Parameterized complexity of games with monotonically ordered ω-regular objectives

Véronique Bruyère[1], Quentin Hautem[1⋆], and Jean-François Raskin[2⋆⋆]

[1]Département d'informatique, Université de Mons (UMONS), Mons, Belgium
[2] Département d'informatique, Université libre de Bruxelles (U.L.B.), Brussels, Belgium

Abstract. In recent years, two-player zero-sum games with multiple objectives have received a lot of interest as a model for the synthesis of complex reactive systems. In this framework, Player 1 wins if he can ensure that all objectives are satisfied against any behavior of Player 2. When this is not possible to satisfy all the objectives at once, an alternative is to use some preorder on the objectives according to which subset of objectives Player 1 wants to satisfy. For example, it is often natural to provide more significance to one objective over another, a situation that can be modelled with lexicographically ordered objectives for instance. Inspired by recent work on concurrent games with multiple ω-regular objectives by Bouyer et al., we investigate in detail turned-based games with monotonically ordered and ω-regular objectives. We study the threshold problem which asks whether player 1 can ensure a payoff greater than or equal to a given threshold w.r.t. a given monotonic preorder. As the number of objectives is usually much smaller than the size of the game graph, we provide a parametric complexity analysis and we show that our threshold problem is in FPT for all monotonic preorders and all classical types of ω-regular objectives. We also provide polynomial time algorithms for Büchi, coBüchi and explicit Muller objectives for a large subclass of monotonic preorders that includes among others the lexicographic preorder. In the particular case of lexicographic preorder, we also study the complexity of computing the values and the memory requirements of optimal strategies.

1 Introduction

Two-player zero-sum games played on directed graphs form an adequate framework for the *synthesis of reactive systems* facing an uncontrollable environment [30]. To model properties to be enforced by the reactive system within its environment, games with Boolean objectives and games with quantitative objectives have been studied, for example games with ω-regular objectives [21] and mean-payoff games [32].

Recently, games with *multiple* objectives have received a lot of attention since in practice, a system must usually satisfy several properties. In this framework, the system wins if it can ensure that *all* objectives are satisfied no matter how the environment behaves. For instance, generalized parity games are studied in [13], multi-mean-payoff games in [31], and multidimensional games with heterogeneous ω-regular objectives in [8].

When multiple objectives are conflicting or if there does not exist a strategy that can enforce all of them at the same time, it is natural to consider trade-offs. A general framework for defining trade-offs between n (Boolean) objectives $\Omega_1, \ldots, \Omega_n$ consists in assigning to each infinite path π of the game a payoff $v \in \{0,1\}^n$ such that $v(i) = 1$ iff π satisfies Ω_i, and then to equip $\{0,1\}^n$ with a preorder $\precsim$ to define a preference between pairs of payoffs: $v \precsim v'$ whenever payoff v' is preferred to payoff v. Because the ideal situation would be to satisfy *all* the objectives together, it is natural to assume that the preorder $\precsim$ has the following *monotonicity* property: if v' is such that whenever $v(i) = 1$ then $v'(i) = 1$, then it should be the case that v' is preferred to v.

As an illustration, let us consider a game in which Player 1 strives to enforce three objectives: Ω_1, Ω_2, and Ω_3. Assume also that Player 1 has no strategy ensuring all three objectives at the same time, that is, Player 1

⋆ Author supported by FRIA fellowship
⋆⋆ J.-F. Raskin is supported by an ERC Starting Grant (279499: inVEST), by an ARC project funded by the Fédération Wallonie-Bruxelles, and by a Professeur Francqui de Recherche grant awarded by the Francqui Fondation.

cannot ensure the objective $\Omega_1 \cap \Omega_2 \cap \Omega_3$. Then several options can be considered, see e.g. [7]. First, we could be interested in a strategy of Player 1 ensuring a maximal subset of the three objectives. Indeed, a strategy that enforces both Ω_1 and Ω_3 should be preferred to a strategy that enforces Ω_3 only. This preference is usually called the *subset preorder*. Now, if Ω_1 is considered more important than Ω_2 itself considered more important than Ω_3, then a strategy that ensures the most important possible objective should be considered as the most desirable. This preference is called the *maximize preorder*. Finally, we could also translate the relative importance of the different objectives into a *lexicographic preorder* on the payoffs: satisfying Ω_1 and Ω_2 would be considered as more desirable than satisfying Ω_1 and Ω_3 but not Ω_2. Those three examples are all monotonic preorders.

In this paper, we consider the following threshold problem: given a game graph G, a set of ω*-regular objectives*[1] $\Omega_1, \dots, \Omega_n$, a monotonic preorder $\precsim$ on the set $\{0,1\}^n$ of payoffs, and a threshold μ, decide whether Player 1 has a strategy such that for all strategies of Player 2, the outcome of the game has payoff v greater than or equal to μ (for the specified preorder), i.e. $\mu \precsim v$. As the number n of objectives is typically much smaller than the size of the game graph G, it is natural to consider a parametric analysis of the complexity of the threshold problem in which the number of objectives and their size are considered to be fixed parameters of the problem. Our main results are as follows.

Contributions. First, we provide *fixed parameter tractable solutions* to the threshold problem for *all* monotonic preorders and for *all* classical types of ω-regular objectives. Our solutions rely on the following ingredients:

1. We show that solving the threshold problem is equivalent to solve a game with a single objective Ω that is a union of intersections of objectives taken among $\Omega_1, \dots, \Omega_n$ (Theorem 8). This is possible by *embedding* the monotonic preorder $\precsim$ in the subset preorder and by translating the threshold μ in preorder $\precsim$ into an antichain of thresholds in the subset preorder. A threshold in the subset preorder is naturally associated with a conjunction of objectives, and an antichain of thresholds leads to a union of such conjunctions.
2. We provide a fixed parameter tractable algorithm to solve games with a single objective Ω as described previously for all types of ω-regular objectives $\Omega_1, \dots, \Omega_n$, leading to a fixed parameter algorithm for our threshold problem (Theorem 7). Those results build on the recent breakthrough of Calude et al. that provides a quasipolynomial time algorithm for parity games as well as their fixed parameter tractability [10], and on the fixed parameter tractability of games with an objective defined by a Boolean combination of Büchi objectives (Proposition 11).

Second, we consider games with a preorder $\precsim$ having a *compact embedding*, with the main condition that the antichain of thresholds resulting from the embedding in the subset preorder is of *polynomial size*. The maximize preorder, the subset preorder, and the lexicographic preorder, given as examples above, all possess this property. For games with a compact embedding, we go beyond fixed parameter tractability as we are able to provide deterministic polynomial time solutions for Büchi, coBüchi, and explicit Muller objectives (Theorem 12). Polynomial time solutions are not possible for the other types of ω-regular objectives as we show that the threshold problem for the lexicographic preorder with reachability, safety, parity, Rabin, Streett, and Muller objectives cannot be solved in polynomial time unless $\mathsf{P} = \mathsf{PSPACE}$ (Theorem 13). Finally, we provide a full picture of the study of games with the lexicographic preorder: complexity class and memory requirements of winning strategies for the threshold problem, and cost of computing values and memory requirements of the related optimal strategies (Table 3).

Related work. In [7], Bouyer et al. investigate concurrent games with multiple objectives leading to payoffs in $\{0,1\}^n$ which are ordered using Boolean circuits. While their threshold problem is slightly more general than ours, their games being concurrent and their preorders being not necessarily monotonic, the algorithms that they provide are nondeterministic and guess witnesses whose size depends polynomially not only in the number of objectives but also in the size of the game graph. Their algorithms are sufficient to establish membership to PSPACE for all classical types of ω-regular objectives but they do not provide a basis

[1] We cover all classical ω-regular objectives: reachability, safety, Büchi, co-Büchi, parity, Rabin, Streett, explicit Muller, or Muller.

for the parametric complexity analysis of the threshold problem. In stark contrast, we provide deterministic algorithms whose complexity only depends polynomially in the size of the game graph. Our new deterministic algorithms are thus instrumental to a finer complexity analysis that leads to fixed parameter tractability for all monotonic preorders and all ω-regular objectives. We also provide tighter lower-bounds for the important special case of lexicographic preorder, in particular for parity objectives.

The particular class of games with multiple Büchi objectives ordered with the maximize preorder has been considered in [1]. The interested reader will find in that paper clear practical motivations for considering multiple objectives and ordering them. The lexicographic ordering of objectives has also been considered in the context of quantitative games: lexicographic mean-payoff games in [6], some special cases of lexicographic quantitative games in [9,22], and lexicographically ordered energy objectives in [14].

Structure of the paper. In Section 2, we present all the useful notions about games with monotonically ordered ω-regular objectives. In Section 3, we show that solving the threshold problem is equivalent to solve a game with a single objective that is a union of intersections of objectives (Theorem 8), and we establish the main result of this paper: the fixed parameter complexity of the threshold problem (Theorem 7). Section 4 is devoted to games with a compact embedding and in particular to the threshold problem for lexicographic games. The last section is dedicated to the study of computing the values and memory requirements of optimal strategies in the case of lexicographic games (Table 3).

2 Preliminaries

We consider zero-sum turn-based games played by two players, $\mathcal{P}_1$ and $\mathcal{P}_2$, on a finite directed graph. Given *several objectives*, we associate with each play of this game a vector of bits called *payoff*, the components of which indicate the objectives that are satisfied. The set of all payoffs being equipped with a *preorder*, $\mathcal{P}_1$ wants to ensure a payoff greater than or equal to a given threshold against any behavior of $\mathcal{P}_2$. In this section we give all the useful notions and the studied problem.

Preorders. Given some non-empty set P, a *preorder* over P is a binary relation $\precsim \subseteq P \times P$ that is reflexive and transitive. The *equivalence relation* $\sim$ associated with $\precsim$ is defined such that $x \sim y$ if and only if $x \precsim y$ and $y \precsim x$. The *strict partial order* $\prec$ associated with $\precsim$ is then defined such that $x \prec y$ if and only if $x \precsim y$ and $x \not\sim y$. A preorder $\precsim$ is *total* if $x \precsim y$ or $y \precsim x$ for all $x, y \in P$. A set $S \subseteq P$ is *upper-closed* if for all $x \in S$, $y \in P$, if $x \precsim y$, then $y \in S$. An *antichain* is a set $S \subseteq P$ of pairwise incomparable elements, that is, for all $x, y \in S$, if $x \neq y$, then $x \not\precsim y$ and $y \not\precsim x$.

Game structures. We give below the definition of a game structure and notations on plays.

Definition 1. *A* game structure *is a tuple $G = (V_1, V_2, E)$ where*

- *(V, E) is a finite directed graph, with $V = V_1 \cup V_2$ the set of vertices and $E \subseteq V \times V$ the set of edges such that*[2] *for each $v \in V$, there exists $(v, v') \in E$ for some $v' \in V$,*
- *(V_1, V_2) forms a partition of V such that V_i is the set of vertices controlled by player $\mathcal{P}_i$ with $i \in \{1, 2\}$.*

A *play* of G is an infinite sequence of vertices $\pi = v_0 v_1 \ldots \in V^\omega$ such that $(v_k, v_{k+1}) \in E$ for all $k \in \mathbb{N}$. We denote by $\mathsf{Plays}(G)$ the set of plays in G. *Histories* of G are finite sequences $\rho = v_0 \ldots v_k \in V^+$ defined in the same way. Given a play $\pi = v_0 v_1 \ldots$, the set $\mathsf{Occ}(\pi)$ denotes the set of vertices that occur in π, and the set $\mathsf{Inf}(\pi)$ denotes the set of vertices visited infinitely often along π, i.e., $\mathsf{Occ}(\pi) = \{v \in V \mid \exists k \geq 0, v_k = v\}$ and $\mathsf{Inf}(\pi) = \{v \in V \mid \forall k \geq 0, \exists l \geq k,\ v_l = v\}$. Given a set $U \subseteq V$ and a set $\Omega \subseteq V^\omega$, we denote by U^c the set $V \setminus U$ and by $\overline{\Omega}$ the set $V^\omega \setminus \Omega$.

[2] This condition guarantees that there is no deadlock. It can be assumed w.l.o.g. for all the problems considered in this article.

Strategies. A *strategy* σ_i for $\mathcal{P}_i$ is a function $\sigma_i \colon V^*V_i \to V$ assigning to each history $\rho v \in V^*V_i$ a vertex $v' = \sigma_i(\rho v)$ such that $(v, v') \in E$. It is *memoryless* if $\sigma_i(\rho v) = \sigma_i(\rho' v)$ for all histories $\rho v, \rho' v$ ending with the same vertex v, that is, if σ_i is a function $\sigma_i \colon V_i \to V$. It is *finite-memory* if it can be encoded by a deterministic *Moore machine* $\mathcal{M} = (M, m_0, \alpha_u, \alpha_n)$ where M is a finite set of states (the memory of the strategy), $m_0 \in M$ is the initial memory state, $\alpha_u \colon M \times V \to M$ is the update function, and $\alpha_n \colon M \times V_i \to V$ is the next-action function. The Moore machine $\mathcal{M}$ defines a strategy σ_i such that $\sigma_i(\rho v) = \alpha_n(\widehat{\alpha}_u(m_0, \rho), v)$ for all histories $\rho v \in V^*V_i$, where $\widehat{\alpha}_u$ extends α_u to histories as expected. The *size* of the strategy σ_i is the size $|M|$ of its machine $\mathcal{M}$. Note that σ_i is memoryless when $|M| = 1$.

The set of all strategies of $\mathcal{P}_i$ is denoted by Σ_i. Given a strategy σ_i of $\mathcal{P}_i$, a play $\pi = v_0 v_1 \ldots$ of G is *consistent* with σ_i if $v_{k+1} = \sigma_i(v_0 \ldots v_k)$ for all $k \in \mathbb{N}$ such that $v_k \in V_i$. Consistency is naturally extended to histories in a similar fashion. Given an *initial vertex* v_0, and a strategy σ_i of each player $\mathcal{P}_i$, we have a unique play consistent with both strategies σ_1, σ_2, called *outcome* and denoted by $\mathsf{Out}(v_0, \sigma_1, \sigma_2)$.

Single objectives and ordered objectives. An *objective for* $\mathcal{P}_1$ is a set of plays $\Omega \subseteq \mathsf{Plays}(G)$. A *game* (G, Ω) is composed of a game structure G and an objective Ω. A play π is *winning* for $\mathcal{P}_1$ if $\pi \in \Omega$, and losing otherwise. As the studied games are zero-sum, $\mathcal{P}_2$ has the opposite objective $\overline{\Omega}$, meaning that a play π is winning for $\mathcal{P}_1$ if and only if it is losing for $\mathcal{P}_2$. Given a game (G, Ω) and an initial vertex v_0, a strategy σ_1 for $\mathcal{P}_1$ is *winning from* v_0 if $\mathsf{Out}(v_0, \sigma_1, \sigma_2) \in \Omega$ for all strategies σ_2 of $\mathcal{P}_2$. Vertex v_0 is thus called *winning* for $\mathcal{P}_1$. We also say that $\mathcal{P}_1$ is winning from v_0 or that he can *ensure* Ω *from* v_0. Similarly the winning vertices of $\mathcal{P}_2$ are those from which $\mathcal{P}_2$ can ensure his objective $\overline{\Omega}$.

A game (G, Ω) is *determined* if each of its vertices is either winning for $\mathcal{P}_1$ or winning for $\mathcal{P}_2$. Martin's theorem [27] states that all games with Borel objectives are determined. The problem of *solving a game* (G, Ω) means to decide, given an initial vertex v_0, whether $\mathcal{P}_1$ is winning from v_0 (or dually whether $\mathcal{P}_2$ is winning from v_0 when the game is determined).

Instead of a *single* objective Ω, one can consider *several* objectives $\Omega_1, \ldots, \Omega_n$ that are *ordered* with respect to a preorder $\precsim$ over $\{0,1\}^n$ in the following way. We first define the payoff of a play as a vector[3] of bits the components of which indicate the objectives that are satisfied.

Definition 2. *Given a game structure* $G = (V_1, V_2, E)$*, and* n *objectives* $\Omega_1, \ldots, \Omega_n \subseteq \mathsf{Plays}(G)$*, the* payoff function $\mathsf{Payoff} \colon \mathsf{Plays}(G) \to \{0,1\}^n$ *assigns a vector of bits to each play* $\pi \in \mathsf{Plays}(G)$*, where for all* $k \in \{1, \ldots, n\}$*,* $\mathsf{Payoff}_k(\pi) = 1$ *if* $\pi \in \Omega_k$ *and* 0 *otherwise.*

Given the preorder $\precsim$ over $\{0,1\}^n$, $\mathcal{P}_1$ prefers a play π to a play π' whenever $\mathsf{Payoff}(\pi') \precsim \mathsf{Payoff}(\pi)$. We call *ordered game* the tuple $(G, \Omega_1, \ldots, \Omega_n, \precsim)$, the payoff function of which is defined w.r.t. the objectives $\Omega_1, \ldots, \Omega_n$ and its values are ordered with $\precsim$. In this context, we are interested in the following problem.

Problem 3. The *threshold problem* for ordered games $(G, \Omega_1, \ldots, \Omega_n, \precsim)$ asks, given a threshold $\mu \in \{0,1\}^n$ and an initial vertex $v_0 \in V$, to decide whether $\mathcal{P}_1$ (resp. $\mathcal{P}_2$) has a strategy to ensure the objective $\Omega = \{\pi \in \mathsf{Plays}(G) \mid \mathsf{Payoff}(\pi) \succsim \mu\}$ from v_0 (resp. $\overline{\Omega} = \{\pi \in \mathsf{Plays}(G) \mid \mathsf{Payoff}(\pi) \not\succsim \mu\}$).[4]

In case $\mathcal{P}_1$ (resp. $\mathcal{P}_2$) has such a winning strategy, we also say that he can *ensure* (resp. *avoid*) *a payoff* $\succsim \mu$.

Classical examples of preorders are the following ones [7]. Let $x, y \in \{0,1\}^n$.

- *Counting*: $x \precsim y$ if and only if $|\{j \mid x_j = 1\}| \leq |\{j \mid y_j = 1\}|$. The aim of $\mathcal{P}_1$ is to maximize the number of satisfied objectives.
- *Subset*: $x \precsim y$ if and only if $\{j \mid x_j = 1\} \subseteq \{j \mid y_j = 1\}$. The aim of $\mathcal{P}_1$ is to maximize the subset of satisfied objectives with respect to the inclusion.
- *Maximise*: $x \precsim y$ if and only if $\max\{j \mid x_j = 1\} \leq \max\{j \mid y_j = 1\}$. The aim of $\mathcal{P}_1$ is to maximize the higher index of the satisfied objectives.
- *Lexicographic*: $x \precsim y$ if and only if either $x = y$ or $\exists j \in \{1, \ldots, n\}$ such that $x_j < y_j$ and $\forall k \in \{1, \ldots, j-1\}$, $x_k = y_k$. The objectives are ranked according to their importance. The aim of $\mathcal{P}_1$ is to maximise the payoff with respect to the induced lexicographic order.

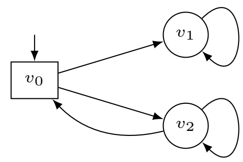

Fig. 1. A simple lexicographic game.

In this article, we *focus on monotonic preorders*. A preorder $\precsim$ is *monotonic* if it is compatible with the subset preorder, i.e. if $\{i \mid x_i = 1\} \subseteq \{i \mid y_i = 1\}$ implies $x \precsim y$. Hence a preorder is monotonic if satisfying more objectives never results in a lower payoff value. This is a *natural property* shared by all the examples of preorders given previously.

Example 4. Consider the game structure G depicted on Figure 1, where circle vertices belong to $\mathcal{P}_1$ and square vertices belong to $\mathcal{P}_2$. We consider the ordered game $(G, \Omega_1, \Omega_2, \precsim)$ with $\Omega_i = \{\pi \in \mathsf{Plays}(G) \mid v_i \in \mathsf{Inf}(\pi)\}$ for $i = 1, 2$ and the lexicographic preorder $\precsim$. Therefore the function Payoff assigns value 1 to each play π on the first (resp. second) bit if and only if π visits infinitely often vertex v_1 (resp. v_2). In this ordered game, $\mathcal{P}_1$ has a strategy to ensure a payoff $\succsim 01$ from v_0. Indeed, consider the memoryless strategy σ_1 that loops in v_1 and in v_2. Then, from v_0, $\mathcal{P}_2$ decides to go either to v_1 leading to the payoff 10, or to v_2 leading to the payoff 01. As $10 \succsim 01$, this shows that any play π consistent with σ_1 satisfies $\mathsf{Payoff}(\pi) \succsim 01$. Notice that while $\mathcal{P}_1$ can ensure a payoff $\succsim 01$ from v_0, he has no strategy to enforce the single objective Ω_1 and similarly no strategy to enforce Ω_2. Remark that $\mathcal{P}_2$ has no strategy avoiding a payoff $\succsim 01$ from v_0, but rather a memoryless winning strategy for the single objective $\overline{\Omega_1}$ from v_0, and similarly for the single objective $\overline{\Omega_2}$.

Homogeneous ω-regular objectives. In the sequel of this article, given a monotonically ordered game $(G, \Omega_1, \dots, \Omega_n, \precsim)$, we want to study the threshold problem described in Problem 3 for *homogeneous ω-regular objectives*, in the sense that all the objectives $\Omega_1, \dots, \Omega_n$ are of the same type, and taken in the following list of well-known ω-regular objectives.
Given a game structure $G = (V_1, V_2, E)$ and a subset U of V called *target set*:

- The *reachability objective* asks to visit a vertex of U at least once, i.e. $\mathsf{Reach}(U) = \{\pi \in \mathsf{Plays}(G) \mid \mathsf{Occ}(\pi) \cap U \neq \emptyset\}$.
- The *safety objective* asks to always stay in the set U, i.e. $\mathsf{Safe}(U) = \{\pi \in \mathsf{Plays}(G) \mid \mathsf{Occ}(\pi) \cap U^c = \emptyset\}$.
- The *Büchi objective* asks to visit infinitely often a vertex of U, i.e. $\mathsf{Buchi}(U) = \{\pi \in \mathsf{Plays}(G) \mid \mathsf{Inf}(\pi) \cap U \neq \emptyset\}$.
- The *co-Büchi objective* asks to eventually always stay in the set U, i.e. $\mathsf{CoBuchi}(U) = \{\pi \in \mathsf{Plays}(G) \mid \mathsf{Inf}(\pi) \cap U^c = \emptyset\}$.

Given a *family* $\mathcal{F} = (F_i)_{i=1}^k$ of sets $F_i \subseteq V$, and a family of *pairs* $((E_i, F_i)_{i=1}^k)$, with $E_i, F_i \subseteq V$:

- The *explicit Muller objective* asks that the set of vertices seen infinitely often is exactly one among the sets of $\mathcal{F}$, i.e. $\mathsf{ExplMuller}(\mathcal{F}) = \{\pi \in \mathsf{Plays}(G) \mid \exists i \in \{1, \dots, k\}, \mathsf{Inf}(\pi) = F_i\}$.
- The *Rabin objective* asks that there exists a pair (E_i, F_i) such that a vertex of F_i is visited infinitely often while no vertex of E_i is visited infinitely often, i.e. $\mathsf{Rabin}((E_i, F_i)_{i=1}^k) = \{\pi \in \mathsf{Plays}(G) \mid \exists i \in \{1, \dots, k\}, \mathsf{Inf}(\pi) \cap E_i = \emptyset$ and $\mathsf{Inf}(\pi) \cap F_i \neq \emptyset\}$.
- The *Streett objective* asks that for each pair (E_i, F_i), a vertex of E_i is visited infinitely often or no vertex of F_i is visited infinitely often, i.e. $\mathsf{Streett}((E_i, F_i)_{i=1}^k) = \{\pi \in \mathsf{Plays}(G) \mid \forall i \in \{1, \dots, k\}, \mathsf{Inf}(\pi) \cap E_i \neq \emptyset$ or $\mathsf{Inf}(\pi) \cap F_i = \emptyset\}$.

[3] Note that in the sequel, we often manipulate equivalently vectors in $\{0, 1\}^n$ and sequences of n bits.
[4] Note that when $n = 1$ and $\precsim$ is the usual order $\leq$ over $\{0, 1\}$, we recover the notion of single objective with the threshold $\mu = 1$.

Given a *coloring* function $p\colon V \to \{0,\dots,d\}$ that associates with each vertex a color, and $\mathcal{F} = (F_i)_{i=1}^k$ a family of subsets F_i of $p(V)$:

- The *parity objective* asks that the minimum color seen infinitely often is even, i.e. $\mathsf{Parity}(p) = \{\pi \in \mathsf{Plays}(G) \mid \min_{v\in\mathsf{Inf}(\pi)} p(v)$ is even$\}$.
- The *Muller objective* asks that the set of colors seen infinitely often is exactly one among the sets of $\mathcal{F}$, i.e. $\mathsf{Muller}(p,\mathcal{F}) = \{\pi \in \mathsf{Plays}(G) \mid \exists i \in \{1,\dots,k\}, p(\mathsf{Inf}(\pi)) = F_i\}$.

In the sequel, we make the *assumption* that the considered preorders are monotonic, and by *ordered game*, we always mean monotonically ordered games. When the objectives of an ordered game are of kind X, we speak of an *ordered X game*, or of a *$\precsim$ X game* if we want to specify the used preorder $\precsim$. As already mentioned, when $n = 1$, an ordered game (with $\precsim$ equal to $\leq$) resumes to a game (G,Ω) with a single objective Ω, that is traditionally called an Ω game. For instance, an ordered game $(G,\Omega_1,\dots,\Omega_n,\precsim)$ where $\Omega_1,\dots,\Omega_n$ are reachability objectives and $\precsim$ is the lexicographic preorder is called a lexicographic reachability game, and when $n = 1$ (G,Ω_1) is called a reachability game.

Note that given an ordered game with n non-homogeneous ω-regular objectives Ω_i, we can always construct a new equivalent ordered parity game, since each objective Ω_i can be translated into a parity objective [21].

Useful results on games with a single objective. Let us end this section by providing some results on games with a single ω-regular objective taken among those defined previously or among the additional ones given herafter. All these results will be useful in the proofs.

Let G be a game structure and $U_1,\dots,U_m$ be m target sets and ϕ be a Boolean formula over variables $x_1,\dots,x_m$. We say that a play π satisfies $(\phi,U_1,\dots,U_m)$ if the truth assignment ($x_i = 1$ if and only $\mathsf{Inf}(\pi)\cap U_i \neq \emptyset$, and $x_i = 0$ otherwise) satisfies ϕ.

- *Boolean combination of Büchi objectives*, or shortly *Boolean Büchi* objective:

$$\mathsf{BooleanBuchi}(\phi,U_1,\dots,U_m) = \{\pi \in \mathsf{Plays}(G) \mid \pi \text{ satisfies } (\phi,U_1,\dots,U_m)\}.$$

All operators $\vee$, $\wedge$, $\neg$ are allowed in Boolean Büchi objectives. However we denote by $|\phi|$ the *size* of ϕ equal to the number of disjunctions and conjunctions inside ϕ, and we say that the Boolean Büchi objective $\mathsf{BooleanBuchi}(\phi,U_1,\dots,U_m)$ is *of size $|\phi|$ and with m variables*. The definition of $|\phi|$ is not the classical one that usually counts the number of operators $\vee,\wedge,\neg$ and variables. This is not a restriction since one can transform any Boolean formula ϕ into one such that negations only apply on variables.

We need to introduce some other kinds of ω-regular objectives with Boolean combinations of objectives that are limited to

- intersections of objectives: like a *generalized reachability* objective $\mathsf{GenReach}(U_1,\dots,U_m)$ or a *generalized Büchi* objective $\mathsf{GenBuchi}(U_1,\dots,U_m)$,
- unions of intersections (UI) of objectives: like a *UI reachability* objective

$$\mathsf{UIReach}(U_{1,1},\dots,U_{l,m}) = \cup_{i=1}^{l} \cap_{j=1}^{m} \mathsf{Reach}(U_{i,j}),$$

  a *UI safety* objective $\mathsf{UISafe}(U_{1,1},\dots,U_{l,m})$, or a *UI Büchi* objective $\mathsf{UIBuchi}(U_{1,1},\dots,U_{l,m})$.

Games (G,Ω) with ω-regular objectives Ω are determined by Martin's theorem [27]. We recall the complexity class of solving those games, as well as the kind (memoryless, finite-memory) of winning strategies for both players. See Theorem 5 and Table 1 below. For each type of objective, the complexity of the algorithms is expressed in terms of the sizes $|V|$ and $|E|$ of the game structure G, the number d of colors (for Parity and Muller), the number k of pairs (for Rabin and $\mathsf{Streett}$), the size $|\mathcal{F}|$ of the family $\mathcal{F}$ (for $\mathsf{ExplMuller}$ and Muller), the size $|\phi|$ of the formula ϕ (for $\mathsf{BooleanBuchi}$), the number m of intersections of objectives (for $\mathsf{GenReach}$ and $\mathsf{GenBuchi}$), and the number m (resp. l) of intersections (resp. unions) in UI objectives (for $\mathsf{UIReach}$, UISafe, and $\mathsf{UIBuchi}$).

Theorem 5. *For games (G, Ω) with ω-regular objectives, we have:*

– *Solving reachability or safety games is* P*-complete (with an algorithm in $O(|V| + |E|)$ time) and both players have memoryless winning strategies* [3,21,25].
– *Solving Büchi or co-Büchi games is* P*-complete (with an algorithm in $O(|V|^2)$ time) and both players have memoryless winning strategies* [12,19,25].
– *Solving explicit Muller games with a family $\mathcal{F}$ is* P*-complete (with an algorithm in $O(|\mathcal{F}| \cdot (|\mathcal{F}| + |V| \cdot |E|)^2)$ time) and exponential memory strategies are necessary and sufficient for both players* [16,23].
– *Solving Rabin (resp. Streett) games with k pairs is* NP*-complete (resp. co-*NP*-complete)* [18] *(with an algorithm in $O(|V|^{k+1} \cdot k!)$ time* [29]*). In Rabin games (resp. Streett games) memoryless strategies are sufficient for $\mathcal{P}_1$ (resp. for $\mathcal{P}_2$)* [17] *and exponential memory strategies are necessary and sufficient for $\mathcal{P}_2$ (resp. $\mathcal{P}_1$)* [16]
– *Solving parity games with d colors is in* UP ∩ co-UP *(with an algorithm in $O(|V|^{\lceil \log(d) \rceil + 6})$ time* [10]*) and both players have memoryless winning strategies* [26].
– *Solving Muller games is* PSPACE*-complete (with an algorithm in $O(|V|^2 \cdot |E| \cdot |V|!)$ time* [28]*) and exponential memory strategy are necessary and sufficient for both players* [16,24].
– *Solving Boolean Büchi games is* PSPACE*-complete (with an algorithm in $O(|\phi| \cdot 2^{O(|V|^2)})$ time and exponential memory strategies are necessary and sufficient for both players* [2].[5]
– *Solving generalized reachability games with m target sets is* PSPACE*-complete (with an algorithm in $O(2^m \cdot (|V| + |E|))$ time) and exponential memory strategies are necessary and sufficient for both players* [20].
– *Solving generalized Büchi games with m target sets is* P*-complete (with an algorithm in $O(m \cdot |V|^2)$ time) and linear memory (resp. memoryless) strategies are necessary and sufficient for $\mathcal{P}_1$ (resp. $\mathcal{P}_2$)* [11].
– *Solving UI reachability and UI safety objectives is* PSPACE*-complete (with an algorithm in $O(2^K \cdot (|V| + |E|))$ time) and exponential memory strategies are necessary and sufficient for both players, where K denotes the number of distinct target sets.*
– *Solving UI Büchi games with an objective $\cup_{i=1}^{l} \cap_{j=1}^{m} \mathsf{Buchi}(U_{i,j})$ is* coNP*-complete (with an algorithm in $O(m^l \cdot |V|^2)$ time), and exponential memory (resp. memoryless) strategies are necessary and sufficient for $\mathcal{P}_1$ (resp. $\mathcal{P}_2$)* [5].

Proof. All the statements follow from the literature except for the case of UI reachability and UI safety games for which we provide a proof. We only consider the reachability case, since the proof is similar for the safety case. First, as solving UI reachability games is harder than solving generalized reachability games (when there is no union), we immediately obtain the lower bounds for the complexity and the memory requirements. Indeed, solving generalized reachability games is PSPACE-complete, and exponential memory strategies are necessary for both players [20].

Let us now prove the upper bounds by following the same approach as proposed in [20] to solve generalized reachability games. Let (G, Ω) be a UI reachability game where $\Omega = \cup_{i=1}^{l} \cap_{j=1}^{m} \mathsf{Reach}(U_{i,j})$. We define the function $f' \colon \{U_{i,j} \mid i \in \{1, \dots, l\}, j \in \{1, \dots, m\}\} \to \{1, \dots, K\}$ that enumerates all the distinct sets $U_{i,j}$. From f', we construct the function $f \colon \{1, \dots, l\} \times \{1, \dots, m\} \to \{1, \dots, N\}$ such that $f(i,j) = k$ if $f'(U_{i,j}) = k$. If $f(i,j) = k$, we abusively write $U_{i,j} = U_k$.

We construct from $G = (V_1, V_2, E)$ a new game structure $G' = (V_1', V_2', E')$ in a way to remember which sets U_k have been visited so far, for $k \in \{1, \dots, K\}$. Formally, $V_i' = V_i \times \{0,1\}^K$ for $i \in \{1,2\}$, and $((v, b_1, \dots, b_K), (v', b_1', \dots, b_K')) \in E'$ if and only if $(v, v') \in E$ and for all k, $b_k' = 1$ if $b_k = 1$ or $v' \in U_k$, and 0 otherwise. With the initial vertex v_0 in G, we associate the initial vertex $(v_0, b_1^0, \dots, b_K^0)$ in G' where $b_k^0 = 1$ if $v_0 \in U_k$ and 0 otherwise. We then have that $\mathcal{P}_1$ is winning in the original UI reachability game from v_0 if and only if $\mathcal{P}_1$ is winning in G' from $(v_0, b_1^0, \dots, b_K^0)$ for the objective $\mathsf{Reach}(U)$ where $U = \{(v, b_1, \dots, b_K) \mid \exists i \in \{1, \dots, l\}, \forall j \in \{1, \dots, m\}, b_{f(i,j)} = 1\}$.

Note that solving this reachability game $(G', \mathsf{Reach}(U))$ can be done in time linear in the size of the game with memoryless winning strategies for both players by [21]. Coming back to the initial UI reachability

[5] The algorithm complexity and the memory requirements do not appear explicitly in [2] but can be deduced straightforwardly thanks to the proposed algorithm.

game, this leads to an algorithm working in $O(2^K \cdot (|V| + |E|))$ time, and to exponential memory winning strategies for both players.

Now, as done for generalized reachability games [20], one can notice that if $\mathcal{P}_1$ is winning for $\mathsf{Reach}(U)$, then he has a strategy to do so within $K \cdot |V|$ steps. Moreover, given a path of this size, one can check in polynomial time if there exists some i such that the path visits all $U_{f(i,j)}$ for $j \in \{1, \dots, m\}$. Thus, we can use an alternating Turing machine that simulates the game for up to $K \cdot |V|$ steps and checks whether $\mathcal{P}_1$ is winning. As the alternating Turing machine works in polynomial time and APTIME = PSPACE, this yields the PSPACE algorithm. □

<table>
<tr><th>Objectives</th><th>Complexity class</th><th>$\mathcal{P}_1$ memory</th><th>$\mathcal{P}_2$ memory</th></tr>
<tr><td>Reachability, safety</td><td rowspan="4">P-complete</td><td colspan="2" rowspan="2">memoryless</td></tr>
<tr><td>Büchi, co-Büchi</td></tr>
<tr><td>Explicit Muller</td><td colspan="2">exponential</td></tr>
<tr><td>Generalized Büchi</td><td>linear</td><td>memoryless</td></tr>
<tr><td>Generalized reachability</td><td rowspan="2">PSPACE-complete</td><td colspan="2" rowspan="2">exponential</td></tr>
<tr><td>UI reachability, UI safe</td></tr>
<tr><td>Parity</td><td>NP ∩ coNP</td><td colspan="2">memoryless</td></tr>
<tr><td>Rabin</td><td>NP-complete</td><td>memoryless</td><td>exponential</td></tr>
<tr><td>Streett</td><td rowspan="2">coNP-complete</td><td rowspan="2">exponential</td><td rowspan="2">memoryless</td></tr>
<tr><td>UI Büchi</td></tr>
<tr><td>Muller</td><td rowspan="2">PSPACE-complete</td><td colspan="2" rowspan="2">exponential</td></tr>
<tr><td>Boolean Büchi</td></tr>
</table>

Table 1. Overview of results on games with a single ω-regular objective. The last two columns indicate the tight memory requirements of the winning strategies.

In the sequel, we need some classical properties on ω-regular objectives that we summarize in the following proposition.

Proposition 6. *1. A safety (resp. co-Büchi, Streett) objective is the complement of a reachability (resp. Büchi, Rabin) objective.*

2. A parity objective is both a Rabin and a Streett objective.

3. Rabin and Streett objectives with one pair are parity objectives with 3 *colors. Thus, a Rabin (resp. Streett) objective is the union (resp. intersection) of parity objectives with* 3 *colors.*

4. The intersection of m (resp. union of l) explicit Muller objectives $\mathsf{ExplMuller}(\mathcal{F}_i)$ *is an explicit Muller objective* $\mathsf{ExplMuller}(\mathcal{F})$ *where* $|\mathcal{F}| \leq \min_{i \in \{1,\dots,m\}}\{|\mathcal{F}_i|\}$ *(resp.* $|\mathcal{F}| \leq \sum_{i=1}^{l} |\mathcal{F}_i|$*).*

5. A parity objective with d colors (resp. Streett objective with k pairs, Rabin objective with k pairs, Muller objective with d colors and a family $\mathcal{F}$) is a Boolean Büchi objective of size at most $\frac{d^2}{2}$ *(resp.* $2 \cdot k$, $d \cdot |\mathcal{F}|$*) and with d (resp.* $2 \cdot k$, d*) variables.*

Proof. First, Item 1 immediately follows from the definitions and Items 2 and 3 are detailed in [13].

Let us consider Item 4. For the intersection we have $\cap_{i=1}^{m} \mathsf{ExplMuller}(\mathcal{F}_i) = \mathsf{ExplMuller}(\mathcal{F})$ where $\mathcal{F} = \cap_{i=1}^{m} \mathcal{F}_i$, and thus $|\mathcal{F}| \leq \min_{i \in \{1,\dots,m\}}\{|\mathcal{F}_i|\}$. For the union we have $\cup_{i=1}^{l} \mathsf{ExplMuller}(\mathcal{F}_i) = \mathsf{ExplMuller}(\mathcal{F})$ with $\mathcal{F} = \cup_{i=1}^{l} \mathcal{F}_i$ with $|\mathcal{F}| \leq \sum_{i=1}^{l} |\mathcal{F}_i|$.

Let us prove the last item by beginning with Muller objectives. It suffices to note that a play belongs to $\mathsf{Muller}(p, \mathcal{F})$ if and only if there exists an element F of $\mathcal{F}$ such that all colors of F are seen infinitely often along the play while no other color is seen infinitely often. This is obviously a Boolean Büchi objective $\mathsf{BooleanBuchi}(\phi, U_1, \dots, U_m)$, where each U_i corresponds to a color, that is, U_i is the set of vertices labeled by this color. Note that, in this case, the size of the related formula ϕ is at most $d \cdot |\mathcal{F}|$. The arguments are similar for parity, Streett and Rabin objectives (for instance, a play belongs to $\mathsf{Parity}(p)$ if and only there exists an even color seen infinitely often along the play and no lower color seen infinitely often). □

3 Fixed parameter complexity of ordered ω-regular games

In this section, we study the fixed parameter tractability of the threshold problem.

Parameterized complexity. A *parameterized language* L is a subset of $\Sigma^* \times \mathbb{N}$, where Σ is a finite alphabet, the second component being the parameter of the language. It is called *fixed parameter tractable* (FPT) if there is an algorithm that determines whether $(x,t) \in L$ in time $f(t) \cdot |x|^c$ time, where c is a constant independent of the parameter t and f is a computable function depending on t only. We also say that L belongs to (the class) FPT. Intuitively, a language is FPT if there is an algorithm running in polynomial time w.r.t the input size times some computable function on the parameter. In this framework, we do not rely on classical polynomial reductions but rather use so called FPT-reductions. An FPT-*reduction* between two parameterized languages $L \subseteq \Sigma^* \times \mathbb{N}$ and $L' \subseteq \Sigma'^* \times \mathbb{N}$ is a function $R : L \to L'$ such that

- $(x,t) \in L$ if and only if $(x',t') = R(x,t) \in L'$, where parameter t' only depends on t, and
- R is computable by an algorithm that takes $f(t) \cdot |x|^c$ time where c is a constant.

Moreover, if L' is in FPT, then L is also in FPT. We refer the interested reader to [15] for more details on parameterized complexity.

Our main result states that the threshold problem is in FPT for all the ordered games of this article. Parameterized complexities are given in Table 2.

Theorem 7. *The threshold problem is in* FPT *for ordered reachability, safety, Büchi, co-Büchi, explicit Muller, Rabin, Streett, parity, and Muller games.*

Objectives	Parameters	Threshold problem
Reachability, Safety	n	$O(2^n \cdot (\lvert V\rvert + \lvert E\rvert))$
Büchi	n	$O(s'(n) \cdot \lvert V\rvert^2)$
co-Büchi	n	$O(s(n) \cdot \lvert V\rvert^2)$
Explicit Muller	n	$O((s(n) \cdot \max_i \lvert\mathcal{F}_i\rvert)^3 \cdot \lvert V\rvert^2 \cdot \lvert E\rvert^2)$
Rabin, Streett	$n,\, k_1, \ldots, k_n$	$O((2^{M_1} \cdot N_1 + M_1^{M_1}) \cdot \lvert V\rvert^5)$
Parity	$n,\, d_1, \ldots, d_n$	$O((2^{M_2} \cdot N_2 + M_2^{M_2}) \cdot \lvert V\rvert^5)$
Muller	$n,\, d_1, \ldots, d_n,\, \lvert\mathcal{F}_1\rvert, \ldots \lvert\mathcal{F}_n\rvert$	$O((2^{M_3} \cdot N_3 + M_3^{M_3}) \cdot \lvert V\rvert^5)$

Table 2. Fixed parameter tractability of ordered games $(G, \Omega_1, \ldots, \Omega_n, \precsim)$: for $i \in \{1,\ldots,n\}$, k_i denotes the number of pairs and d_i denotes the number of colors of p_i. Sizes $s(n)$ and $s'(n)$ are resp. upper bounded by 2^n and 2^{2^n}. For $j \in \{1,2,3\}$, $M_j = 2^{m_j}$, where $m_1 = \sum_{i=1}^n 2 \cdot k_i$, and $m_2 = m_3 = \sum_{i=1}^n d_i$; and $N_1 = s(n) \cdot \sum_{i=1}^n 2 \cdot k_i$, $N_2 = s(n) \cdot \sum_{i=1}^n \frac{d_i^2}{2}$, $N_3 = s(n) \cdot \sum_{i=1}^n |\mathcal{F}_i| \cdot d_i$.

The proof of this theorem needs to show that solving the threshold problem for an ordered game $(G, \Omega_1, \ldots, \Omega_n, \precsim)$ is equivalent to solving a game (G, Ω) with a single objective Ω equal to the union of intersections of objectives taken in $\{\Omega_1, \ldots, \Omega_n\}$. It also needs to show that solving Boolean Büchi games is in FPT.

Monotonic preorders embedded in the subset preorder. We here present a *key tool* of this paper: solving the threshold problem for an ordered game $(G, \Omega_1, \ldots, \Omega_n, \precsim)$ is equivalent to solving a game (G, Ω) with a single objective Ω equal to the union of intersections of objectives taken in $\{\Omega_1, \ldots, \Omega_n\}$. The arguments are the following ones. (1) We consider the set $\{0,1\}^n$ of payoffs ordered with $\precsim$ as well as

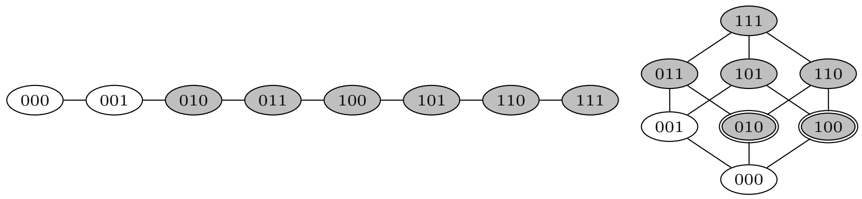

Fig. 2. Gray nodes represent the set of payoffs $\nu \succsim \mu = 010$ for the lexicographic preorder and its embedding for the subset preorder. The elements of $\mathsf{M}(\mu) = \{010, 100\}$ are doubly circled nodes.

ordered with the subset preorder $\subseteq$ (see the example of Figure 2 where $\precsim$ is the lexicographic preorder). To any payoff $\nu \in \{0,1\}^n$, we associate the set $\delta_\nu = \{i \in \{1,\dots,n\} \mid \nu_i = 1\}$ containing all indices i such that objective Ω_i is satisfied. (2) Consider the set of payoffs $\nu \succsim \mu$ embedded in the set $\{0,1\}^n$ ordered with $\subseteq$. By monotonicity of $\precsim$, we obtain an upper-closed set S that can be represented by the antichain of its *minimal elements* (with respect to $\subseteq$), that we denote by $\mathsf{M}(\mu)$. (3) $\mathcal{P}_1$ can ensure a payoff $\succsim \mu$ if and only if he has a strategy such that any consistent outcome π has a payoff $\nu^* \supseteq \nu$ for some $\nu \in \mathsf{M}(\mu)$, equivalently such that π satisfies (at least) the conjunction of the objectives Ω_i such that $\nu_i = 1$. (4) The objective Ω of $\mathcal{P}_1$ is thus a disjunction (over $\nu \in \mathsf{M}(\mu)$) of conjunctions (over $i \in \delta_\nu$) of objectives Ω_i. This statement is formulated in the next theorem (see again Figure 2).

Theorem 8. *Let $(G, \Omega_1, \dots, \Omega_n, \precsim)$ be an ordered game, $\mu \in \{0,1\}^n$ be some threshold, and v_0 be an initial vertex. Then, $\mathcal{P}_1$ can ensure a payoff $\succsim \mu$ from v_0 in $(G, \Omega_1, \dots, \Omega_n, \precsim)$ if and only if $\mathcal{P}_1$ has a winning strategy from v_0 in the game (G, Ω) with the objective $\Omega = \cup_{\nu \in \mathsf{M}(\mu)} \cap_{i \in \delta_\nu} \Omega_i$.* □

Note that we obtain the following corollary as a direct consequence of Theorem 8 and Martin's theorem [27].

Corollary 9. *Let $(G, \Omega_1, \dots, \Omega_n)$ be an ordered game. If $\Omega_1, \dots, \Omega_n$ are Borel sets, then $\mathcal{P}_1$ has a strategy to ensure a payoff $\succsim \mu$ from v_0 if and only if it is not the case that $\mathcal{P}_2$ has a strategy to avoid a payoff $\succsim \mu$ from v_0.* □

Parameterized complexity of Boolean Büchi games. In order to show that solving the threshold problem for ordered games is in FPT, we need to recall some known results of parameterized complexity for games with a single objective and to prove that solving Boolean Büchi games belongs to FPT.

It is proved in [20] that generalized reachability games belong to FPT. Parity, Rabin, Streett, and Muller games are shown to be FPT-interreducible in [4]. Very recently, Calude and al. provided a quasipolynomial time algorithm for parity games and showed that parity games are in FPT [10]. It follows that Rabin, Streett, and Muller games also belong to FPT. All these results are summarized in the next theorem with the related complexities.

Theorem 10. *Solving generalized reachability, parity, Rabin, Streett, and Muller games is in FPT. Generalized reachability (resp. parity, Muller) games are solvable with an algorithm running in $O(2^m \cdot (|V| + |E|))$ (resp. $O(|V|^5) + g(d)$, $O((d^d \cdot |V|)^5)$) time, where parameter m is the number of reachability objectives, parameter d is the number of colors, and g is some computable function.*

Proposition 11. *Solving Boolean Büchi games (G, Ω) is in FPT, with an algorithm in $O(2^M \cdot |\phi| + (M^M \cdot |V|)^5)$ time with $M = 2^m$ such that m is the number of variables of ϕ in the Boolean Büchi objective Ω.*

Proof. Let us show the existence of an FPT-reduction from Boolean Büchi games to Muller games. For this purpose, consider a Boolean Büchi game (G, Ω) with $\Omega = \mathsf{BooleanBuchi}(\phi, U_1, \dots, U_m)$, where ϕ is a Boolean formula over variables $x_1, \dots, x_m$, and m is seen as a parameter. We build an adequate Muller game $(G, \mathsf{Muller}(p, \mathcal{F}))$ on the same game structure and parameterized by the number of colors. The coloring function p and the family $\mathcal{F}$ are constructed as follows.

To any vertex $v \in V$, we associate the vector $\mu^v \in \{0,1\}^m$ such that $\mu^v_i = 1$ if $v \in U_i$ and 0 otherwise. Intuitively, we keep track for all i, whether a vertex belongs to U_i or not. For each $j \in \{0, \ldots, 2^m - 1\}$ we denote by $\mathsf{bin}(j)$ its binary encoding on m bits (this notation is extended to subsets of integers). Consider the coloring function $p\colon V \to \{0, \ldots, 2^m - 1\}$ that associates with each vertex v the color $p(v)$ such that $\mathsf{bin}(p(v)) = \mu^v$. The number M of colors is thus equal to 2^m. One can notice that $(*)$ a play π visits a vertex $v \in U_i$ if and only if π visits a color j with a binary encoding $\mathsf{bin}(j) = \mu$ such that $\mu_i = 1$.

To any subset F of $p(V)$, we associate the truth assignment $\chi(F) \in \{0,1\}^m$ of variables $x_1, \ldots, x_m$ such that for all i, $\chi(F)_i = 1$ if there exists $\mu \in \mathsf{bin}(F)$ with $\mu_i = 1$, and 0 otherwise. The idea (by $(*)$) is that the set F of colors visited infinitely often by a play π corresponds to the set $\mathsf{Inf}(\pi)$ of vertices visited infinitely often, such that $\chi(F)_i = 1$ if and only if $\mathsf{Inf}(\pi) \cap U_i \neq \emptyset$. We then define $\mathcal{F} = \{F \subseteq p(V) \mid \chi(F) \models \phi\}$, that is, $\mathcal{F}$ corresponds to the set of all truth assignments satisfying ϕ.

In this way we have the announced FPT-reduction: First, parameter $M = 2^m$ only depends on parameter m. Second, we have that $\mathcal{P}_1$ is winning in $(G, \mathsf{BooleanBuchi}(\phi, U_1, \ldots, U_m))$ from an initial vertex v_0 if and only if he is winning in $(G, \mathsf{Muller}(p, \mathcal{F}))$ from v_0. Indeed, a play π satisfies $(\phi, U_1, \ldots, U_m)$ if and only if the truth assignment ($x_i = 1$ if and only $\mathsf{Inf}(\pi) \cap U_i \neq \emptyset$, and $x_i = 0$ otherwise) satisfies ϕ. This is equivalent to have that $F = p(\mathsf{Inf}(\pi))$ belongs to $\mathcal{F}$ (by definition of $\chi(F)$), that is, π belongs to $\mathsf{Muller}(p, \mathcal{F})$. Third, the construction of the Muller game is in $O(2^{2^m} \cdot |\phi|)$ time since it requires $O(|V| + |E|)$ time for the game structure, $O(m \cdot |V|)$ time for the coloring function p, and $O(2^{2^m} \cdot |\phi|)$ time for the family $\mathcal{F}$.

From this FPT-reduction and by Theorem 10, we have an algorithm solving the Boolean Büchi game in $O(2^M \cdot |\phi| + (M^M \cdot |V|)^5)$ time, where $M = 2^m$. ◻

Proof of Theorem 7. By Theorem 8, we know that solving the threshold problem for an ordered game $(G, \Omega_1, \ldots, \Omega_n, \precsim)$ is equivalent to solving a game (G, Ω) with a single objective $\Omega = \cup_{\nu \in \mathsf{M}(\mu)} \cap_{i \in \delta_\nu} \Omega_i$. Thanks to this equivalence, we provide a proof of Theorem 7 with the parameterized complexities given in Table 2. This proof uses two sizes depending on the number n of objectives:

- the *size* $s(n)$ of $\mathsf{M}(\mu)$. It is upper bounded by 2^n (an antichain of maximum size in the subset preorder over $\{0,1\}^n$ is of exponential size $\binom{n}{\lfloor \frac{n}{2} \rfloor}$).
- the *size* $s'(n)$ defined as follows. In case of Büchi objectives Ω_i, we need to rewrite the objective $\cup_{\nu \in \mathsf{M}(\mu)} \cap_{i \in \delta_\nu} \Omega_i$ in conjunctive normal form $\cap_k \cup_l \Omega'_{k,l}$ with $\Omega'_{k,l} \in \{\Omega_1, \ldots, \Omega_n\}$. We denote by $s'(n)$ the size of this disjunction. It is bounded by 2^{2^n}.

In Section 4 we will show that, for several objectives, we can go beyond the fixed parameter tractability of Theorem 7 by providing polynomial time algorithms when the sizes $s(n)$ and $s'(n)$ are polynomial in n.

Proof (of Theorem 7). By Theorem 8, solving the threshold problem for an ordered game $(G, \Omega_1, \ldots, \Omega_n, \precsim)$ is equivalent to solving a classical game (G, Ω) with $\Omega = \cup_{\nu \in \mathsf{M}(\mu)} \cap_{i \in \delta_\nu} \Omega_i$. We have $|\mathsf{M}(\mu)| = s(n)$ and $|\delta_\nu| \leq n \; \forall \nu \in \mathsf{M}(\mu)$. Recall that $s(n) \leq 2^n$ and $s'(n) \leq 2^{2^n}$.

We first show that the threshold problem for ordered reachability, safety, Büchi, co-Büchi, and explicit Muller games is in FPT with parameter n.

- If each Ω_i is a reachability (resp. safety) objective, then (G, Ω) is a UI reachability (resp. safety) game that can be solved in $O(2^n \cdot (|V| + |E|))$ time by Theorem 5.
- If Ω is a union of intersections of Büchi objectives, then it can be rewritten as the intersection of unions of Büchi objectives which is a generalized Büchi objective with at most $s'(n)$ target sets. The latter game is solved in $O(s'(n) \cdot |V|^2) = O(2^{2^n} \cdot |V|^2)$ time by Theorem 5. The union of intersections of co-Büchi objectives is the complementary of a generalized Büchi objective with at most $s(n)$ target sets, leading to an algorithm in $O(s(n) \cdot |V|^2) = O(2^n \cdot |V|^2)$ time.
- If each Ω_i is an explicit Muller objective $\mathsf{ExplMuller}(\mathcal{F}_i)$ then Ω is again an explicit Muller objective $\mathsf{ExplMuller}(\mathcal{F})$ where $|\mathcal{F}| \leq \sum_{\nu \in \mathsf{M}(\mu)} \min_{j \in \delta_\nu} |\mathcal{F}_j|$ by Item 4 of Proposition 6. The latter game is solved in $O(|\mathcal{F}| \cdot (|V| \cdot |E| + |\mathcal{F}|)^2)$ time by Theorem 5. We thus get a complexity in $O((s(n) \cdot \max_i |\mathcal{F}_i|)^3 \cdot |V|^2 \cdot |E|^2) = O(2^{3n} \cdot (\max_i |\mathcal{F}_i|)^3 \cdot |V|^2 \cdot |E|^2)$ time.

We now show that the threshold problem for ordered parity, Rabin, Streett, and Muller games is in FPT thanks to Proposition 11.

- If each Ω_i is a parity objective with d_i colors, then Ω is a Boolean Büchi objective of size $|\phi| \leq s(n) \cdot \sum_{i=1}^{n} \frac{d_i^2}{2}$, and with $m = \sum_{i=1}^{n} d_i$ variables by Proposition 6 and as $\cup_{\nu \in \mathsf{M}(\mu)} \{\Omega_i \mid i \in \delta_\nu\} \subseteq \{\Omega_1, \ldots, \Omega_n\}$. By Proposition 11, the latter game is in FPT with an algorithm in $O(2^M \cdot |\phi| + (M^M \cdot |V|)^5)$ time with $M = 2^m$. Thus the threshold problem for ordered parity games is in FPT with parameters $n, d_1, \ldots, d_n$, with an algorithm in $O((2^M \cdot N + M^M) \cdot |V|^5)$ time with $N = s(n) \cdot \sum_{i=1}^{n} \frac{d_i^2}{2}$.
- The arguments are similar for ordered Rabin, Streett, and Muller games. The only differences are the bound on size $|\phi|$, the number m of variables and the parameters (see Table 2). For Rabin and Streett games, the parameters are $n, k_1, \ldots, k_n$ and we have $|\phi| \leq s(n) \cdot \sum_{i=1}^{n} 2 \cdot k_i = N$ and $m = \sum_{i=1}^{n} 2 \cdot k_i$. For Muller games, the parameters are $n, d_1, \ldots, d_n, |\mathcal{F}_1|, \ldots, |\mathcal{F}_n|$ and we have $|\phi| \leq s(n) \cdot \sum_{i=1}^{n} |\mathcal{F}_i| \cdot d_i = N$ and $m = \sum_{i=1}^{n} d_i$.

□

4 Ordered games with a compact embedding

In the previous section, we have shown that solving the threshold problem for ordered ω-regular games is in FPT. This result depends on sizes $s(n)$ and $s'(n)$ which vary with the number n of objectives. In this section, we study ordered games such that these sizes are polynomial in n.

Preorders with a compact embedding in the subset preorder. An ordered game $(G, \Omega_1, \ldots, \Omega_n, \precsim)$ has a *compact embedding* (in the subset preorder) if both sizes $s(n)$ and $s'(n)$ are polynomial in n. While the threshold problem is in FPT for ordered Büchi, co-Büchi, and explicit Muller games, it becomes polynomial as soon as their preorder has a compact embedding. This is a direct consequence of Table 2, rows 2-4.

Theorem 12. *The threshold problem is solved in polynomial time for ordered Büchi, co-Büchi, and explicit Muller games with a compact embedding.*

One can easily prove that ordered games using the subset or the maximize preorder have a compact embedding. We will later prove that this also holds for the lexicographic preorder. Nevertheless it is not the case for the counting preorder. Indeed solving the threshold problem for counting Büchi games is co-NP-complete [7].

Recall that solving the threshold problem for ordered Büchi games reduces to solving some UI Büchi game (by Theorem 8). Whereas solving the latter games is coNP-complete by Theorem 5, solving the threshold problem for ordered Büchi games is only polynomial when they have a compact embedding (see Theorem 12).

There is no hope to extend Theorem 12 to the other ω-regular objectives studied in this article, unless P = PSPACE. Indeed, we have PSPACE-hardness of the threshold problem for the following lexicographic games.

Theorem 13. *(1) Lexicographic games have a compact embedding and (2) the threshold problem is* PSPACE-*hard for lexicographic reachability, safety, Rabin, Streett, parity, and Muller games.*

Note that we obtain the following corollary as a direct consequence of Theorems 12 and 13.

Corollary 14. *The threshold problem for lexicographic Büchi, co-Büchi and explicit Muller games is* P-*complete and is* PSPACE-*complete for lexicographic safety, reachability, parity, Streett, Rabin and Muller games.*

Proof. The P-membership for lexicographic Büchi, co-Büchi, and explicit Muller games games follows from Theorem 12 and Part (1) of Theorem 13, while P-hardness follows from Theorem 5 as with $n = 1$ lexicographic Büchi, co-Büchi and explicit Muller games are respectively (classical) Büchi, co-Büchi, and explicit Muller games. For the other ω-regular objectives, PSPACE-hardness follows from Part (2) of Theorem 13, while PSPACE-membership follows from the work of Bouyer et al. [7]. □

The rest of this section is devoted to the proof of Theorem 13.

Lexicographic games. We now focus on the lexicographic preorder $\precsim$. Let us first provide several useful terminology and comments on this preorder. Recall that the lexicographic preorder is monotonic. It is also total, hence $x \sim y$ if and only if $x = y$, and $x \prec y$ if and only if $\neg(y \precsim x)$. Given a vector $x \in \{0,1\}^n$, we denote by $\overline{x}$ the *complement* of x, i.e. $\overline{x}_i = 1 - x_i$, for all $i \in \{1, \ldots, n\}$. We denote by $x - 1$ the *predecessor* of $x \neq 0^n$, that is, the greatest vector which is strictly smaller than x. We define the *successor* $x + 1$ of x similarly. In the sequel, as the threshold problem is trivial for $x = 0^n$, we do not consider this threshold. By abuse of notation, we keep writing $x \in \{0,1\}^n$ without mentioning that $x \neq 0^n$. We denote by $\mathsf{Last}_1(x)$ the last index i of x such that $x_i = 1$, i.e. $\mathsf{Last}_1(x) = \max\{i \in \{1, \ldots, n\} \mid x_i = 1\}$. Note that $\mathcal{P}_1$ can ensure a payoff $\succsim x \neq 0^n$ if and only if he can ensure a payoff $\succ x - 1$, and when $\mathcal{P}_2$ can avoid a payoff $\succsim x$, we rather say that $\mathcal{P}_2$ can *ensure* a payoff $\prec x$.

We now prove that the lexicographic games have a compact embedding (Part (1) of Theorem 13): we first show that $s(n)$ is polynomial in Proposition 15, and we then show that $s'(n)$ is also polynomial in Proposition 18.

Proposition 15. *Let $x \in \{0,1\}^n$. Then the set $\mathsf{M}(x)$ is equal to $\{x\} \cup \{y^j \in \{0,1\}^n \mid x_j = 0 \wedge j < \mathsf{Last}_1(x)\}$, where for all $j \in \{1, \ldots, \mathsf{Last}_1(x) - 1\}$, we define the vector $y^j \in \{0,1\}^n$ as equal to $x_1 \ldots x_{j-1}10^{n-j}$ (x and y^j share the same (possibly empty) prefix $x_1 \ldots x_{j-1}$). Moreover, $s(n) = |\mathsf{M}(x)| \leq n$.*

Example 16. Consider the vector $x = 0010100$ such that $\mathsf{Last}_1(x) = 5$. Then, the set $\mathsf{M}(x)$ is equal to $\{x\} \cup \{1000000, 0100000, 0011000\}$.

Proof (of Proposition 15). We recall that $\mathsf{M}(x)$ is the set of minimal elements (with respect to the subset preorder $\subseteq$) of the set of payoffs $y \succsim x$ embedded in the set $\{0,1\}^n$ ordered with $\subseteq$. Let us show both inclusions between $\mathsf{M}(x)$ and $M = \{x\} \cup \{y^j \in \{0,1\}^n \mid x_j = 0 \wedge j < \mathsf{Last}_1(x)\}$.

Let $y \in \mathsf{M}(x)$. If $y = x$, then trivially $y \in M$. Otherwise, assume $y \succ x$ and let j be the first index such that $y_j = 1$ and $x_j = 0$. Note that $x_1 \ldots x_{j-1} = y_1 \ldots y_{j-1}$ since $y \succ x$. We associate with y the vector $y^j = y_1 \ldots y_{j-1}10^{n-j}$. Note that $y^j \succ x$. By minimality of y and by construction of y^j, we obtain $y = y^j$ showing that $y \in M$.

For the second inclusion, as the lexicographic preorder is monotonic, we have $x \in \mathsf{M}(x)$. Now, consider some $y^j \in M$ such that $x_j = 0$ and $j < \mathsf{Last}_1(x)$. Let us show that y^j belongs to $\mathsf{M}(x)$, that is, $y^j \succsim x$ and there is no $y \succsim x$, $y \neq y^j$, such that $y \subset y^j$ (i.e. $\{i \mid y_i = 1\} \subset \{i \mid y^j_i = 1\}$). First, we clearly have $y^j \succsim x$ since $y^j = x_1 \ldots x_{j-1}10^{n-j}$ and $x_j = 0$. Towards a contradiction, assume now that there exists some $y \succsim x$, $y \neq y^j$, such that $y \subset y^j$. Let i be the first index such that $y_i = 0$ and $y^j_i = 1$. As $y \subset y^j$, we have $i \leq j$. If $i < j$, then y has $x_1 \ldots x_{i-1}0$ as prefix, $y^j_i = x_i = 1$, showing that $y \prec x$ in contradiction with $y \succsim x$. If $i = j$, then $y = x_1 \ldots x_{j-1}0^{n-j+1}$, and again $y \prec x$ since $j < \mathsf{Last}_1(x)$ by construction of y^j. ◻

In order to show that $s'(n)$ is polynomial in Proposition 18, we need to proof the following proposition that establishes the link between the duality between a payoff defined with some objectives and the payoff defined with the opposite objectives.

Proposition 17. *Let $(G, \Omega_1, \ldots, \Omega_n, \precsim)$ be a lexicographic game, $\mu \in \{0,1\}^n$ be a threshold, and v_0 be an initial vertex. Then $\mathcal{P}_1$ can ensure a payoff $\succeq \mu$ in the lexicographic game $(G, \Omega_1, \ldots, \Omega_n, \precsim)$ if and only if $\mathcal{P}_1$ can ensure a payoff $\precsim \overline{\mu}$ in the lexicographic game $(G, \overline{\Omega_1}, \ldots, \overline{\Omega_n}, \precsim)$.*

Proof. Recall that $\mathcal{P}_1$ can ensure a payoff $\succeq \mu$ from v_0 in the lexicographic game $(G, \Omega_1, \ldots, \Omega_n, \precsim)$ if and only if he has a winning strategy from v_0 for the objective $\{\pi \mid \mathsf{Payoff}(\pi) = \nu \succeq \mu\}$. Moreover, for any i, $\nu_i = 1$ if and only if $\pi \in \Omega_i$, i.e. $\pi \notin \overline{\Omega_i}$, and $\nu_i = 0$ if and only if $\pi \notin \Omega_i$, i.e. $\pi \in \overline{\Omega_i}$. Thus, $\overline{\nu}_i = 0$ iff $\pi \notin \overline{\Omega_i}$ and $\overline{\nu}_i = 1$ iff $\pi \in \overline{\Omega_i}$. Then, as $\nu \succeq \mu$ iff $\overline{\nu} \preceq \overline{\mu}$, we have that $\mathcal{P}_1$ is winning from v_0 for the objective $\{\pi \mid \mathsf{Payoff}(\pi) = \nu \succeq \mu\}$ if and only if $\mathcal{P}_1$ has a strategy to ensure a payoff $\preceq \overline{\mu}$ from v_0 in the lexicographic game $(G, \overline{\Omega_1}, \ldots, \overline{\Omega_n}, \precsim)$. ◻

Proposition 18. *Let $(G, \Omega_1, \ldots, \Omega_n, \precsim)$ be a lexicographic Büchi game and $\mu \in \{0,1\}^n$. Then, the objective $\Omega = \cup_{\nu \in \mathsf{M}(\mu)} \cap_{i \in \delta_\nu} \Omega_i$ can be rewritten in conjunctive normal form with a conjunction of size $s'(n) \leq n$.*

Proof. By Proposition 17 and Item 1 of Proposition 6, $\mathcal{P}_1$ can ensure a payoff $\gtrsim \mu$ in $(G, \Omega_1, \ldots, \Omega_n, \precsim)$ if and only if $\mathcal{P}_1$ can ensure a payoff $\precsim \overline{\mu}$ in the lexicographic co-Büchi game $(G, \overline{\Omega_1}, \ldots, \overline{\Omega_n}, \precsim)$. By Thereom 8 and Corollary 9, equivalently, $\mathcal{P}_2$ cannot satisfy the objective $\cup_{\nu \in \mathsf{M}(\overline{\mu}+1)} \cap_{i \in \delta_\nu} \overline{\Omega_i}$. This is equivalent to say that $\mathcal{P}_1$ can satisfy the complement of the latter objective, that is, the objective $\cap_{\nu \in \mathsf{M}(\overline{\mu}+1)} \cup_{i \in \delta_\nu} \Omega_i$. We have $|\mathsf{M}(\overline{\mu}+1)| \leq n$ by Proposition 15. ◻

We finally prove Part (2) of Theorem 13.

Proof (of Theorem 13, Part (2)). Let us study the complexity lower bounds.

- The PSPACE-hardness of the threshold problem for lexicographic reachability (resp. safety) games is obtained thanks to a polynomial reduction from solving generalized reachability games which is PSPACE-complete by Theorem 5. Let (G, Ω) be a generalized reachability game with $\Omega = \mathsf{GenReach}(U_1, \ldots, U_n)$. Let $(G, \Omega_1, \ldots, \Omega_n, \precsim)$ be the lexicographic reachability (resp. safety) game with $\Omega_i = \mathsf{Reach}(U_i)$ (resp. $\Omega_i = \mathsf{Safe}(U_i^c)$) $\forall i$.
  - Reachability: We have that $\mathcal{P}_1$ is winning in (G, Ω) from v_0 if and only if $\mathcal{P}_1$ can ensure a payoff $\gtrsim \mu = 1^n$ from v_0 in the lexicographic reachability game $(G, \Omega_1, \ldots, \Omega_n, \precsim)$.
  - Safety: We claim that $\mathcal{P}_1$ is winning in (G, Ω) from v_0 if and only if $\mathcal{P}_1$ can ensure a payoff $\gtrsim \mu = 0^{n-1}1$ from v_0 in the lexicographic safety game. This follows from the determinacy of generalized reachability games, and from the fact that $\mathcal{P}_1$ can ensure a payoff $\gtrsim \mu$ from v_0 in the lexicographic safety game if and only if $\mathcal{P}_2$ is losing in the generalized reachability game (G, Ω) from v_0.
- The hardness of the threshold problem for lexicographic parity games is obtained thanks to a polynomial reduction from solving games (G, Ω) the objective Ω of which is a union of a Rabin objective and a Streett objective, which is known to be PSPACE-complete [2]. Let $\Omega = \mathsf{Rabin}((E_i, F_i)_{i=1}^{n_1}) \cup \mathsf{Streett}((E_i, F_i)_{i=n_1+1}^{n})$. As any Rabin (resp. Streett) objective is the union (resp. intersection) of parity objectives by Item 3 of Proposition 6, we can rewrite Ω as $\Omega = \cup_{i=1}^{n_1}(\mathsf{Parity}(p_i)) \cup (\cap_{i=n_1+1}^{n} \mathsf{Parity}(p_i))$, where all p_i are coloring functions. Let $(G, \Omega_1, \ldots, \Omega_n, \precsim)$ be the lexicographic parity game where $\Omega_i = \mathsf{Parity}(p_i)$ for all i. We claim that $\mathcal{P}_1$ is winning in the game (G, Ω) from v_0 if and only if $\mathcal{P}_1$ can ensure a payoff $\gtrsim \mu$ from v_0 in the lexicographic parity game $(G, \Omega_1, \ldots, \Omega_n, \precsim)$ where $\mu = 0^{n_1}1^{n-n_1}$. Indeed, if a play π satisfies $\mathsf{Payoff}(\pi) \gtrsim \mu$ then either $\mathsf{Payoff}(\pi) = \mu$ in which case $\pi \in \cap_{i=n_1+1}^{n} \mathsf{Parity}(p_i)$, i.e. π satisfies the Streett objective, or $\mathsf{Payoff}(\pi) \succ \mu$ in which case there exists $1 \in \{1, \ldots, n_1\}$ such that $\pi \in \mathsf{Parity}(p_i)$, i.e. π satisfies the Rabin objective. Conversely, if a play π satisfies the Streett or the Rabin objective then $\mathsf{Payoff}(\pi) \gtrsim \mu$ since $\mathsf{Payoff}(\pi) \gtrsim \mu$ (resp. $\succ \mu$) as soon as π satisfies the Streett (resp. Rabin) objective.
- As parity objectives are a special case of Rabin (Streett) objectives by Item 2 of Proposition 6, the lower bound follows (from the previous item) for both lexicographic Rabin and Streett games.
- Lexicographic Muller games with $n = 1$ and $\mu = 1$ are a special case of Muller games and solving the latter games is PSPACE-complete by Theorem 5.

This completes the proof. ◻

5 Values and optimal strategies in lexicographic games

In this section, we first recall the notion of values and optimal strategies. We then show how to compute the values in lexicographic games, and what are the memory requirements for the related optimal strategies. This yields a *full picture* of the study of lexicographic games, see Table 3. In this table, the first row indicates the complexity of the threshold problem (Corollary 14) and the remaining rows summarize the results on the values and the optimal strategies (Theorem 20 herafter).

Values and optimal strategies. In a lexicographic game, one can define the best reward that $\mathcal{P}_1$ can ensure from a given vertex, that is, the highest threshold μ for which $\mathcal{P}_1$ can ensure a payoff $\gtrsim \mu$. Dually, we can also define the worth reward that $\mathcal{P}_2$ can ensure. More precisely, if there exists some $\mu \in \{0, 1\}^n$ and two

	Reachability	Safety	Büchi	Co-Büchi	Explicit Muller	Parity	Rabin	Streett	Muller
Threshold problem	PSPACE-complete		P-complete			PSPACE-complete			
Values	exponential		polynomial			exponential			
$\mathcal{P}_1$ memory	exponential		linear	memoryless	exponential				
$\mathcal{P}_2$ memory			memoryless	linear					

Table 3. Overview of the results on lexicographic games with ω-regular objectives. The second row indicates the complexity time of computing the values. The third and last rows indicate the tight memory requirements of the winning and optimal strategies for both players.

strategies $\sigma_1^* \in \Sigma_1, \sigma_2^* \in \Sigma_2$ such that $\mathsf{Payoff}(\mathsf{Out}(v, \sigma_1, \sigma_2^*)) \precsim \mu \precsim \mathsf{Payoff}(\mathsf{Out}(v, \sigma_1^*, \sigma_2))$ for all strategies $\sigma_1 \in \Sigma_1, \sigma_2 \in \Sigma_2$, then μ is called the *value* $\mathsf{Val}(v)$ of v and σ_1^*, σ_2^* are called *optimal* strategies from v. Note that the play $\pi = \mathsf{Out}(v, \sigma_1^*, \sigma_2^*)$ consistent with both optimal strategies has payoff $\mathsf{Payoff}(\pi) = \mathsf{Val}(v)$. The lexicographic game $(G, \Omega_1, \dots, \Omega_n, \precsim)$ is called *determined*[6] if $\mathsf{Val}(v)$ exists for every $v \in V$.

Note that in the following definition, the infimum and supremum functions are applied with $\precsim$.

Definition 19. *Given a lexicographic game* $(G, \Omega_1, \dots, \Omega_n, \precsim)$*, for every vertex* $v \in V$*, the* upper value $\overline{\mathsf{Val}}(v)$ *and the* lower value $\underline{\mathsf{Val}}(v)$ *are defined as:*

$$\overline{\mathsf{Val}}(v) = \inf_{\sigma_2 \in \Sigma_2} \sup_{\sigma_1 \in \Sigma_1} \mathsf{Payoff}(\mathsf{Out}(v, \sigma_1, \sigma_2)) \textit{ and } \underline{\mathsf{Val}}(v) = \sup_{\sigma_1 \in \Sigma_1} \inf_{\sigma_2 \in \Sigma_2} \mathsf{Payoff}(\mathsf{Out}(v, \sigma_1, \sigma_2)).$$

The lexicographic game $(G, \Omega_1, \dots, \Omega_n, \precsim)$ is *determined* if, for every $v \in V$, $\underline{\mathsf{Val}}(v) = \overline{\mathsf{Val}}(v)$. In this case, we write $\mathsf{Val}(v) = \overline{\mathsf{Val}}(v) = \underline{\mathsf{Val}}(v)$ and we call $\mathsf{Val}(v)$ the *value* of v. Note that the inequality $\underline{\mathsf{Val}}(v) \precsim \overline{\mathsf{Val}}(v)$ always holds. If $\mathcal{P}_1$ (resp. $\mathcal{P}_2$) can ensure a payoff $\succsim \underline{\mathsf{Val}}(v)$ (resp. $\precsim \overline{\mathsf{Val}}(v)$) from v, his related winning strategy σ_1^* (resp. σ_2^*) is called *optimal from* v:

$$\underline{\mathsf{Val}}(v) \precsim \mathsf{Payoff}(\mathsf{Out}(v, \sigma_1^*, \sigma_2)) \quad \forall \sigma_2 \in \Sigma_2 \text{ and } \mathsf{Payoff}(\mathsf{Out}(v, \sigma_1, \sigma_2^*)) \precsim \overline{\mathsf{Val}}(v) \quad \forall \sigma_1 \in \Sigma_1.$$

Notice that in case of determinacy, the play $\pi = \mathsf{Out}(v, \sigma_1^*, \sigma_2^*)$ consistent with both optimal strategies has payoff $\mathsf{Payoff}(\pi) = \mathsf{Val}(v)$.

We have the following theorem for lexicographic games. The rest of the section is devoted to its proof.

Theorem 20. *(1) The value of each vertex in lexicographic Büchi, co-Büchi, and explicit Muller games can be computed with a polynomial time algorithm, and with an exponential time algorithm for lexicographic reachability, safety, parity, Rabin, Streett, and Muller games.*

(2) The following assertions hold for both winning strategies of the threshold problem and optimal strategies. Linear memory strategies are necessary and sufficient for $\mathcal{P}_1$ *(resp.* $\mathcal{P}_2$*) while memoryless strategies are sufficient for* $\mathcal{P}_2$ *(resp.* $\mathcal{P}_1$*) in lexicographic Büchi (resp. co-Büchi) games. Exponential memory strategies are both necessary and sufficient for both players in lexicographic reachability, safety, explicit Muller, parity, Rabin, Streett, and Muller games.*

Proof of Part (1) of Theorem 20. When the objectives are Borel sets, the following proposition states the determinacy of lexicographic games as well as the existence of optimal strategies. It also states that an algorithm for the threshold problem leads to an algorithm for computing the values with a complexity multiplied by n. Hence the first part of Theorem 20 immediately follows from Corollary 14 and Proposition 21.

Proposition 21. *Let* $(G, \Omega_1, \dots, \Omega_n)$ *be a lexicographic game. If* $\Omega_1, \dots, \Omega_n$ *are Borel sets, then the lexicographic game is determined and has optimal strategies. Moreover, if the threshold problem can be solved with an algorithm of complexity* $\mathbb{C}$*, then for all* $v \in V$*, the value* $\mathsf{Val}(v)$ *can be computed with an algorithm of complexity* $n \cdot \mathbb{C}$*.*

[6] The notion of determinacy is different from the one given previously for games (G, Ω) with a single objective Ω.

Proof. Let us show first that the lexicographic game $(G, \Omega_1, \ldots, \Omega_n)$ is determined. To this end, we use the following Folk property: if there exists some $\alpha \in \{0,1\}^n$ and two strategies $\sigma_1^* \in \Sigma_1, \sigma_2^* \in \Sigma_2$ such that $\mathsf{Payoff}(\mathsf{Out}(v, \sigma_1, \sigma_2^*)) \precsim \alpha \precsim \mathsf{Payoff}(\mathsf{Out}(v, \sigma_1^*, \sigma_2))$ for all strategies $\sigma_1 \in \Sigma_1, \sigma_2 \in \Sigma_2$, then $\alpha = \mathsf{Val}(v)$ and σ_1^*, σ_2^* are optimal strategies from v. Let v be a vertex. The set of thresholds μ is partitioned between the two players according to whether $\mathcal{P}_1$ (resp. $\mathcal{P}_2$) can ensure a payoff $\succsim \mu$ (resp. $\prec \mu + 1$) from v by Corollary 9. Let α be the highest threshold that $\mathcal{P}_1$ can ensure and σ_1^* be the corresponding winning strategy. By definition of α, $\mathcal{P}_2$ can ensure a payoff $\prec \mu + 1$ with a winning strategy σ_2^*. It follows that $\mathsf{Payoff}(\mathsf{Out}(v, \sigma_1, \sigma_2^*)) \precsim \alpha \precsim \mathsf{Payoff}(\mathsf{Out}(v, \sigma_1^*, \sigma_2))$ for all strategies $\sigma_1 \in \Sigma_1, \sigma_2 \in \Sigma_2$, and therefore we have $\mathsf{Val}(v) = \alpha$.

When the threshold problem is decidable, the procedure to compute the value $\mathsf{Val}(v)$ works as follows. The idea is to solve the threshold problem for different thresholds from vertex v in a way to compute the highest threshold μ for which $\mathcal{P}_1$ can ensure a payoff $\succsim \mu$. This threshold μ is the value $\mathsf{Val}(v)$.

First, we test whether $\mathcal{P}_1$ can ensure a payoff $\succsim 10^{n-1}$ from v. If this is the case, we set bit μ_1 to 1 and to 0 otherwise. Then, for $i \in \{2, \ldots, n\}$, we successively test whether $\mathcal{P}_1$ can ensure a payoff $\succsim \mu_1 \ldots \mu_{i-1} 10^{n-i}$ from v and we set bit μ_i to 1 if this is the case and to 0 otherwise. Thus, after those n solutions to the threshold problem, we obtain a threshold $\mu = \mu_1 \ldots \mu_n$ for which $\mathcal{P}_1$ can ensure a payoff $\succeq \mu$ from v. The complexity of the algorithm computing μ is thus in $n \cdot \mathbb{C}$. By using again the previous Folk property with the computed μ, we have that $\mu = \mathsf{Val}(v)$.

This concludes the proof. ◻

Remark 22. When the procedure given in the proof of Proposition 21 computes the value μ of a given vertex v, it also computes at the same time an optimal strategy from this vertex for both players. Indeed, the optimal strategy σ_1^* of $\mathcal{P}_1$ from v is his winning strategy (for the threshold problem) that ensures a payoff $\succsim \mu$ and that the optimal strategy σ_2^* of $\mathcal{P}_2$ from v is his winning strategy that ensures a payoff $\prec \mu + 1$. Notice that in this procedure σ_1^* (resp. σ_2^*) is the winning stategy of $\mathcal{P}_1$ (resp. $\mathcal{P}_2$) for the last bit μ_i set to 1 (resp. to 0). Therefore, in order to study some properties on optimal strategies (such as memory requirements), it is sufficient to study winning strategies for the threshold problem.

Example 23. Let us consider the lexicographic reachability game $(G, \Omega_1, \Omega_2, \Omega_3, \precsim)$ depicted on Figure 3 where $\Omega_1 = \mathsf{Reach}(\{v_1\})$, $\Omega_2 = \mathsf{Reach}(\{v_2, v_4\})$ and $\Omega_3 = \mathsf{Reach}(\{v_5\})$ and $\precsim$ is the lexicographic order. We apply the procedure described in Proposition 21 to compute $\mathsf{Val}(v_0)$ and the corresponding optimal strategies. For this purpose, we begin by testing whether $\mathcal{P}_1$ can ensure a payoff $\succsim 100$ from v_0. This is not the case since $\mathcal{P}_2$ can prevent him from visiting vertex v_1 by going from v_0 to v_2. In particular, this strategy of $\mathcal{P}_2$ ensures a payoff $\prec 100$. We fix $\mu_1 = 0$ and we now test whether $\mathcal{P}_1$ can ensure a payoff $\succsim 010$. This is the case since by going from v_3 to v_4, $\mathcal{P}_1$ is guaranteed to visit vertex v_4. We thus set $\mu_2 = 1$. The final test made is whether $\mathcal{P}_1$ can ensure a payoff $\succsim 011$. This is possible with the strategy that consists in going from v_3 to v_5. Indeed, the two possible outcomes consistent with this strategy are $v_0 v_1 v_3 v_5^\omega$ and $v_0 v_2 v_3 v_5^\omega$. The payoff of the first play is $101 \succsim 011$ while the latter payoff is 011. Hence we set $\mu_3 = 1$ and we get $\mathsf{Val}(\mathsf{v_0}) = 011$. The corresponding optimal strategies are to go from v_3 to v_5 for $\mathcal{P}_1$ (to ensure a payoff $\succsim 011$) and to go from v_0 to v_2 for $\mathcal{P}_2$ (to ensure a payoff $\prec 100$). Note that the outcome from v_0 consistent with those strategies is $v_0 v_2 v_3 v_5^\omega$ and that its payoff is indeed $\mathsf{Val}(v_0)$.

Proof of Part (2) of Theorem 20. Thanks to Remark 22, in order to prove the second part of Theorem 20, it is sufficient to study the memory requirements of winning strategies as those of optimal strategies are the same. Upper bounds on the memory sizes are obtained by analyzing the various reductions done in the proof of Theorem 7 in the case of a preorder with a compact embedding. Lower bounds for lexicographic Büchi and co-Büchi games are obtained thanks to a reduction from generalized Büchi games, and for the other lexicographic games thanks to the reductions proposed in the proof of Part (2) of Theorem 13.

Proof (of Part (2) of Theorem 20). By Remark 22, we only study the memory requirements of winning strategies for the threshold problem. The proof is split into two parts dealing first with the upper bounds and then with the lower bounds.

Concerning the upper bounds, we first recall that for any ordered game with a compact embedding, the proof of Theorem 7 yields a reduction of

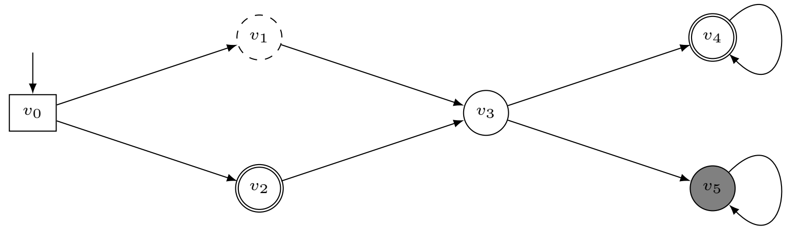

Fig. 3. The value of vertex v_0 is equal to 011.

- lexicographic reachability (resp. safety) games to UI reachability (resp. UI safety) games,
- lexicographic Büchi (resp. co-Büchi) games to (resp. the complement of) generalized Büchi games,
- lexicographic explicit Muller games to explicit Muller games and
- lexicographic parity, Rabin, Streett, and Muller games to Boolean Büchi games.

Those reductions do not modify the initial game structure and winning strategies for the games obtained by the reductions are winning strategies for the threshold problem of the original lexicographic games. As linear memory (resp. memoryless) strategies are sufficient for $\mathcal{P}_1$ (resp. $\mathcal{P}_2$) in generalized Büchi games and exponential memory strategies are sufficient for both players in Explicit Muller, Boolean Büchi, UI reachability, and UI safety games by Theorem 5, we obtain the expected upper bounds.

Concerning the lower bounds, we rely on the reductions proposed in the proof of Part (2) of Theorem 13.

- As there is a reduction from solving generalized reachability (resp. explicit Muller, Muller) games to solving the threshold problem for lexicographic reachability and safety (resp. explicit Muller, Muller) games, exponential memory is necessary for both players by Theorem 5.
- There is a reduction from solving games the objective of which is a union of a Rabin and a Streett objective to the threshold problem for lexicographic parity games. Thus, the latter problem is harder than solving both Rabin and Streett games, which implies that exponential memory is necessary for both players in lexicographic parity games by Theorem 5. This is also the case for lexicographic Rabin and Streett games, since parity objectives are a special case of Rabin (Streett) objectives by Item 2 of Proposition 6.
- It remains to show that $\mathcal{P}_1$ (resp. $\mathcal{P}_2$) needs linear memory in lexicographic Büchi (resp. co-Büchi) games. This is obtained thanks to Theorem 5 and the following reductions from generalized Büchi games. Let (G, Ω) with $\Omega = \mathsf{GenBuchi}(U_1, \ldots, U_n)$
  - Büchi case: We have that $\mathcal{P}_1$ is winning in (G, Ω) from v_0 if and only if $\mathcal{P}_1$ can ensure a payoff $\succeq 1^n$ from v_0 in the lexicographic Büchi game $(G, \mathsf{Buchi}(U_1), \ldots, \mathsf{Buchi}(U_n), \precsim)$.
  - Co-Büchi case: Note that any play π belongs to $\mathsf{GenBuchi}(U_1, \ldots, U_n)$ if and only if $\pi \notin \mathsf{CoBuchi}(U_i^c)$ for all i, i.e. $\mathsf{Payoff}(\pi) = 0^n$. Hence, $\mathcal{P}_1$ is winning in (G, Ω) from v_0 if and only if, taking on the role of $\mathcal{P}_2$, he can ensure a payoff $\prec \mu = 0^{n-1}1$ in the lexicographic co-Büchi game $(G, \mathsf{CoBuchi}(U_1^c), \ldots, \mathsf{CoBuchi}(U_n^c), \precsim)$.

This finishes the proof. □

6 Conclusion

In this paper, we have studied the parameterized complexity of the threshold problem for monotonically ordered games with ω-regular objectives when the set of objectives is taken as a parameter. This latter result is particularly relevant as in practice, the number of objectives is usually restricted. We have also studied the special case of the lexicographic order, and given a full ficture of the study of lexicographic games. In particular, we have shown how to compute the values in lexicographic games.

As future work, we would like to investigate notions of equilibria for those games, as well as subgame perfection: a strategy is subgame perfect if it ensures the maximal value that is achievable in every subgame. Also, we would like to study lexicographic games with quantitative objectives.

Acknowledgments. We would like to thank Antonia Lechner for useful discussions.

References

1. R. Alur, A. Kanade, and G. Weiss. Ranking automata and games for prioritized requirements. In *CAV Proceedings*, volume 5123 of *LNCS*, pages 240–253. Springer, 2008.
2. R. Alur, S. La Torre, and P. Madhusudan. Playing games with boxes and diamonds. In *CONCUR Proceedings*, volume 2761 of *LNCS*, pages 127–141. Springer, 2003.
3. C. Beeri. On the membership problem for functional and multivalued dependencies in relational databases. *ACM Trans. Database Syst.*, 5(3):241–259, 1980.
4. H. Björklund, S. Sandberg, and S. Vorobyov. On fixed-parameter complexity of infinite games. Technical Report 2003-038, Department of Information Technology, Uppsala University, Aug. 2003.
5. R. Bloem, K. Chatterjee, K. Greimel, T. A. Henzinger, and B. Jobstmann. Robustness in the presence of liveness. In *CAV Proceedings*, volume 6174 of *LNCS*, pages 410–424. Springer, 2010.
6. R. Bloem, K. Chatterjee, T. A. Henzinger, and B. Jobstmann. Better quality in synthesis through quantitative objectives. In *CAV Proceedings*, volume 5643 of *LNCS*, pages 140–156. Springer, 2009.
7. P. Bouyer, R. Brenguier, N. Markey, and M. Ummels. Concurrent games with ordered objectives. In *FOSSACS Proceedings*, volume 7213 of *LNCS*, pages 301–315. Springer, 2012.
8. V. Bruyère, Q. Hautem, and J. Raskin. On the complexity of heterogeneous multidimensional games. In *CONCUR Proceedings*, volume 59 of *LIPIcs*, pages 11:1–11:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.
9. V. Bruyère, N. Meunier, and J. Raskin. Secure equilibria in weighted games. In *CSL-LICS Proceedings*, pages 26:1–26:26. ACM, 2014.
10. C. S. Calude, S. Jain, B. Khoussainov, W. Li, and F. Stephan. Deciding parity games in quasipolynomial time. In *STOC Proceedings*, pages 252–263. ACM, 2017.
11. K. Chatterjee, W. Dvorák, M. Henzinger, and V. Loitzenbauer. Conditionally optimal algorithms for generalized Büchi games. In *MFCS Proceedings*, volume 58 of *LIPIcs*, pages 25:1–25:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.
12. K. Chatterjee and M. Henzinger. Efficient and dynamic algorithms for alternating büchi games and maximal end-component decomposition. *J. ACM*, 61(3):15:1–15:40, 2014.
13. K. Chatterjee, T. A. Henzinger, and N. Piterman. Generalized parity games. In *FOSSACS Proceedings*, volume 4423 of *LNCS*, pages 153–167. Springer, 2007.
14. T. Colcombet, M. Jurdzinski, R. Lazic, and S. Schmitz. Perfect half space games. In *LICS Proceedings*, pages 1–11. IEEE Computer Society, 2017.
15. R. G. Downey and M. R. Fellows. *Parameterized Complexity*. Springer Publishing Company, Incorporated, 2012.
16. S. Dziembowski, M. Jurdzinski, and I. Walukiewicz. How much memory is needed to win infinite games? In *IEEE Proceedings*, pages 99–110. IEEE Computer Society, 1997.
17. E. A. Emerson. Automata, tableaux and temporal logics (extended abstract). In *Logics of Programs Proceedings*, volume 193 of *LNCS*, pages 79–88. Springer, 1985.
18. E. A. Emerson and C. S. Jutla. The complexity of tree automata and logics of programs (extended abstract). In *FOCS Proceedings*, pages 328–337. IEEE Computer Society, 1988.
19. E. A. Emerson and C. S. Jutla. Tree automata, mu-calculus and determinacy (extended abstract). In *FOCS*, pages 368–377. IEEE Computer Society, 1991.
20. N. Fijalkow and F. Horn. Les jeux d'accessibilité généralisée. *Technique et Science Informatiques*, 32:931–949, 2013.
21. E. Grädel, W. Thomas, and T. Wilke, editors. *Automata, Logics, and Infinite Games: A Guide to Current Research [outcome of a Dagstuhl seminar, February 2001]*, volume 2500 of *LNCS*. Springer, 2002.
22. J. Gutierrez, A. Murano, G. Perelli, S. Rubin, and M. Wooldridge. Nash equilibria in concurrent games with lexicographic preferences. In *IJCAI Proceedings*, pages 1067–1073. ijcai.org, 2017.
23. F. Horn. Explicit Muller games are PTIME. In *FSTTCS Proceedings*, volume 2 of *LIPIcs*, pages 235–243. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2008.

24. P. Hunter and A. Dawar. Complexity bounds for regular games. In *MFCS Proceedings*, volume 3618 of *LNCS*, pages 495–506. Springer, 2005.
25. N. Immerman. Number of quantifiers is better than number of tape cells. *J. Comput. Syst. Sci.*, 22:384–406, 1981.
26. M. Jurdzinski. Deciding the winner in parity games is in UP $\cap$ co-UP. *Inf. Process. Lett.*, 68(3):119–124, 1998.
27. D. A. Martin. Borel determinacy. *Annals of Mathematics*, 102:363–371, 1975.
28. R. McNaughton. Infinite games played on finite graphs. *Ann. Pure Appl. Logic*, 65(2):149–184, 1993.
29. N. Piterman and A. Pnueli. Faster solutions of rabin and streett games. In *LICS Proceedings*, pages 275–284. IEEE Computer Society, 2006.
30. A. Pnueli and R. Rosner. On the synthesis of a reactive module. In *POPL Proceedings*, pages 179–190. ACM Press, 1989.
31. Y. Velner, K. Chatterjee, L. Doyen, T. A. Henzinger, A. M. Rabinovich, and J. Raskin. The complexity of multi-mean-payoff and multi-energy games. *Inf. Comput.*, 241:177–196, 2015.
32. U. Zwick and M. Paterson. The complexity of mean payoff games on graphs. *Theor. Comput. Sci., 158*, 959:343–359, 1996.