follow ρ_{NE}, and she uses $(|A|-1)S$ bits to know how the others are supposed to play and thus detect when someone has deviated and who. To be able to take part in the coalition "punishing" the deviator, she uses $(|A|-1)kT$ bits to remember the t^i_{-a} for all a but herself; and she uses $(|A|-1)kD$ bits to know when a t^i_{-a} is optimal. Since the two phases of the play, *i.e.* following ρ_{NE} and "punishing" a deviator, are not simultaneous, the memory can be repurposed: altogether $|A| \max(S, kD+kT)+1$ bits suffice, where the $+1$ is used to remember the current phase. $\square$

Using uniformly finite memory for several future *threshold games:*. Lemma 23 is a complex variant of Lemma 18.1 that considers future games. It is not invoked in our main results but will be useful in Subsection 5.2 to further weaken the assumption that optimality is regular.

Lemma 23. Fix a player a and her preference such that for all one-player games g_a, player a wins her winnable future threshold games in g_a via uniformly finite memory. Then in all antagonistic games involving a and b, for all finite memory strategies s_b, there are finitely many finite-memory strategies $t_1, \dots, t_n$ such that for all histories h, one of the t_i is a best response by a to s_b starting at h.

Proof. (This proof is similar to that of Lemma 18.1.) Let g be such a game involving a and b, and let s_b be a finite-memory strategy of b. Let g' be defined wrt g as in the proof of Lemma 18.1. Only player a is playing in g', so by uniformity assumption let finite-memory strategies $t'_1, \dots, t'_n$ be such that all winnable future threshold games in g' is won by some t'_i. Using t'_i and s_b one can construct a finite-memory strategy t_i : player a feeds the history h in g to s_b and thus keeps track of the corresponding history h' in g'. Player a then uses t'_i to compute a move in g', which corresponds to a unique possible move to be played in g. This defines a strategy t_i in g. Now for all h in g, some t'_i is an optimal strategy for a at h' in g', and the corresponding t_i is is a best response by a to s_b starting at h. $\square$

Main construction:. Lemma 24 below concludes that a player a has a subgame optimal strategy. Assumptions 1 and 2 suffice to construct the candidate strategy, whereas the other assumptions are only used to prove subgame optimality. Assumption 3 is used to build uniformly finite memory best responses by b for all histories. Then Assumption 4a (resp. 4b) allows us to conclude quickly (resp. to continue the proof). Assumptions 3 and 4(b)ii are partly redundant due to our factoring out of two theorems and proofs. Note that determinacy need not hold.

Lemma 24. Let a and b be the players of an antagonistic game g. Let us assume the following:

1. player a wins her winnable future threshold games in g via k strategies using T bits each,
2. Optimality of such a strategy is regular using D bits,

3. players b wins her winnable future threshold games either in g or in every one-player game using uniformly finite memory.
4. Either of the following holds:
   (a) a's preference is Mont;
   (b) i. optimality is regular for b's strategies in g, and
       ii. Each of players a and b wins her winnable threshold games in every one-player game using uniformly finite memory.

Then player a has a subgame-optimal strategy in g using $kD + kT + \log k$ bits.

Proof. Let $t_1, \ldots, t_k$ be the strategies from Assumption 1. By Lemma 14, for all h one of the t_i is optimal at h. By Assumption 2 there exist automata $A_1, \ldots, A_k$ using D bits that decide their respective optimality depending on h.

We define a strategy s for player a as follows: always store the index of one of the t_i, and follow the selected t_i until it ceases to be optimal at some history. Then select an optimal t_j and follow it. Storing the index of the strategy requires $\log k$ bits; simulating all the k strategies in parallel uses kT bits; and deciding optimality kD bits, so $kD + kT + \log k$ bits suffice to implement this strategy.

It remains to show that s is indeed subgame-optimal.

To show that $\gamma_a(h, s^h) = \Gamma_a(h)$ for all h, let $h_0 \in \mathcal{H}$, let $\rho \in \mathcal{H}(s^{h_0})$. In a slight notation overload, for all histories h let t_h be the one strategy t_i that our construction of s follows at h. Also, let $(h_n)_{n\geq 1}$ be such that the n-th change occurring strictly after h_0 occurs at history $h'_n := h_0\hat{}h_1\hat{}\ldots\hat{}h_n$, and let us make a case disjunction on whether $(h_n)_{n\geq 1}$ is finite or infinite. First case, player a changes strategies finitely many times along ρ, say N times. Applying Lemma 13.3 N times yields $h_1\hat{}\ldots\hat{}h_N\hat{}\gamma_a(h'_N, t_{h'_N}^{h'_N}) \subseteq \cdots \subseteq h_1\hat{}\gamma_a(h'_1, t_{h'_1}^{h'_1}) \subseteq \gamma_a(h_0, t_{h_0}^{h_0}) \subseteq \Gamma_a(h_0)$. So $\rho \in \Gamma_a(h_0)$ since $\rho \in h_1\hat{}\ldots\hat{}h_N\hat{}\gamma_a(h'_N, t_{h'_N}^{h'_N})$.

Second case, player a changes strategies infinitely many times along ρ. Regardless of which disjunct of Assumption 3 holds, there are finitely many finite-memory strategies r_i for b in g such that for all h one of the r_i, which we call r_h, is a best-response by b to t_h. So (r_h, t_h) induces a minimum ρ_h of $\Gamma_a(h)$.

Since the t_i and the r_i are finitely many and are finite-memory strategies, the ρ_h are also finitely many, and regular.

Since a changes strategies at h'_{n+1} we have $h'_n\hat{}\rho_{h'_n} \prec_a h'_{n+1}\hat{}\rho_{h'_{n+1}}$ (or $\rho_{h'_n} \prec_a^{h'_n} h_{n+1}\hat{}\rho_{h'_{n+1}}$). By finiteness some ρ_h must occur infinitely often, let ρ' be the constant run of the the corresponding subsequence $\rho_{h'_{\varphi(n)}}$. So $h'_{\varphi(0)}\hat{}\rho' \prec_a \cdots \prec_a h'_{\varphi(n)}\hat{}\rho' \prec_a \dots$. If Assumption 4a holds, a's preference is Mont and therefore $h_0\hat{}\rho' \prec_a \rho$, *i.e.* $\rho \in \Gamma_a(h_0)$ since $h_0\hat{}\rho' \in \Gamma_a(h_0)$.

Let us now deal with the case where Assumption 4b holds. By Lemma 18.2 and Assumption 4(b)ii (for b) we can assume wlog that ρ is regular. By Assumption 4(b)i, suitable r_h can also be chosen via a finite automaton, in which case $h'_{\varphi(n)}$ can be decomposed as h_0uw^n. So $h_0u\hat{}\rho' \prec_a h_0uw\hat{}\rho' \prec_a \cdots \prec_a h_0uw^n\hat{}\rho' \prec_a \dots$. By Assumption 4(b)ii (for a) and contraposition of Lemma 20 we have $h_0u\hat{}\rho' \prec_a \rho$, *i.e.* $\rho \in \Gamma_a(h_0)$ since $h_0\hat{}\rho' \in \Gamma_a(h_0)$. □

5. Discussion

5.1. Comparison to previous work

As mentioned above, a similar but weaker result (compared to our Lemma 22) has previously been obtained by Brihaye, De Pril and Schewe [1],[4, Theorem 4.4.14]. They use cost functions rather than preference relations. Our setting of strict weak orders is strictly more general [3]. However, even if both frameworks are available, it is more convenient for us to have results formulated via preference relations rather than cost functions: Cost functions can be translated immediately into preferences, whereas translating preferences to cost functions is more cumbersome. In particular, it can be unclear to what extend *nice* preferences translate into *nice* cost functions. Note also that prefix-linearity for strict weak orders is more general than prefix-linearity for cost functions. We

[3] For example, the lexicographic combination of two payoff functions can typically not be modeled as a payoff function, as $\mathbb{R} \times \{0,1\}$ (with lexicographic order) does not embed into $\mathbb{R}$ as a linear order, cf. Subsection 6.2.

will see in Subsection 5.2 that prefix-linearity implies the optimality-is-regular property by a very simple argument.

As a second substantial difference, [4, Theorem 4.4.14] requires either prefix-independent cost functions and finite-memory determinacy of the induced threshold games, or prefix-linear cost functions and optimal positional strategies in the induced antagonistic games. In particular,[4, Theorem 4.4.14] cannot be applied to bounded energy parity games, where finite prefixes of the run do impact the overall value for the players, and where at least the protagonist requires memory to execute a winning strategy.

Before [4, 1], it had already been stated by Paul and Simon [15] that multi-player multi-outcome Muller games have Nash equilibria consisting of finite memory strategies. As (two-player win/lose) Muller games are finite-memory determined [16], and the corresponding preferences are obviously prefix independent, this result is also a consequence of [4, Theorem 4.4.14]. Another result subsumed by [4, Theorem 4.4.14] (and subsequently by our main theorem) is found in [17] by Brihaye, Bruyère and De Pril.

5.2. Exploring optimality is regular

We shall discuss the optimality-is-regular property and its relationship to some other, established properties of preferences. We will in particular show that it is not dispensable in our main theorem (Example 31), as in its absence, uniform finite-memory determinacy no longer implies the existence of finite memory subgame-optimal strategies. On the other hand, the optimality-is-regular property is also not necessary, as shown by Example 32.

Recall that a preference relation $\prec \subseteq [\mathcal{H}] \times [\mathcal{H}]$ is called *prefix-linear*, if $\rho \prec \rho' \Leftrightarrow h\hat{}\rho \prec h\hat{}\rho'$ for all $\rho, \rho', h\hat{}\rho \in [\mathcal{H}]$. It is *prefix-independent*, if $\rho \prec \rho' \Leftrightarrow h\hat{}\rho \prec \rho'$ and $\rho' \prec \rho \Leftrightarrow \rho' \prec h\hat{}\rho$ for all $\rho, \rho', h\hat{}\rho \in [\mathcal{H}]$. Clearly, a prefix-independent preference is prefix-linear.

As a further generalization, we will consider *automatic-piecewise prefix-linear* preferences $\prec$. Here, there is an equivalence relation on $\mathcal{H}$ with equivalence classes (pieces for short) in $\overline{\mathcal{H}}$ and satisfying three constraints: First, the histories in the same piece end with the same vertex. Second, there exists a deterministic finite automaton, without accepting states, that reads histories and such that two histories are equivalent iff reading them leads to the same states. Third, for all $h\hat{}\rho, h\hat{}\rho', h'\hat{}\rho, h'\hat{}\rho' \in [\mathcal{H}]$, if $\overline{h'} = \overline{h} \in \overline{\mathcal{H}}$, then $h\hat{}\rho \prec h\hat{}\rho' \Leftrightarrow h'\hat{}\rho \prec h'\hat{}\rho'$.

The extension to automatic-piecewise prefix-linear preferences ensures that e.g. safety and reachability games are also covered. In fact, most of the common payoff functions considered in the literature give rise to automatic-piecewise prefix-linear preferences. Examples include mean-payoff, discounted payoff, Muller, mean-payoff parity and bounded energy parity games (see Subsection 6.1 below for the latter).

Of the popular winning conditions, many are actually prefix-independent, such as parity, Muller, mean-payoff, cost-Parity, cost-Streett [18], etc. Clearly, any combination of prefix-independent conditions itself will be prefix-independent. Typical examples of non-prefix independent, but prefix-linear conditions are reachability, energy, and discounted payoff. Combining prefix-linear conditions not necessarily yields another prefix-linear condition. However, we can easily verify that combining a reachability or energy condition with any prefix-linear condition yields an automatic-piecewise prefix-linear condition (provided that energy is bounded).

Proposition 25. Automatic-piecewise prefix-linear preferences have the optimality-is-regular property.

Proof. Assume automatic-piecewise prefix-linear preferences. Whether a finite memory strategy is optimal at some history depends only on its memory content at that history, and on the piece the history falls into. □

Proposition 26. A preference $\prec$ is automatic-piecewise prefix-linear iff there is a finite set $\mathcal{M}$ of regular sets $M \subseteq \mathcal{H}$ such that:

$$\forall p, q \in C^\omega \quad \exists M \in \mathcal{M} \quad \forall h \in \mathcal{H} \quad (h \in M \Leftrightarrow h\hat{}p \prec h\hat{}q)$$

Proof. If $\prec$ is automatic-piecewise prefix-linear, then the pieces satisfy the property of $\mathcal{M}$. Conversely, given some finite set $\mathcal{M}$ with the given property, we can consider the finest partition where each part is some boolean combination of elements of $\mathcal{M}$. This partition is automatic, and witnesses automatic-piecewise prefix linearity $\prec$. $\square$

Definition 27. We say that $\prec$ has the *weak oir* property if

$$\forall p, q \in C^\omega \cap \mathrm{reg} \quad \exists M \in \mathrm{Reg} \quad \forall h \in \mathcal{H} \quad (h \in M \Leftrightarrow h\hat{}p \prec h\hat{}q)$$

where Reg are the regular subsets of C^*.

Observation 28. The OIR property implies the weak OIR property.

Proof. Given regular p and q, we can construct finite graphs P and Q, such that the only infinite run through P has colors p, and the only infinite run through Q has colors q. We further use a clique K where all colors appear, controlled by a different player. Then we merge them together as follows: The game starts in K, from where a choice vertex v controlled by the protagonist can be reached. The vertex v has two outgoing edges, to P and to Q.

Now we use the OIR property on the constant strategy going to P. At some history h ending in v, we find that $h\hat{}p \prec h\hat{}q$ iff this constant strategy is not optimal at h. By using the construction with all finitely many different choices for the color of v, we obtain the full claim. $\square$

Proposition 29. Let g be an antagonistic game involving a and b, and let us assume the following.

1. for all one-player games g_b, player b wins her winnable future threshold games in g_b via uniformly finite memory.
2. Player a wins her winnable future threshold games in g using uniformly finite memory.
3. $\prec_a$ has the weak OIR property

Then $\prec_a$ has also the OIR property.

Proof. Let s_a be a finite-memory strategy for a in some antagonistic game g involving a and b. We want to construct an automaton that decides whether s_a

is optimal at some input history. By Assumption 1 and Lemma 23, there are finitely many finite-memory strategies s_b^i such that for every history h, one of the s_b^i is a best response by b to s_a at h. Depending on the h from which we start, playing some fixed s_b^i against s_a yields a tail from some finitely many tails p_{ij}. Moreover we can decide which j from h (and i) in an automatic way.

By Assumption 2 and Lemma 14 there are finitely many finite-memory strategies t_a^k for a such that for all histories h, one of the t_a^k is optimal at h. By Assumption 1 and Lemma 23 again, there are finitely many finite-memory strategies t_b^l such that for all histories h and strategies t_a^k, one of the t_b^l is a best response to t_a^k. Depending on the h from which we start, playing some fixed t_b^j against t_a^i yields a tail from some finitely many tails q_{klm}. Moreover we can decide which m from h (and k, l) in an automatic way.

Now, s_a is not optimal at some history h iff one of the t_a^i does better at h, which is equivalent to

$$\min_i h\hat{}p_{ij(i,h)} \prec_a \max_k \min_l h\hat{}q_{klm(k,l,h)}$$

By invoking the weak OIR property for the finitely many tails p_{ij} and q_{klm}, and using that j and m depend on h in an automatic way, we can test this property with a finite automaton. $\square$

Corollary 30. If all future threshold games are uniformly finite-memory determined, then $\prec$ has the weak OIR property iff it has the OIR property.

Example 31. Fix some non-regular set $A \subseteq \mathbb{N}$. We define a payoff function $P_A : \{0,1\}^\omega \to \mathbb{N} \cup \{+\infty\}$ as follows:

$$P_A(p) = \begin{cases} +\infty & p = 0^\omega \\ 2n+1 & \exists q \in \{0,1\}^\omega \ (p = 0^n 10q \wedge n \in A) \vee (p = 0^n 11q \wedge n \notin A) \\ 2n & \exists q \in \{0,1\}^\omega \ (p = 0^n 10q \wedge n \notin A) \vee (p = 0^n 11q \wedge n \in A) \end{cases}$$

The induced preference $\prec_A$ is given by $p \prec_A q$ iff $P_A(p) < P_A(q)$.

Claim: The threshold games for $\prec_A$ are uniformly finite-memory determined.

Proof. If the protagonist can win the safety game for staying on vertices colored 0, he has a positional strategy for doing so. This strategy would win all winnable threshold games on that graph. If he can not win that game, then the opponent can force a 1 within $k \leq n$ moves (where n is the size of the graph), and she can do so positionally. In particular, the protagonist loses all threshold games for thresholds $0^j 11q$ or $0^j 10q$ for $j > k$. As all threshold games here have ω-regular winning conditions, and there are only finitely many cases left, uniform finite-memory determinacy follows. □

Claim: The games built with $\prec_A$ do not have the optimality-is-regular property, and the player with preference $\prec_A$ does not always have a finite-memory subgame-optimal strategy.

Proof. Consider the game graph depicted in Figure 2, with the protagonist (with preference $\prec_A$) controlling the diamond vertex and the opponent the circle vertices. Further, consider the positional strategy where the protagonist always goes to the vertex labeled 0. If there were an automaton that decides whether this strategy is optimal after some history, then by applying this automaton to histories of the form $0^n 1$ allows us to decide whether $n \in A$, a contradiction to the choice of A being non-regular. Similarly, by inspecting the choice a finite-memory subgame-optimal strategy of the player makes after some history of the form $0^n 1$ allows us to decide whether $n \in A$, again a contradiction. □

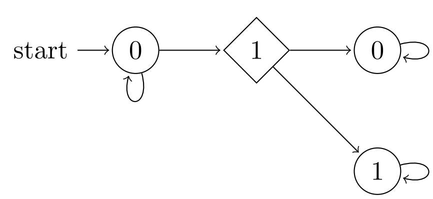


Figure 2: The graph for the game in Examples 31 and 32

Example 32. Fix some non-regular set $B \subseteq \mathbb{N}$. We define a payoff function

$P_B : \{0,1\}^\omega \to \mathbb{N} \cup \{-\infty, +\infty\}$ as follows:

$$P_B(p) = \begin{cases} +\infty & p = 0^\omega \\ 2n+1 & n \in B \wedge \exists q \in \{0,1\}^\omega \quad p = 0^n 10q \\ 2n & n \notin B \vee \exists q \in \{0,1\}^\omega \quad p = 0^n 11q \end{cases}$$

The induced preference $\prec_B$ is given by $p \prec_B q$ iff $P_B(p) < P_B(q)$.

Claim: The player with preference $\prec_B$ always has a finite-memory subgame-optimal strategy.

Proof. The player first plays a safety game where he tries to stay on vertices colored 0 as long as possible. If a 1 is ever reached, and the player can choose, he will go a vertex colored 0 in the next step. □

Claim: The games built with $\prec_B$ do not have the optimality-is-regular property.

Proof. Consider the again game graph depicted in Figure 2, with the protagonist (with preference $\prec_B$) controlling the diamond vertex and the opponent the circle vertices. Further, consider the positional strategy where the protagonist always goes to the vertex labeled 1. If there were an automaton that decides whether this strategy is optimal after some history, then by applying this automaton to histories of the form $0^n 1$ allows us to decide whether $n \in B$, a contradiction to the choice of B being non-regular.

□

5.3. On the Mont condition

If we consider *all* games on finite graphs involving a certain set of preferences, we saw by Lemma 20 that the regular-Mont condition comes for free in our setting. If we explicitly assume the regular-Mont condition to hold, it suffices on the other hand to make assumptions merely about games played on some fixed graph. We shall now give an example that shows that merely assuming optimality is regular and the uniform finite-memory determinacy of the future threshold games played on a given graph does not suffice to conclude the existence of Nash equilibria.

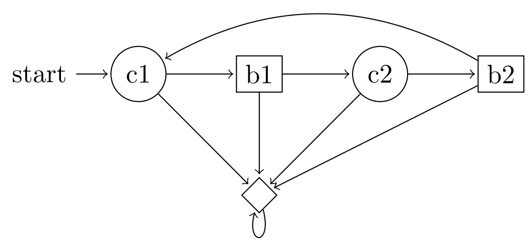


Figure 3: The graph for the game in Example 33

Example 33 ([4])**.** The game g in Figure 3 involves Player 1 (2) who owns the circle (box) vertices. Who owns the diamond is irrelevant. The payoff for Player 1 (2) is the number of visits to a box (circle) vertex, if this number is finite, and is -1 otherwise.

Claim: All future threshold games in g are determined via positional strategies.

Proof. Let s_1 be the positional strategy where Player 1 chooses $b1$ when in $c1$ and the diamond when in $c2$, let s_2 be the positional strategy where Player 1 chooses the diamond in $c1$ and b_2 in c_2. Let s_∞ be the positional strategy where Player 1 always chooses the diamond. Likewise, let t_1 be the positional strategy of Player 2 going to c_2 in b_1 and to diamond in b_2, let t_2 go to diamond in b_1 and to c_1 in b_2, and let t_∞ always go to the diamond.

Consider some history h ending in $c1$, which has seen n occurrences of box vertices. By playing s_1, Player 1 can win the future threshold game starting after h for all threshold $k \leq n+1$. His opponent can win for all higher thresholds by playing t_2. By symmetry of the game, we see that also for histories ending in $c2$, $b1$ or $b2$, the player can win all winnable future threshold games using

[4]This example is based on an example communicated to the authors by Axel Haddad and Thomas Brihaye, which in turn is based on a construction in [19].

one of s_1, s_2, t_1, t_2, and his opponent can win the remaining ones using the counterpart strategy. □

Claim: The game g has no Nash equilibrium (so in particular, neither optimal strategies nor finite memory Nash equilibrium).

Proof. In the run induced by a putative NE, one of the players has to choose the diamond at some point (to avoid payoff -1), but by postponing this choice to next time, the player can increase her payoff by 1. □

5.4. On uniform finite-memory determinacy

We have seen that the existence of a uniform memory bound sufficient to win all winnable threshold games is necessary for the existence of finite-memory Nash equilibria. A typical example for a class of games failing this condition is found in mean-payoff parity games (which, being prefix-independent, have the optimality-is-regular property). The same example, however, also works as a discounted-payoff parity game.

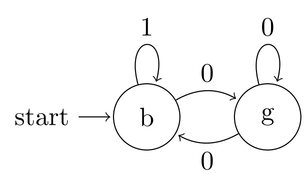


Figure 4: The graph for Example 34

Example 34. Let g be the one-player game in Figure 4. The payoff of a run that visits the vertex g infinitely often is the limit (inferior or superior) of the average payoff. It is zero if g is visited finitely many times only. For any threshold $t \in \mathbb{R}$, if $t < 1$, the player has a winning finite-memory strategy: cycle p times in b, where $p > \frac{1}{1-t}$, visit g once, cycle p times in b, and so on. If $t \geq 1$, the player has no winning strategy at all. So the thresholds games of g, and likewise for the future game of g, are finite-memory determined. The game has no finite-memory Nash equilibrium nonetheless, since the player can get a payoff as closed to 1 as she wants, but not 1.

Uniform finite-memory determinacy can sometimes be recovered by considering ε-versions instead: We partition the payoffs in blocks of size ε, and let the player be indifferent within the same block. Clearly any Nash equilibrium from the ε-discretized version yields an ε-Nash equilibrium of the original game. If the original preferences were prefix-independent, the modified preferences still are. Moreover, as there are now only finitely many relevant threshold games per graph, their uniform finite-memory determinacy follows from mere finite-memory determinacy. In such a situation, our result allows us to conclude that multi-player multi-outcome games have finite memory ε-Nash equilibria. For example, we obtain:

Corollary 35. Multi-player multi-outcome mean-payoff parity games have finite memory ε-Nash equilibria.

Here a multi-player multi-outcome mean-payoff parity game is understood to be a game where each player has payoff labels associated with it, and each vertex some priority. Players have some strict weak order preference on the pairs of the $\limsup$ or $\liminf$ of their average payoff on a prefix and the least priority seen infinitely often, which is consistent with the usual order on the payoff component (i.e. getting more payoff while keeping the same priority is always better). A ε-Nash equilibrium is one where no player can improve by changing the priority, and no player can improve their payoff by more than ε.

6. Applications

We shall briefly mention two classes of games covered by our main theorem, but not by the results from [4, 1], (bounded) energy parity games and games with the lexicographic product of mean-payoff and reachability preferences. We leave the investigation whether winning conditions defined via LTL[$\mathcal{F}$] or LTL[$\mathcal{D}$] formulae [20] match the criteria of Theorem 8 to future work. Another area of prospective examples to explore are multi-dimensional objectives as studied e.g. in [21, 22].

6.1. Energy parity games

Energy games were first introduced in [23]: Two players take turns moving a token through a graph, while keeping track of the *current energy level*, which will be some integer. Each move either adds or subtracts to the energy level, and if the energy level ever reaches 0, the protagonist loses. These conditions were later combined with parity winning conditions in [5] to yield energy parity games as a model for a system specification that keeps track of gaining and spending of some resource, while simultaneously conforming to a parity specification.

In both [23] and [5] the energy levels are a priori not bounded from above. This is a problem for the applicability of Theorem 12, since unbounded energy preferences do not have the optimality-is-regular property, as shown in Example 36 below.

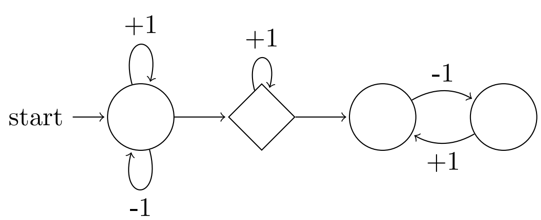


Figure 5: The graph for Example 36

Example 36. Consider the game depicted in Figure 5. The protagonist controls the diamond vertex, the opponent the circle vertices. Energy deltas are denoted on the edges.

Claim: There is no finite automaton that decides whether the strategy of the protagonist that goes right straight-away on reaching diamond is optimal after some history in the energy game.

Proof. This strategy is optimal provided that the current energy level is not equal to the least energy level ever reached. In that case, taking the self-loop at the diamond vertex once and then going right would be preferable. But deciding whether the current energy level is equal to the least energy level essentially requires counting, and can thus not be done by a finite automaton. □

In [24], two versions of bounded energy conditions were investigated: Either any energy gained in excess of the upper bound is just lost (as in e.g. recharging a battery), or gaining energy in excess of the bound leads to a loss of the protagonist (as in e.g. refilling a fuel tank without automatic spill-over prevention). We are only concerned with the former, and define our multi-player multi-outcome version as follows:

Definition 37. A multi-player multi-outcome energy parity game (MMEP game) is a game where each color is a tuple of ordered pairs in $\mathbb{Z} \times \mathbb{N}$, one pair for each player. The first (second) component of a pair is called an energy delta (a priority), noted δ^a_v (π^a_v).

The preferences are described as follows. Each player a has an upper energy bound $E^a_{\max} \in \mathbb{N}$. The cumulative energy values E^a_n for player a in a run $\rho = v_0v_1\dots$ are defined by $E^a_0 := \min(E^a_{\max}, \delta^a_{v_0})$ and $E^a_{n+1} := \min\{E^a_{\max}, E^a_n + \delta^a_{v_{n+1}}\}$. Player a only cares about the least priority occurring infinitely many times together with $E^a = \min_{n\in\mathbb{N}} E^a_n$. He has some strict weak order $\prec_a$ on such pairs, which must respect $E < E' \Rightarrow (\pi, E') \not\prec_a (\pi, E)$.

The threshold games arising from MMEP games are the disjunctive bounded-energy parity games, which we define next:

Definition 38. A disjunctive bounded-energy parity game is a two-player win/lose game where colors are ordered pairs in $\mathbb{Z} \times \mathbb{N}$.

The winning condition is defined as follows. Let $E_{\max} \in \mathbb{N}$. The cumulative energy values E_n in a run $\rho = v_0v_1\dots$ are defined by $E_0 := \min(E_{\max}, \delta_{v_0})$ and $E_{n+1} := \min\{E_{\max}, E_n + \delta_{v_{n+1}}\}$.

Given a run $\rho = v_0v_1\dots$, we consider two values: The least priority π occurring infinitely many times, and the least cumulative energy value reached, $E = \min_{n\in\mathbb{N}} E_n$. The winning condition is given by some family $(E^i_{\min}, P_i)_{i\in I}$ of energy thresholds and sets of priorities. Player 1 wins iff there is some $i \in I$ with $E > E^i_{\min}$ and $\pi \in P_i$. We will write $B_{\min} = \min_{i\in I} E^i_{\min}$.

The usual bounded-energy parity games are the special case where $|I| = 1$ (one can then of course easily rearrange the priorities such that the ones in P_i

are even, and the others odd). Some remarks on the relevance of disjunctions of winning conditions follow below on Page 38.

Theorem 39. Disjunctive bounded-energy parity games are finite-memory determined, and $\log |E_{\max} - B_{\min}|$ bits of memory suffices for Player 1, and $2 \log |E_{\max} - B_{\min}|$ bits for Player 2.

Proof. We take the product of V with the set $Q := \{B_{\min}, \dots, E_{\max}\} \times \{B_{\min}, \dots, E_{\max}\}$, where there is an edge from (v, e_0, e_1) to (u, e'_0, e'_1) iff there an edge from v to u in the original graph, $e'_0 := \min\{E_{\max}, \max\{B_{\min}, e_0 + \delta_v\}\}$ and $e'_1 = \min\{e_1, e'_0\}$. Essentially, we keep track of both the current energy level and of the least energy level ever encountered as part of the vertices. Note that there never can be an edge from some (v, e_0, e_1) to (u, e'_0, e'_1) where $e'_1 > e_1$.

Let $T_k := \{\pi \mid \exists i \; \pi \in P_i \wedge E^i_{\min} \leq B_{\min} + k\}$. Now we first consider the subgraph induced by the vertices of the form $(v, e_0, B_{\min})$; and then the parity game played on this subgraph with winning priorities T_0. As parity games admit positional strategies that win from every possible vertex, we can fix such strategies for both players on the subgraph. Let A^0_0 be the set of vertices in this subgraph where Player 1 wins, and let B^0_0 be the set of vertices in this subgraph where Player 2 wins.

We proceed by a reachability analysis: Let A^{i+1}_0 be A^i_0 together with all vertices controlled by Player 1 that have an outgoing edge into A^i_0, and all vertices controlled by Player 2 where all outgoing edges go into A^i_0. We extend the positional strategies of the players to A^{i+1}_0 by letting Player 1 pick some witnessing outgoing edge, and Player 2 some arbitrary edge. Likewise, we define B^{i+1}_0 be B^i_0 together with all vertices controlled by Player 1 where all going edges go into B^i_0, and all vertices controlled by Player 2 with some outgoing edge into B^i_0, and extend the strategies analogously.

It remains to define the strategies in the subgraph induced by $V \setminus (A^{|V|}_0 \cup B^{|V|}_0)$. In the next stage, we consider the subgraph of this subgraph induced by the vertices of the form $(v, e_0, B_{\min} + 1)$, and again consider a parity game played there, this time with winning priorities in T_1, and so on.

Iterating the parity-game and reachability analysis steps will yield positional optimal strategies for both players on the whole expanded graph.

Now consider the winning sets and strategies of both players: If Player 1 wins from some vertex (v, e_0, e_1), then he also wins from any (v, e_0, e_1') where $e_1 < e_1'$ – for the only difference is the lowest energy level ever reached, which can only benefit, but not harm, Player 1. Moreover, as (v, e_0, e_1') cannot be reached from (v, e_0, e_1) at all, Player 1 can safely play the same vertex u in the original graph at (v, e_0, e_1') as he plays at (v, e_0, e_1). For fixed v, e_0, let e_1 be minimal such that Player 1 wins from (v, e_0, e_1). Then we can change his strategy such that he plays the same vertex in the underlying graph from any (v, e_0, e_1').([5])

By using $\log |E_{\max} - B_{\min}|$ bits of memory, Player 1 can play his positional strategy from above in the original game. Likewise, Player 2 can play her positional strategy from the expanded graph in the original game using $2 \log |E_{\max} - B_{\min}|$ bits of memory. □

We need one last simple lemma, and then will be able to apply Theorem 12 to energy parity games.

Lemma 40. The valuation-preference combinations in MMEP games are automatic-piecewise prefix-independent with at most nE^2 pieces, where n is the size of the graph and E is the maximum difference between the energy maximum and the energy minimum for some player.

Proof. The pieces are defined by the current vertex, the current energy level, and the least ever energy level, i.e. the values E_n^a and $\min_{j \leq n} E_n^a$. As energy is bounded, both enrgy levels can easily be computed by a finite automaton. If h and h' end with the same vertex and share the same current energy level and least energy level, then the least energy level reached in hp is equal to the

[5]This trick does not work for Player 2, because we would need to consider the maximal e_1 where she wins (instead of the minimal one for Player 1), but then the backward induction from the middle of the proof goes in the "wrong direction".

least energy level reached in $h'p$. As the least priority seen infinitely many times depends only on the tail, but never on the finite prefix, we see that for h and h' being the same piece, hp and $h'p$ are interchangeable for the player. □

Corollary 41. All multiplayer multioutcome energy parity games have Nash equilibria in finite memory strategies. Let A be the set of players, n the size of the graph and let E be the maximum difference between the energy maximum and the energy minimum for some player. Then $2|A|E^2 n \log E^2 n + 1$ bits of memory suffice.

Proof. By Theorem 39 and Lemma 40 the prerequisites of Theorem 12 are given. The Mont condition is trivially true, as there are no infinite ascending chains in the preferences. From Theorem 39 we see that the parameter T in Theorem 12 can be chosen as $2 \log E$. By Lemma 40 we can chose $D = \log E^2 n$ and $k = E^2 n$. First note that the claim holds for $n = 1$ or $E = 1$ (by positional determinacy of parity games). Second, for $2 \leq n$ and $2 \leq E$ we have $\log nE^2 \leq nE^2 \log n$, so straightforward calculus shows the general claim. □

Algorithmic considerations

The proof of Theorem 39 immediately gives rise to an algorithm computing the winning strategies in disjunctive bounded-energy parity game while using an oracle for winning strategies in parity games. Using e.g. the algorithm for solving parity games from [25], which has a runtime of $n^{O(\sqrt{n})}$, we obtain a runtime of $(nE)^{O(\sqrt{nE})}$, if we set $E := |E_{\max} - B_{\min}|$. Unfortunately, only the binary representation of E will need to be present in the input – E itself can easily be exponential in the size of the input.

If we assume W to be fixed[6], we arrive at a Cook reduction of solving disjunctive bounded-energy parity games to solving parity games. This in particular implies that the decision problem for disjunctive bounded-energy parity games with bounded weights is in $\mathrm{P}^{(\mathrm{UP} \cap \text{co-UP})}$.

[6] Which is poly-time equivalent to W being given in unary.

Disjunctions of winning conditions

Parity and Muller conditions are easily seen to be closed under conjunction and disjunction, as these just correspond to intersection and union respectively of the relevant sets of winning priorities, or sets of winning sets of vertices visited infinitely many often. As long as just a single notion of energy (or respectively payoff) is available, likewise energy, mean-payoff and discounted payoff conditions are closed under conjunction and disjunction, as here these logical connectives just correspond to minimum and maximum on the threshold values. Subsequently, despite the high relevance of boolean operations on winning conditions, it is unsurprising that they have received little attention in the literature so far.

For energy parity conditions, which are themselves a conjunction of parity and energy conditions, the considerations above immediately imply closure under conjunction. The disjunction of two energy parity conditions, however, is not necessarily equivalent to an energy parity condition. In fact, we even see a qualitative difference with respect to the memory requirements for Player 2: It was shown by Chatterjee and Doyen that in an (unbounded) energy parity game, if Player 2 has a winning strategy, she has a positional one. This translates directly to the corresponding result for bounded energy parity games. For disjunctive energy parity games, Player 2 might require memory to win though:

Example 42. In the energy parity game depicted in Figure 6, Player 2 controls all vertices. Player 1 wins with any priority, if the energy stays above 5, and wins with priority 0 if the energy stays above 0. Each vertex is marked with π/δ, where π is the priority and δ the energy delta.

Any positional strategy of Player 2 is winning for Player 1, but by e.g. alternating the sole choice she has, Player 2 can win using one bit of memory.

6.2. Lexicographic product of Mean-payoff and reachability

In our second example, each player has both a mean-payoff goal and a reachability objective. Maximizing the mean-payoff, however, takes precedence, and the reachability objective only becomes relevant as a tie-breaker; i.e. we consider

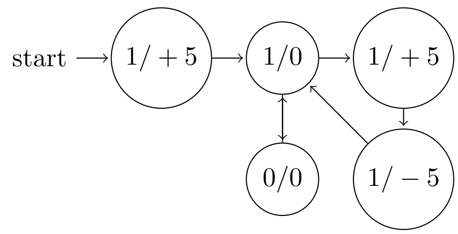


Figure 6: The graph for the game in Example 42

the lexicographic combination of the mean-payoff preferences with the reachability objective. These preferences are of particular interest as they cannot be expressed via a payoff function[7], hence are an example for why considering preferences instead of payoff functions is useful.

Proposition 43. The lexicographic product of mean-payoff preferences and a reachability objective cannot be expressed as a payoff function.

Proof. It is straight-forward to construct an example of a game where any mean payoff in $[0, 1]$ in any combination with reaching or not reaching the reachability objective is realizable. If there were an equivalent payoff function for this game, it would induce an order embedding ι of $[0, 1] \times_{\text{lex}} \{0, 1\}$ into $\mathbb{R}$. As $(x, 0) \prec (x, 1)$ for any $x \in [0, 1]$, we would find that $\iota(x, 0) < \iota(x, 1)$. Thus, there has to be some rational q_x with $\iota(x, 0) < q_x < \iota(x, 1)$. Moreover, as $(x, 1) \prec (y, 0)$ for $x < y$, we find that $q_x \neq q_y$ for $x \neq y$. But then $x \mapsto q_x$ would be an injection from $[0, 1]$ to $\mathbb{Q}$, which cannot exist for reasons of cardinality. $\square$

To deal with lexicographic products, the following will be very useful:

Lemma 44. The weak OIR property is preserved by lexicographic products.

Proof. Let $\prec_1$ and $\prec_2$ have the OIR property, and let $p, q \in C^\omega \cap \text{reg}$. By assumption there exist regular sets M_1, M_1', and M_2 such that for all $h \in \mathcal{H}$ we

[7] The von Neumann Morgenstern utility theorem [26] does not apply, as the continuity axiom is not satisfied.

have $(h \in M_1 \Leftrightarrow h\hat{}p \prec_1 h\hat{}q)$ and $(h \in M_1' \Leftrightarrow h\hat{}q \prec_1 h\hat{}p)$ and $(h \in M_2 \Leftrightarrow h\hat{}p \prec_2 h\hat{}q)$. So $h\hat{}p(\prec_1 \times_{lex} \prec_2)h\hat{}q$ iff $h \in M_1 \cup (M_2 \setminus M_1')$. $\square$

By Proposition 29, it only remains to show that the induced threshold games are uniformly finite-memory determined. We show a slightly stronger result:

Lemma 45. In the threshold games where the preferences are the lexicographic product of mean-payoff and reachability, either Player 1 has a winning strategy using one bit of memory, or Player 2 has a positional winning strategy.

Proof. If the threshold is of the form $(x, 1)$, then the reachability component is irrelevant, and the game is equivalent to the threshold game of a mean-payoff game. As these are positionally determined, the claim is immediate.

If the threshold is of the form $(x, 0)$, then Player 1 has two ways of winning: Either get mean-payoff strictly more than x, or reach the target set and obtain a mean-payoff of at least x. We can consider for each vertex the value of the mean-payoff game starting there. Player 1 has a positional strategy that obtains more than x mean-payoff from all vertices where this is possible, and this strategy ensures that this region is never left. If this region includes the starting vertex, we are done. Otherwise, note that Player 1 cannot enter this region from the outside, and Player 2 has no incentive to ever enter this region from the outside. Thus, we can restrict our attention to the game played on the induced subgraph on the complement.

In the remaining game, we consider those vertices in the target set where the mean-payoff obtainable is x. From all vertices where Player 1 can force the play to reach one of these, he can win with a strategy using a single bit of memory: Play towards such a vertex in a positional way, then flip the bit and follow a positional strategy ensuring mean-payoff at least x. Again, if the starting vertex is covered, we are done. Else, note that Player 1 cannot reach the region from the outside, and Player 2 has no incentive to.

In the game remaining after the second step, we again compute the obtainable mean-payoff values. Player 2's refusal to enter the region in round 2 might increase the mean-payoff Player 1 can obtain above x, and thus let him win after all. In any case, if both players follow positional optimal strategies for mean-payoff games in the remaining part, they will win the mean-payoff plus reachability game if they can at all. $\square$

Corollary 46. The multi-player multi-outcome games where preferences are lexicographic products of mean-payoff and reachability objectives have finite-memory Nash equilibria.

Acknowledgements

We are in debt to Nathanaël Fijalkow for presenting our work on our behalf at the Strategic Reasoning Workshop, as well as to Axel Haddad and Thomas Brihaye for a very useful conversation.

References

[1] T. Brihaye, J. De Pril, S. Schewe, Multiplayer cost games with simple Nash equilibria, in: Logical Foundations of Computer Science, LNCS, 2013, pp. 59–73. doi:10.1007/978-3-642-35722-0_5.

[2] N. Bulling, V. Goranko, How to be both rich and happy: Combining quantitative and qualitative strategic reasoning about multi-player games (extended abstract), in: Proc. of Strategic Reasoning, 2013. doi:10.4204/EPTCS.112.8.
URL http://www.arxiv.org/abs/1303.0789

[3] O. Kupferman, On high-quality synthesis, in: S. A. Kulikov, J. G. Woeginger (Eds.), 11th International Computer Science Symposium in Russia, CSR 2016, Springer International Publishing, 2016, pp. 1–15. doi:10.1007/978-3-319-34171-2_1.

[4] J. D. Pril, Equilibria in multiplayer cost games, Ph.D. thesis, Université de Mons (2013).

[5] K. Chatterjee, L. Doyen, Energy parity games, Theor. Comput. Sci. 458 (2012) 49–60. doi:10.1016/j.tcs.2012.07.038.

[6] S. Le Roux, A. Pauly, Infinite sequential games with real-valued payoffs, in: CSL-LICS '14, ACM, 2014, pp. 62:1–62:10. doi:10.1145/2603088.2603120.

[7] S. Le Roux, Infinite sequential Nash equilibria, Logical Methods in Computer Science 9 (2). doi:10.2168/LMCS-9(2:3)2013.

[8] J. F. Mertens, Repeated games, in: Proc. Internat. Congress Mathematicians, American Mathematical Society, 1987, pp. 1528–1577.

[9] S. Le Roux, A. Pauly, Weihrauch degrees of finding equilibria in sequential games, in: A. Beckmann, V. Mitrana, M. Soskova (Eds.), Evolving Computability, Vol. 9136 of Lecture Notes in Computer Science, Springer, 2015, pp. 246–257. doi:10.1007/978-3-319-20028-6_25.

[10] F. Thuijsman, T. E. S. Raghavan, Perfect information stochastic games and related classes, International Journal of Game Theory 26 (3) (1997) 403–408. doi:10.1007/BF01263280.
URL http://dx.doi.org/10.1007/BF01263280

[11] E. Grädel, M. Ummels, Solution concepts and algorithms for infinite multiplayer game, in: K. Apt, R. van Rooij (Eds.), New Perspectives on Games and Interaction, Vol. 4 of Texts in Logic and Games, Amsterdam University Press, 2008, pp. 151–178.

[12] S. Le Roux, A. Pauly, Extending finite memory determinacy to multiplayer games, in: A. Lomuscio, M. Y. Vardi (Eds.), Proc. Strategic Reasoning, 2016, Vol. 218 of EPTCS, Open Publishing Association, 2016, pp. 27–40. doi:10.4204/EPTCS.218.3.

[13] D. A. Martin, Borel determinacy, Annals of Mathematics 102 (2) (1975) pp. 363–371. doi:10.2307/1971035.

[14] H. Gimbert, W. Zielonka, Games where you can play optimally without any memory, in: M. Abadi, L. de Alfaro (Eds.), CONCUR 2005. Proceedings, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 428–442. `doi:10.1007/11539452\_33`.
URL `http://dx.doi.org/10.1007/11539452_33`

[15] S. Paul, S. Simon, Nash Equilibrium in Generalised Muller Games, in: R. Kannan, K. N. Kumar (Eds.), IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, Vol. 4 of Leibniz International Proceedings in Informatics (LIPIcs), Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2009, pp. 335–346. `doi:10.4230/LIPIcs.FSTTCS.2009.2330`.
URL `http://drops.dagstuhl.de/opus/volltexte/2009/2330`

[16] Y. Gurevich, L. Harrington, Trees, automata and games, in: Proc. STOC, 1982. `doi:10.1145/800070.802177`.

[17] T. Brihaye, V. Bruyère, J. De Pril, Equilibria in quantitative reachability games, in: Proc. of CSR, Vol. 6072 of LNCS, Springer, 2010. `doi:10.1007/978-3-642-13182-0\_7`.

[18] N. Fijalkow, M. Zimmermann, Parity and streett games with costs, Logical Methods in Computer Science 10 (2). `doi:10.2168/LMCS-10(2:14)2014`.

[19] T. Brihaye, G. Geeraerts, A. Haddad, B. Monmege, To reach or not to reach? Efficient algorithms for total-payoff games, arXiv 1407.5030 (2014).
URL `http://arxiv.org/abs/1407.5030`

[20] S. Almagor, U. Boker, O. Kupferman, Formalizing and reasoning about quality, Journal of the ACM`doi:10.1145/2875421`.

[21] Y. Velner, K. Chatterjee, L. Doyen, T. A. Henzinger, A. Rabinovich, J.-F. Raskin, The complexity of multi-mean-payoff and multi-energy games, Information and Computation 241 (2015) 177 – 196. `doi:10.1016/j.ic.2015.03.001`.

URL http://www.sciencedirect.com/science/article/pii/S0890540115000164

[22] L. Clemente, J. Raskin, Multidimensional beyond worst-case and almost-sure problems for mean-payoff objectives, in: 30th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2015, Kyoto, Japan, July 6-10, 2015, 2015, pp. 257–268. doi:10.1109/LICS.2015.33.

[23] A. Chakrabarti, L. de Alfaro, T. A. Henzinger, M. Stoelinga, Resource interfaces, in: R. Alur, I. Lee (Eds.), Embedded Software, Vol. 2855 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2003, pp. 117–133. doi:10.1007/978-3-540-45212-6_9.

[24] P. Bouyer, U. Fahrenberg, K. G. Larsen, N. Markey, J. Srba, Infinite runs in weighted timed automata with energy constraints, in: F. Cassez, C. Jard (Eds.), Formal Modeling and Analysis of Timed Systems, Vol. 5215 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2008, pp. 33–47. doi:10.1007/978-3-540-85778-5_4.

[25] M. Jurdzinski, M. Paterson, U. Zwick, A deterministic subexponential algorithm for solving parity games, SIAM J. Comput. 38 (4) (2008) 1519–1532. doi:10.1137/070686652.

[26] J. von Neumann, O. Morgenstern, Theory of Games and Economic Behavior, Princeton University Press, 1944.