Random Fruits on the Zielonka Tree

Florian Horn

LIAFA, Université Paris 7, Case 7014, 2 place Jussieu, F-75251 Paris 5, France
Lehrstuhl für Informatik VII, RWTH, Ahornstraße 55, 52056 Aachen, Germany
horn@liafa.jussieu.fr

Abstract. Stochastic games are a natural model for the synthesis of controllers confronted to adversarial and/or random actions. In particular, ω-regular games of infinite length can represent reactive systems which are not expected to reach a correct state, but rather to handle a continuous stream of events. One critical resource in such applications is the memory used by the controller. In this paper, we study the amount of memory that can be saved through the use of randomisation in strategies, and present matching upper and lower bounds for stochastic Muller games.

1 Introduction

A stochastic game arena is a directed graph with three kinds of states: Eve's, Adam's and random states. A token circulates on this arena: when it is in one of Eve's states, she chooses its next location among the successors of the current state; when it is in one of Adam's states, he chooses its next location; and when it is in a random state, the next location is chosen according to a fixed probability distribution. The result of playing the game for ω moves is an infinite path of the graph. A play is winning either for Eve or for Adam, and the "winner problem" consists in determining whether one of the players has a winning strategy, from a given initial state. Closely related problems concern the computation of winning strategies, as well as determining the nature of these strategies: pure or randomised, with finite or infinite memory. There has been a long history of using arenas without random states (2-player arenas) for modelling and synthesising reactive processes [BL69,PR89]: Eve represents the controller, and Adam the environment. Stochastic ($2\frac{1}{2}$-player) arenas [Con92,deA97], with the addition of random states, can also model uncontrollable actions that happen according to a random law, rather than by choice of an actively hostile environment. The desired behaviour of the system is traditionally represented as an ω-regular winning condition, which naturally expresses the temporal specifications and fairness assumptions of transition systems [MP92]. From this point of view, the complexity of the winning strategies is a central question, since they represent possible implementations of the controllers in the synthesis problem. In this paper, we focus on an important normal form of ω-regular conditions, namely *Muller* winning conditions (see [Tho95] for a survey).

In the case of 2-player Muller games, a fundamental determinacy result of Büchi and Landweber ensures that, from any initial state, one of the players has a winning strategy [BL69]. Gurevich and Harrington used the LAR (*latest appearance record*) structure of McNaughton to extend this result to strategies with memory factorial in the size of the game [GH82]. Zielonka refines the LAR construction into a tree, and derives from it an elegant algorithm to compute winning regions in 2-player Muller games [Zie98]. An insightful analysis of the Zielonka tree by Dziembowski, Jurdzinski, and Walukiewicz leads to optimal (and asymmetrical) memory bounds for non-randomised winning strategies in 2-player Muller games [DJW97]. Chatterjee extends algorithm and bounds to the case of non-randomised strategies

over $2\frac{1}{2}$-player arenas [Cha07]. However, the lower bound on memory does not hold for *randomised strategies*, even in non-stochastic arenas. In particular, Chatterjee, de Alfaro, and Henzinger show that Eve only needs to consider memoryless randomised strategies when the condition is upward-closed [CdAH04]. Chatterjee extends this result in [Cha07], showing that conditions with non-trivial upward-closed subsets admit randomised strategies with less memory than non-randomised ones. Majumdar gives a global lower bound for all Muller conditions with a fixed number of colours [Maj03], while we showed that in the case of Streett games, the upper bound for non-randomised strategy is still valid for randomised ones [Hor07].
Our contributions. The memory bounds of [Cha07] are not tight in general, even in the case of 2-player arenas. We give here matching upper and lower bounds, for any Muller condition. We compute a number $r_{\mathcal{F}}$ from the Zielonka Tree of a Muller condition $\mathcal{F}$, and we show that:

- there is a randomised winning strategy with $r_{\mathcal{F}}$ memory in every $2\frac{1}{2}$-player game $(\mathcal{A}_{\mathcal{F}}, \mathcal{F})$ (Theorem 9).
- there is a 2-player game $(\mathcal{A}_{\mathcal{F}}, \mathcal{F})$ where any randomised winning strategy for Eve has at least $r_{\mathcal{F}}$ memory states (Theorem 20).

Furthermore, the witness arenas we build in the proof of Theorem 20 are notably smaller (exponentially smaller, in some cases) than the arenas built in [DJW97], even though the problem of *polynomial* arenas remains open.
Outline of the paper. Section 2 recalls the classical notions in the area, while Section 3 presents former results on memory bounds and randomised strategies. The next two sections present our main results. In Section 4, we introduce the number $r_{\mathcal{F}}$ and show that it is an upper bound on the memory needed to win in any $2\frac{1}{2}$-game $(\mathcal{A}_{\mathcal{F}}, \mathcal{F})$. In Section 5, we show that this bound is tight. Finally, in Section 6, we derive some consequences from our result: we characterise the class of Muller conditions that admit memoryless randomised strategies, and we show that for each Muller condition, at least one of the players cannot improve its memory through randomisation.

2 Definitions

We consider turn-based stochastic two-player Muller games. We recall here several classical notions in the field, and refer the reader to [Tho95,deA97] for more details.
Probability Distribution. For a finite set A, a *probability distribution* on A is a function $\alpha : A \to [0,1]$ such that $\sum_{a \in A} \alpha(a) = 1$. We denote the set of probability distributions on A by $\mathcal{D}(A)$. Given a distribution $\alpha \in \mathcal{D}(A)$, we denote by $\mathrm{Supp}(\alpha) = \{a \in A \mid \alpha(a) > 0\}$ the support of α.
Arenas. A *turn-based stochastic finite arena* (*$2\frac{1}{2}$-player arena*) $\mathcal{A}$ over a set of colours $\mathcal{C}$ consists of a directed finite graph $(\mathcal{S}, \mathcal{T})$, a partition $(\mathcal{S}_E, \mathcal{S}_A, \mathcal{S}_R)$ of $\mathcal{S}$, a probabilistic transition function $\delta : S_R \to \mathcal{D}(S)$ such that $t \in \mathrm{Supp}(\delta(s)) \Leftrightarrow (s,t) \in \mathcal{T}$, and a partial colouring function $\chi : \mathcal{S} \rightharpoonup \mathcal{C}$.

The states in $\mathcal{S}_E$ (resp. $\mathcal{S}_A$, $\mathcal{S}_R$) are *Eve's states* (resp. *Adam's states*, *random states*), and are graphically represented as $\bigcirc$'s (resp. $\square$, $\triangle$) in figures. The *turn-based deterministic arenas* (*2-player* arenas) are the special case of $2\frac{1}{2}$-player arenas with $\mathcal{S}_R = \emptyset$.

A set $U \subseteq \mathcal{S}$ of states is called *δ-closed* if for every random state $u \in U \cap \mathcal{S}_R$, if $(u,t) \in \mathcal{T}$, then $t \in U$. It is *live* if for every non-random state $u \in U \cap (\mathcal{S}_E \cup \mathcal{S}_A)$, there is a state $t \in U$ such that $(u,t) \in \mathcal{T}$. A subset U that is live and δ-closed induces a *subarena* of $\mathcal{A}$, denoted by $\mathcal{A} \restriction U$. A set $U \subseteq \mathcal{S}$ that is not a subarena is called a *partial subarena*.

Plays and Strategies. An infinite path, or *play*, over the arena $\mathcal{A}$ is an infinite sequence $\rho = \rho_0\rho_1\dots$ of states such that $(\rho_i, \rho_{i+1}) \in \mathcal{T}$ for all $i \in \mathbb{N}$. The set of *occurring states* is $\mathrm{Occ}(\rho) = \{s \mid \exists i \in \mathbb{N}, \rho_i = s\}$, and the set of *limit states* is $\mathrm{Inf}(\rho) = \{s \mid \exists^\infty i \in \mathbb{N}, \rho_i = s\}$. We write Ω for the set of all plays, and Ω_s for the set of plays that start from the state s.

A *strategy with memory* M for Eve on the arena $\mathcal{A}$ is a (possibly infinite) transducer $\sigma = (M, \sigma^{\mathtt{n}}, \sigma^{\mathtt{u}})$, where $\sigma^{\mathtt{n}}$ is the "next-move" function from $(\mathcal{S}_E \times M)$ to $\mathcal{D}(\mathcal{S})$ such that $\mathrm{Supp}(\sigma^{\mathtt{n}}(s, m)) \subseteq \mathcal{T}(s)$ and $\sigma^{\mathtt{u}}$ is the "memory-update" function, from $(\mathcal{S} \times M)$ to $\mathcal{D}(M)$. Notice that both the move and the update are randomised: strategies whose memory is deterministic are a different, less compact, model. The strategies for Adam are defined likewise. We denote by Σ and T the set of all strategies for Eve and Adam, respectively. A strategy σ is pure if it does not use randomisation. It is *finite-memory* if M is a finite set, and *memoryless* if M is a singleton. Notice that strategies defined in the usual way as functions from $\mathcal{S}^*$ to $\mathcal{S}$ can be defined as strategies with infinite memory: the set of memory states is $\mathcal{S}^*$, the memory update is $\sigma^{\mathtt{u}}(s, w) \mapsto ws$, so the next-move can use the full prefix as argument.

Once a starting state $s \in \mathcal{S}$ and strategies $\sigma \in \Sigma$ and $\tau \in T$ are fixed, the outcome of the game is a random walk $\rho_s^{\sigma,\tau}$ for which the probabilities of events are uniquely fixed (an *event* is a measurable set of paths). For an event $P \in \Omega$, we denote by $\mathbb{P}_s^{\sigma,\tau}(P)$ the probability that a play belongs to P if it starts from s and Eve and Adam follow the strategies σ and τ.

A play is consistent with σ if for each position i such that $w_i \in \mathcal{S}_E$, $\mathbb{P}_{w_0}^{\sigma,\tau}(\rho_{i+1} = w_{i+1} \mid \rho_0 = w_0 \dots \rho_i = w_i) > 0$. The set of plays consistent with σ is denoted by Ω^σ. Similar notions can be defined for Adam's strategies.

Traps and Attractors. The *attractor of Eve to the set* U, denoted $\mathrm{Attr}_E(U)$, is the set of states where Eve can guarantee that the token reaches the set U with a positive probability. It is defined inductively by:

$$\begin{array}{rcl}
\mathrm{Attr}_E^0(U) &=& U \\
\mathrm{Attr}_E^{i+1}(U) &=& \quad \mathrm{Attr}_E^i(U) \\
&\cup& \{s \in \mathcal{S}_E \cup \mathcal{S}_R, \exists t \in \mathrm{Attr}_E^i(U) \mid (s,t) \in \mathcal{T}\} \\
&\cup& \{s \in \mathcal{S}_A \mid \forall t, (s,t) \in E \Rightarrow t \in \mathrm{Attr}_E^i(U)\} \\
\mathrm{Attr}_E(U) &=& \bigcup_{i>0} \mathrm{Attr}_E^i(U)
\end{array}$$

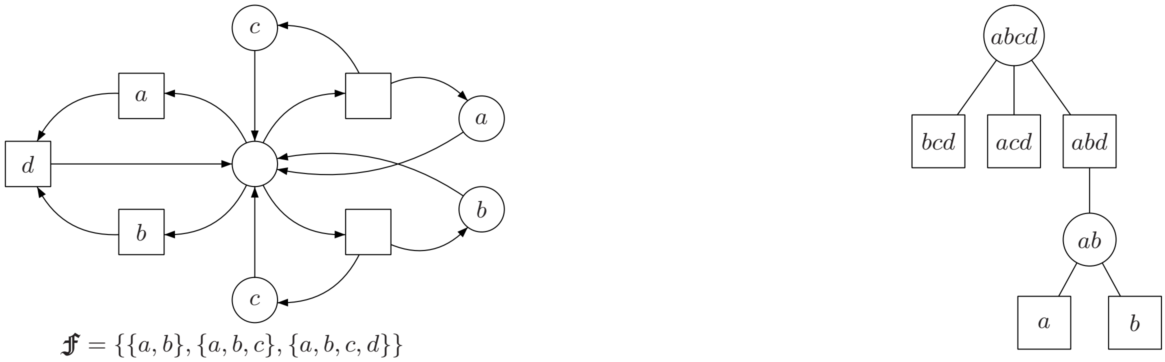
The corresponding *attractor strategy to* U *for Eve* is a pure and memoryless strategy a_U such that for any state $s \in \mathcal{S}_E \cap (\mathrm{Attr}_E(U) \setminus U)$, $s \in \mathrm{Attr}_E^{i+1}(U) \Rightarrow a_U(s) \in \mathrm{Attr}_E^i(U)$.

The dual notion of *trap for Eve* denotes a set from where Eve cannot escape, unless Adam allows her to do so: a set U is a trap for Eve if and only if $\forall s \in U \cap (\mathcal{S}_E \cup \mathcal{S}_R), (s,t) \in \mathcal{T} \Rightarrow t \in U$ and $\forall s \in U \cap \mathcal{S}_A, \exists t \in U, (s,t) \in \mathcal{T}$. Notice that a trap is a "strong" notion — the token can never leave it if Adam does not allow it to do so, while an attractor is a "weak" one — the token can avoid the target even if Eve uses the attractor strategy. Notice also that a trap (for either player) is always a subarena.

Winning Conditions. A *zero-sum boolean winning condition* (*winning condition*) is a subset Φ of Ω. Eve *wins a play* ρ if $\rho \in \Phi$. Adam wins if $\rho \in \Omega \setminus \Phi$. We consider ω-regular winning conditions formalised as *Muller conditions*. A Muller condition is determined by a subset $\mathcal{F}$ of the power set $\mathcal{P}(\mathcal{C})$ of colours. The plays winning for Eve for such a condition are the ones where the set of limit colours belongs to $\mathcal{F}$:

$$\Phi_{\mathcal{F}} = \{\rho \in \Omega | \chi(Inf(\rho)) \in \mathcal{F}\}$$

An example of Muller game is given in Figure 1(a). We use it throughout the paper to describe various notions and results.



(a) The game $\mathfrak{G} = (\mathfrak{A}, \mathfrak{F})$

(b) Zielonka Tree of $\mathfrak{F}$

Fig. 1. Recurring Example

Winning Strategies. Given a winning condition Φ and a state $s \in \mathcal{S}$, a strategy $\sigma \in \Sigma$ is *sure winning for Eve from s* (resp. *almost-sure winning*, *positive winning*) if for every strategy $\tau \in T$, we have $\Omega_s^{\sigma,\tau} \subseteq \Phi$ (resp. $\mathbb{P}_s^{\sigma,\tau}(\Phi) = 1$, $\mathbb{P}_s^{\sigma,\tau}(\Phi) > 0$). The *pure winning region* of Eve is the set of states from where she has a pure winning strategy. The *almost sure winning region* and the *positive winning region*, as well as all these notions for Adam are defined in a similar way.

3 Former results in memory bounds and randomisation

3.1 Pure strategies

There has been intense research since the sixties on the non-stochastic setting, *i.e.* pure strategies and 2-player arenas. Büchi and Landweber showed the determinacy of Muller games in [BL69]. Gurevich and Harrington used the LAR (Latest Appearance Record) of McNaughton to prove their *Forgetful Determinacy* theorem [GH82], which shows that a memory of size $|\mathcal{C}|!$ is sufficient for any game that uses only colours from $\mathcal{C}$, even when the arena is infinite. This result was later refined by Zielonka in [Zie98], using a representation of the Muller conditions as trees:

Definition 1 (Zielonka Tree of a Muller condition). *The Zielonka Tree $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$ of a winning condition $\mathcal{F} \subseteq \mathcal{P}(\mathcal{C})$ is defined inductively as follows:*

1. *If $\mathcal{C} \notin \mathcal{F}$, then $\mathcal{Z}_{\mathcal{F},\mathcal{C}} = \mathcal{Z}_{\overline{\mathcal{F}},\mathcal{C}}$, where $\overline{\mathcal{F}} = \mathcal{P}(\mathcal{C}) \setminus \mathcal{F}$.*
2. *If $\mathcal{C} \in \mathcal{F}$, then the root of $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$ is labelled with $\mathcal{C}$. Let $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_k$ be all the maximal sets in $\{U \notin \mathcal{F} \mid U \subseteq \mathcal{C}\}$. Then we attach to the root, as its subtrees, the Zielonka trees of $\mathcal{F} \upharpoonright \mathcal{C}_i$,* i.e. *the $\mathcal{Z}_{\mathcal{F}\upharpoonright\mathcal{C}_i,\mathcal{C}_i}$, for $i = 1 \ldots k$.*

Hence, the Zielonka tree is a tree with nodes labelled by sets of colours. A node of $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$ is an Eve node if it is labelled with a set from $\mathcal{F}$, otherwise it is an Adam node.

A later analysis of this construction by Dziembowski, Jurdzinski and Walukiewicz in [DJW97] led to an optimal and asymmetrical bound on the memory needed by the players to define winning strategies:

Definition 2 **(Number $m_{\mathcal{F}}$ of a Muller condition).** *Let $\mathcal{F} \subseteq \mathcal{P}(\mathcal{C})$ be a Muller condition, and $\mathcal{Z}_{\mathcal{F}_1,\mathcal{C}_1}, \mathcal{Z}_{\mathcal{F}_2,\mathcal{C}_2}, \ldots, \mathcal{Z}_{\mathcal{F}_k,\mathcal{C}_k}$ be the subtrees attached to the root of the tree $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$. We define the number $m_{\mathcal{F}}$ inductively as follows:*

$$m_{\mathcal{F}} = \begin{cases} 1 & \text{if } \mathcal{Z}_{\mathcal{F},\mathcal{C}} \text{ does not have any subtrees,} \\ \max\{m_{\mathcal{F}_1}, m_{\mathcal{F}_2}, \ldots, m_{\mathcal{F}_k}\} & \text{if } \mathcal{C} \notin \mathcal{F} \text{ (Adam node),} \\ \sum_{i=1}^{k} m_{\mathcal{F}_i} & \text{if } \mathcal{C} \in \mathcal{F} \text{ (Eve node).} \end{cases}$$

Theorem 3 **([DJW97]).** *In any 2-player Muller game $\mathcal{G}$ with winning condition $\mathcal{F}$, Eve has a pure strategy $\sigma_{\mathcal{G}}$ winning from every state in her winning region and with memory at most $m_{\mathcal{F}}$. Furthermore, there is a 2-player arena $\mathcal{A}_{\mathcal{F}}$ such that any strategy winning for Eve in every state of her winning region has a memory of size at least $m_{\mathcal{F}}$.*

In [Cha07], Chatterjee showed that Theorem 3 can be extended to the setting of randomised games, still with pure strategies:

Theorem 4 **([Cha07]).** *For any $2\frac{1}{2}$-player Muller game $\mathcal{G}$ with the winning condition $\mathcal{F}$, Eve has a pure strategy $\sigma_{\mathcal{G}}$ almost surely winning from every state in her winning region and with memory at most $m_{\mathcal{F}}$.*

3.2 Memory reduction through randomisation

Randomised strategies are more general than pure strategies. It is thus a legitimate question to ask whether the lower bound of Theorem 3 still holds for randomised strategies. It turns out that it is not the case. In [CdAH04], a first result showed that upward-closed conditions admit memoryless randomised strategies, while they don't admit memoryless pure strategies:

Theorem 5 **([CdAH04]).** *For any $2\frac{1}{2}$-player Muller game $\mathcal{G}$ with an upward-closed winning condition $\mathcal{F}$, Eve has a randomised memoryless strategy $\sigma_{\mathcal{G}}$ winning from every state in her winning region.*

This result was later extended in [Cha07], by removing the leaves attached to a node of the Zielonka Tree representing an upward-closed subcondition:

Definition 6 **([Cha07]).** *Let $\mathcal{F} \subseteq \mathcal{P}(\mathcal{C})$ be a Muller condition, and $\mathcal{Z}_{\mathcal{F}_1,\mathcal{C}_1}, \mathcal{Z}_{\mathcal{F}_2,\mathcal{C}_2}, \ldots, \mathcal{Z}_{\mathcal{F}_k,\mathcal{C}_k}$ be the subtrees attached to the root of the tree $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$. We define the number $m^U_{\mathcal{F}}$ inductively as follows:*

$$m^U_{\mathcal{F}} = \begin{cases} 1 & \text{if } \mathcal{Z}_{\mathcal{F},\mathcal{C}} \text{ does not have any subtrees,} \\ 1 & \text{if } \mathcal{F} \text{ is upward-closed,} \\ \max\{m^U_{\mathcal{F}_1}, m^U_{\mathcal{F}_2}, \ldots, m^U_{\mathcal{F}_k}\} & \text{if } \mathcal{C} \notin \mathcal{F} \text{ (Adam node),} \\ \sum_{i=1}^{k} m^U_{\mathcal{F}_i} & \text{if } \mathcal{C} \in \mathcal{F} \text{ (Eve node).} \end{cases}$$

Theorem 7 **([Cha07]).** *For any $2\frac{1}{2}$-player Muller game $\mathcal{G}$ with the winning condition $\mathcal{F}$, Eve has a randomised strategy $\sigma_{\mathcal{G}}$ winning from every state in her winning region and with memory at most $m^U_{\mathcal{F}}$.*

4 Randomised Upper Bound

The upper bound of Theorem 7 is not tight for all conditions. For example, the number $m^U_{\mathfrak{F}}$ of the condition $\mathfrak{F}$ in Figure 1(b) is three, while there is always a winning condition with two memory states. We present here yet another number for any Muller condition $\mathcal{F}$, denoted $r_{\mathcal{F}}$, that we compute from the Zielonka Tree:

Definition 8 (Number $r_{\mathcal{F}}$ of a Muller condition). *Let $\mathcal{F} \subseteq \mathcal{P}(\mathcal{C})$ be a Muller condition, where the root has $k+l$ children, l of them being leaves. We denote by $\mathcal{Z}_{\mathcal{F}_1,\mathcal{C}_1}, \mathcal{Z}_{\mathcal{F}_2,\mathcal{C}_2}, \cdots, \mathcal{Z}_{\mathcal{F}_k,\mathcal{C}_k}$ the non-leaves subtrees attached to the root of $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$. We define $r_{\mathcal{F}}$ inductively as follows:*

$$r_{\mathcal{F}} = \begin{cases} 1 & \text{if } \mathcal{Z}_{\mathcal{F},\mathcal{C}} \text{ does not have any subtrees,} \\ \max\{1, r_{\mathcal{F}_1}, r_{\mathcal{F}_2}, \ldots, r_{\mathcal{F}_k}\} & \text{if } \mathcal{C} \notin \mathcal{F} \text{ (Adam node),} \\ \sum_{i=1}^{k} r_{\mathcal{F}_i} & \text{if } \mathcal{C} \in \mathcal{F} \text{ (Eve node) and } l = 0, \\ \sum_{i=1}^{k} r_{\mathcal{F}_i} + 1 & \text{if } \mathcal{C} \in \mathcal{F} \text{ (Eve node) and } l > 0. \end{cases}$$

The first remark is that if $\emptyset \in \mathcal{F}$, $r_{\mathcal{F}}$ is equal to $m_{\mathcal{F}}$: as the leaves belong to Eve, and the fourth case cannot occur. In the other case, the intuition is that we merge leaves if they are siblings. For example, the number $r_{\mathfrak{F}}$ for our recurring example is two: one for the leaves labelled *bcd* and *acd*, and one for the leaves labelled a and b. The number $m_{\mathfrak{F}}$ is four (one for each leaf), and $m^U_{\mathfrak{F}}$ is three (one for the leaves labelled a and b, and one for each other leaf). This section will be devoted to the proof of Theorem 9:

Theorem 9 (Randomised upper bound). *If Eve has a winning strategy in the 2-$\frac{1}{2}$ player Muller game $(\mathcal{A}_{\mathcal{F}}, \mathcal{F})$, she has a winning strategy with memory $r_{\mathcal{F}}$.*

Let $\mathcal{G} = (\mathcal{F}, \mathcal{A})$ be a game defined on the set of colours $\mathcal{C}$ such that Eve wins from any initial node. We describe in the next three subsections a recursive procedure to compute a winning strategy for Eve with $r_{\mathcal{F}}$ memory states in each non-trivial case in the definition of $r_{\mathcal{F}}$. To this end, we fix a strategy $\tau \in T$ for Adam and an initial state $s_0 \in \mathcal{S}$. We use two lemmas — Lemmas 10 and 12 — that derive directly from similar results in [DJW97] and [Cha07]. The application of these principles to the game $\mathfrak{G}$ in Figure 1 builds a randomised strategy with two memory states *left* and *right*. In *left*, Eve sends the token to ($\nwarrow$ *or* $\swarrow$) and in *right*, to ($\nearrow$ *or* $\searrow$). The memory switches from *right* to *left* with probability one when the token visits a c, and from *left* to *right* with probability $\frac{1}{2}$ at each step.

4.1 $\mathcal{C}$ is winning for Adam

In the case where Adam wins the set $\mathcal{C}$, the construction of σ relies on Lemma 10:

Lemma 10. *Let $\mathcal{F} \subseteq \mathcal{P}(\mathcal{C})$ be a Muller winning condition such that $\mathcal{C} \notin \mathcal{F}$, and $\mathcal{A}$ be a $2\frac{1}{2}$-player arena such that Eve wins everywhere. There are subarenas $\mathcal{A}_1 \ldots \mathcal{A}_n$ such that:*

- *$i \neq j \Rightarrow \mathcal{A}_i \cap \mathcal{A}_j = \emptyset$;*
- *$\forall i, \mathcal{A}_i$ is a trap for Adam in the subarena $\mathcal{A} \setminus \mathrm{Attr}_E\left(\cup_{j=1}^{i-1} \mathcal{A}_j\right)$;*
- *$\forall i, \chi(\mathcal{A}_i)$ is included in the label E_i of a child of the root of $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$, and Eve wins everywhere in $(\mathcal{A}_i, \mathcal{F} \upharpoonright E_i)$;*

– $\mathcal{A} = \mathrm{Attr}_E(\cup_{j=1}^{n} \mathcal{A}_j)$.

Let the subarenas $\mathcal{A}_i$ be the ones whose existence is proved in this lemma. We denote by σ_i the winning strategy for Eve in $\mathcal{A}_i$, and by a_i the attractor strategy for Eve to $\mathcal{A}_i$ in the arena $\mathcal{A} \setminus \mathrm{Attr}(\cup_{j=1}^{i-1} \mathcal{A}_j)$. We identify the memory states of the σ_i, so their union has the same cardinal as the largest of them. For a state s, if $i = \min\{j \mid s \in \underset{E}{\mathrm{Attr}}(\cup_{k=1}^{j} \mathcal{A}_k)\}$, we define $\sigma(s, m)$ by:

– if $s \in \mathcal{A}_i$
  - $\sigma^{\mathtt{u}}(s, m) = \sigma_i^{\mathtt{u}}(s, m)$
  - $\sigma^{\mathtt{n}}(s, m) = \sigma_i^{\mathtt{n}}(s, m)$

– if $s \in \mathrm{Attr}_E(\cup_{k=1}^{i} \mathcal{A}_k) \setminus \mathcal{A}_i$
  - $\sigma^{\mathtt{u}}(s, m) = m$
  - $\sigma^{\mathtt{n}}(s, m) = a_i(s)$

By induction hypothesis over the number of colours, we can assume that the strategies σ_i have $r_{\mathcal{F}_i}$ memory states. The strategy σ uses $\max\{r_{\mathcal{F}_i}\}$ memory states.

Proposition 11. $\mathbb{P}_{s_0}^{\sigma,\tau}(\exists i, \mathrm{Inf}(\rho) \subseteq \mathcal{A}_i) = 1$.

Proof. The subarenas $\mathcal{A}_i$ are embedded traps, defined in such a way that the token can escape an $\mathcal{A}_i$ only by going to the attractor of a smaller one. Eve has thus a positive probability of reaching an $\mathcal{A}_j$ with $j < i$. Thus, if the token escapes one of the A_i infinitely often, the token has probability one to go to an $\mathcal{A}_j$ with $j < i$. By argument of minimality, after a finite prefix, the token will stay in one of the traps forever. $\square$

The strategy σ_i is winning from any state in $\mathcal{A}_i$. As Muller conditions are prefix-independent, it follows from Proposition 11 that σ is also winning from any state in $\mathcal{A}$.

4.2 $\mathcal{C}$ is winning for Eve, and the root of $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$ has no leaves among its children.

In this case, the construction relies on the following lemma:

Lemma 12. *Let $\mathcal{F} \subseteq \mathcal{P}(\mathcal{C})$ be a Muller winning condition such that $\mathcal{C} \in \mathcal{F}$, $\mathcal{A}$ a $2\frac{1}{2}$-player arena coloured by $\mathcal{C}$ such that Eve wins everywhere, and A_i the label of a child of the root in $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$. Then, Eve wins everywhere on the subarena $\mathcal{A} \setminus \mathrm{Attr}_E(\chi^{-1}(\mathcal{C} \setminus A_i))$ with the condition $\mathcal{F} \upharpoonright A_i$.*

Eve has a strategy σ_i that is winning from each state in $\mathcal{A} \setminus \mathrm{Attr}_E(\chi^{-1}(\mathcal{C} \setminus A_i))$. In this case, the set of memory states of σ is $M = \cup_{i=1}^{k}(i \times M^i)$. The "next-move" and "memory-update" functions $\sigma^{\mathtt{n}}$ and $\sigma^{\mathtt{u}}$ for a memory state $m = (i, m^i)$ are defined below:

– if $s \in \chi^{-1}(\mathcal{C} \setminus A_i)$
  - $\sigma^{\mathtt{u}}(s, m^i) = (i + 1, m^{i+1})$ where m^{i+1} is any state in M^{i+1}
  - if $s \in S_E$, $\sigma^{\mathtt{n}}(s, m^i)$ is any successor of s in $\mathcal{A}$

– if $s \in \mathrm{Attr}_E(\chi^{-1}(\mathcal{C} \setminus A_i))$
  - $\sigma^{\mathtt{u}}(s, m^i) = (i, m^i)$
  - $\sigma^{\mathtt{n}}(s, m^i) = a_i(s)$

– if $s \in \mathcal{A} \setminus \mathrm{Attr}_E(\chi^{-1}(\mathcal{C} \setminus A_i))$

- $\sigma^{\mathtt{u}}(s, m^i) = (i, \sigma_i^{\mathtt{u}}(s, m^i))$
- $\sigma^{\mathtt{n}}(s, m^i) = \sigma_i^{\mathtt{n}}(s, m^i)$

Once again, we can assume that the memory M_i of the strategy σ_i is of size $r_{\mathcal{F}\upharpoonright A_i}$. Here, however, the memory set of σ is the disjoint union of the M_i's. Thus, σ uses memory $\sum_{i=1}^{k}\{r_{\mathcal{F}\upharpoonright A_i}\}$.

Proposition 13. *Let* $\mathfrak{uc}$ *be the event "the top-level memory of* σ *is ultimately constant". Then,* $\mathbb{P}_{s_0}^{\sigma,\tau}(\rho \in \Phi_{\mathcal{F}} \mid \mathfrak{uc}) = 1$.

Proof. We call i the value of the top-level memory at the limit. After a finite prefix, the token stops visiting $\chi^{-1}(\mathcal{C} \setminus A_i)$. Thus, with probability one, it also stops visiting $\mathrm{Attr}_E(\chi^{-1}(\mathcal{C} \setminus A_i))$. From this point on, the token stays in the arena $\mathcal{A}_i$, where Eve plays with the winning strategy σ_i. Thus, $\mathbb{P}^{\sigma,\tau}(\rho \in \Phi_{\mathcal{F}\upharpoonright A_i} \mid \mathfrak{uc}) = 1$, and, as $\Phi_{\mathcal{F}\upharpoonright A_i} \subseteq \Phi_{\mathcal{F}}$, Proposition 13 follows.

Proposition 14. *If the top-level memory takes each value in* $1 \ldots k$ *infinitely often, then surely,* $\forall i \in 1 \ldots k, \chi(\mathrm{Inf}(\rho)) \nsubseteq A_i$.

Proof. The update on the top-level memory follows a cycle on $1 \ldots k$, leaving i only when the token visits $\chi^{-1}(\mathcal{C} \setminus A_i)$. Thus, in order for the top-level memory to change continuously, the token has to visit each of the $\chi^{-1}(\mathcal{C} \setminus A_i)$ infinitely often. Proposition 14 follows. □

4.3 $\mathcal{C}$ is winning for Eve, and the root of $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$ has at least one leaf in its children.

As in the previous section, the construction relies on Lemma 12. In fact, the construction for children which are not leaves, labelled $A_1, \ldots, A_k$, is exactly the same. The difference is that we add here a single memory state —0— that represents all the leaves (labelled $A_{-1}, \ldots, A_{-l}$). The memory states are thus updated modulo $k+1$, and not modulo k. The "next-move" function of σ when the top-level memory is 0 is an even distribution over all the successors in A of the current state. The "memory-update" function has probability $\frac{1}{2}$ to stay into 0, and $\frac{1}{2}$ to go to $(1, m_1)$, for some memory state $m_1 \in M_1$. Thus, σ uses memory $\sum_{i=1}^{k} r_{\mathcal{F}_i} + 1$. We prove now that σ is winning. The structure of the proof is the same as in the former section, with some extra considerations for the memory state 0.

Proposition 15. *Let* $\mathfrak{uc}$ *be the event "the top-level memory of* σ *is ultimately constant and different from 0". Then,* $\mathbb{P}_{s}^{\sigma,\tau}(\rho \in \Phi_{\mathcal{F}} \mid \mathfrak{uc}) = 1$.

Proof. The proof is exactly the same as the one of Proposition 13. □

Proposition 16. *The event "the top-level memory is ultimately constant and equal to 0" has probability 0.*

Proof. When the top-level memory is 0, the memory-update function has probability $\frac{1}{2}$ at each step to switch to 1. Proposition 16 follows. □

Proposition 17 considers the case where the top-level memory evolves continuously. By definition of the memory update, this can happen only if all the memory states are visited infinitely often.

Proposition 17. *Let $\mathfrak{ec}$ be the event "the top-level memory takes each value in $0 \ldots k$ infinitely often". Then, $\forall i \in -l \ldots k, \mathbb{P}_s^{\sigma,\tau}(\chi(\mathrm{Inf}(\rho)) \subseteq \mathcal{C}_i \mid \mathfrak{ec}) = 0$.*

Proof. As in the proof of Proposition 14, from the fact that the memory is equal to each of the $i \in 1 \ldots k$ infinitely often, we can deduce that the token surely visits each of the $\mathcal{C} \setminus A_i$ infinitely often. We only need to show that, with probability one and for any $j \in 1 \ldots l$, the set of limit states is not included in A_{-j}. The Zielonka Trees of the conditions $\mathcal{F} \restriction A_{-j}$ are leaves. This means that they are trivial conditions, where all the plays are winning for Adam. Consequently, in this case, Lemma 12 guarantees that $\mathrm{Attr}_E(\chi^{-1}(\mathcal{C} \setminus A_{-j}))$ is the whole arena. The definition of σ in the memory state (0) is to play legal moves at random. There is thus a positive probability that Eve will play according to the attractor strategy a_j long enough to guarantee a positive probability that the token visits $\chi^{-1}(\mathcal{C} \setminus A_{-j})$. To be precise, for any $s \in S$, this probability is greater than $(2 \cdot |S|)^{-|S|}$. Thus, with probability one, the token visits each $\chi^{-1}(\mathcal{C} \setminus A_{-j})$ infinitely often. Proposition 17 follows. $\square$

The initial case, where the Zielonka tree is reduced to a leaf, is trivial: the winner does not depend on the play. Thus, Theorem 9 follows from the results of Sections 4.1, 4.2, and 4.3.

5 Lower Bound

In this section, we consider lower bounds on memory, *i.e.* if we fix a Muller condition $\mathcal{F}$ on a set of colours $\mathcal{C}$, the minimal size of the memory set that is enough to define randomised winning strategies for Eve on any arena coloured by the set $\mathcal{C}$. In his thesis, Majumdar showed the following theorem:

Theorem 18 ([Maj03]). *For any set of colours $\mathcal{C}$, there is a Muller game $\mathcal{G}_\mathcal{C} = (\mathcal{A}_\mathcal{C}, \mathcal{F}_\mathcal{C})$ such that Eve wins, and every randomised almost-sure winning strategy for her in $\mathcal{G}_\mathcal{C}$ has a memory of size at least $\frac{|\mathcal{C}|}{2}!$.*

However, this is a general lower bound on *all* Muller conditions, while we aim to find specific lower bounds for *each* condition. We showed, in [Hor07], that in the special case of Streett Games, the lower bound on pure strategies of [DJW97] still holds for randomised strategies:

Theorem 19 ([Hor07]). *For any $k \in \mathbb{N}$, there is a Streett arena $\mathcal{A}_k$ of index k such that any randomised strategy winning for Eve in every state of her winning region has a memory of size at least $k!$.*

We prove here that there is a lower bound for each Muller condition that matches the upper bound of Theorem 9. This result is formalised as Theorem 20:

Theorem 20. *Let $\mathcal{F}$ be a Muller condition on $\mathcal{C}$. There is a 2-player arena $\mathcal{A}_\mathcal{F}$ over $\mathcal{C}$ such that Eve has a winning strategy, and any randomised winning strategy for Eve in $(\mathcal{A}_\mathcal{F}, \mathcal{F})$ uses memory at least $r_\mathcal{F}$.*

As the construction of the upper bound was based on the Zielonka tree, the lower bound is based on the Zielonka *DAG*:

Definition 21. *The Zielonka DAG $\mathcal{D}_{\mathcal{F},\mathcal{C}}$ of a winning condition $\mathcal{F} \subseteq \mathcal{P}(\mathcal{C})$ is derived from $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$ by merging the nodes which share the same label.*

Notice that the computation of $r_\mathcal{F}$ (Definition 8) is as natural on the DAG as on the tree.

5.1 Cropped DAGs

The relation between $r_{\mathcal{F}}$ and the shape of $\mathcal{D}_{\mathcal{F},\mathcal{C}}$ is asymmetrical: it depends directly on the number of children of Eve's nodes, and not at all on the number of children of Adam's nodes. The notion of *cropped DAG* is the next logical step: a sub-DAG where Eve's nodes keep all their children, while each node of Adam keeps only one child:

Definition 22. *A DAG $\mathcal{E}$ is a cropped DAG of a Zielonka DAG $\mathcal{D}_{\mathcal{F},\mathcal{C}}$ if and only if*

- *The nodes of $\mathcal{E}$ are a subset of the nodes of $\mathcal{D}_{\mathcal{F},\mathcal{C}}$. Furthermore, the owner and label of a node in $\mathcal{E}$ are its owner and label in $\mathcal{D}_{\mathcal{F},\mathcal{C}}$.*
- *There is only one node without predecessor in $\mathcal{E}$, which we call the root of $\mathcal{E}$. It is the root of $\mathcal{D}_{\mathcal{F},\mathcal{C}}$, if it belongs to Eve; otherwise, it is one of its children.*
- *The children of a node of Eve in $\mathcal{E}$ are exactly its children in $\mathcal{D}_{\mathcal{F},\mathcal{C}}$.*
- *A node of Adam has exactly one child in $\mathcal{E}$, chosen among his children in $\mathcal{D}_{\mathcal{F},\mathcal{C}}$, provided there is one. If it has no children in $\mathcal{D}_{\mathcal{F},\mathcal{C}}$, it has no children in $\mathcal{E}$.*

A cropped DAG has the general form of a Zielonka DAG: the nodes belong to either Eve or Adam, and they are labelled by sets of states. We can thus compute the number $r_{\mathcal{E}}$ of a cropped DAG $\mathcal{E}$ in a natural way. In fact, this number has a more intuitive meaning in the case of cropped DAGs: if the leaves belong to Eve, it is the number of branches; if Adam owns the leaves, it is the number of branches with the leaf removed. There is also a direct link between the cropped DAGs of a Zielonka DAG $\mathcal{D}_{\mathcal{F},\mathcal{C}}$ and the number $r_{\mathcal{F}}$: in a cropped DAG, there is one child for each internal node of Adam; in the recursive definition of $r_{\mathcal{F}}$, there is a maximum over the values of the children. Proposition 23 follows directly:

Proposition 23. *Let $\mathcal{F}$ be a Muller condition on $\mathcal{C}$, and $\mathcal{D}_{\mathcal{F},\mathcal{C}}$ be its Zielonka DAG. Then for any cropped DAG $\mathcal{E}$ of $\mathcal{D}_{\mathcal{F},\mathcal{C}}$, we have $r_{\mathcal{E}} \leq r_{\mathcal{F}}$. Furthermore, there is a cropped DAG $\mathcal{E}^*$ such that $r_{\mathcal{E}^*} = r_{\mathcal{F}}$.*

5.2 From cropped DAGs to arenas

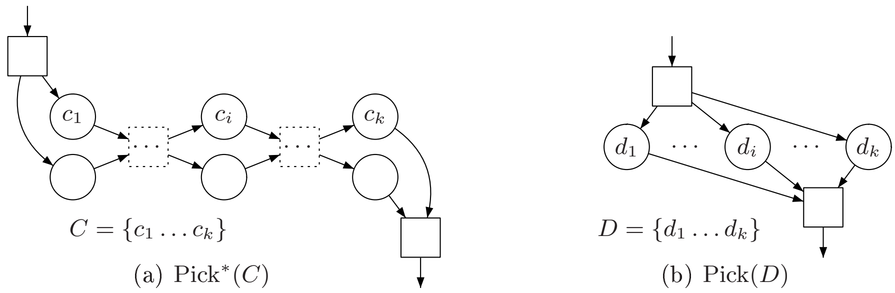
From any cropped DAG $\mathcal{E}$ of $\mathcal{D}_{\mathcal{F},\mathcal{C}}$, we define an arena $\mathcal{A}_{\mathcal{E}}$ which follows roughly the structure of $\mathcal{E}$: the token starts from the root, goes towards the leaves, and then restarts from the root. In her nodes, Eve can choose to which child she wants to go. Adam's choices, on the other hand, consists in either stopping the current traversal or allowing it to proceed.

We present first two "macros", depending on a subset of $\mathcal{C}$:

- in Pick$^*(C)$, Adam can visit any subset of colours in C;
- in Pick(D), he must visit exactly one colour in D.

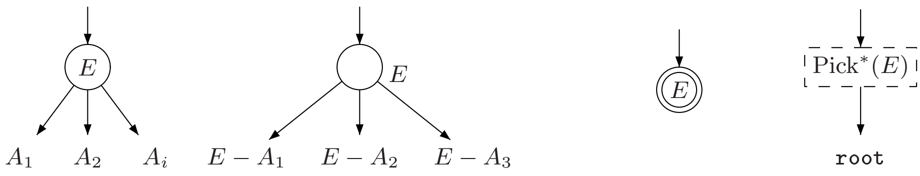
Both are represented in Figure 2, and they are the only occasions where colours are visited in $\mathcal{A}_{\mathcal{E}}$: all the other states are colourless.

Eve's states in the arena $\mathcal{A}_{\mathcal{E}}$ are in bijection with the nodes of $\mathcal{E}$. Likewise, each outgoing transition corresponds to a child of the corresponding node. But the successors of these states are not themselves in bijection with the nodes of Adam: if a single node of Adam A is the child of two different nodes of Eve E and F, we must use the construction of Figure 4 twice: one for $E - A$ and one for $F - A$. In states corresponding to leaves, Eve has no decision to take; Adam can visit any colours in the label of the leaf (Pick* procedure). The token is then sent back to the root. These cases are described in Figure 3.



(a) Pick*(C) (b) Pick(D)

Fig. 2. Pick*(C) and Pick(D)



(a) Node "E" (b) Leaf "E"

Fig. 3. Eve chooses where to go ...

Adam's options on a given node, on the other hand, do not involve the choice of a child: by Definition 22, Adam's nodes in $\mathcal{E}$ have but one child. Instead, he can either stop the current traversal, or, if the current node is not a leaf, allow it to proceed to its only child.

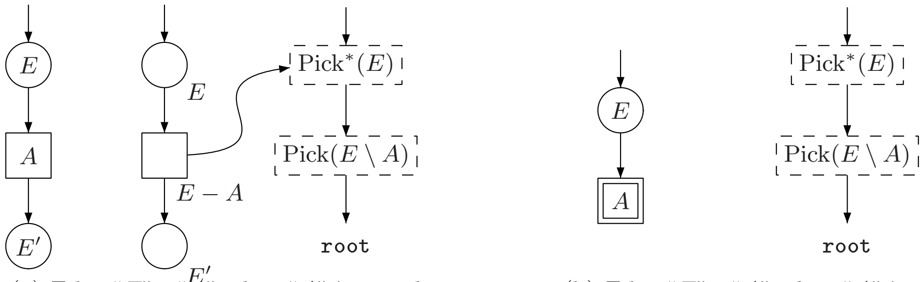
If he chooses to stop, Adam has to visit some coloured states before the token is sent back to the root. The available choices depend on the labels of both the current and the *former* nodes — which is why there are as many copies of Adam's nodes in $\mathcal{A}_{\mathcal{E}}$ as they have parents in $\mathcal{E}$. If the parent is labelled by E, and the current node by A, the token goes through Pick$^*(E)$ and Pick$(E \setminus A)$. Adam can thus choose any number of colours in E, as long as he chooses at least one outside of A.

Notice that if Adam does not stop the traversal, the token is sent to the *unique* state corresponding to the child of the current node. This is why the size of these arenas are roughly DAG-sized, instead of tree-sized.

5.3 Winning strategy, branch strategies, passive strategy

We describe a winning strategy σ for Eve in the game $(\mathcal{A}_{\mathcal{E}}, \mathcal{F})$. Its memory states are the branches of $\mathcal{E}$, and do not change during a traversal. Her moves in the memory state $b = E_1A_1 \dots E_\ell(A_\ell)$ follow the branch b: in the state E_i, Eve chooses the successor corresponding to the transition $E_i - A_i$. Notice that Adam cannot diverge from the branch, as his nodes have at most one child. When he chooses to stop the traversal, Eve updates her memory. For example, if he stops at the ith step, while Eve is in the memory state $b = E_1A_1 \dots E_\ell(A_\ell)$, the update is done as follows:

- If E_i has zero or one child in $\mathcal{E}$, the memory is unchanged;



(a) Edge "E" - "A" when "A" is a node   (b) Edge "E" - "A" when "A" is a leaf

Fig. 4. . . . and Adam chooses when to stop.

– otherwise, the new memory branch has $E_1A_1 \ldots E_iA$ as a prefix, where A is the next child of E_i, or the first one if A_i was the last.

Proposition 24. *The strategy σ is surely winning for Eve in the game $(\mathcal{A}_\mathcal{E}, \mathcal{F})$.*

Proof. Let ρ be a play consistent with σ. We denote by i the smallest integer such that Adam stops infinitely often a traversal at the ith step.

After a finite prefix, the first $2i - 1$ nodes in the memory branch are constant, and we denote them by $E_1A_1E_2 \ldots E_i$. From this point on, whatever Adam does, he can only choose colours in E_i. Furthermore, each time he chooses i, he must choose a state outside of the current A_i, which changes afterwards to the next, in a circular way.

So, in the end, $\mathrm{Inf}(\rho) \subseteq E_i$, and, for any child A of E_i in $\mathcal{E}$, $\mathrm{Inf}(\rho) \nsubseteq A$. Thus ρ is winning for Eve. Proposition 24 follows.

Obviously, Adam has no winning strategy in $\mathcal{A}_\mathcal{E}$. However, we describe the class of *branch strategies* for him, whose point is to punish any attempt of Eve to win with less than $m_\mathcal{F}$ or $r_\mathcal{F}$ memory states. There is one such strategy τ_b for each branch b in $\mathcal{E}$ (whence the name), and the principle is that τ_b stops the traversal as soon as Eve deviates from b:

Definition 25. *The branch strategy τ_b for Adam in $\mathcal{A}_\mathcal{E}$, corresponding to the branch $b = E_1A_1E_2 \ldots E_\ell(A_\ell)$ in $\mathcal{E}$, is a positional strategy whose moves are described below.*

- *In a state $E - A$ such that $\exists i, E = E_i \wedge A \neq A_i$: stop the traversal and visit the colours of A_i;*
- *in a state $E - A$ such that $\exists i, E = E_i \wedge A = A_i$: send the token to E_{i+1};*
- *in the state $E_\ell - A_\ell$: visit E_ℓ;*
- *in the leaf E_ℓ: visit E_ℓ.*

Notice that no move is given for a state $E - A$ such that $\forall i, E \neq E_i$. The reason is that these states are not reachable from the root when Adam plays τ_b, so, in the limit, what he does in these states doesn't matter. Notice also that when Adam chooses to stop a traversal in a state $E_i - A$, he *can* visit exactly the colours of A_i: as A and A_i are maximal subsets of E_i, there is at least one state in $A_i \setminus A$ that he can pick in the $\mathrm{Pick}(E_i \setminus A)$ area.

We informally describe one last strategy for Adam: the *passive strategy*, in which he never stops a traversal before it reaches a leaf, and then plays at random in the Pick / Pick* part.

5.4 Winning against branch strategies

We define the *set of branches of a memory state m* as the branches of $\mathcal{E}$ that have a positive probability to be traversed when Eve is in the memory state m and Adam plays with a passive strategy.

The notion of "branch of a memory state" carries to the case of randomised strategies, but not its unicity: even if Eve starts in the same memory state and Adam plays with a passive strategy, the random decisions can lead to different branches. We consider thus the *set of branches of a memory state m*: they are the branches that have a positive probability to be traversed when Eve is in the memory state m and Adam plays with a passive strategy.

Proposition 26. *Let $\sigma = (M, \sigma^{\mathrm{n}}, \sigma^{\mathrm{u}})$ be an almost-sure winning strategy for Eve in $(\mathcal{A}_\mathcal{E}, \mathcal{F})$. Then σ has memory at least $r_\mathcal{E}$.*

Proof. The idea is that different memory states are necessary to deal with the branch strategies. However, as we will see, a single memory state can sometimes deal with several branch strategies.

Let $b = E_1 A_1 \dots E_\ell (A_\ell)$ be a branch of $\mathcal{E}$ and τ_b be the corresponding branch strategy for Adam. Consider what happens if Eve plays σ and Adam plays τ_b. By definition of τ_b, the set of colours visited in a traversal of ρ is one of the A_i's, or E_ℓ if and only if Eve plays along b. So, as σ wins against τ_b, there is at least one memory state m such that b is a branch of m. However, there can be other branches for m, as long as they lead to visits to A_ℓ, and not another A_i *i.e.*when the other branches are siblings or nephews of b. Consequently, a memory state m is suitable against τ_b if b is a branch of m and $E_1 A_1 \dots E_\ell$ is a prefix of all the branches of m.

It follows that a single memory state can be suitable against two strategies τ_b and $\tau_{b'}$ corresponding to the branches $b = E_1 A_1 \dots E_\ell A_\ell$ and $b' = E'_1 A'_1 \dots E'_{\ell'} A'_{\ell'}$ only if they are siblings, *i.e.*$\ell = \ell'$ and $\forall i < \ell, E_i = E'_i$

There are $r_\mathcal{E}$ equivalence classes for this relation in $\mathcal{E}$. Hence, there must be at least $r_\mathcal{E}$ memory states in M. Proposition 26 follows.

By Proposition 23, there is a cropped DAG $\mathcal{E}$ of $\mathcal{D}_{\mathcal{F},\mathcal{C}}$ such that $r_\mathcal{E} = r_\mathcal{F}$. So, in general, Eve needs randomised strategies with memory $r_\mathcal{F}$ in order to win games whose winning condition is $\mathcal{F}$, which completes the proof of Theorem 20.

6 Conclusions

We have provided a better and tight bound for the memory needed to define almost sure winning randomised strategies. This allows us to characterise the class of games which admit randomised memoryless strategies

Corollary 27 (Randomised Memoryless Conditions). *Eve can restrict herself to randomised memoryless strategies for a Muller condition $\mathcal{F}$ if and only if her nodes in the Zielonka Tree $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$ cannot have more than one child, unless all these children are leaves.*

This yields a `NP` algorithm for the winning problem of such games: randomised memoryless strategy are polynomial witnesses; and solving $1\frac{1}{2}$-player Muller games is PTIME [CdAH04]. Another consequence of our result is that for each Muller condition, at least one of the players cannot improve its memory through randomisation.

Corollary 28. *For any Muller condition $\mathcal{F}$, the player who wins the plays where no colour is visited infinitely often needs as much memory for randomised strategies as for pure strategies.*

A third collateral result is the size of the witness arena in our proof of Theorem 20, which is roughly equivalent to the size of the Zielonka DAG. In [DJW97], the size of the arena was roughly the size of the Zielonka tree, which can be exponentially larger. However, the question of memory bounds in arenas of polynomial size in the number of colours remains unanswered, except for some special cases (Majumdar's "global" lower bound [Maj03] and Streett games [Hor07]).

We intend now to consider the case of games whose winning condition is a regular language, in order to get cheaper alternatives to the use of the Muller normal form.

References

[BL69] J. Richard Büchi and Lawrence H. Landweber. Solving sequential conditions by finite-state strategies. *Transactions of the American Mathematical Society*, 138:295–311, 1969.

[CdAH04] Krishnendu Chatterjee, Luca de Alfaro, and Thomas A. Henzinger. Trading memory for randomness. In *Proceedings of the 1st International Conference on Quantitative Evaluation of Systems, QEST'04*, pages 206–217, 2004.

[Cha07] Krishnendu Chatterjee. Optimal strategy synthesis in stochastic muller games. In *Proceedings of the 10th International Conference on Foundations of Software Science and Computation Structures, FOSSACS'07*, pages 138–152, 2007.

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

[deA97] Luca de Alfaro. *Formal Verification of Probabilistic Systems.* PhD thesis, Stanford University, 1997.

[DJW97] Stefan Dziembowski, Marcin Jurdzinski, and Igor Walukiewicz. How much memory is needed to win infinite games? In *Proceedings of the 12th Symposium on Logic in Computer Science, LICS'97*, pages 99–110, 1997.

[GH82] Yuri Gurevich and Leo Harrington. Trees, automata, and games. In *Proceedings of the 14th Annual ACM Symposium on Theory of Computing, STOC'82*, pages 60–65, 1982.

[Hor07] Florian Horn. Dicing on the Streett. *Information Processing Letters*, 104(1):1–9, 2007.

[Maj03] Rupak Majumdar. *Symbolic Algorithms for Verification and Control.* PhD thesis, Berkeley University, 2003

[MP92] Zohar Manna and Amir Pnueli. *The temporal Logic of Reactive and Concurrent Systems: Specification.* Springer, 1992.

[PR89] Amir Pnueli and Roni Rosner. On the synthesis of a reactive module. In *Proceedings of the 16th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 179–190, 1989.

[Tho95] Wolfgang Thomas. On the synthesis of strategies in infinite games. In *Proceedings of the 24th International Symposium on Theoretical Aspects of Computer Science, STACS'07*, pages 1–13, 1995.

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