

# Using games and universal trees to characterise the nondeterministic index of tree languages

Olivier Idir ✉ 

Université Paris Cité, CNRS, IRIF, France

Karoliina Lehtinen ✉ 

CNRS, Université Aix-Marseille, LIS, Marseille, France

---

## Abstract

The parity index problem of tree automata asks, given a regular tree language  $L$  and a set of priorities  $J$ , is  $L$   $J$ -feasible, that is, recognised by a nondeterministic parity automaton with priorities  $J$ ? This is a long-standing open problem, of which only a few sub-cases and variations are known to be decidable. In a significant but technically difficult step, Colcombet and Löding reduced the problem to the uniform universality of distance-parity automata. In this article, we revisit the index problem using tools from the parity game literature.

We add some counters to Lehtinen’s register game, originally used to solve parity games in quasipolynomial time, and use this novel game to characterise  $J$ -feasibility. This provides a alternative proof to Colcombet and Löding’s reduction.

We then provide a second characterisation, based on the notion of attractor decompositions and the complexity of their structure, as measured by a parameterised version of their Strahler number, which we call  $n$ -Strahler number. Finally, we rephrase this result using the notion of universal tree extended to automata: a guidable automaton recognises a  $[1, 2j]$ -feasible language if and only if it admits a universal tree with  $n$ -Strahler number  $j$ , for some  $n$ . In particular, a language recognised by a guidable automaton  $A$  is Büchi-feasible if and only if there is a uniform bound  $n \in \mathbb{N}$  such that all trees in the language admit an accepting run with an attractor decomposition of width bounded by  $n$ . Equivalently, the language is Büchi-feasible if and only if  $A$  admits a *finite* universal tree.

While we do not solve the decidability of the index problem, our work makes the state-of-the-art more accessible and brings to light the deep relationships between the  $J$ -feasibility of a language and attractor decompositions, universal trees and Lehtinen’s register game.

**2012 ACM Subject Classification** Theory of computation  $\rightarrow$  Automata over infinite objects

**Keywords and phrases** Tree automata, parity automata, Mostowski index, Strahler number, attractor decomposition, universal trees

**Digital Object Identifier** 10.4230/LIPIcs.ICALP.2025.159

**Category** Track B: Automata, Logic, Semantics, and Theory of Programming

**Related Version** This is a full version of an article of the same name published at ICALP 2025

**Acknowledgements** We are grateful to Marcin Jurdziński for insightful discussions about attractor decompositions at Dagstuhl Seminar 24231, and of course to Dagstuhl for enabling such discussions.

## 1 Introduction

Finite-state [automata](#) running on infinite structures are fundamental to the theory of verification and synthesis, where they model non-terminating systems. The complexity of an [automaton](#) is measured not only by the size of its state-space, but also by the complexity of the acceptance condition. For instance, while the membership and non-emptiness questions for Büchi and coBüchi tree automata are in PTIME, for parity automata they are fixed-parameter tractable in the number of priorities in the parity condition, called its index [2]. In the modal  $\mu$ -calculus, the logic corresponding to parity tree automata, the alternation



© Olivier Idir and Karoliina Lehtinen;

licensed under Creative Commons License CC-BY 4.0

52nd International Colloquium on Automata, Languages, and Programming (ICALP 2025).

Editors: Keren Censor-Hillel, Fabrizio Grandoni, Joel Ouaknine, and Gabriele Puppis; Article No. 159; pp. 159:1–159:18



Leibniz International Proceedings in Informatics

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

depth of a formula – that is, the nesting depth of alternating least and greatest fixpoints – coincides with the index of the corresponding parity automaton.

While for nondeterministic automata over infinite words, the Büchi acceptance condition suffices to recognise all  $\omega$ -regular languages [22], the classes of languages recognised by parity tree automata of each index form an infinite hierarchy, often called the parity, Mostowski, or Rabin-Mostowski index hierarchy. In other words, no fixed parity index suffices to recognise all  $\omega$ -regular tree languages, and this is the case for both nondeterministic [18], and alternating [1, 16] tree automata. A language is said to be  $J$ -feasible if it is recognised by a nondeterministic parity automaton of index  $J$ . The nondeterministic index of an  $\omega$ -regular tree language is the minimal index  $J$  for which it is  $J$ -feasible. The decidability of the index of a language is one of the major open problems in automata theory.

In the case of infinite words, the *deterministic* index of a language is decidable in PTIME [20]. In the case of infinite trees, however, not much is known. For languages given by deterministic parity automata, deciding their nondeterministic index is decidable [21]. Similarly, deciding if a language is recognisable with a safety/reachability condition can be done in EXPTIME [25]. CoBüchi-feasibility, as well as the weak feasibility of Büchi languages, are also decidable [3, 24]. For the restricted class of game automata (which can be seen as the closure of deterministic automata under complementation and composition), the nondeterministic and alternating index problems are decidable [10]. The most recent advance on the topic is that the *guidable* index of a language is decidable [19], where *guidable* automata, introduced by Colcombet and Löding [4], restrict the nondeterminism of the automaton without the loss of expressivity imposed by determinism.

The general nondeterministic index problem remains wide open. However, in a significant step, in 2008, Colcombet and Löding [4] reduced the index problem of a tree language to the uniform universality of distance-parity automata. This remarkable result is, however, quite technical. In this article we present a similar result, (from which Colcombet and Löding’s result can be obtained as a corollary, see Remark 12), using variations of known tools from the parity game literature – namely, attractor decompositions, universal trees, the register index of parity games, and Strahler numbers. These are all notions that (re-)emerged in the aftermath of Calude et al.’s first quasipolynomial algorithm for solving parity games [2] to provide clarity on the newly established complexity bound. Here, we demonstrate that these tools also provide insight into the index hierarchy by using them to reformulate Colcombet and Löding’s result and give an alternative proof.

Let us discuss each of these notions in more detail, in order to state our results.

**The register index of parity games and  $J$ -feasibility.** Parity games are infinite two-player games in which two players, Adam and Eve, take turns moving a token along the edges of a graph labelled with integer priorities. Eve’s goal is to ensure that the infinite path taken by the token satisfies the parity condition, that is, that the highest priority occurring infinitely often along the path is even. The acceptance of a tree by a parity tree automaton is determined by whether Eve wins a parity game based on the input tree and the automaton.

In this article, we use the data-structure introduced in Lehtinen’s quasipolynomial parity game algorithm [15]. Lehtinen reduces solving a parity game to solving a new game, in which Eve must map the original game’s priorities into a smaller priority range using a purpose-built data-structure, while guaranteeing that the sequence of outputs in this smaller range still satisfies the parity condition. Lehtinen shows that for a parity game of size  $n$ , Eve wins if and only if she also wins this new game with output range  $O(\log n)$ , which can be

solved in quasipolynomial time.

Here we extend this game to the acceptance parity games of nondeterministic parity tree automata, that is, parity games with unbounded or even infinite arenas. We furthermore add some counters (inspired by the Colcombet and Löding construction), which give Eve some additional (but bounded) leeway in her mapping. We obtain a game that we call the parity transduction game  $\mathcal{T}_J^n(G)$ , played over a parity game  $G$ , parameterised by the output priority range  $J$ , and the bound  $n$  on the counters.

Our first contribution is showing that the  $J$ -feasibility of the language of a guidable automaton  $A$  (and we can always make the input guidable) is characterised by the existence of an integer  $n$  such that the parity transduction game with parameters  $J$  and  $n$  coincides with the acceptance game of  $A$ , written  $\mathcal{G}(A, t)$  for an input tree  $t$ . In other words, a language is  $J$ -feasible whenever there is a uniform parameter  $n$ , such that whenever Eve wins the acceptance game  $\mathcal{G}(A, t)$ , she also wins the transduction game over it, with output range  $J$  and parameter  $n$ .

- **Theorem 1.** *Given a guidable automaton  $A$ ,  $J$  an index, the following are equivalent:*
- $\mathcal{L}(A)$  is  $J$ -feasible.
  - There exists  $n \in \mathbb{N}$  such that for all  $\Sigma$ -tree  $t$ ,  $t \in \mathcal{L}(A)$  if and only if Eve wins  $\mathcal{T}_J^n(\mathcal{G}(A, t))$ .

This corresponds to our version of the Colcombet-Löding reduction. We then proceed to reinterpret this characterisation in terms of *attractor decompositions* and *universal trees*.

**Attractor decompositions** describe the structure of Eve's winning strategies in a parity game, or, equivalently, of accepting runs of a parity tree automaton. While this notion appears at least implicitly in many seminal works, e.g. Zielonka's algorithm for parity games [26], Kupferman and Vardi's automata transformations [14] and Klarlund's [13] proof of Rabin's complementation theorem, it has more recently been explicitly studied for mean payoff parity games [6] and parity games [7, 12].

While similar in spirit and structure to progress-measures [11], which count the number of odd priorities that might occur before a higher priority, attractor decompositions are more suitable for parity games on infinite arenas, where Eve might see an unbounded number of odd priorities in a row, as long as she is advancing in the attractor of some larger even priority. While progress measures, bounded by the size of a finite game, can be seen as a way to reduce parity games to *safety* games, here we use attractor decompositions with bounded structure to reduce the priority range of the parity condition. Like progress-measures, attractor decompositions have a tree-like structure, where the play only moves to the right if a suitably high even priorities occurs. The structure of these trees turns out to be closely tied to the index of a language.

**$n$ -Strahler number** The Strahler number of a tree  $t$  consists in the largest  $h$  such that  $t$  admits a complete binary tree of height  $h$  as a minor. Daviaud, Jurdziński and Thejaswini [8] proved an equivalence between the output range that Eve needs in Lehtinen's game, called the game's register index, and the Strahler-number of the attractor decompositions of Eve's strategies. Inspired by this, we define, for  $n \in \mathbb{N}$ , the  $n$ -Strahler number of a tree  $t$ , that consists in the largest  $h$  such that  $t$  admits a complete  $(n+1)$ -ary tree of height  $h$  as a minor (by subtree deletion and single-child contraction; we do not allow edge contraction in the presence of siblings). The Strahler number corresponds to our 1-Strahler number. Our

second characterisation of the index of a languages is based on the  $n$ -Strahler number of attractor decompositions.

► **Theorem 2.** *Given a **guidable nondeterministic parity tree automaton**  $A$ , the following are equivalent:*

- $L(A)$  is  $[1, 2j]$ -feasible.
- There is an  $n \in \mathbb{N}$  such that for all  $t \in L(A)$  there exists a **run** of  $A$  on  $t$  with an **attractor decomposition** of  $n$ -Strahler number at most  $j$ .

In particular, Büchi feasibility coincides with the existence of a uniform bound on the width (i.e branching degree) of attractor decompositions needed by Eve. Finally, we restate this result in terms of *universal trees*, extended to automata, as follows.

**Universal trees** Given a set  $\mathcal{T}$  of ordered trees of bounded depth, a tree  $U$  is said to be universal for  $\mathcal{T}$  if all  $t \in \mathcal{T}$  can be obtained from  $U$  by removing subtrees. We then say that  $t$  is isomorphically embedded in  $U$ . This elegant notion emerged in the analysis of quasi-polynomial time parity game algorithms, as a unifying combinatorial structure that can be extracted from the different algorithms [5].

We say that an ordered tree  $U$  is universal for an automaton  $A$  if for all regular trees in the language of  $A$ , there exists an accepting run with an attractor decomposition (seen as a tree) that can be isomorphically embedded in  $U$ .

Then, the  $[1, 2j]$ -feasibility of the language of a guidable automaton  $A$  is characterised by the existence of an ordered tree universal for  $A$  of  $n$ -Strahler  $j$ , for some  $n \in \mathbb{N}$ . Büchi-feasibility is equivalent to the existence of a *finite* universal tree for  $A$ .

► **Theorem 3.** *Given a **guidable nondeterministic parity tree automaton**  $A$ , the following are equivalent:*

- $L(A)$  is  $[1, 2j]$ -feasible.
- There exists an  $n \in \mathbb{N}$  and a tree  $U$  of  $n$ -Strahler number at most  $j$  that is **universal** for  $A$ .

While our work does not give us the decidability of the index problem, it provides new tools for tackling it and makes the state-of-the-art more accessible by relating it to other familiar concepts. We hope that the deep link between the index of a language and the structure of attractor decompositions will be helpful for future work. The remarkably simple characterisation of Büchi feasible languages, as those with attractor decompositions of bounded width, or, equivalently a finite universal tree, is particularly encouraging, as deciding Büchi-feasibility is the next challenge for advancing on the index problem.

## 2 Preliminaries

The set of natural numbers  $\{0, 1, \dots\}$  is denoted  $\mathbb{N}$ , the set of strictly positive numbers is denoted  $\mathbb{N}^+$ . The disjoint union of two sets  $A$  and  $B$  is denoted  $A \sqcup B$ . An **alphabet** is a finite non-empty set  $\Sigma$  of elements, called letters.  $\Sigma^*$  and  $\Sigma^\omega$  denote the sets of finite and infinite **words** over  $\Sigma$ , respectively. For  $u$  a (possibly infinite) **word** and  $n \in \mathbb{N}$ , the **word**  $u|_n$  consists of the first  $n$  letters of  $u$ . For  $u$  and  $v$  finite **words**,  $u \cdot v$  denotes the concatenation of  $u$  and  $v$ . The length of a finite **word**  $u$  is written  $|u|$ .

⌈ An *index*  $[i, j]$  is a non-empty finite range of natural numbers  $I = \{i, i+1, \dots, j\} \subseteq \mathbb{N}$ . Elements  $c \in I$  are called *priorities*. We say that an infinite sequence of *priorities*  $(c_n)_{n \in \mathbb{N}}$  is *parity accepting* (or simply accepting) if  $\limsup_{n \rightarrow \infty} c_n \equiv 0 \pmod{2}$ , else it is *parity rejecting* (or rejecting).

## 2.1 Parity games

⌈ For  $I$  an *index*,  $(V, E)$  a graph with  $V$  a countable set of vertices and  $L : E \rightarrow I$  an edge labeling, we call  $G = (V, E, L)$  a *I-graph*, or a *parity graph*. We work with graphs in which every vertex has at least one successor. A graph (or tree) is said *finitely branching* if all its vertices have a finite number of exiting edges.

A graph is said *even* if all its infinite paths are *parity accepting*. For  $G = (V, E, L)$  a *parity graph* and  $V' \subseteq V$ , the graph  $G|V'$  is the subgraph restricted to the vertices in  $V'$ . Similarly, for  $E' \subseteq E$ , the graph  $G \setminus E'$  corresponds to  $(V, E \setminus E', L')$  with  $L'$  the restriction of  $L$  to  $E \setminus E'$ .

⌈ Let  $G = (V, E, L)$  a *parity graph*, and  $E' \subseteq E$ . The *attractor* of  $E'$  in  $G$  is the set  $\text{Attr}(E', G) := \{v \in V \mid \forall \text{ infinite path } \rho \text{ from } v \text{ in } G, \rho \text{ has an edge in } E'\}$ . Similarly, if  $V' \subseteq V$ , we define its *attractor* as the set  $\text{Attr}(V', G)$  of vertices from which all infinite paths eventually pass by  $V'$ . Note that  $V' \subseteq \text{Attr}(V', G)$ .

⌈ A *parity game* played by players Eve and Adam consists in a *parity graph*  $\mathcal{G} = (V, E, L)$  with a partition of  $V$  in two sets:  $V = V_E \sqcup V_A$ , controlled respectively by Eve and Adam. A *play* of  $\mathcal{G}$  starting in  $v \in V$  consists in an infinite sequence of edges  $\rho := (e_i)_{i \in \mathbb{N}}$  forming an infinite path starting in  $v$ . A *play*  $(e_i)_{i \in \mathbb{N}}$  is *winning* for Eve (or simply *winning*) if  $(L(e_i))_{i \in \mathbb{N}}$  is *parity accepting*, else it is said to be *losing* (for Eve, and *winning* for Adam).

⌈ A *strategy* for Eve consists of a function  $\sigma : E^* \rightarrow E$  such that, for all play  $\rho$ , for all  $n \in \mathbb{N}$ , if  $\rho|_n$  ends in a vertex  $v \in V_E$ ,  $\sigma(\rho|_n)$  is an edge from  $v$ . A *play*  $\rho$  is said to be *consistent with* the strategy  $\sigma$  if for all  $n$ ,  $\rho|_n$  ending in a vertex of  $V_E$  implies that  $\rho|_{n+1} = \rho|_n \sigma(\rho|_n)$ . We say that a Eve *strategy*  $\sigma$  is *winning* from vertex  $v \in V$  if all plays *consistent with*  $\sigma$  starting in  $v$  are *winning*. We similarly define *strategies* for Adam, winning when all plays *consistent with* them are *winning* for Adam.

⌈ Parity games enjoy positional *determinacy*: one of the players always wins with a strategy that only depends on the current position [9].

A *strategy* for Eve in a game  $\mathcal{G} = (V_E \sqcup V_A, E, L)$  induces an Adam-only *game*  $\mathcal{G}'$  played on the unfolding of  $\mathcal{G}$ , from which are removed all the edges that Eve does not choose. This game can be seen as a *parity graph*, as the partition of the vertex set is now a trivial one, and it is *even* if and only if Eve's strategy is winning.

## 2.2 Attractor decomposition

An *attractor decomposition* of an *even parity graph*  $G$  is a recursive partitionning of  $G$ . The intuition is that it identifies subgames of  $G$  in which the top priorities  $h$  (even) and  $h-1$  (odd) do not occur and orders them so that a path must always eventually either stay within a subgame (and never see  $h-1$  again), advance in the order (potentially seeing  $h-1$  finitely many times in between by advancing through the attractor of a subgame), or see the higher even priority  $h$ . Each subgame is then decomposed recursively, with respect to the priority  $h-2$ . As the number of subgames is countable, such a decomposition witnesses that the *parity graph* is indeed *even*. An *attractor decomposition* has a tree-like structure, induced by the order on the subgames (which corresponds to the order of sibling nodes), and their sub-decompositions.

Given a **parity graph**  $G = (V, E, L)$  with maximal priority at most some even  $h$ , and  $\kappa$  an ordinal, a  $(h\text{-level}, \kappa\text{-width})\text{-attractor decomposition}$  of  $G$ , if it exists, is recursively defined to be  $D = (H, A_0, \{(S_i, A_i, D_i)\}_{0 < i < \ell})$  where:

- $\ell \leq \kappa$ ,
- $H \subseteq E$  is the set of edges in  $G$  of priority  $h$ ,
- $A_0 = \text{Attr}(H, G)$ ,
- For every  $0 < i < \ell$ , let  $V_i = V \setminus \bigcup_{j < i} A_j$  and  $G_i = (G \setminus H) \upharpoonright V_i$ . Then:
  - $S_i \subseteq V_i$  is non-empty, such that  $(G \setminus H) \upharpoonright S_i$  only contains edges with priorities up to  $h-2$ , has no terminal vertices and is closed under successors in  $G_i$ ,
  - $A_i$  is  $\text{Attr}(S_i, G_i)$ ,
  - $D_i$  is a  $((h-2)\text{-level}, \kappa\text{-width})\text{-attractor decomposition}$  of  $(G \setminus H) \upharpoonright S_i$ ,
- $V = \bigcup_{i < \ell} A_i$ ,
- A  $(0\text{-height}, \kappa\text{-width})\text{-}$  and a  $(h\text{-level}, 0\text{-width})\text{-attractor decomposition}$  is just  $(H, V)$ : the entire graph is in the attractor of the edges of highest priority.

► **Lemma 4.** *Given a **parity graph**  $G$  that admits an **attractor decomposition**  $(H, A_0, \{(S_i, A_i, D_i)\}_{0 < i < \kappa})$ , the set  $A_j$  is unreachable from  $A_i$  in  $G \upharpoonright \bigcup_{0 < \ell < \kappa} A_\ell$  for all  $i$  and  $j$  such that  $0 < i < j < \kappa$ .*

**Proof.** TOPROVE 0 ◀

► **Lemma 5.** *A **parity graph** is **even** if and only if it admits an **attractor decomposition**.*

**Proof.** TOPROVE 1 ◀

## 2.3 $\Sigma$ -trees and automata

A  **$\Sigma$ -tree** (or just tree) is a function  $t : \{0, 1\}^* \rightarrow \Sigma$ . The set of all  $\Sigma$ -trees is denoted  $Tr_\Sigma$ . A tree is **regular** if it is finitely representable, that is, if it is the unfolding of a rooted graph. We denote  $\text{Reg}_\Sigma$  the set of **regular** trees of  $Tr_\Sigma$ .

An infinite word  $b \in \{0, 1\}^\omega$  is called a **branch**. Given a tree  $t \in Tr_\Sigma$ , a **path**  $p$  (along a branch  $b$ ) is a sequence  $(p_i)_{i \in \mathbb{N}} := (t(b \upharpoonright_i))_{i \in \mathbb{N}}$ .

A **nondeterministic  $I$ -parity tree automaton** (also called  $I$ -automaton, or **automaton** of index  $I$ ) is a tuple  $A = (\Sigma, Q_A, q_{i,A}, \Delta_A, \Omega_A)$ , where  $\Sigma$  is an alphabet,  $Q_A$  a finite set of **states**,  $q_{i,A} \in Q_A$  an initial **state**,  $\Delta_A \subseteq Q_A \times \Sigma \times Q_A \times Q_A$  a transition relation; and  $\Omega_A : \Delta_A \rightarrow I^2$  a **priority** mapping over the edges. A **transition**  $(q, a, q_0, q_1) \in \Delta_A$ , is said to be from the **state**  $q$  and over the letter  $a$ . By default, all **automata** in consideration are complete, that is, for each state  $q \in Q_A$  and letter  $a \in \Sigma$ , there is at least one transition from  $q$  over  $a$  in  $\Delta_A$ . When an automaton  $A$  is known from the context, we skip the subscript and write just  $Q, \Delta$ , etc.

For  $q, q' \in Q$ , a **path** from  $q$  to  $q'$  is a finite **transition** sequence  $(q_j, a_j, q_{j,0}, q_{j,1})_{j < N} \in \Delta^N$  such that  $q = q_0$ , and  $\forall j < N, q_{j+1} \in \{q_{j,0}, q_{j,1}\}$  with  $q_{j+1} = q'$ .

A tree is said to be **accepted** by an **automaton**  $A$  if Eve wins a game defined by the product of this tree and the automaton, in which Eve chooses the transitions in  $A$  and Adam chooses the direction in  $t$ . More formally, given a tree  $t \in Tr_\Sigma$ , and an  $I$ -**automaton**  $A$ , the **acceptance game** of  $A$  on  $t$ , also denoted  $\mathcal{G}(A, t)$ , is the **parity game** obtained by taking the product of  $A$  and  $t$ . Its arena consists in  $\{0, 1\}^* \times (Q_A \cup \Delta_A)$ , where all the positions of the shape  $\{0, 1\}^* \times Q_A$  are controlled by Eve, and the others by Adam.



- When in a position  $(w, q) \in \{0, 1\}^* \times Q_A$ , Eve chooses a transition  $e \in \Delta_A$  of the shape  $(q, t(w), q_0, q_1)$ , and the play proceeds to the state  $(w, e)$ . All these transitions have for label the minimal priority in  $I$ .
- Let  $q \in Q_A$  and  $e = (q, a, q_0, q_1) \in \Delta_A$ . In a position  $(w, e)$ , Adam chooses either 0 or 1, and the games then moves towards either  $(w \cdot 0, q_0)$  or  $(w \cdot 1, q_1)$ . For  $\Omega_A(e) = (i_0, i_1)$ , these transitions have priorities  $i_0$  and  $i_1$ , respectively.
- We say that  $t$  is *accepted* by  $A$  if Eve wins  $\mathcal{G}(A, t)$ . The set of *trees accepted* by  $A$  is called the *language* of  $A$  and is denoted  $\mathcal{L}(A)$ . We say that  $A$  *recognizes*  $\mathcal{L}(A)$ .
- If we fix a *strategy* for Eve, the *acceptance game* becomes an Adam-only game, called a *run* of  $A$  on  $t$ . We observe that it is played on a *parity graph* in the shape of a binary tree. We thus observe that a *run* can be considered as a tree in  $Tr_{\Delta_A}$ . This *run* is won by Adam if and only if there exists a *parity rejecting branch*. In this case, it called a *rejecting run*, else it is an *accepting run*.
- If  $A$  is an *I-automaton*, such a *run* over  $t$  induces an  $I$ -labelling of  $t$ , which, for convenience, we consider to be on edges.
- A set of trees  $L \subseteq Tr_{\Sigma}$  is an  *$\omega$ -regular tree language* if it is of the form  $\mathcal{L}(A)$  for some *automaton*  $A$ . It is said to be *I-feasible* if furthermore  $A$  is of *index*  $I$ .

## 2.4 Guidable automata

The notion of a *guidable automata* was first introduced in [4]. Intuitively, they are *automata* that fairly simulates all language equivalent *automata*. *Guidable automata* are fully expressive [4, Theorem 1] and are more manageable than general nondeterministic automata.

- Fix two *automata*  $A$  and  $B$  over the same *alphabet*  $\Sigma$ . A *guiding function* from  $B$  to  $A$  is a function  $g : Q_A \times \Delta_B \rightarrow \Delta_A$  such that  $g(p, (q, a, q_0, q_1)) = (p, a, p_0, p_1)$  for some  $p_0, p_1 \in Q_A$  (i.e. the function  $g$  is compatible with the *state*  $p$  and the letter  $a$ ).
- If  $\rho \in Tr_{\Delta_B}$  is a *run* of  $B$  over a *tree*  $t \in Tr_{\Sigma}$  then we define the *run*  $g(\rho) \in Tr_{\Delta_A}$  as follows. We define inductively  $q : \{0, 1\}^* \rightarrow Q_A$  in the following fashion:  $q(\varepsilon) = q_{i,A}$ , and supposing  $q(u)$  to be defined, for  $g(q(w), \rho(w)) = (q(w), t(w), q_0, q_1)$ , we let  $q(u \cdot 0), q(u \cdot 1)$  to be respectively  $q_0, q_1$ . We can then define the run  $g(\rho) \in Tr_{\Delta_A}$  as

$$g(\rho) : u \mapsto g(q(u), \rho(u)).$$

Notice that directly by the definition, the *tree*  $g(\rho)$  is a *run* of  $A$  over  $t$ .

- We say that a *guiding function*  $g : Q_A \times \Delta_B \rightarrow \Delta_A$  *preserves acceptance* if whenever  $\rho$  is an *accepting run* of  $B$  then  $g(\rho)$  is an *accepting run* of  $A$ . We say that an *automaton*  $B$  *guides* an *automaton*  $A$  if there exists a *guiding function*  $g : Q_A \times \Delta_B \rightarrow \Delta_A$  which *preserves acceptance*. In particular, it implies that  $\mathcal{L}(B) \subseteq \mathcal{L}(A)$ .
- An *automaton*  $A$  is *guidable* if it can be *guided* by any *automaton*  $B$  such that  $L(B) = L(A)$  (in fact one can equivalently require that  $L(B) \subseteq L(A)$ , see [17, Remark 4.5]). We will use the following fundamental theorem, stating that *guidable automata* are as expressive as *non-deterministic ones*.

► **Theorem 6** ([4, Theorem 1]). *For every regular tree language  $L$ , there exists a guidable automaton recognizing  $L$ . Moreover, such an automaton can be effectively constructed from any non-deterministic automaton for  $L$ .*

## 2.5 Ordered trees

- We define inductively *ordered trees* of finite *depth*. They are either the *leaf* tree  $\langle \rangle$  of *depth*  $(\langle \rangle) = 1$ , or a tree  $T = \langle (T_k)_{k \in K} \rangle$  where  $\forall k, T_k$  is an ordered tree of finite *depth*, and  $K$  is a

well-ordered countable set. The *children*, *siblings* and *subtree* relation  $\sqsubseteq$  are defined in the usual way. We denote  $\prec$  the order relation between the *siblings* of a tree  $\langle (T_k)_{k \in K} \rangle$ . That is, for  $k, k' \in K$ , we have  $T_k \prec T_{k'}$  when  $k < k'$  for  $<$  the well-order of  $K$ . By abuse of notation, we say that  $T_1 \prec T_2$  if  $T_1 \sqsubseteq T'_1, T_2 \sqsubseteq T'_2$  and  $T'_1 \prec T'_2$ .

From their definitions, it is clear that *attractor decompositions* are tree-shaped. To make this explicit, the *tree-shape* of an *attractor decomposition*  $D = (H, A_0, \{(S_i, A_i, D_i)\}_{0 < i < \kappa})$  is defined inductively as  $\langle \rangle$  if  $\kappa = 0$ , else, defining  $(T_i)_{0 < i < \kappa}$  the *tree-shapes* of the  $(D_i)_{0 < i < \kappa}$ ,  $D$  has *tree-shape*  $\langle (T_i)_{0 < i < \kappa} \rangle$ . Observe that the *width* of an *attractor decomposition* corresponds to an upper-bound on the branching degree of its *tree-shape*.

We extend the notion of *tree-shape* to *runs*: a run has *tree-shape*  $t$  if it has an *attractor decomposition* of *tree-shape*  $t$ .

We say that an *ordered tree*  $T = \langle (T_i)_{i \in I} \rangle$  is *isomorphically embedded* in a tree  $T' = \langle (T'_j)_{j \in J} \rangle$  if either  $I$  is empty, or of there exists  $\phi : I \rightarrow J$ , strictly increasing, such that  $\forall i \in I, T_i$  is *isomorphically embedded* in  $T'_{\phi(i)}$ . Intuitively, this implies the existence of a map from the *subtrees* of  $T$  to the *subtrees* of  $T'$ , where the root of  $T$  is mapped onto the root of  $T'$ , and the children of every node must be mapped injectively and in an order-preserving way onto the children of its image.

Let  $\mathcal{T}$  be a set of *ordered trees*. We say that a tree  $U$  is *universal for*  $\mathcal{T}$  if all the trees of  $\mathcal{T}$  can be *isomorphically embedded* in  $U$ .

### 3 Game characterisation of the parity index

In this section we define *priority transduction games*, based on the register games from Lehtinen's algorithm in [15], augmented with some counters. We characterise the *J-feasibility* of a *language*  $\mathcal{L}(A)$ , where  $A$  is a *guidable automaton*, by the existence of a uniform bound  $n \in \mathbb{N}$  such that a *tree* is in  $\mathcal{L}(A)$  if and only Eve wins the *J-priority transduction game* on  $\mathcal{G}(A, t)$ , with counters bounded by  $n$ .

The idea of these *priority transduction games* is that in addition to playing the *acceptance game* of an  $I$ -automaton  $A$  over a tree  $t$ , which has priorities in  $I$ , Eve must map these priorities on-the-fly into the *index*  $J$ . In the original games from [15], she does so by choosing at each turn a register among roughly  $\frac{|J|}{2}$  registers. Each register stores the highest priority seen since the last time it was chosen. Then, the output is a priority in  $J$  which depends on both the register chosen and the parity of the value stored in it. Here, the mechanism is similar, except that we additionally have counters that allow Eve to delay outputting odd priorities a bounded number of times.

Intuitively, the registers, which store the highest priority seen since the last time they were chosen, determine the magnitude of the output, while their content's parity decides the output's parity. This allows Eve to strategically pick registers so that odd priorities get eclipsed by higher even priorities occurring soon after. However, a large odd priority occurring infinitely often will force Eve to produce odd outputs infinitely often. The counters give Eve some error margin, whereby she can pick a register containing an odd value *without* outputting an odd priority, up to  $n$  times in a row.

Formally, for  $J$  a *priority index* (of minimal value assumed to be 1 or 2 for convenience),  $n \in \mathbb{N}$ , the *J, n-priority transduction game* is a game played by Eve and Adam, over an *I-parity graph*  $G = (V, E, L)$  for  $I$  an *index*. It has two parameters,  $J$  the output *index* and  $n$  the bound of its *counters*, and is denoted  $\mathcal{T}_J^n(G)$ . A configuration of the game corresponds to a position  $p \in V$ , a value in  $I$  for each *register*  $r_j$  for even  $2j \in J$  (if  $1 \in J$ , there is an additional *register*  $r_0$ ), and a value between 0 and  $n$  for each *counters*  $c_{i,j}$  with  $i$  odd  $\in I, j$



such that  $r_j$  is a **register**.

Starting from some initial vertex  $p_0 \in V$  with **counters** set to 0 and **registers** set to the maximal even priority in  $I$ , the game proceeds as follows at step  $l$  :

- Adam chooses an exiting edge  $e = (p, p') \in E$  ; the position becomes  $p'$ .
- Eve chooses a **register**  $r_j$ .
- The game produces the output  $w_l$  :
  - if  $j = 0$ ,  $w_l = 1$  (recall that  $r_0$  is a **register** iff  $1 \in J$ ). Else,
  - if  $r_j$  is even,  $w_l = 2j$ .
  - Else, if  $c_{r_j, j} = n$ , it is said to reach  $n + 1$  before being reset:  $w_l = 2j + 1$  and  $c_{r_j, j} := 0$ . If  $2j + 1 \notin J$ , Eve loses instantly.
  - else,  $w_l = 2j$  and  $c_{r_j, j} := c_{r_j, j} + 1$
- If  $L(e)$  is even, let  $i := L(e)$  be the label of the current edge, else Eve chooses an odd  $i$  such that  $L(e) \leq i$  (choice  $\sharp$ ). Then the following updates occur :
  - Smaller **counters** are reset :  $\forall i' < i, c_{i', j} := 0$  and  $\forall j' < j, c_{r_j, j'} := 0$ ,
  - **Registers** get updated :  $r_j := i$  and  $\forall j' > j, r_{j'} := \max(i, r_{j'})$ ,

Eve wins if the infinite sequence of outputs  $(w_l)_{l \in \mathbb{N}}$  is **parity accepting**, else Adam wins.

In order to explain two aspects of this game that were not covered in the initial intuition: the minimal **register**  $r_0$  allows Eve to wait, for a finite but unbounded time, for better priorities to override the register contents. It thus corresponds to outputting a minimal odd priority. The choice  $\sharp$  allows her to break a sequence dominated by many identical odd priorities  $i$  in a sequence with some greater odd  $i'$  in between, resetting the **counters** albeit at the cost of witnessing a greater odd priority.<sup>1</sup>

Let  $\mathcal{G}$  be a **parity game** of index  $I$ ,  $n \in \mathbb{N}$ ,  $J$  an **index**. We define the game  $\mathcal{T}_J^n(\mathcal{G})$  as the game  $\mathcal{T}_J^n$  where, instead of following a path of **parity graph** chosen by Adam, it follows an ongoing play of  $\mathcal{G}$  where the player owning the current position  $q$  chooses its move in  $\mathcal{G}$  at each step, before Eve chooses her **register**. It corresponds to the composition of  $\mathcal{T}_J^n$  with the game  $\mathcal{G}$ . If we fix a strategy  $\sigma$  for Eve in  $\mathcal{G}$ , we observe that  $\mathcal{T}_J^n(\mathcal{G})$  corresponds exactly to the **priority transduction game**  $\mathcal{T}_J^n$  over the Adam-only game  $\mathcal{G}_\sigma$  induced by  $\sigma$  in  $\mathcal{G}$ , and that Eve wins  $\mathcal{T}_J^n(\mathcal{G})$  if and only if she wins  $\mathcal{T}_J^n(\mathcal{G}_\sigma)$ .

Note that  $\mathcal{T}_J^n(\mathcal{G})$  is a parity game, and therefore **determined**.

We show that this transduction game characterises the **index** of a **regular tree language**.

► **Theorem 1.** *Given a **guidable automaton**  $A$ ,  $J$  an **index**, the following are equivalent:*

- $\mathcal{L}(A)$  is  $J$ -feasible.
- There exists  $n \in \mathbb{N}$  such that for all  $\Sigma$ -tree  $t$ ,  $t \in \mathcal{L}(A)$  if and only if Eve wins  $\mathcal{T}_J^n(\mathcal{G}(A, t))$ .

For the upward implication, it suffices to observe that the transduction game is captured by a finite state  $J$ -automaton describing the register contents and counter values (bounded by  $n$ ), with nondeterministic choices corresponding to Eve's choices, and a  $J$ -parity condition corresponding to the outputs. Then, the  $J$ -automaton equivalent to  $A$  is the composition of  $A$  with this  $J$ -automaton. The details, which are as one would expect, are in the appendix.

► **Lemma 7.** *Let  $J$  be a **priority index** and let  $A$  be a **guidable automaton** such that there exists  $n \in \mathbb{N}$  such that for all  $\Sigma$ -tree  $t$ ,  $t \in \mathcal{L}(A)$  if and only if Eve wins  $\mathcal{T}_J^n(\mathcal{G}(A, t))$ . Then there exists an **automaton** of **index**  $J$  such that  $\mathcal{L}(B) = \mathcal{L}(A)$ .*

<sup>1</sup> This is a technical adjustment that is convenient in the proof of Proposition 15

The rest of the section focuses on the downward implication of Theorem 1. We first show that Eve can only win  $\mathcal{T}_J^n(G)$  for  $G$  an **even parity graph**, which implies that Eve loses  $\mathcal{T}_J^n(\mathcal{G}(A, t))$  for any  $t \notin \mathcal{L}(A)$ .

► **Lemma 8.** *Let  $G$  a **parity graph**. If  $G$  is not **even**, then for all  $J, n$ , Adam wins  $\mathcal{T}_J^n(G)$*

**Proof.** TOPROVE 2 ◀

Then, it remains to show that if the language of guidable  $A$  is  $J$ -feasible, then for some  $n \in \mathbb{N}$  Eve wins  $\mathcal{T}_J^n(\mathcal{G}(A, t))$  for all  $t \in \mathcal{L}(A)$ . To do so, we first analyse the relation between guided and guiding runs, and show that the preservation of global acceptance implies a more local version that restricts differences in the parity of the dominant priority over long segments of both runs. We will then use this to show that Eve can win the transduction game by using a run of  $A$  guided by an accepting run of an equivalent  $J$ -automaton and choosing registers corresponding to the priorities of the guiding run.

The following lemma, obtained by a simple pumping argument (see Appendix) expresses that between all pumpable pairs of states, that is, pairs of states that are not distinguished by either run, if the guiding run is dominated by an even priority, then so is the guided one.

► **Lemma 9.** *Let  $A, B$  be **automata**, let  $t \in \text{Tr}_\Sigma$ , let  $\rho_A, \rho_B$  be **accepting runs** over  $t$  of  $A$  and  $B$ , respectively, where  $\rho_A$  is **guided by**  $\rho_B$ . We consider these **runs** as trees in  $\text{Tr}_{\Delta_A}, \text{Tr}_{\Delta_B}$  respectively. Given  $u, v \in \{0, 1\}^*, \{0, 1\}^+$  such that  $\rho_A(u) = \rho_A(u \cdot v)$ , and such that  $\rho_B(u) = \rho_B(u \cdot v)$ , if the greatest priority encountered between positions  $u$  and  $u \cdot v$  in  $\rho_B$  is even, so is the greatest priority encountered in this segment in  $\rho_A$ .*

◀ We now capture this relation with the notion of a labelling being  $n$ -bounded by the other. Let  $L_I : E \rightarrow I$  and  $L_J : E \rightarrow J$  be two labellings of a graph  $G = (V, E)$  (or tree) and let  $n \in \mathbb{N}$ . We say that  $L_I$  is  **$n$ -bound** by  $L_J$  if there is no finite path  $\pi$  in  $G$ , segmented into consecutive paths  $\pi_0, \pi_1, \dots, \pi_n$  such that for some odd  $i$  and some even  $j$ , the maximal priority on the  $L_I$ - and  $L_J$ -labels of each  $\pi_m, m \in [0, n]$  are  $i$  and  $j$ , respectively.

From Lemma 9 we obtain that the labelling induced by a guided run is  $n$ -bound by the one induced by its guide, with  $n$  the product of the sizes of the two automata:

► **Lemma 10.** *Let  $A$  a guidable automaton. If  $\mathcal{L}(A)$  is  $J$ -feasible witnessed by an **automaton**  $B$ , for  $n := |A||B| + 1$ , for all  $\Sigma$ -tree  $t \in \mathcal{L}(A)$ , for  $\rho_A$  the **run** of  $A$  on  $t$  **guided by** an **accepting run**  $\rho_B$  of  $B$  over  $t$ , the labelling  $L_A$  induced by  $\rho_A$  is  **$n$ -bound** by the labelling  $L_B$  induced by  $\rho_B$ .*

We use  **$n$ -boundedness** to show that Eve can use a run  $\rho_B$  of an equivalent  $J$ -automaton to choose her **registers** to win in  $\mathcal{T}_J^{n+1}(\rho_A)$  for  $\rho_A$  **accepting run** of  $A$  guided by  $\rho_B$ .

► **Lemma 11.** *Let  $I, J$  be indices,  $n \in \mathbb{N}^+$ , and  $\rho_I : E \rightarrow I$  and  $\rho_J : E \rightarrow J$  two **even**  $I$ - and  $J$ -labelling of the same graph  $(E, V)$ . If  $\rho_I$  is  **$n$ -bound** by  $\rho_J$ , Eve wins  $\mathcal{T}_J^{n+1}(\rho_I)$ .*

**Proof.** TOPROVE 3 ◀

Lemma 8 implies that Eve loses  $\mathcal{T}_J^{n+1}(\mathcal{G}(A, t))$  for  $t \notin \mathcal{L}(A)$ . If  $\mathcal{L}(A)$  is  $J$ -feasible as witnessed by a  $J$ -automaton  $B$ , then for all  $t \in \mathcal{L}(A)$ , from Lemma 10, Eve has a run  $\rho_A$  that is  $n$ -bounded by an accepting run of  $B$ , which, from Lemma 11, implies that Eve wins  $\mathcal{T}_J^{n+1}(\mathcal{G}(A, t))$ , concluding the proof of Theorem 1.

► **Remark 12.** To obtain Colcombet and Löding's result from ours, it suffices to encode the transduction game as a distance-parity automaton that on an input tree  $t$  computes a bound  $n$  on the counters such that Eve wins  $\mathcal{T}_J^n(\mathcal{G}(A, t))$ . Then, like in [4, Lemma 3], there is a distance-parity automaton that is uniformly universal if and only if  $A$  is  $J$ -feasible.

## 4 Characterisation via attractor decompositions

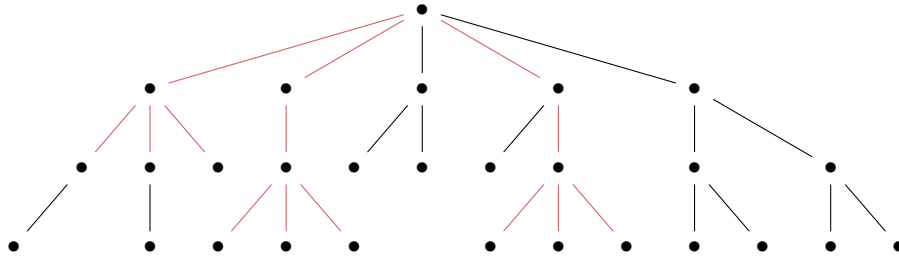
### 4.1 Strahler number

The Strahler number of a tree, given by the height of the largest full binary tree that appears as a minor, measures the arborescence of a tree. We generalise this notion.

Let  $n \in \mathbb{N}$ . The  *$n$ -Strahler number* of  $T$  a tree of finite depth, denoted  $\mathcal{S}_n(T)$ , is defined by recurrence:

- if  $T = \langle \rangle$ ,  $\mathcal{S}_n(T) = 1$ .
- Else,  $T = \langle (T_k)_{k \in K} \rangle$ . We consider  $m := \max\{\mathcal{S}_n(T_k) \mid k \in K\}$ . If there are at least  $n + 1$   $T_k$ 's of  *$n$ -Strahler number*  $m$ ,  $\mathcal{S}_n(T) = m + 1$ . Else,  $\mathcal{S}_n(T) = m$ .

The  *$n$ -Strahler number* of  $T$  is at most its *depth*. Having a  *$n$ -Strahler number*  $k$  is equivalent to having a complete  $n$ -ary tree of depth  $k$  as a minor, for the operations of child deletion and replacing a node by one of its *children*. Figure 1 gives an example.



■ **Figure 1** An *ordered tree* of depth 4, of *3-Strahler number* 3, as exemplified by the red edges.

We say that a *parity game*  $G$  has  *$n$ -Strahler number*  $j$  if there exists a strategy  $\sigma_G$ , winning for Eve, such that the resulting *parity graph* admits an *attractor decomposition* of tree-shape with  *$n$ -Strahler number*  $j$ .

In the next two sections we prove each direction of the following theorem, using Theorem 1 for the upward implication.

► **Theorem 2.** *Given a *guidable nondeterministic parity tree automaton*  $A$ , the following are equivalent:*

- $L(A)$  is  $[1, 2j]$ -feasible.
- There is an  $n \in \mathbb{N}$  such that for all  $t \in L(A)$  there exists a *run* of  $A$  on  $t$  with an *attractor decomposition* of  *$n$ -Strahler number* at most  $j$ .

► **Remark 13.** Note that this theorem, based on a range  $[1, 2j]$ , is less precise than Theorem 1, which handles all ranges  $J$ . This is because the parity of the minimal and maximal priorities are not reflected in the tree-shape of the attractor decomposition. For example, if there is a uniform bound on the lengths of paths in attractors, then there is no need for a minimal odd priority. The maximal even priority on the other hand is not required if there are no edges that go from  $A_i$  to  $A_j$  with  $j > i$ . While the extremal parities are hard to characterise from the attractor decompositions, they are neatly captured by the transduction game.

#### 4.1.1 From feasibility to attractor decompositions

Let  $A$  be a *guidable automaton* of index  $I$ . If  $\mathcal{L}(A)$  is  $[1, 2j]$  feasible by some automaton  $B$ , by Lemma 10, there exists  $n \in \mathbb{N}$  and a *run*  $\rho_B$  *guiding*  $A$  such that the resulting *run*  $\rho_A$  is  *$n$ -bound* by  $\rho_B$ . From this, we exhibit an *attractor decomposition* of  $G$  of  *$n$ -Strahler*

number  $j$ . More precisely these runs over  $\mathcal{G}(A, t)$  and  $\mathcal{G}(B, t)$  are considered as an  $I$ -tree and a  $[1, 2j]$ -tree, respectively. We will use these two trees in order to exhibit an attractor decomposition of  $\mathcal{G}(A, t)$  of  $n$ -Strahler number  $j$ .

► **Proposition 14.** *Given a tree-shaped graph  $G = (V, E)$ , finitely-branching and without terminal vertices, two indices  $I = [0, 2i]$  and  $J = [1, 2j]$  and labellings  $\rho_I : E \rightarrow I$  and  $\sigma_J : E \rightarrow J$  such that  $(G, \rho_I)$  and  $(G, \sigma_J)$  are even parity graphs, if  $\rho_I$  is  $n$ -bound by  $\sigma_J$ , then  $(G, \rho_I)$  admits an attractor decomposition of  $n$ -Strahler number at most  $j$ .*

**Proof.** TOPROVE 4 ◀

#### 4.1.2 From attractor decomposition to feasibility

For the backward direction of Theorem 2, we show that if Eve has a winning strategy in a game with a corresponding attractor decomposition of  $n$ -Strahler number  $h$ , then she can win the corresponding priority transduction game with  $h$  registers and counters going up to  $n$ . Then, using Theorem 1, we obtain the required implication.

► **Proposition 15.** *Given a game  $G$  and  $n \in \mathbb{N} \setminus \{0\}$ , if  $G$  has  $n$ -Strahler number  $h$ , then Eve wins  $\mathcal{T}_{[1, 2h]}^{n+1}(G)$ .*

**Proof.** TOPROVE 5 ◀

► **Remark 16.** Eve also has a winning strategy in  $\mathcal{T}_{[1, 2h]}^n(G)$ , but the proof is more elaborate, as we need to do a case analysis of the behaviour of the last counter incrementation.

If Eve has such an attractor decomposition over all the games  $\mathcal{G}(A, t)$  for  $t \in \mathcal{L}(A)$ , the corresponding  $n$  is a uniform bound such that Eve wins all the  $\mathcal{T}_{[1, 2j]}^n$ . From this, we conclude the proof of Theorem 2 using the upwards direction of Theorem 1.

### 5 Characterisation via universal trees

We now show that the previous characterisations of  $J$ -feasibility of a guidable automaton  $A$  can be reformulated in terms of the existence of a universal tree for  $A$ . Note that in this section we use both trees, which are binary, infinite and inputs to automata, and ordered trees, which are of potentially infinite branching but finite height and describe attractor decompositions.

◻ We say that an ordered tree is *universal for an automaton  $A$*  if it is universal for some set of ordered trees  $T$  such that for all regular trees  $t \in L(A)$ , Eve has a strategy in  $\mathcal{G}(A, t)$  with an attractor decomposition of tree-shape in  $T$ .

► **Theorem 3.** *Given a guidable nondeterministic parity tree automaton  $A$ , the following are equivalent:*

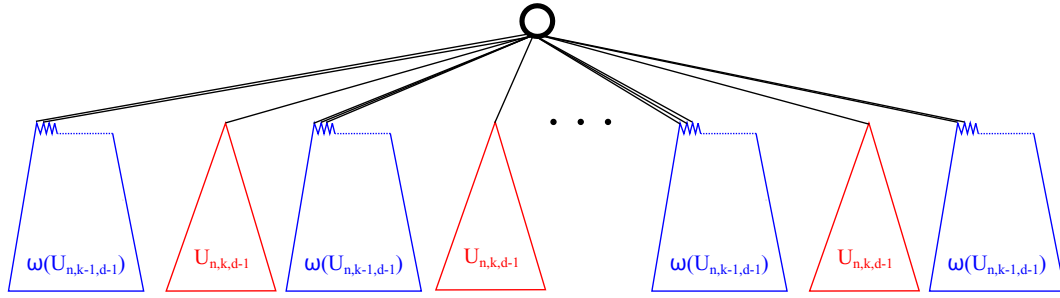
- $L(A)$  is  $[1, 2j]$ -feasible.
- There exists an  $n \in \mathbb{N}$  and a tree  $\mathcal{U}$  of  $n$ -Strahler number at most  $j$  that is universal for  $A$ .

To prove this theorem, we show that that for fixed  $n, j, d \in \mathbb{N}^+$ , there is an infinite ordered tree  $\mathcal{U}$  of  $n$ -Strahler number  $j$  and depth  $d$  that is universal for the set of finite ordered trees of  $n$ -Strahler number at most  $j$  and depth at most  $d$ . Over regular trees, because of the positionality of parity games, Eve's strategies can be chosen to be regular, which implies that their attractor decompositions can be finite, making  $\mathcal{U}$  universal for guidable automata

recognising a  $[1, 2j]$ -feasible language. For the other direction, we recall that if two tree automata are equivalent over **regular trees**, they are equivalent over all **trees** [23].

Let  $n, k, d \in \mathbb{N}^+$ . We define recursively the **universal tree**  $\mathcal{U}_{n,k,d}$  of  $n$ -Strahler number  $k$  and **depth**  $d$  as follows, where  $\omega(T)$  denotes the repetition of  $\omega$  times the **ordered tree**  $T$ :

- $\mathcal{U}_{n,1,1} := \langle \rangle$
- When  $d < k$ ,  $\mathcal{U}_{n,k,d}$  is undefined.
- Else,  $d \geq k$ , and by denoting  $U := \mathcal{U}_{n,k-1,d-1}$ , we have  
 $\mathcal{U}_{n,k,d} := \langle \omega(U), \mathcal{U}_{n,k,d-1}, \omega(U), \dots, \mathcal{U}_{n,k,d-1}, \omega(U) \rangle$ , with  $n$  repetitions of  $\mathcal{U}_{n,k,d-1}$  (or, if it is not defined, no such repetition). Similarly, if  $U = \mathcal{U}_{n,k-1,d-1}$  is undefined due to  $k$  being equal to 0, these children are omitted.



■ **Figure 2** The recursion step in the construction of  $\mathcal{U}_{\alpha,n,k,d}$ .

An example of such a construction can be found in figure 2. Observe that  $\mathcal{U}_{n,k,d}$  has **width** greater than  $\omega$  as soon as  $2 \leq k \leq d$ . Furthermore,  $\forall n, k, d$ ,  $\mathcal{U}_{\alpha,n,k,d}$  has **depth**  $d$ , and we establish that it also has  $n$ -Strahler number exactly  $k$ :

► **Lemma 17.** For all  $n, k, d \in \mathbb{N} \setminus \{0\}$ , if  $k \leq d$ , then  $\mathcal{U}_{n,k,d}$  is defined, and  $\mathcal{S}_n(\mathcal{U}_{n,k,d}) = k$ .

**Proof.** TOPROVE 6 ◀

We can now prove its **universality**:

► **Lemma 18.** Let  $\mathcal{T}$  be a set of finite **ordered trees**, all of **depth** bounded by  $d$  and  $n$ -Strahler number at most  $k$ . Then  $\mathcal{U}_{n,\min(k,d),d}$  is **universal** for  $\mathcal{T}$ .

**Proof.** TOPROVE 7 ◀

► **Remark 19.** As established by Rabin [23, Theorem 20], a non-empty tree automaton accepts a regular tree, therefore, if two **automata** are equivalent in  $\text{Reg}_\Sigma$ , they are equivalent over all **trees**. This notably implies that, for  $A$  an automaton and  $J$  an index,  $\mathcal{L}(A)$  is  $J$ -feasible over  $\text{Reg}_\Sigma$  if and only if it is  $J$ -feasible over all **trees**.

► **Lemma 20.** Let  $A$  a **guidable I-automaton**, let  $t$  a **regular tree** in  $\mathcal{L}(A)$ . If  $\mathcal{L}(A)$  is  $[1, 2j^*]$ -feasible, then there exists a finite **attractor decomposition** of  $\mathcal{G}(A, t)$  of  $n$ -Strahler number at most  $j^*$ .

**Proof.** TOPROVE 8 ◀

We finally obtain the direct implication of Theorem 3 from this lemma and Lemma 18.

For the converse direction, by Proposition 15, Eve wins all the  $\mathcal{T}_{[1,2j]}^{n+1}(\mathcal{G}(A, t))$  for  $t \in \mathcal{L}(A) \cap \text{Reg}_\Sigma$ . Conversely, for  $t \in \mathcal{L}(A)^C \cap \text{Reg}_\Sigma$ , by Lemma 8, Eve loses. Therefore, by Lemma 7 (restricting ourselves to the [regular trees](#)), we can construct  $B$  a  $[1, 2j]$ -automaton recognizing  $\mathcal{L}(A) \cap \text{Reg}_\Sigma$  over the [regular trees](#). Therefore  $\mathcal{L}(A)$  is  $[1, 2j]$ -feasible over the [regular trees](#), and by Remark 19 is thus  $[1, 2j]$ -feasible.

## 6 Conclusion

We have given three closely related new characterisations of the  $J$ -feasibility of  $\omega$ -regular tree languages: one via the [transduction game](#), one via [attractor decompositions](#) and one via [universal trees](#). While we do not solve the decidability of the index problem, our work brings to light the deep relationships between the tools we are used to manipulate in the context of solving [parity games](#), such as [attractor decompositions](#), [universal trees](#) and Lehtinen's register game, and the  $J$ -feasibility of a language. In particular, the  $n$ -Strahler number turns out to have great explanatory power by relating the [transduction game](#), the structure of [attractor decompositions](#) and the [index](#) of a language.

The Büchi case, which is at the frontier of the state of the art, is particularly appealing because of its simplicity: the language of a [guidable automaton](#)  $A$  is Büchi [feasible](#) if and only if there is a finite bound  $n$  such that Eve can win in the acceptance games with strategies with [attractor decompositions](#) of width at most  $n$ , or, equivalently, if  $A$  admits a [finite universal tree](#). We hope that these insights will help unlock the next steps in tackling this long-standing open problem.

---

## References

- 1 Julian C. Bradfield. The modal  $\mu$ -calculus alternation hierarchy is strict. *Theor. Comput. Sci.*, 195(2):133–153, 1998. doi:10.1016/S0304-3975(97)00217-X.
- 2 Cristian S. Calude, Sanjay Jain, Bakhadyr Khoussainov, Wei Li, and Frank Stephan. Deciding parity games in quasipolynomial time. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 252–263. ACM, 2017. doi:10.1145/3055399.3055409.
- 3 Thomas Colcombet, Denis Kuperberg, Christof Löding, and Michael Vanden Boom. Deciding the weak definability of Büchi definable tree languages. In Simona Ronchi Della Rocca, editor, *Computer Science Logic 2013 (CSL 2013), CSL 2013, September 2-5, 2013, Torino, Italy*, volume 23 of *LIPIcs*, pages 215–230. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2013. doi:10.4230/LIPIcs.CSL.2013.215.
- 4 Thomas Colcombet and Christof Löding. The non-deterministic Mostowski hierarchy and distance-parity automata. In Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfssdóttir, and Igor Walukiewicz, editors, *Automata, Languages and Programming, 35th International Colloquium, ICALP 2008, Reykjavik, Iceland, July 7-11, 2008, Proceedings, Part II - Track B: Logic, Semantics, and Theory of Programming & Track C: Security and Cryptography Foundations*, volume 5126 of *Lecture Notes in Computer Science*, pages 398–409. Springer, 2008. URL: [https://doi.org/10.1007/978-3-540-70583-3\\_33](https://doi.org/10.1007/978-3-540-70583-3_33), doi:10.1007/978-3-540-70583-3\_33.
- 5 Wojciech Czerwinski, Laure Daviaud, Nathanaël Fijalkow, Marcin Jurdzinski, Ranko Lazic, and Pawel Parys. Universal trees grow inside separating automata: Quasi-polynomial lower bounds for parity games. In Timothy M. Chan, editor, *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, pages 2333–2349. SIAM, 2019. doi:10.1137/1.9781611975482.142.



- 6 Laure Daviaud, Marcin Jurdzinski, and Ranko Lazic. A pseudo-quasi-polynomial algorithm for mean-payoff parity games. In Anuj Dawar and Erich Grädel, editors, *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018, Oxford, UK, July 09-12, 2018*, LICS '18, pages 325–334. ACM, jul 2018. doi:10.1145/3209108.3209162.
- 7 Laure Daviaud, Marcin Jurdzinski, and Karoliina Lehtinen. Alternating weak automata from universal trees. In Wan J. Fokkink and Rob van Glabbeek, editors, *30th International Conference on Concurrency Theory, CONCUR 2019, August 27-30, 2019, Amsterdam, the Netherlands*, volume 140 of *LIPIcs*, pages 18:1–18:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.CONCUR.2019.18.
- 8 Laure Daviaud, Marcin Jurdzinski, and K. S. Thejaswini. The Strahler number of a parity game. In Artur Czumaj, Anuj Dawar, and Emanuela Merelli, editors, *47th International Colloquium on Automata, Languages, and Programming, ICALP 2020, July 8-11, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 168 of *LIPIcs*, pages 123:1–123:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.ICALP.2020.123.
- 9 E. Allen Emerson and Charanjit S. Jutla. Tree automata, mu-calculus and determinacy (extended abstract). In *32nd Annual Symposium on Foundations of Computer Science, San Juan, Puerto Rico, 1-4 October 1991*, pages 368–377. IEEE Computer Society, 1991. doi:10.1109/SFCS.1991.185392.
- 10 Alessandro Facchini, Filip Murlak, and Michal Skrzypczak. Rabin-Mostowski index problem: A step beyond deterministic automata. In *28th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2013, New Orleans, LA, USA, June 25-28, 2013*, pages 499–508. IEEE Computer Society, 2013. doi:10.1109/LICS.2013.56.
- 11 Marcin Jurdzinski and Ranko Lazic. Succinct progress measures for solving parity games. *CoRR*, abs/1702.05051:1–9, 2017. URL: <http://arxiv.org/abs/1702.05051>, arXiv:1702.05051, doi:10.48550/arXiv.1702.05051.
- 12 Marcin Jurdzinski, Rémi Morvan, and K. S. Thejaswini. Universal algorithms for parity games and nested fixpoints. In Jean-François Raskin, Krishnendu Chatterjee, Laurent Doyen, and Rupak Majumdar, editors, *Principles of Systems Design - Essays Dedicated to Thomas A. Henzinger on the Occasion of His 60th Birthday*, volume 13660 of *Lecture Notes in Computer Science*, pages 252–271. Springer, 2022. doi:10.1007/978-3-031-22337-2\_12.
- 13 Nils Klarlund. Progress measures for complementation of omega-automata with applications to temporal logic. In *32nd Annual Symposium on Foundations of Computer Science, San Juan, Puerto Rico, 1-4 October 1991*, pages 358–367. IEEE Computer Society, 1991. doi:10.1109/SFCS.1991.185391.
- 14 Orna Kupferman and Moshe Y. Vardi. Weak alternating automata and tree automata emptiness. In Jeffrey Scott Vitter, editor, *Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing, Dallas, Texas, USA, May 23-26, 1998*, STOC '98, pages 224–233. New York, NY, USA, 1998. ACM. doi:10.1145/276698.276748.
- 15 Karoliina Lehtinen. A modal  $\mu$  perspective on solving parity games in quasi-polynomial time. In Anuj Dawar and Erich Grädel, editors, *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018, Oxford, UK, July 09-12, 2018*, LICS '18, pages 639–648. New York, NY, USA, 2018. ACM. doi:10.1145/3209108.3209115.
- 16 Giacomo Lenzi. A hierarchy theorem for the  $\mu$ -calculus. In Friedhelm Meyer auf der Heide and Burkhard Monien, editors, *Automata, Languages and Programming, 23rd International Colloquium, ICALP96, Paderborn, Germany, 8-12 July 1996, Proceedings*, volume 1099 of *Lecture Notes in Computer Science*, pages 87–97. Springer, 1996. doi:10.1007/3-540-61440-0\_119.
- 17 Christof Löding. *Logic and automata over infinite trees*. PhD thesis, RWTH Aachen, Germany, 2009. URL: [https://www.lics.rwth-aachen.de/global/show\\_document.asp?id=aaaaaaaaabqdrw](https://www.lics.rwth-aachen.de/global/show_document.asp?id=aaaaaaaaabqdrw).
- 18 Damian Niwinski. On fixed-point clones (extended abstract). In Laurent Kott, editor, *Automata, Languages and Programming, 13th International Colloquium, ICALP86, Rennes, France, July 15-19, 1986, Proceedings*, volume 226 of *Lecture Notes in Computer Science*,

- pages 464–473. Springer, 1986. URL: [https://doi.org/10.1007/3-540-16761-7\\_96](https://doi.org/10.1007/3-540-16761-7_96), doi: 10.1007/3-540-16761-7\_96.
- 19 Damian Niwinski and Michal Skrzypczak. On guidable index of tree automata. In Filippo Bonchi and Simon J. Puglisi, editors, *46th International Symposium on Mathematical Foundations of Computer Science, MFCS 2021, August 23-27, 2021, Tallinn, Estonia*, volume 202 of *LIPIcs*, pages 81:1–81:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.MFCS.2021.81.
  - 20 Damian Niwinski and Igor Walukiewicz. Relating hierarchies of word and tree automata. In Michel Morvan, Christoph Meinel, and Daniel Krob, editors, *STACS 98, 15th Annual Symposium on Theoretical Aspects of Computer Science, Paris, France, February 25-27, 1998, Proceedings*, volume 1373 of *Lecture Notes in Computer Science*, pages 320–331. Springer, 1998. doi:10.1007/BFb0028571.
  - 21 Damian Niwinski and Igor Walukiewicz. Deciding nondeterministic hierarchy of deterministic tree automata. In Ruy J. G. B. de Queiroz and Patrick Cégielski, editors, *Proceedings of the 11th Workshop on Logic, Language, Information and Computation, WoLLIC 2004, Fontainebleau, France, July 19-22, 2004*, volume 123 of *Electronic Notes in Theoretical Computer Science*, pages 195–208. Elsevier, 2004. doi:10.1016/j.entcs.2004.05.015.
  - 22 Michael O. Rabin. Decidability of second-order theories and automata on infinite trees. *Bulletin of the American Mathematical Society*, 74:1025–1029, 1968. URL: <https://api.semanticscholar.org/CorpusID:6015948>.
  - 23 Michael O. Rabin. Automata on infinite objects and Church’s problem, 1972. URL: <https://api.semanticscholar.org/CorpusID:119846488>.
  - 24 Michal Skrzypczak and Igor Walukiewicz. Deciding the topological complexity of Büchi languages. In Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi, editors, *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, volume 55 of *LIPIcs*, pages 99:1–99:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPIcs.ICALP.2016.99.
  - 25 Igor Walukiewicz. Deciding low levels of tree-automata hierarchy. In Ruy J. G. B. de Queiroz, Luiz Carlos Pereira, and Edward Hermann Haeusler, editors, *9th Workshop on Logic, Language, Information and Computation, WoLLIC 2002, Rio de Janeiro, Brazil, July 30 - August 2, 2002*, volume 67 of *Electronic Notes in Theoretical Computer Science*, pages 61–75. Elsevier, 2002. doi:10.1016/S1571-0661(04)80541-3.
  - 26 Wiesław Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees. *Theor. Comput. Sci.*, 200(1-2):135–183, 1998. doi:10.1016/S0304-3975(98)00009-7.

## A Appendix

### A.1 Proofs from Section 3

► **Lemma 7.** Let  $J$  be a priority *index* and let  $A$  be a *guidable automaton* such that there exists  $n \in \mathbb{N}$  such that for all  $\Sigma$ -tree  $t$ ,  $t \in \mathcal{L}(A)$  if and only if Eve wins  $\mathcal{T}_J^n(\mathcal{G}(A, t))$ . Then there exists an *automaton of index*  $J$  such that  $\mathcal{L}(B) = \mathcal{L}(A)$ .

**Proof.** TOPROVE 9 ◀

► **Lemma 8.** Let  $G$  a *parity graph*. If  $G$  is not *even*, then for all  $J, n$ , Adam wins  $\mathcal{T}_J^n(G)$

**Proof.** TOPROVE 10 ◀

► **Lemma 9.** Let  $A, B$  be *automata*, let  $t \in \text{Tr}_\Sigma$ , let  $\rho_A, \rho_B$  be *accepting runs* over  $t$  of  $A$  and  $B$ , respectively, where  $\rho_A$  is *guided by*  $\rho_B$ . We consider these *runs* as trees in  $\text{Tr}_{\Delta_A}, \text{Tr}_{\Delta_B}$  respectively. Given  $u, v \in \{0, 1\}^*, \{0, 1\}^+$  such that  $\rho_A(u) = \rho_A(u \cdot v)$ , and such

that  $\rho_B(u) = \rho_B(u \cdot v)$ , if the greatest priority encountered between positions  $u$  and  $u \cdot v$  in  $\rho_B$  is even, so is the greatest priority encountered in this segment in  $\rho_A$ .

**Proof.** TOPROVE 11 ◀

► **Lemma 10.** Let  $A$  a guidable automaton. If  $\mathcal{L}(A)$  is  $J$ -feasible witnessed by an automaton  $B$ , for  $n := |A||B| + 1$ , for all  $\Sigma$ -tree  $t \in \mathcal{L}(A)$ , for  $\rho_A$  the run of  $A$  on  $t$  guided by an accepting run  $\rho_B$  of  $B$  over  $t$ , the labelling  $L_A$  induced by  $\rho_A$  is  $n$ -bound by the labelling  $L_B$  induced by  $\rho_B$ .

**Proof.** TOPROVE 12 ◀

## A.2 Proofs from Section 4

### Proof of Proposition 14

We begin with three technical lemmas: Lemma 21 on the attractor of a union of disjoint sets, and Lemma 22 that builds an attractor decomposition once all the subparts  $S_k$  have been identified. The last one exhibits an attractor decomposition with more structure than usual, in our specific case where the underlying graph is a tree.

► **Lemma 21.** Let  $\kappa$  be an ordinal, let  $G = (V, E, L)$  a finitely-branching parity graph, and  $(S_k)_{k < \kappa}$ ,  $\kappa$  disjoint subsets of  $V^*$ . Then, by defining iteratively, for  $k \leq \kappa$ ,  $V_k := V \setminus \bigcup_{j < k} A_j$  and  $A_k := \text{Attr}(S_k, G|V_k)$ , we have that  $\text{Attr}(\bigsqcup_{k < \kappa} S_k, G) = \bigsqcup_{k < \kappa} A_k$ .

**Proof.** TOPROVE 13 ◀

► **Lemma 22.** Let  $G = (V, E, L)$  a finitely-branching parity game of maximal even parity  $h$ . Let  $H$  be the set of its transitions labelled by  $h$ , and  $A_0 := \text{Attr}(H, G)$ . and  $(S_k)_{1 \leq k \leq \kappa}$  a family of disjoint subsets of  $V$ . If  $(S_k)_{1 \leq k \leq \kappa}$  is such that

- $\forall 1 \leq k \leq \kappa$ ,  $S_k$  is closed under successor in  $(G \setminus H) \setminus \text{Attr}(\bigcup_{k' < k} S_{k'}, G \setminus H)$ .
- $\forall 1 \leq k \leq \kappa$ ,  $(G \setminus H)|S_k$  is a subgame containing priorities up to  $h - 2$ , with an attractor decomposition  $D_i$  of level  $h - 2$
- $\text{Attr}(\bigcup_{0 \leq k \leq \kappa} S_k, G \setminus H) = V$ .

Then, by defining iteratively, for  $1 \leq k \leq \kappa$ ,  $V_k := V \setminus \bigcup_{j < k} A_j$  and  $A_k := \text{Attr}(S_k, (G \setminus H)|V_k)$ ,  $(H, A_0, (S_k, A_k, D_k)_{1 \leq k \leq \kappa})$  is an attractor decomposition of  $G$  (up to neglecting the empty  $S_k$ , of empty attractors).

**Proof.** TOPROVE 14 ◀

We say that an attractor decomposition  $(H, A_0, (S_i, A_i, D_i)_{0 < i < \ell})$  is *tight* if, for all  $i, j < \ell$  with  $j < i$ , all paths from  $S_i$  to  $S_j$  in  $G \setminus H$  are dominated by  $h - 1$ , and the  $(D_i)_{(0 < i < \ell)}$  are also *tight*.

► **Lemma 23.** If  $G$  is an even parity game such that the underlying graph is a tree, we can build a tight attractor decomposition  $D = (H, A_0, (S_i, A_i, D_i)_{0 < i < \ell})$  of  $G$ .

**Proof.** TOPROVE 15 ◀

We can then proceed with Proposition 14, in which we build the attractor decomposition.

► **Proposition 14.** Given a tree-shaped graph  $G = (V, E)$ , finitely-branching and without terminal vertices, two indices  $I = [0, 2i]$  and  $J = [1, 2j]$  and labellings  $\rho_I : E \rightarrow I$  and  $\sigma_J : E \rightarrow J$  such that  $(G, \rho_I)$  and  $(G, \sigma_J)$  are even parity graphs, if  $\rho_I$  is  $n$ -bound by  $\sigma_J$ , then  $(G, \rho_I)$  admits an attractor decomposition of  $n$ -Strahler number at most  $j$ .

**Proof.** TOPROVE 16 ◀

### A.2.1 Proof from Section 5

► **Proposition 15.** *Given a game  $G$  and  $n \in \mathbb{N} \setminus \{0\}$ , if  $G$  has  $n$ -Strahler number  $h$ , then Eve wins  $\mathcal{T}_{[1,2h]}^{n+1}(G)$ .*

**Proof.** TOPROVE 17

