PLAYING SAFE, TEN YEARS LATER

THOMAS COLCOMBET ◉, NATHANAËL FIJALKOW ◉, AND FLORIAN HORN

[a] CNRS, IRIF, Université de Paris, Paris, France

[b] CNRS, LaBRI, Bordeaux, France and University of Warsaw, Warsaw, Poland

[c] CNRS, IRIF, Université de Paris, Paris, France

ABSTRACT. We consider two-player games over graphs and give tight bounds on the memory size of strategies ensuring safety objectives. More specifically, we show that the minimal number of memory states of a strategy ensuring a safety objective is given by the size of the maximal antichain of left quotients with respect to language inclusion. This result holds for all safety objectives without any regularity assumptions. We give several applications of this general principle. In particular, we characterize the exact memory requirements for the opponent in generalized reachability games, and we prove the existence of positional strategies in games with counters.

1. INTRODUCTION

Graphs games provide a mathematical framework to model reactive systems (we refer to [GTW02] for a survey on the topic, and to [FBB+23] for a recent textbook). We focus here on the Synthesis Problem to motivate the problem we consider, which is to characterize the amount of memory required in games with safety objectives.

The synthesis problem. The inputs of the synthesis problem are a *system* and a *specification*. The expected output is a *controller* for the system, that ensures the specification. We describe here a classical and well-studied approach to solve the synthesis problem through game theory. We model the system as a graph, whose vertices represent states and edges represent transitions. Its evolution consists in interactions between a controller and an environment, which is turned into a game on the graph between two players, Eve and Adam. If in a given state, the controller can choose the evolution of the system, then the corresponding vertex is controlled by Eve. If the system evolves in an uncertain way, we consider the worst-case scenario, where Adam controls those states. A pebble is initially placed on the vertex representing the initial state of the system, then Eve and Adam move this pebble along the edges. The sequence built describes a run of the system: Eve tries to ensure that it satisfies the specification. So, in order to synthesize a controller, we are interested in whether Eve can ensure this objective and what resources she needs. In particular, the most salient question is: what is the size of a *minimal* controller satisfying the

specification? Since a controller is here given by a strategy for Eve, this is equivalent to the following question: what is the minimal amount of memory used by a winning strategy? The following diagram shows the correspondence between the notions from the synthesis problem (left-hand side) and the game-theoretic notions (right-hand side).

$$\underbrace{\mathbb{S}}_{\text{system}}, \underbrace{\mathcal{C}}_{\text{controller}} \models \underbrace{\Phi}_{\text{specification}} \qquad \Longleftrightarrow \qquad \underbrace{\mathcal{A}}_{\text{arena (graph)}}, \underbrace{\sigma}_{\text{strategy}} \models \underbrace{W}_{\text{objective}}$$

Safety specifications. Since we consider non-terminating sequences, a specification is given by a language of infinite words. Of special interest are the specifications asserting that "nothing bad" ever happens; such specifications are called *safety specifications*. A safety specification is induced by a (possibly infinite and non-regular) set of bad prefixes P, and the specification is met by a run if no prefixes belong to P. Although quite simple, the safety specifications proved useful in both theory and practice, and are actively studied (we refer to [Kup14] for a survey).

Our contribution. In this paper, our goal is to characterize the memory requirements of (arbitrary) safety winning objectives. The reader with a background in game theory may be surprised, as it is well known that "safety games are positionally determined", implying that the quantity above is constant equal to one. The subtlety is that our setting is (much) more general than the classical notion of safety games: in a safety game, the goal is to avoid a set of forbidden edges, while a safety objective defines a set of forbidden prefixes independent of the arena. We show the following general principle:

> For a safety winning objective W, the minimal number of memory states of a winning strategy is exactly the cardinal of the maximal antichain of left quotients of W.

We refer to Section 4 for the missing definitions. Note that this result holds for all safety winning objectives, without any regularity assumption. The characterisation above holds for graphs with finite degree, however we can state and prove a variant lifting this assumption, but assuming that the set of left quotients is well founded with respect to inclusion. We state our main results in Section 3, and prove them in Section 4. We give several examples and applications in Section 5. For instance, it allows to characterize the memory requirements for the opponent in generalized reachability games, and to prove the existence of positional strategies in games with counters. This journal version additionally discusses related works in Section 6, many of them posterior to the publication of the conference version of this article [CFH14] in 2014.

2. Definitions

The games we consider are played on an *arena* $\mathcal{A} = (V, (V_\exists, V_\forall), E, c)$, consisting of a (finite or infinite) graph (V, E), a partition $(V_\exists, V_\forall)$ of the vertex set V: a vertex in $V_\exists$ belongs to Eve and in $V_\forall$ to Adam, and a coloring function $c : E \to A$ mapping edges to a color from a finite alphabet A. When drawing arenas, we will use circles for vertices owned by Eve and squares for those owned by Adam. Throughout this paper, we make the cosmetic assumption that graphs do not have dead-ends: for every vertex $v \in V$, there exists an edge $(v, v') \in E$.

Game. A *play* π is an infinite word of edges $e_0 \cdot e_1 \cdots$ that are consecutive: for all i, $e_i = (\_, v) \in E$ and $e_{i+1} = (v, \_) \in E$ for some $v \in V$. The prefix of length k of π is denoted π_k. A play π induces an infinite sequence of colors $c(\pi)$, obtaining by applying the coloring function c component-wise. We define *winning objectives* for a player by giving a set of infinite sequences of colors $W \subseteq A^\omega$. As we are interested in zero-sum games, *i.e.* where the winning objectives of the two players are opposite, if the winning objective for Eve is W, then the winning objective for Adam is $A^\omega \setminus W$. A *game* is a couple $\mathcal{G} = (\mathcal{A}, W)$ where $\mathcal{A}$ is an arena and W a winning objective.

Strategy. A *strategy* for a player is a function that prescribes, given a finite history of the play, the next move. Formally, a strategy for Eve is a function $\sigma : E^* \cdot V_\exists \to E$ such that for all $\pi \in E^*$ and $v \in V_\exists$ we have $\sigma(\pi \cdot v) = (v, \_) \in E$. Strategies for Adam are defined similarly, and usually denoted τ. Once a game $\mathcal{G} = (\mathcal{A}, W)$, a starting vertex v_0 and strategies σ for Eve and τ for Adam are fixed, there is a unique play $\pi(v_0, \sigma, \tau)$, which is said winning for Eve if its image by c belongs to W.

A strategy σ for Eve is *winning* if for all strategies τ for Adam, $\pi(q_0, \sigma, \tau)$ is winning. We say that Eve wins the game $\mathcal{G}$ from v_0 if she has a winning strategy from v_0, and denote $\mathcal{W}_E(\mathcal{G})$ the set of vertices from where Eve wins; we often say that $v \in \mathcal{W}_E(\mathcal{G})$ is winning. We define similarly $\mathcal{W}_A(\mathcal{G})$ for Adam to be the set of vertices from where Adam wins.

Memory. A *memory structure* is a deterministic state machine that reads the sequence of edges and abstracts its relevant informations into a memory state. Formally, a memory structure $\mathcal{M} = (M, m_0, \mu)$ for an arena consists of a set M of memory states, an initial memory state $m_0 \in M$ and an update function $\mu : M \times E \to M$. The update function takes as input the current memory state and the chosen edge to compute the next memory state. It can be extended to a function $\mu^* : E^* \to M$ by defining $\mu^*(\varepsilon) = m_0$ and $\mu^*(\pi \cdot e) = \mu(\mu^*(\pi), e)$. Given a memory structure $\mathcal{M}$ and a next-move function $\nu : V_\exists \times M \to E$, we can define a strategy σ for Eve by $\sigma(\pi \cdot v) = \nu(v, \mu^*(\pi \cdot v))$. A strategy with memory structure $\mathcal{M}$ has finite memory if M is a finite set, and we write $|\mathcal{M}|$ for the size of M. It is *memoryless*, or *positional* if M is a singleton: it only depends on the current vertex. Hence a memoryless strategy can be described as a function $\sigma : V_\exists \to E$.

An arena and a memory structure induce an expanded arena where the current memory state is computed online. Formally, the arena $\mathcal{A} = (V, (V_\exists, V_\forall), E, c)$, the memory structure $\mathcal{M}$ for $\mathcal{A}$ and a new coloring function $c' : E \times M \to A$ induce an expanded arena $\mathcal{A} \times \mathcal{M} = (V \times M, (V_\exists \times M, V_\forall \times M), E \times \mu, c')$, where $E \times \mu$ is defined by: $((v, m), (v', m')) \in E \times \mu$ if $(v, v') \in E$ and $\mu(m, (v, v')) = m'$.

From a memoryless strategy in $\mathcal{A} \times \mathcal{M}$, we can build a strategy in $\mathcal{A}$ using $\mathcal{M}$ as memory structure, which behaves as the original strategy. This key observation will be used several times in the paper.
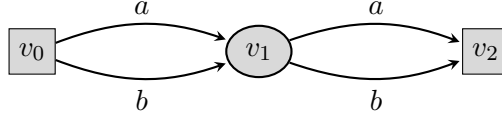
FIGURE 1. Memory is necessary for safety objectives: to avoid seeing both a and b, Eve must choose the same letter as Adam did in the first move, which requires two memory states.

3. STATEMENTS OF THE RESULTS

In this section, we consider a safety objective[1] W and compute the following quantity:

$$\operatorname{mem}(W) \ \dot{=} \ \sup_{\substack{\mathcal{G}=(\mathcal{A},W) \text{ game} \\ v_0 \text{ initial vertex}}} \ \inf_{\substack{\sigma \text{ winning strategy from } v_0 \\ \text{with memory structure } \mathcal{M}}} \ |\mathcal{M}| \ .$$

In words, $\operatorname{mem}(W)$ is the necessary and sufficient number of memory states for constructing a winning strategy in games with objective W. Equivalently:

- *upper bound:* for all games $\mathcal{G} = (\mathcal{A}, W)$, if Eve has a winning strategy from an initial vertex v_0, then she has a winning strategy using at most $\operatorname{mem}(W)$ memory states,
- *lower bound:* there exists a game $\mathcal{G} = (\mathcal{A}, W)$ and an initial vertex v_0 where Eve has a winning strategy, but no winning strategy using less than $\operatorname{mem}(W)$ memory states.

Note that we place no restrictions on the size of the games involved: they may be infinite. We explain now why our study of games with safety objectives is (much) more general than the classical notion of safety games.

Consider an arena $\mathcal{A}$: a safety *condition* is given by a subset $B \subseteq E$ of forbidden edges, inducing the winning condition

$$\{\rho = a_0 \cdot a_1 \cdots \in E^\omega \mid \text{for all } i, \ a_i \notin B\} \ .$$

Let A a set of colours. A safety *objective* is given by a subset $P \subseteq A^*$ of forbidden prefixes of colors, inducing the winning objective

$$\operatorname{Safe}(P) = \{\rho = a_0 \cdot a_1 \cdots \in A^\omega \mid \text{for all } i, \ a_0 \cdot a_1 \cdots a_i \notin P\} \ .$$

For each arena $\mathcal{A}$ equipped with a colouring function $c : E \to A$, this induces the winning condition

$$\{\rho = a_0 \cdot a_1 \cdots \in E^\omega \mid c(\rho) = c(a_0) \cdot c(a_1) \cdots \in \operatorname{Safe}(P)\}.$$

In other words, a condition is defined directly on an arena, while an objective is defined independently of the arena on the sequence of colours and induces a condition for each arena equipped with a colouring function. As an example, consider the safety objective $\operatorname{Safe}(P)$ over the colours $A = \{a, b, c\}$ where $P = A^* \cdot a \cdot A^* \cdot b \cup A^* \cdot b \cdot A^* \cdot a$: the objective $\operatorname{Safe}(P)$ specifies that a and b cannot be both seen along a play. This cannot be expressed by a safety condition, and $\operatorname{mem}(\operatorname{Safe}(P)) \geq 2$, as shown in Figure 1.

Lemma 1 (Folklore). *Let $\mathcal{G}$ be a game with a safety condition, and v_0 an initial vertex. If Eve has a winning strategy, then she has a positional winning strategy.*

---

[1]To be defined in this section.

Safety objectives form a very expressive class of objectives, as we will demonstrate in Section 5. The notion of safety objective originates from topological studies of the set of infinite words: the safety objectives are the closed sets for the Cantor topology, denoted Σ_1 in the corresponding Borel hierarchy.

Let us fix a safety objective $W = \mathrm{Safe}(P)$ induced by $P \subseteq A^*$. Let $w \in A^*$, define its left quotient as:

$$w^{-1}W = \{\rho \in A^\omega \mid w \cdot \rho \in W\}.$$

We denote $\mathrm{Res}(W)$ the set of left quotients of W. We mention some special left quotients: the initial one, $\varepsilon^{-1}W$ (equal to W), and the empty one, obtained as $w^{-1}W$ for any $w \in P$. As a small abuse of notations, from now on by "left quotient" we mean "non-empty left quotient of W".

From a left quotient $w^{-1}W$ and a letter $a \in A$, we define $(w^{-1}W) \cdot a$ as $(w \cdot a)^{-1}W$: it is easy to check that this is well defined (independent of the representant w chosen). Recall that $\mathrm{Res}(W)$ is finite if and only if W is regular, and in such case it can be used to describe the set of states of the minimal deterministic automaton recognizing W.

Before stating our main results, we need some order-theoretic definitions. The width of a partially ordered set $(X, \leq)$ is the cardinal of a maximal antichain of X with respect to $\leq$, *i.e.* the cardinal of a maximal set of pairwise incomparable elements. Recall that an antichan is a subset of X such that any two distinct elements are incomparable, and a chain is a subset of X such that any two elements are comparable. We say that $(X, \leq)$ is well founded if every chain contains a minimal element.

We write $\mathrm{mem}_{\text{fin-deg}}(W)$ for the quantity defined as $\mathrm{mem}(W)$ but restricting to arenas of finite degree: for every vertex, there are finitely many outgoing edges.

Theorem 1. *For all safety objectives W,*
- $\mathrm{mem}_{\text{fin-deg}}(W)$ *is the width of* $(\mathrm{Res}(W), \subseteq)$.
- *if* $(\mathrm{Res}(W), \subseteq)$ *is well-founded (in particular if W is regular), then* $\mathrm{mem}(W)$ *is the width of* $(\mathrm{Res}(W), \subseteq)$.

We present in 5 an example, called the outbidding games, showing that the well-founded assumption is necessary.

An objective W is half-positional if $\mathrm{mem}(W) = 1$. In the case of safety objectives, we obtain the following characterization:

Corollary 1. *For all safety objectives W, W is half-positional over arenas of finite degree if and only if the inclusion is a linear order over $\mathrm{Res}(W)$.*

If furthermore $(\mathrm{Res}(W), \subseteq)$ is well-founded, then W is half-positional (over all arenas) if and only if the inclusion is a linear order over $\mathrm{Res}(W)$.

4. Proofs

A First Upper Bound.

Lemma 2. *For all safety objectives $W = \mathrm{Safe}(P)$, for all games $\mathcal{G} = (\mathcal{A}, W)$, if Eve has a winning strategy from an initial vertex v_0, then she has a winning strategy using at most $|\mathrm{Res}(W)|$ memory states.*

Consequently,

$$\mathrm{mem}_{\textit{fin-deg}}(W) \;\leq\; \mathrm{mem}(W) \;\leq\; |\mathrm{Res}(W)| \;.$$

Proof. We construct a memory structure $\mathcal{M}$, as follows: $\mathcal{M} = (\mathrm{Res}(W), W, \nu)$, where $\nu(w^{-1}W, a) = (w^{-1}W) \cdot a$. At any point in the game, the memory state computed by $\mathcal{M}$ is the current left quotient. Let $\mathcal{A}$ be an arena. We construct the expanded arena $\mathcal{A} \times \mathcal{M}$ equipped with the coloring function $c' : E \times \mathrm{Res}(W) \to \{0, 1\}$ defined by:

$$c'(\_, w^{-1}W) = \begin{cases} 0 & \text{if } w^{-1}W = \emptyset \text{ (equivalently, } w \in P) \;, \\ 1 & \text{otherwise } . \end{cases}$$

We attach to $\mathcal{A} \times \mathcal{M}$ the safety condition induced by $B = \{0\}$, giving rise to the game $\mathcal{G} \times \mathcal{M} = (\mathcal{A} \times \mathcal{M}, \mathrm{Safe}(B))$. First observe that by construction, the plays in $\mathcal{A} \times \mathcal{M}$ are of the form $(e_0, c(e_0)^{-1}W) \cdot (e_1, c(e_0 \cdot e_1)^{-1}W) \cdots (e_k, c(e_0 \cdot e_1 \cdots e_k)^{-1}W)$, so by definition of c' a play is winning is $\mathcal{G} \times \mathcal{M}$ if and only if its projection (on the first component) is winning in $\mathcal{G}$.

It follows that a winning strategy for Eve in $\mathcal{G}$ from v_0 induces a winning strategy in $\mathcal{G} \times \mathcal{M}$ from (v_0, W). Now, thanks to Lemma 1, since Eve wins in $\mathcal{G} \times \mathcal{M}$, she has a positional winning strategy. This induces a winning strategy in $\mathcal{G}$ using $\mathcal{M}$ as memory structure, concluding the proof of Lemma 2. ∎

The game $\mathcal{G} \times \mathcal{M}$ defined above will be an important tool in the proofs to follow. We will also rely on the following remark: assume we want to prove that a strategy σ is winning. Then it is enough to show that for all plays π consistent with σ, for all k, $c(\pi_k)^{-1}W \neq \emptyset$, where π_k is the prefix of π of length k. This simple observation follows from the definition of safety objectives.

A Tighter Upper Bound. The memory structure $\mathcal{M}$ is not optimal. A first remark is that the empty left quotient (which exists if $W \neq A^\omega$) can be removed from the memory states as the game is lost. This is why by "left quotient" we mean "non-empty left quotient of W".

The second remark is the following: let L_1 and L_2 two left quotients of W, such that $L_1 \subseteq L_2$. With the same notations as above, consider a vertex v in the arena $\mathcal{A}$. If Eve wins from (v, L_1) in $\mathcal{G} \times \mathcal{M}$, then she also wins from (v, L_2): indeed, she can play as she would have played from (v, L_1). Since this ensures from v that all plays are winning for L_1, then a fortiori they are winning for L_2.

This suggests to restrict the memory states only to *minimally winning* left quotients with respect to inclusion. Two issues arise:

- which left quotients are winning depends on the current vertex, so the semantics of a memory state can no longer be *one* left quotient, but rather a left quotient for each possible vertex,
- there may not exist *minimally winning* left quotients.

For the sake of presentation, we first show how to deal with the first issue, assuming the second issue does not appear. Specifically, in the following lemma, we assume that $\mathrm{Res}(W)$ is finite (*i.e.* W is regular), implying the existence of minimally winning left quotients. We will later drop this assumption.

Lemma 3 (Upper bound in the regular case). *Let W a safety objective, we assume that $\mathrm{Res}(W)$ is finite.*

For all games $\mathcal{G} = (\mathcal{A}, W)$, if Eve has a winning strategy from an initial vertex v_0, then she has a winning strategy using at most K memory states, where K is the width of $(\mathrm{Res}(W), \subseteq)$.

The key intuition to have is the following: our goal is to construct a strategy σ such that for a vertex v, we associate to every memory state i a left quotient $L_i(v)$ ensuring that in a play π consistent with σ, then $L_i(v)$ is an under-approximation of $c(\pi)^{-1}W$, meaning $L_i(v) \subseteq c(\pi)^{-1}W$.

Proof. We use the same notations as for the proof of Lemma 2, and construct a smaller memory structure together with a winning strategy using this memory structure. In this proof, by winning we mean winning in the game $\mathcal{G} \times \mathcal{M}$.

Let K be the cardinal of the maximal antichain of left quotients of W. We construct a memory structure $\mathcal{M}^* = (\{1, \ldots, K\}, 1, \mu)$, and a strategy σ induced by the next-move function ν.

Let v be a vertex in $\mathcal{A}$. We consider the set of minimal left quotients L such that (v, L) is winning. (Here we use the finiteness of $\mathrm{Res}(W)$ to guarantee the existence of such left quotients.) This is an antichain, so there are at most K of them, we denote them by $L_1(v), \ldots, L_p(v)$, for some $p \leq K$. The *key* property is that for every left quotient L such that (v, L) is winning, there exists i such that $L_i(v) \subseteq L$. Furthermore, we choose $L_1(v_0)$ such that $L_1(v_0) \subseteq W$. (Indeed, by assumption (v_0, W) is winning.)

We define the update function: $\mu(i, (v, v'))$ is a j such that $L_j(v') \subseteq L_i(v) \cdot c(v, v')$. Note that in general, such a j may not exist; it does exist if $(v', L_i(v) \cdot c(v, v'))$ is winning, and we will prove that this will always be the case when playing the strategy σ.

We define the next-move function ν (inducing σ). Let $v \in V_\exists$, and consider $(v, L_i(v))$: since Eve wins from there, there exists an edge $(v, v') \in E$ such that $(v', L_i(v) \cdot c(v, v'))$ is winning. Define $\nu(v, i)$ to be this v'.

We show that the strategy σ is winning. Consider a play $\pi = (v_0, v_1) \cdot (v_1, v_2) \cdots$ consistent with σ, and $i_0 \cdot i_1 \cdots$ the sequence of memory states assumed along this play. Denote π_k the prefix of π of length k, we prove that for all k, $L_{i_k}(v_k) \subseteq c(\pi_k)^{-1}W$. Note that by definition, $(v_k, L_{i_k}(v_k))$ is winning, so $L_{i_k}(v_k) \neq \emptyset$, implying that $c(\pi_k)^{-1}W \neq \emptyset$.

We proceed by induction. For $k = 0$, it follows from $L_1(v_0) \subseteq W$. Let $k > 0$, the induction hypothesis is $L_{i_{k-1}}(v_{k-1}) \subseteq c(\pi_{k-1})^{-1}W$. We distinguish two cases.

- Either v_{k-1} belongs to Eve, then by construction of σ we have that $(v_k, L_{i_{k-1}}(v_{k-1}) \cdot c(v_{k-1}, v_k))$ is winning. It follows that the update function is well defined, and $L_{i_k}(v_k) \subseteq L_{i_{k-1}}(v_{k-1}) \cdot c(v_{k-1}, v_k)$, which together with the induction hypothesis implies $L_{i_k}(v_k) \subseteq c(\pi_k)^{-1}W$.
- Or v_{k-1} belongs to Adam. Since Adam cannot escape $\mathcal{W}_E(\mathcal{G} \times \mathcal{M})$, we have that $(v_k, L_{i_{k-1}}(v_{k-1}) \cdot c(v_{k-1}, v_k))$ is winning, and the same reasoning concludes.

It follows that the strategy σ is winning, concluding the proof of Lemma 3. ∎

We now get rid of the regularity assumption. This means that for a vertex v, there may not be a minimal left quotient L such that (v, L) is winning. We present two ways to get around this difficulty: either an assumption on W, or an assumption on the games.

Lemma 4 (Upper bound – well founded assumption). *Let W a safety objective, we assume that $(\mathrm{Res}(W), \subseteq)$ is well-founded.*

For all games $\mathcal{G} = (\mathcal{A}, W)$, *if Eve has a winning strategy from an initial vertex* v_0, *then she has a winning strategy using at most* K *memory states, where* K *is the width of* $(\mathrm{Res}(W), \subseteq)$.

Consequently, $\mathrm{mem}(W)$ *is smaller than or equal to the width of* $(\mathrm{Res}(W), \subseteq)$.

Note that if W is regular, then $(\mathrm{Res}(W), \subseteq)$ is finite, hence well-founded. The proof of Lemma 4 is the same as for Lemma 3, once we note that the well founded assumption implies that for any vertex v, there exists a minimal left quotient L such that (v, L) is winning.

Lemma 5 (Upper bound – finite degree assumption). *Let* W *a safety objective.*

For all games $\mathcal{G} = (\mathcal{A}, W)$ *with finite degree, if Eve has a winning strategy from an initial vertex* v_0, *then she has a winning strategy using at most* K *memory states, where* K *is the width of* $(\mathrm{Res}(W), \subseteq)$.

Consequently, $\mathrm{mem}_{fin\text{-}deg}(W)$ *is smaller than or equal to the width of* $(\mathrm{Res}(W), \subseteq)$.

The proof of Lemma 5 extends the proof of Lemma 3, getting rid of the well founded assumption. To this end, we need an equivalent formulation of the width of an ordered set $(X, \leq)$: recall that it is defined as the cardinal of a maximal antichain of X. Thanks to Dilworth's theorem, the width is also the smallest number of maximal chains required to cover all elements. This point of view will be useful in proving our results, and as we will discuss in the related works section, in further extending them.

Before going into the technical details, let us highlight the main difficulty of the proof. Under the previous assumptions, for a vertex v, we could associate to every memory state i a left quotient $L_i(v)$ and ensure that in a play π consistent with the strategy we construct, $L_i(v)$ is an under-approximation of $c(\pi)^{-1}W$, meaning $L_i(v) \subseteq c(\pi)^{-1}W$. In this proof, we will not be able to associate to every memory state a single left quotient, but rather a chain of left quotients, and ensure that some left quotient in this chain is an under-approximation.

Proof. We use the same notations as for the proof of Lemma 3, and construct a memory structure together with a winning strategy using this memory structure. Let K be the width of $(\mathrm{Res}(W), \subseteq)$. We construct a memory structure $\mathcal{M}^* = (\{1, \ldots, K\}, 1, \mu)$, and a strategy σ induced by the next-move function ν.

Let v be a vertex in $\mathcal{A}$. We consider the set $\mathcal{W}(v)$ of left quotients L such that (v, L) is winning. Then $(\mathcal{W}(v), \subseteq)$ is an ordered set of width at most K: thanks to the characterisation above of the width, we can split $\mathcal{W}(v)$ into maximal decreasing (finite or infinite) chains of left quotients, denoted $\ell_1(v), \ldots, \ell_p(v)$, for some $p \leq K$. Note that this decomposition is not unique. Up to renumbering, let us assume that $W \in \ell_1(v_0)$: indeed, by assumption (v_0, W) is winning.

We say that (v, ℓ) is winning if for all $L \in \ell$, we have that (v, L) is winning. For ℓ a chain of left quotients and $a \in A$, we define $\ell \cdot a$ component-wise. Note that even if ℓ is infinite, it may be that $\ell \cdot a$ is finite.

We define the update function: $\mu(i, (v, v'))$ is a j as follows.

- If $\ell_i(v) \cdot c(v, v')$ is finite, denote $L \cdot c(v, v')$ its last element. Choose j such that $L \cdot c(v, v') \in \ell_j(v')$. Note that in general, such a j may not exist; it does exist if $(v', \ell_i(v) \cdot c(v, v'))$ is winning, and we will prove that this will always be the case when playing the strategy σ.
- If $\ell_i(v) \cdot c(v, v')$ is infinite, then choose j such that $\ell_j(v')$ has an infinite intersection with $\ell_i(v) \cdot c(v, v')$. Such a j exists without any assumption.

We define the next-move function ν (inducing σ). Let $v \in V_\exists$, and consider $(v, \ell_i(v))$. Let $L \in \ell_i(v)$, Eve wins from (v, L), so there exists an edge $(v, v') \in E$ such that $(v', L \cdot c(v, v'))$ is winning, we say that $(v, v') \in E$ is good for L. Since $\mathcal{W}(v')$ is upward closed, if $(v, v') \in E$ is good for L, then it is good for every L' such that $L \subseteq L'$. We argue that there exists an edge $(v, v') \in E$ that is good for all $L \in \ell_i(v)$, *i.e.* such that $(v', \ell_i(v) \cdot c(v, v'))$ is winning; define $\nu(v, i)$ to be this v'. There are two cases:

- Either $\ell_i(v)$ is finite, denote L its last element. Since $\ell_i(v)$ is decreasing, an edge good for L is good for all $L' \in \ell_i(v)$.
- Or $\ell_i(v)$ is infinite. The vertex v has finite degree, so there exists an edge which is good for infinitely many $L \in \ell_i(v)$. Since $\ell_i(v)$ is decreasing, it is good for all $L' \in \ell_i(v)$.

We show that the strategy σ is winning. Consider a play $\pi = (v_0, v_1) \cdot (v_1, v_2) \cdots$ consistent with σ, and $i_0 \cdot i_1 \cdots$ the sequence of memory states assumed along this play. Denote π_k the prefix of π of length k, we prove that for all k, there exists $L \in \ell_{i_k}(v_k)$ such that $L \subseteq c(\pi_k)^{-1}W$. Note that by definition, $(v_k, \ell_{i_k}(v_k))$ is winning, so $c(\pi_k)^{-1}W \neq \emptyset$.

We proceed by induction. For $k = 0$, it follows from $W \in \ell_1(v_0)$. Let $k > 0$, the induction hypothesis implies the existence of $L \in \ell_{i_{k-1}}(v_{k-1})$ such that $L \subseteq c(\pi_{k-1})^{-1}W$. We distinguish two cases, and denote $c_k = c(v_{k-1}, v_k)$.

- Either v_{k-1} belongs to Eve, then by construction of σ we have that $(v_k, \ell_{i_{k-1}}(v_{k-1}) \cdot c_k)$ is winning. It follows that the update function is well defined, and:
  (1) If $\ell_{i_{k-1}}(v_{k-1}) \cdot c_k$ is finite, denote $L' \cdot c_k$ its last element, we have $L' \cdot c_k \in \ell_{i_k}(v_k)$. Since $L' \cdot c_k$ is the last element of $\ell_{i_{k-1}}(v_{k-1}) \cdot c_k$, it follows that $L' \subseteq L$. We have thus $L' \cdot c_k \subseteq L \cdot c_k$, so $L' \subseteq c(\pi_k)^{-1}W$, and $L' \cdot c_k \in \ell_{i_k}(v_k)$.
  (2) If $\ell_{i_{k-1}}(v_{k-1}) \cdot c_k$ is infinite, $\ell_{i_k}(v_k)$ has an infinite intersection with $\ell_{i_{k-1}}(v_{k-1}) \cdot c_k$. So there exists $L' \subseteq L$ with $L' \in \ell_{i_{k-1}}(v_{k-1})$ such that $L' \cdot c_k$ is in this intersection. We have $L' \cdot c_k \in \ell_{i_k}(v_k)$ and $L' \subseteq c(\pi_k)^{-1}W$.
- Or v_{k-1} belongs to Adam. Since Adam cannot escape $\mathcal{W}_E(\mathcal{G} \times \mathcal{M})$, we have that $(v_k, \ell_{i_{k-1}}(v_{k-1}) \cdot c_k)$ is winning, and the same reasoning concludes.

It follows that the strategy σ is winning, concluding the proof of Lemma 5. ■

A Matching Lower Bound.

Lemma 6 (Lower bound). *For all safety objectives W, there exists a game $\mathcal{G} = (\mathcal{A}, W)$ with finite degree and an initial vertex v_0 where Eve has a winning strategy, but no winning strategy using less than K memory states, where K is the width of $(\mathrm{Res}(W), \subseteq)$.*

Consequently, $\mathrm{mem}_{fin\text{-}deg}(W)$ and $\mathrm{mem}(W)$ are both greater than or equal to the width of $(\mathrm{Res}(W), \subseteq)$.

Proof. Let W a safety objective, and K the wdth of $(\mathrm{Res}(W), \subseteq)$. Consider $\{w_1^{-1}W, \ldots, w_K^{-1}W\}$ an antichain of left quotients of W. Note that since there are countably many finite words, K is either finite or countable infinite, but not larger. For $i \neq j$, there exists $u_{i,j} \in A^\omega$ such that $u_{i,j} \in w_i^{-1}W$ and $u_{i,j} \notin w_j^{-1}W$.

We describe the game, illustrated in Figure 2. A play consists in three steps:

(1) From v_0 to v_0': Adam chooses a word in $\{w_1, \ldots, w_K\}$;
(2) From v_0', Eve chooses between K options $v_1, \ldots, v_K$;
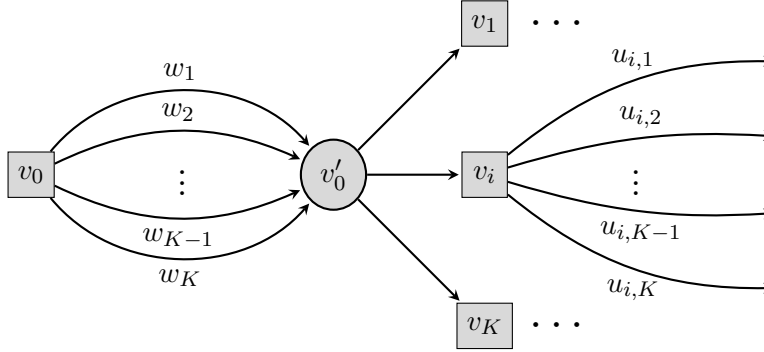(3) From v_i, Adam chooses a word in $\{u_{i,j} : j \neq i\}$.

FIGURE 2. The lower bound.

More formally: the game between v_0 and v_0' is a finitely branching and well founded tree whose branches are labeled with the words $w_1, \ldots, w_K$, and the leaves are identified with v_0'. Both for the first and third step labeling an edge by a word is only for graphical representation: formally, we use a sequence of edges labeled by the corresponding sequence of colours.

We first show that Eve has a winning strategy from v_0, using K memory states. It consists in choosing the ith option whenever Adam chooses the word w_i: whatever Adam chooses at the third step, $w_i \cdot u_{i,j} \in W$.

We now show that there exists no winning strategy using less than K memory states. Indeed, such a strategy will not comply with the above strategy and for some $i \neq j$, choose the jth option if Adam chooses w_i. Then Adam wins by playing $u_{j,i}$, since $w_i \cdot u_{j,i} \notin W$. ∎

Theorem 1 easily follows from combining the upper bound and lower bound lemmas.

5. EXAMPLES AND APPLICATIONS

In this section, we instantiate Theorem 1 on different examples. We chose four examples:

- The outbidding objective shows the difference between graphs with finite degree and graphs with infinite degree; in particular, it gives a counter example to Lemma 5 when dropping the finite degree assumption,
- The energy objective is a non-regular half-positional safety objective,
- The generalized safety objective is a regular safety objective for which the partially ordered set of left quotients has a nice well-known combinatorial structure,
- The boundedness objective is a central piece in the theory of regular cost functions.

When representing the partial order $(\text{Res}(W), \subseteq)$ for a given W, we use the following convention: a black edge from L to L' means that $L \subseteq L'$, and a dotted edge labeled a from L to L' means that $L' = L \cdot a$, so the dotted structure is the minimal (although possibly infinite) deterministic automaton recognizing W.

Outbidding Games. Let $A = \{a, b, c\}$ and $W = \{a^n \cdot b^p \cdot c^\omega \mid n \leq p\} \cup \{a^\omega\} \cup a^* \cdot b^\omega$. It is a non-regular safety objective, called the outbidding objective. The figure 3 represents the partial order $(\text{Res}(W), \subseteq)$. Its width is three: there are two incomparable infinite increasing sequences of left quotients, $((a^n)^{-1}W)_{n \in \mathbb{N}}$ and $((b \cdot a^n)^{-1}W)_{n \in \mathbb{N}}$, and $c^{-1}W$.
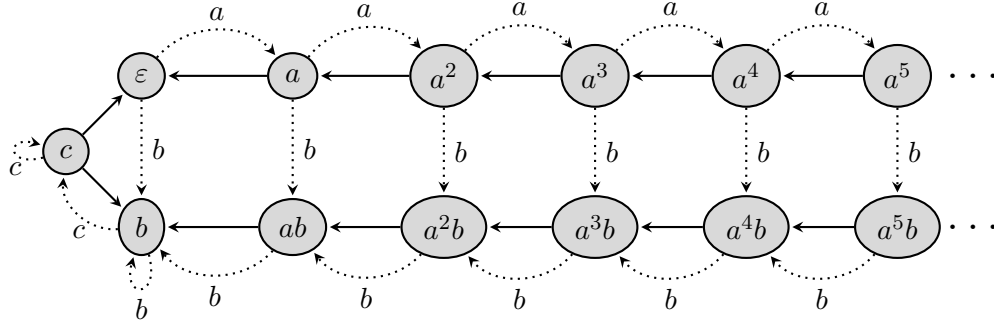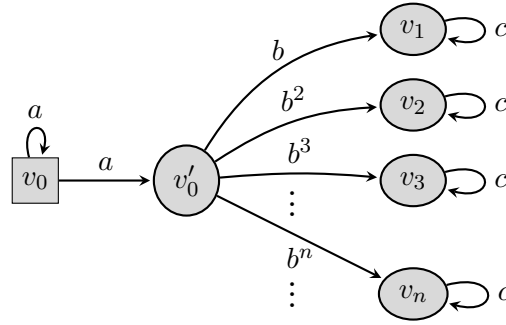
FIGURE 3. The outbidding objective: more b's than a's.

FIGURE 4. An outbidding game with infinite degree where Eve needs infinite memory to win.

Hence thanks to Theorem 1, $\mathrm{mem}_{\mathrm{fin\text{-}deg}}(W) = 3$. However, there exists an outbidding game where Eve wins but needs infinite memory. This does not contradict Theorem 1, as this game, represented in Figure 4, has a vertex of infinite degree. It goes as follows: first Adam picks a number n, and then Eve takes over: she has to pick a number p, higher than or equal to n. A finite memory strategy can only choose from finitely many options, hence cannot win against all strategies of Adam.

Energy Games. The setup for the energy objective is the following: assume we are monitoring a resource. We denote by A the set of actions on this resource, which is any monotonic function $f : \mathbb{N} \to \mathbb{N}$, as for instance:

- consuming one unit of the resource,
- reloading by one unit,
- emptying the resource,
- consuming half of the current energy level.

Define the energy objective by

$$W = \{w = w_0 w_1 \cdots \mid \text{ the energy level remains always non-negative}\} .$$

It is a non-regular safety objective. Energy games and several variants have been extensively studied [BFL$^+$08, CD12, CdAHS03]. The Figure 5 represents the partial order $(\mathrm{Res}(W), \subseteq)$, with only two actions: a reloads by one unit, and b consumes one unit. In general, if the

actions are monotonic, then the left quotients are totally ordered by inclusion, so thanks to Theorem 1 we have $\text{mem}_{\text{fin-deg}}(W) = 1$.
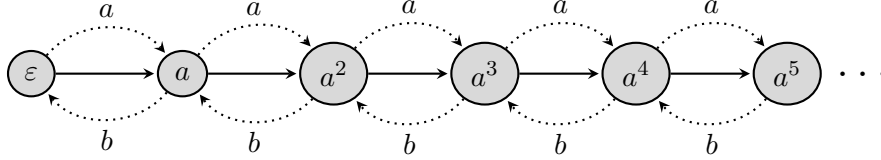

FIGURE 5. The energy objective: always more a's than b's.

Corollary 2 ([BFL$^+$08])**.** *The energy games are half-positional.*

Generalized Safety Games. This example originates from the study of generalized reachability games [FH10, FH13]. A generalized reachability objective is a (finite) conjunction of reachability objectives. Here we take the opponent's vantage point: a generalized safety objective is a (finite) disjunction of (internal) safety objectives. Specifically, let $A = \{\bot, 1, \ldots, k\}$: each letter is a color, and $\bot$ is uncolored. Let $W = \{w = w_0 w_1 \cdots \mid \exists i \in \{1, \ldots, k\}, \forall n, w_n \neq i\}$, it is satisfied if at least one color is not seen along the play. It is a safety objective. The figure 6 represents the partial order $(\text{Res}(W), \subseteq)$ for $k = 3$. The left quotients are all the strict subsets of $\{1, \ldots, k\}$. The width of this partial order is $\binom{k}{\lfloor k/2 \rfloor}$, according to the well known Sperner's Lemma from combinatorics.
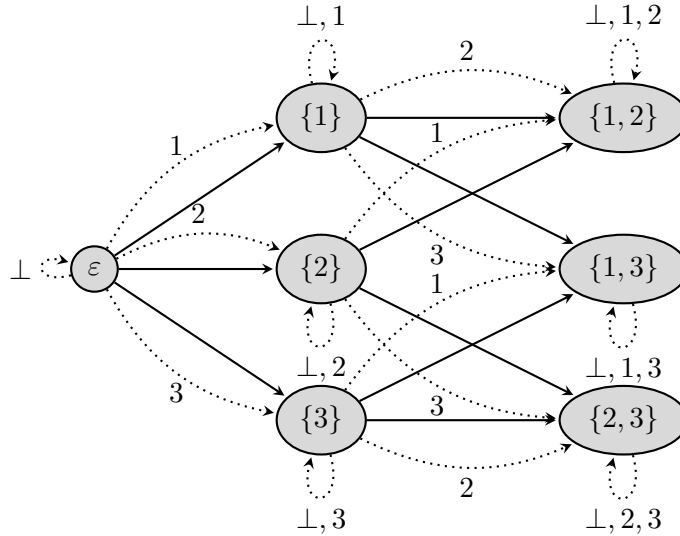

FIGURE 6. The generalized safety objective.

Corollary 3 ([FH10, FH13])**.** *For all generalized safety games with k colors, if Eve has a winning strategy, then she has a winning strategy with $\binom{k}{\lfloor k/2 \rfloor}$ memory states.*

Furthermore, for all k, there exists a generalized safety game with k colors where Eve has a winning strategy using $\binom{k}{\lfloor k/2 \rfloor}$ memory states, but none using less memory states.

Games with Counters. This example originates from the theory of regular cost functions [Col13]. Let $N \in \mathbb{N}$, and define the boundedness objective W_N involving a counter as follows:
$$W_N = \{w \mid \text{the counter value in } w \text{ remains bounded by } N\} .$$
For the set of actions, we consider any monotonic action (even non-regular), *i.e.* function $f : \mathbb{N} \to \mathbb{N}$ such that if $i \leq j$ then $f(i) \leq f(j)$, as for instance:
- leaving the counter value unchanged,
- incrementing the counter value by one,
- resetting the counter value to zero,
- dividing the counter value by two, rounded down,
- increasing the counter value to the next power of two.

The objective W_N is a regular safety objective.

Corollary 4 ([Col13])**.** *The boundedness games are half-positional.*

6. Related Works

The study of memory requirements in games has a long history.

Before 2014 – the publication of the conference version of this paper. The first general result in this direction is about Muller objectives: in [DJW97], the authors show how to compute the exact memory requirements by looking at the so-called Zielonka tree [Zie98]. This is orthogonal to our results, as the Muller objectives only specify the limit behaviour (what is seen infinitely often), whereas we consider here only the behaviours in the finite. Gimbert and Zielonka [GZ05] (see also [Gim07]) gave a characterization of all payoff functions (extending objectives to a quantitative setting) for which both players have memoryless optimal strategies over finite games.

Eryk Kopczyński investigated memory requirements for ω-regular objectives [Kop09], asking the following question: given a ω-regular objective, can we compute its memory requirement? This fascinating question that we call the Kopczyński programme remains open to this day. One of the many contributions of Kopczyński in this line of work was to introduce the notion of *chromatic memory*: the memory structure only considers the sequence of colours, and not the sequence of edges. Kopczyński constructed an algorithm to compute the *chromatic* memory requirements of an ω-regular objective, and conjectured that this coincides with general memory requirements.

After 2014. Casares [Cas22] has disproved Kopczyński's conjecture, and in particular has shown that chromatic memory requirements are hard to compute: for Muller objectives, deciding whether there is a memory structure of size k becomes NP-complete and equivalent to minimizing transition-based deterministic Rabin automata. In this direction, Casares, Colcombet and Lehtinen [CCL22] showed that computing general memory requirements for Muller objectives is equivalent to minimizing good-for-games automata. A result by Bouyer, Randour, and Vandenhove [BRV22] provides a link between the chromatic memory requirements of all ω-regular objectives (not only Muller objectives) and their representation as transition-based deterministic parity automata, but with less tight bounds on the minimal memory structures. Bouyer et al. [BLO$^+$22] extended the results of Gimbert and Zielonka [GZ05] to chromatic finite memory.

A promising line of work extending the approach initiated in this paper uses the notion of universal graphs, initially introduced for understanding algorithms solving games [CFGO22],

and later used by Ohlmann for giving a characterisation of all positionally determined objectives [Ohl21, Ohl22]. The proof of the characterisation is indeed very close to the techniques we present here: indeed universal graphs form a generic way of reducing to safety games. The characterisation has then been extended to finite memory by Casares and Ohlmann [CO23].

The subsequent works most related to the present results are by Bouyer, Casares, Randour and Vandenhove [BCRV22], and by Bouyer, Fijalkow, Randour, and Vandenhove [BFRV23]. The first give a characterization of half-positional objectives recognized by deterministic Büchi automata, and the second characterizations of chromatic memory requirements for open and closed objectives, as well as complexity-theoretic results about computing these requirements. Both are technically rooted in the very ideas developed in this paper. In particular, Dilworth's theorem and the use of covering chains (developed in Lemma 5) is at the heart of the characterization given in [BFRV23].

Technically unrelated but close in spirit, the article [BLT22] establishes the existence of finite-memory optimal strategies from topological properties of objectives. There are major differences with our work: their framework is different (they study *concurrent* games that are not played *on graphs*), and their aim is to establish the existence of finite-memory optimal strategies for many objectives, but not to understand precisely the memory requirements of some class of objectives.

Conclusion and Perspectives

We considered general safety objectives and characterized their memory requirements. Specifically, the memory requirements of a safety objective W is the width of the partially ordered set $(\mathrm{Res}(W), \subseteq)$. This is the first general result characterizing the memory requirements for *some* non-regular objectives, based on their topological properties. Back in 2014, we hoped that this would be a stepping stone for obtaining memory requirements characterizations for other classes of objectives. As we have discussed in the related works section this hope has materialised, with many results along this line. We now only hope that there will be more: the long-term goal is indeed to characterise memory requirements for all ω-regular objectives, a research programme started by Kopczyński around 2006 [Kop09] and far from over, and even more active than ever.

References

[BCRV22] Patricia Bouyer, Antonio Casares, Mickael Randour, and Pierre Vandenhove. Half-positional objectives recognized by deterministic Büchi automata. In Bartek Klin, Slawomir Lasota, and Anca Muscholl, editors, *CONCUR*, volume 243 of *LIPIcs*, pages 20:1–20:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. `doi:10.4230/LIPIcs.CONCUR.2022.20`.

[BFL+08] Patricia Bouyer, Ulrich Fahrenberg, Kim Guldstrand Larsen, Nicolas Markey, and Jirí Srba. Infinite runs in weighted timed automata with energy constraints. In Franck Cassez and Claude Jard, editors, *FORMATS*, volume 5215 of *Lecture Notes in Computer Science*, pages 33–47. Springer, 2008.

[BFRV23] Patricia Bouyer, Nathanaël Fijalkow, Mickael Randour, and Pierre Vandenhove. How to play optimally for regular objectives? In *ICALP*, volume 261 of *LIPIcs*, pages 118:1–118:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023. `doi:10.4230/LIPIcs.ICALP.2023.118`.

[BLO+22] Patricia Bouyer, Stéphane Le Roux, Youssouf Oualhadj, Mickael Randour, and Pierre Vandenhove. Games where you can play optimally with arena-independent finite memory. *Logical Methods in Computer Science*, 18(1), 2022. `doi:10.46298/lmcs-18(1:11)2022`.

[BLT22]   Patricia Bouyer, Stéphane Le Roux, and Nathan Thomasset. Finite-memory strategies in two-player infinite games. In Florin Manea and Alex Simpson, editors, *CSL*, volume 216 of *LIPIcs*, pages 8:1–8:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. `doi:10.4230/LIPIcs.CSL.2022.8`.

[BRV22]   Patricia Bouyer, Mickael Randour, and Pierre Vandenhove. Characterizing omega-regularity through finite-memory determinacy of games on infinite graphs. In Petra Berenbrink and Benjamin Monmege, editors, *STACS*, volume 219 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 16:1–16:16, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. `doi:10.4230/LIPIcs.STACS.2022.16`.

[Cas22]   Antonio Casares. On the minimisation of transition-based Rabin automata and the chromatic memory requirements of Muller conditions. In *CSL*, volume 216, pages 12:1–12:17, 2022. `doi:10.4230/LIPIcs.CSL.2022.12`.

[CCL22]   Antonio Casares, Thomas Colcombet, and Karoliina Lehtinen. On the size of good-for-games Rabin automata and its link with the memory in Muller games. In Mikołaj Bojanczyk, Emanuela Merelli, and David P. Woodruff, editors, *ICALP*, volume 229 of *LIPIcs*, pages 117:1–117:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. `doi:10.4230/LIPIcs.ICALP.2022.117`.

[CD12]   Krishnendu Chatterjee and Laurent Doyen. Energy parity games. *Theoretical Computer Science*, 458:49–60, 2012.

[CdAHS03]   Arindam Chakrabarti, Luca de Alfaro, Thomas A. Henzinger, and Mariëlle Stoelinga. Resource interfaces. In Rajeev Alur and Insup Lee, editors, *EMSOFT*, volume 2855 of *Lecture Notes in Computer Science*, pages 117–133. Springer, 2003.

[CFGO22]   Thomas Colcombet, Nathanaël Fijalkow, Pawel Gawrychowski, and Pierre Ohlmann. The theory of universal graphs for infinite duration games. *Logical Methods in Computer Science*, 18(3), 2022. `doi:10.46298/lmcs-18(3:29)2022`.

[CFH14]   Thomas Colcombet, Nathanaël Fijalkow, and Florian Horn. Playing safe. In Venkatesh Raman and S. P. Suresh, editors, *FSTTCS*, volume 29 of *LIPIcs*, pages 379–390. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2014. `doi:10.4230/LIPIcs.FSTTCS.2014.379`.

[CO23]   Antonio Casares and Pierre Ohlmann. Characterising memory in infinite games. In *ICALP*, volume 261 of *LIPIcs*, pages 122:1–122:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023. `doi:10.4230/LIPIcs.ICALP.2023.122`.

[Col13]   Thomas Colcombet. Regular cost functions, part I: Logic and algebra over words. *Logical Methods in Computer Science*, 9(3), 2013.

[DJW97]   Stefan Dziembowski, Marcin Jurdziński, and Igor Walukiewicz. How much memory is needed to win infinite games? In *LICS*, pages 99–110. IEEE Computer Society, 1997.

[FBB+23]   Nathanaël Fijalkow, Nathalie Bertrand, Patricia Bouyer-Decitre, Romain Brenguier, Arnaud Carayol, John Fearnley, Hugo Gimbert, Florian Horn, Rasmus Ibsen-Jensen, Nicolas Markey, Benjamin Monmege, Petr Novotný, Mickael Randour, Ocan Sankur, Sylvain Schmitz, Olivier Serre, and Mateusz Skomra. Games on graphs. *CoRR*, abs/2305.10546, 2023. `arXiv:2305.10546`, `doi:10.48550/arXiv.2305.10546`.

[FH10]   Nathanaël Fijalkow and Florian Horn. The surprizing complexity of reachability games. *CoRR*, abs/1010.2420, 2010.

[FH13]   Nathanaël Fijalkow and Florian Horn. Les jeux d'accessibilité généralisée. *Technique et Science Informatiques*, 32(9-10):931–949, 2013.

[Gim07]   Hugo Gimbert. *Jeux Positionnels*. PhD thesis, Université Paris 7, 2007.

[GTW02]   Erich Grädel, Wolfgang Thomas, and Thomas Wilke, editors. *Automata, Logics, and Infinite Games: A Guide to Current Research [outcome of a Dagstuhl seminar, February 2001]*, volume 2500 of *Lecture Notes in Computer Science*. Springer, 2002.

[GZ05]   Hugo Gimbert and Wiesław Zielonka. Games where you can play optimally without any memory. In *CONCUR 2005*, pages 428–442, 2005. `doi:10.1007/11539452_33`.

[Kop09]   Eryk Kopczyński. *Half-Positional Determinacy of Infinite Games*. PhD thesis, University of Warsaw, 2009.

[Kup14]   Orna Kupferman. Variations on safety. In Erika Ábrahám and Klaus Havelund, editors, *TACAS*, volume 8413 of *Lecture Notes in Computer Science*, pages 1–14. Springer, 2014.

[Ohl21]   Pierre Ohlmann. *Monotonic graphs for parity and mean-payoff games.* PhD thesis, IRIF – Research Institute on the Foundations of Computer Science, 2021.

[Ohl22]   Pierre Ohlmann. Characterizing positionality in games of infinite duration over infinite graphs. In Christel Baier and Dana Fisman, editors, *LICS*, pages 22:1–22:12. ACM, 2022. `doi:10.1145/3531130.3532418`.

[Zie98]   Wiesław Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees. *Theoretical Computer Science*, 200(1-2):135–183, 1998.