Random Fruits on the Zielonka Tree

Florian Horn

RANDOM FRUITS ON THE ZIELONKA TREE

FLORIAN HORN [1]

[1] CWI, Amsterdam, The Netherlands
E-mail address: `f.horn@cwi.nl`

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 [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 states that, from any initial state, one of the players has a winning strategy [BL69]. Gurevich and Harrington used the *latest appearance record* (LAR) 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 the winning regions [Zie98]. An insightful analysis of the Zielonka tree by Dziembowski, Jurdzinski, and Walukiewicz leads to optimal (and asymmetrical) memory bounds for pure (non-randomised) winning strategies [DJW97]. Chatterjee extended these bounds to the case of pure strategies over $2\frac{1}{2}$-player arenas [Cha07b]. However, the lower bound on memory does not hold for *randomised strategies*, even in non-stochastic arenas: Chatterjee, de Alfaro, and Henzinger show that memoryless randomised strategies are enough for to deal with upward-closed winning conditions [CdAH04]. Chatterjee extends this result in [Cha07a], showing that conditions with non-trivial upward-closed subsets admit randomised strategies with less memory than pure ones.

Our contributions. The memory bounds of [Cha07a] are not tight in general, even for 2-player arenas. We give here matching upper and lower bounds for any Muller condition $\mathcal{F}$, in the form of a number $r_\mathcal{F}$ computed from the Zielonka tree of $\mathcal{F}$:

- if Eve has a winning strategy in a $2\frac{1}{2}$-player game $(\mathcal{A}, \mathcal{F})$, she has a randomised winning strategy with memory $r_\mathcal{F}$ (Theorem 4.2);
- 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 5.2).

Furthermore, the witness arenas we build in the proof of Theorem 5.2 are significantly smaller than 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 $(\mathcal{A}, \mathcal{F})$. In Section 5, we show that this bound is tight. Finally, in Section 6, we characterise the class of Muller conditions that admit memoryless randomised strategies, and 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. A probability distribution γ over a set X is a function from X to $[0, 1]$ such that $\sum_{x \in X} \gamma(x) = 1$. The set of probability distributions over X is denoted by $\mathcal{D}(X)$.

Arenas. A $2\frac{1}{2}$-*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 $\delta(s)(t) > 0 \Leftrightarrow (s, t) \in \mathcal{T}$, 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. $\square$, $\triangle$). A 2-*player arena* is an arena where $\mathcal{S}_R = \emptyset$.

A set $U \subseteq \mathcal{S}$ of states is δ-*closed* if for every random state $u \in U \cap \mathcal{S}_R$, $(u, t) \in \mathcal{T} \to t \in U$. 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 live and δ-closed subset U induces a *subarena* of $\mathcal{A}$, denoted by $\mathcal{A} \restriction U$.

Plays and Strategies. An infinite path, or *play*, over the arena $\mathcal{A}$ is an infinite sequence $\rho = \rho_0\rho_1\ldots$ of states such that $(\rho_i, \rho_{i+1}) \in \mathcal{T}$ for all $i \in \mathbb{N}$. The set of states occurring infinitely often in a play ρ is denoted by $\mathrm{Inf}(\rho) = \{s \mid \exists^\infty i \in \mathbb{N}, \rho_i = s\}$. 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 $\sigma = (M, \sigma^{\mathbf{n}}, \sigma^{\mathbf{u}})$, where $\sigma^{\mathbf{n}}$ is the "next-move" function from $(\mathcal{S}_E \times M)$ to $\mathcal{D}(\mathcal{S})$ and $\sigma^{\mathbf{u}}$ 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. 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 $\mathcal{S}^*$ and the memory update is $\sigma^{\mathbf{u}}(s, w) \mapsto ws$.

Once a starting state $s \in \mathcal{S}$ and strategies $\sigma \in \Sigma$ for both players 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 $w_i \in \mathcal{S}_E$, $\mathbb{P}_{w_0}^{\sigma,\tau}(\rho_{i+1} = w_{i+1} \mid \rho_0 = w_0 \ldots \rho_i = w_i) > 0$. 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:

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

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 *winning condition* is a subset Φ of Ω. A play ρ is *winning for Eve* if $\rho \in \Phi$, and *winning for Adam* otherwise. 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, and Eve wins a play if and only if the set of colours visited infinitely often belongs to $\mathcal{F}$: $\Phi_\mathcal{F} = \{\rho \in \Omega \mid \chi(\mathrm{Inf}(\rho)) \in \mathcal{F}\}$. An example of Muller game is given in Figure 1(a). We use it throughout the paper to describe various notions and results.

Winning Strategies. A strategy σ for Eve is *surely winning* (or *sure*) from a state s for the winning condition Φ if any play consistent with σ belongs to Φ, and *almost-surely winning* (or *almost-sure*) if for any strategy τ for Adam, $\mathbb{P}_s^{\sigma,\tau}(\Phi) = 1$. The *sure* and *almost-sure* regions are the sets of states from which she has a sure (resp. almost-sure) strategy.
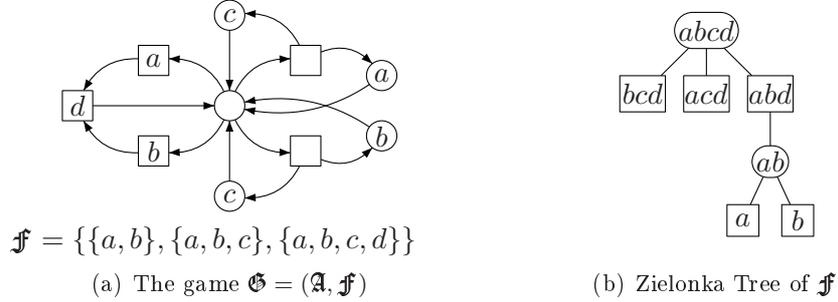
$$\mathit{f\!f} = \{\{a,b\},\{a,b,c\},\{a,b,c,d\}\}$$

(a) The game $\mathfrak{G} = (\mathfrak{A}, \mathit{f\!f})$ (b) Zielonka Tree of $\mathit{f\!f}$

Figure 1: Recurring Example

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 3.1 (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 $\mathcal{F} \restriction \mathcal{C}_i$, *i.e.* the $\mathcal{Z}_{\mathcal{F}\restriction\mathcal{C}_i, \mathcal{C}_i}$, 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 sure strategies:

Definition 3.2 (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:

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

Theorem 3.3 ([DJW97]). *If Eve has a sure strategy in a 2-player Muller game with the winning condition $\mathcal{F}$, she has a pure sure strategy with at most $m_\mathcal{F}$ memory states. Furthermore, there is a 2-player arena $\mathcal{A}_\mathcal{F}$ such that Eve has a sure strategy, but none of her sure strategies have less than $m_\mathcal{F}$ memory states.* ∎

Theorem 3.4 ([Cha07b])*. If Eve has an almost-sure strategy in a $2\frac{1}{2}$-player Muller game with the winning condition $\mathcal{F}$, she has a pure almost-sure with at most $m_{\mathcal{F}}$ memory states.* ∎

3.2. Memory reduction through randomisation

Randomised strategies are more general than pure strategies, and in some cases, they are also more compact. In [CdAH04], a first result showed that upward-closed conditions admit memoryless randomised strategies, while they don't admit memoryless pure strategies:

Theorem 3.5 ([CdAH04])*. If Eve has an almost-sure strategy in a $2\frac{1}{2}$-player Muller game with an upward-closed winning condition, she has a randomised almost-sure strategy.* ∎

This result was later extended in [Cha07a], by removing the leaves attached to a node of the Zielonka Tree representing an upward-closed subcondition:

Definition 3.6 ([Cha07a])**.** Let $\mathcal{F} \subseteq \mathcal{P}(\mathcal{C})$ be a Muller condition, and $\mathcal{Z}_{\mathcal{F}_1,\mathcal{C}_1}, \mathcal{Z}_{\mathcal{F}_2,\mathcal{C}_2}, \dots,$ $\mathcal{Z}_{\mathcal{F}_k,\mathcal{C}_k}$ be the subtrees attached to the root of the tree $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$. We define the number $m_{\mathcal{F}}^U$ inductively as follows:

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

Theorem 3.7 ([Cha07a])*. If Eve has an almost-sure strategy in a $2\frac{1}{2}$-player Muller game with the winning condition $\mathcal{F}$, she has a randomised almost-sure strategy with at most $m_{\mathcal{F}}^U$ memory states.* ∎

4. Randomised Upper Bound

The upper bound of Theorem 3.7 is not tight for all conditions. For example, the number $m_{\mathfrak{f}}^U$ of the condition $\mathfrak{f}$ in Figure 1(b) is three, while there is always an almost-sure strategy 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 4.1 (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 $\mathcal{Z}_{\mathcal{F}_1,\mathcal{C}_1}, \mathcal{Z}_{\mathcal{F}_2,\mathcal{C}_2}, \dots,$ $\mathcal{Z}_{\mathcal{F}_k,\mathcal{C}_k}$ the non-leaves subtrees attached to the root of $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$. We define $r_{\mathcal{F}}$ inductively as follows:

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

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, the fourth case cannot occur. In the other case, the intuition is that we merge leaves if they are siblings. For example, the number $r_{\mathpzc{f}}$ 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 $m_{\mathpzc{f}}$ is four (one for each leaf), and $m_{\mathpzc{f}}^U$ 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 4.2:

Theorem 4.2 (Randomised upper bound). *If Eve has an almost-sure strategy in a $2\text{-}\frac{1}{2}$ player Muller game with the winning condition $\mathcal{F}$), she has an almost-sure 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 an almost-sure strategy for Eve with $r_{\mathcal{F}}$ memory states in each non-trivial case in the definition of $r_{\mathcal{F}}$. We use two lemmas — Lemmas 4.3 and 4.5 — that derive directly from similar results in [DJW97] and [Cha07b]. 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* $\nearrow$) 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 4.3:

Lemma 4.3. *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 $(\mathcal{A}_i, \mathcal{F} \restriction E_i)$;*
- *$\mathcal{A} = \mathrm{Attr}_E(\cup_{j=1}^n \mathcal{A}_j)$.*

Let the subarenas $\mathcal{A}_i$ be the ones whose existence is proved in this lemma. We denote by σ_i the almost-sure 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 $s \in \mathcal{A}_i$
  - $\sigma^{\mathrm{u}}(s, m) = \sigma_i^{\mathrm{u}}(s, m)$
  - $\sigma^{\mathrm{n}}(s, m) = \sigma_i^{\mathrm{n}}(s, m)$
- if $s \in \mathrm{Attr}_E(\cup_{k=1}^i \mathcal{A}_k) \setminus \mathcal{A}_i$
  - $\sigma^{\mathrm{u}}(s, m) = m$
  - $\sigma^{\mathrm{n}}(s, m) = a_i(s)$

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

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. ∎

The strategy σ_i is almost-sure from any state in $\mathcal{A}_i$. As Muller conditions are prefix-independent, it follows from Proposition 4.4 that σ is also almost-sure 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 4.5. *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 almost-sure 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 σ^{n} and σ^{u} for a memory state $m = (i, m^i)$ are defined below:

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

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', so σ's needs the sum of the $\{r_{\mathcal{F} \upharpoonright A_i}\}$'s.

Proposition 4.6. *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 almost-sure 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 4.6 follows. ∎

Proposition 4.7. *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 4.7 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 4.5. 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 almost-sure. The structure of the proof is the same as in the former section, with some extra considerations for the memory state 0.

Proposition 4.8. *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 4.6. ∎

Proposition 4.9. *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 4.9 follows. ∎

Proposition 4.10 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 4.10. *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 4.7, 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 $\mathcal{F} \upharpoonright A_{-j}$ are leaves. This means that they are trivial conditions, where all the plays are winning for Adam. Consequently, in this case, Lemma 4.5 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 4.10 follows. ∎

The initial case, where the Zielonka tree is reduced to a leaf, is trivial: the winner does not depend on the play. Thus, Theorem 4.2 follows from 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 almost-sure strategies for Eve on any arena coloured by the set $\mathcal{C}$. In his thesis, Majumdar showed the following theorem:

Theorem 5.1 ([Maj03]). *For any set of colours $\mathcal{C}$, there is a 2-player Muller game $\mathcal{G}_{\mathcal{C}} = (\mathcal{A}_{\mathcal{C}}, \mathcal{F}_{\mathcal{C}})$ such that Eve has an almost sure, but none of her almost-sure strategies have less than $\frac{|\mathcal{C}|}{2}!$ memory states.*

However, this is a general lower bound on *all* Muller conditions, while we aim to find specific lower bounds for *each* condition. We prove here that there is a lower bound for each Muller condition that matches the upper bound of Theorem 4.2:

Theorem 5.2. *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 sure strategy, but none of her almost-sure strategies have less than $r_{\mathcal{F}}$ memory states.*

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

Definition 5.3. 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.

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 5.4. 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 nodes of $\mathcal{D}_{\mathcal{F},\mathcal{C}}$, with the same owner and label.
- 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 $\mathcal{E}$.

Cropped DAG resemble Zielonka DAGs: 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. Furthermore, there is a direct link between the cropped DAGs of a Zielonka DAG $\mathcal{D}_{\mathcal{F},\mathcal{C}}$ and the number $r_{\mathcal{F}}$:

Proposition 5.5. *Let $\mathcal{F}$ be a Muller condition on $\mathcal{C}$, and $\mathcal{D}_{\mathcal{F},\mathcal{C}}$ be its Zielonka DAG. Then 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 first present 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.
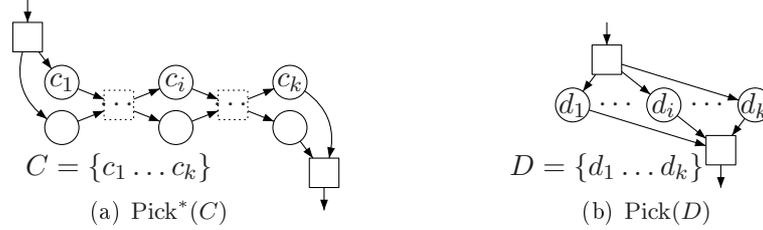

(a) Pick*(C)   (b) Pick(D)

Figure 2: Pick*(C) and Pick(D)

Eve's states in the arena $\mathcal{A}_{\mathcal{E}}$ are in bijection with her nodes in $\mathcal{E}$. Adam's nodes, on the other hand, are in bijection with the pairs parent-child of $\mathcal{E}$, where the parent belongs to Eve and the child to Adam.

In the state corresponding to the node n, Eve can send the token to any state of the form $n - c$. In states corresponding to leaves, Eve has no decision to take, and Adam can visit any colours in the label of the leaf (Pick* procedure). The token is then sent back to the root.

Adam's moves do not involve the choice of a child: by Definition 5.4, 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 \ A). Adam can thus choose any number of colours in E, as long as he chooses at least one outside of A.

(a) Edge "E" - "A" when "A" is a node   (b) Edge "E" - "A" when "A" is a leaf
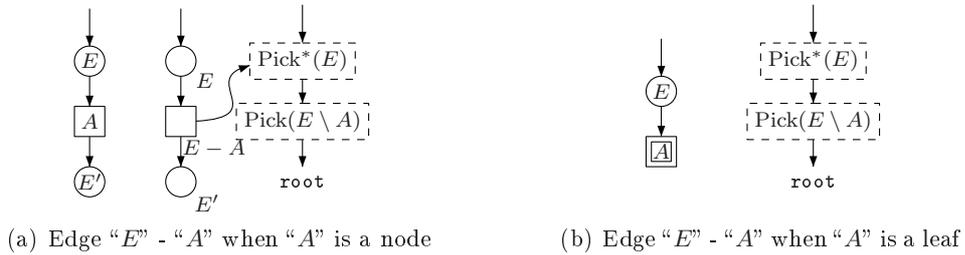
Figure 3: Adam's states in $\mathcal{A}_{\mathcal{E}}$.

5.3. Winning strategy and branch strategies

We first describe a sure 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. If the current memory state is

$b = E_1 A_1 \ldots E_\ell(A_\ell)$, Eve's moves follow the branch b: in E_i, she goes to $E_i - A_i$. When Adam stops the traversal at the ith step, Eve updates her memory as follows:

- If E_i has zero or one child in $\mathcal{E}$, the memory is unchanged;
- otherwise, the new memory branch has $E_1 A_1 \ldots E_i A$ as a prefix, where A is the next child of E_i, or the first one if A_i was the last.

Proposition 5.6. *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 traversals stops infinitely often at the ith step. After a finite prefix, the first $2i - 1$ nodes in the memory branch are constant, and we denote them by $E_1 A_1 E_2 \ldots E_i$. From this point on, the colours visited belong to E_i. Furthermore, each time a traversal stops at step i, a state is visited outside of the current A_i, which changes afterwards to the next, in a circular way. It follows that $\mathrm{Inf}(\rho) \subseteq E_i$, and, for any child A of E_i in $\mathcal{E}$, $\mathrm{Inf}(\rho) \nsubseteq A$. Thus ρ is winning for Eve. Proposition 5.6 follows. ∎

Obviously, Adam has no winning strategy in $\mathcal{A}_\mathcal{E}$. However, we describe the class of *branch strategies*, whose point is to punish any attempt of Eve to win with less than $r_\mathcal{F}$ 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 5.7. The branch strategy τ_b for Adam in $\mathcal{A}_\mathcal{E}$, corresponding to the branch $b = E_1 A_1 E_2 \ldots E_\ell(A_\ell)$ 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 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 $E_\ell - A_\ell$, or the leaf E_ℓ: visit the colours of E_ℓ.

No move is given for a state $E - A$ such that $\forall i, E \neq E_i$, as these states are not reachable from the root when Adam plays τ_b. 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.

5.4. Winning against branch strategies

The key idea of the proof of Theorem 5.2 is that if two branches b and b' of $\mathcal{E}$ are too different, Eve needs different memory states to win against τ_b and $\tau_{b'}$.

Proposition 5.8. *Let $\sigma = (M, \sigma^\mathrm{n}, \sigma^\mathrm{u})$ be an almost-sure strategy for Eve in $(\mathcal{A}_\mathcal{E}, \mathcal{F})$. Then σ has memory at least $r_\mathcal{E}$.*

Proof. Let $b = E_1 A_1 \ldots E_\ell(A_\ell)$ be a branch of $\mathcal{E}$ and τ_b be the corresponding branch strategy for Adam. By definition of τ_b, the set of colours visited in a traversal consistent with τ_b is one of the A_i's, or E_ℓ if and only if Eve plays along b. As σ is almost-sure, there must be a memory state m such that Eve has a positive probability to play along b. It is also necessary to ensure that none of the A_i's is visited infinitely often, with the possible exception of A_ℓ. So, if Eve has a positive to play along a branch b' when she is in the memory state m, $E_1 A_1 \ldots E_\ell$ must be a prefix of b'. It follows that a single memory state can be suitable against two strategies τ_b and $\tau_{b'}$ with $b = E_1 A_1 \ldots E_\ell(A_\ell)$ and $b' = E_1' A_1' \ldots E_{\ell'}'(A_{\ell'}')$ only if $\ell = \ell'$ and $\forall i \leq \ell, E_i = E_i'$. By Definition 4.1, the underlying equivalence relation has $r_\mathcal{E}$ equivalence classes. Proposition 5.8 follows. ∎

By Proposition 5.5, 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 $\mathcal{F}$. This completes the proof of Theorem 5.2.

6. Conclusion

We have provided better and tight bounds for the memory needed to define almost sure winning randomised strategies. This allows us to characterise the class of Muller conditions which admit randomised memoryless strategies:

Corollary 6.1. *Eve admits randomised memoryless almost-sure strategies for a Muller condition $\mathcal{F}$ if and only if all her nodes in $\mathcal{Z}_{\mathcal{F},\mathcal{C}}$ have either one child, or only leave children.* ∎

This yields a NP algorithm for the winner problem of such games, as 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 6.2. *Let $\mathcal{F}$ be a Muller condition. If $\emptyset \in \mathcal{F}$, Eve needs as much memory for randomised strategy as for pure strategies. Otherwise, Adam does.* ∎

Our proof of lower bound also improves on the size of the witness arena: it is roughly equivalent to the size of the Zielonka DAG, instead of the size of the Zielonka tree. Whether these bounds still hold for arenas of size polynomial in the number of colours remains an open question, except for special cases like Streett games [Hor07].

References

[BL69]   J. Richard Büchi and Lawrence H. Landweber. Solving sequential conditions by finite-state strategies. *TAMS*, 138:295–311, 1969.

[CdAH04] Krishnendu Chatterjee, Luca de Alfaro, and Thomas A. Henzinger. Trading memory for randomness. In *QEST*, pages 206–217, 2004.

[Cha07a]   Krishnendu Chatterjee. Optimal Strategy Synthesis in Stochastic Muller Games. In *FOSSACS*, pages 138–152, 2007.

[Cha07b]   Krishnendu Chatterjee. Stochastic Müller Games are PSPACE-Complete. In *FSTTCS*, pages 436–448, 2007.

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

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

[GH82]   Yuri Gurevich and Leo Harrington. Trees, automata, and games. In *STOC*, pages 60–65, 1982.

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

[Maj03]   Rupak Majumdar. *Symbolic Algorithms for Verification and Control*. PhD thesis, Berkeley, 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 *POPL*, pages 179–190, 1989.

[Tho95]   Wolfgang Thomas. On the synthesis of strategies in infinite games. In *STACS*, pages 1–13, 1995.

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