On Unambiguous Regular Tree Languages of Index (0,2)

Jacques Duparc[1], Kevin Fournier[1,2], and Szczepan Hummel*[3]

1 **Université de Lausanne, Switzerland**
jacques.duparc@unil.ch
2 **Université Paris Diderot, France**
kevin.fournier@imj-prg.fr
3 **University of Warsaw, Poland**
shummel@mimuw.edu.pl

Abstract

Unambiguous automata are usually seen as a natural class of automata in-between deterministic and nondeterministic ones. We show that in case of infinite tree languages, the unambiguous ones are topologically far more complicated than the deterministic ones. We do so by providing operations that generate a family $\left(\mathcal{A}^b_\alpha\right)_{\alpha<\varphi_2(0)}$ of unambiguous automata such that:

1. It respects the strict Wadge ordering: $\alpha < \beta$ if and only if $\mathcal{A}^b_\alpha <_W \mathcal{A}^b_\beta$. This can be established without the help of any determinacy principle, simply by providing effective winning strategies in the underlying games.
2. Its length $(\varphi_2(0))$ is the first fixpoint of the ordinal function that itself enumerates all fixpoints of the ordinal exponentiation $x \mapsto \omega^x$: an ordinal tremendously larger than $(\omega^\omega)^3 + 3$ which is the height of the Wadge hierarchy of *deterministic* tree languages as uncovered by Filip Murlak.
3. The priorities of all these parity automata only range from 0 to 2.

1998 ACM Subject Classification F.1.1 Models of Computation, F.1.3 Complexity Measures and Classes, F.4.1 Mathematical Logic

Keywords and phrases Tree Automata, Unambiguity, Wadge Hierarchy

Digital Object Identifier 10.4230/LIPIcs.CSL.2015.534

1 Introduction

An unambiguous automaton is a nondeterministic automaton that admits at most one accepting run on each input. By definition, the class of languages recognized by unambiguous automata includes the class of languages recognized by deterministic automata and is included in the class of languages recognized by nondeterministic automata. Depending on the context, some of these inclusions may be strict. For example, in the case of finite automata on finite words, none of these inclusions is strict, because every regular language is recognized by a deterministic finite automaton. The picture is still trivial for infinite words if we consider the parity condition, but becomes more interesting for Büchi automata. While not every ω-regular (nondeterministic) language is recognized by deterministic Büchi automaton, it always can be recognized by an unambiguous automaton ([2]). From the algorithmic perspective unambiguous automata may be considered as a trade off between succinctness

* The author was partially supported by Polish National Science Centre grant no. 2014-13/B/ST6/03595.

24th EACSL Annual Conference on Computer Science Logic (CSL 2015).
Editor: Stephan Kreutzer; pp. 534–548
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

and efficiency. It is best understood for finite words. One can find example of unambiguous automaton exponentially more succinct than the corresponding deterministic one, while universality, equivalence and inclusion decision problems are in P for unambiguous and are $PSPACE$-complete for nondeterministic automata (see [9] for more details and further references).

In this paper we concentrate on infinite trees and parity condition. On the one hand, it is easy to observe that unambiguous automata are more expressive than the deterministic ones in this context: consider for example the language "exists exactly one branch with infinitely many labels a". On the other hand, it took a while to determine whether there are regular languages that are inherently ambiguous: it was shown by Niwiński and Walukiewicz in [14] (later described in [7] and [8]) that unambiguous automata do not recognize all nondeterministic languages. Still, unambiguous automata, although poorly understood so far, can occur to be an important intermediate model, as many questions seem to be very hard to answer for nondeterministic automata. For example no algorithm is known that calculates Rabin-Mostowski index for a given regular language.

The tool to measure the position of unambiguous languages of infinite trees in between deterministic and nondeterministic ones is given by descriptive set theory through the notion of *topological complexity*. It is well known that deterministic parity tree automata recognize only languages in the $\mathbf{\Pi}^1_1$ class (coanalytic sets), whereas nondeterministic automata recognize some languages that are neither analytic, nor coanalytic. The expressive power of nondeterministic automata is nonetheless bounded by the second level of the projective hierarchy, and, by Rabin's complementation result ([15]), all nondeterministic languages are in fact in the $\mathbf{\Delta}^1_2$ class. In [12], the third author gives an unambiguous language G which is $\mathbf{\Sigma}^1_1$-complete, and constructs from it an unambiguous language that is outside both $\mathbf{\Pi}^1_1$ and $\mathbf{\Sigma}^1_1$. A finer topological complexity measure is therefore needed: the Wadge hierarchy, which relies on the notion of reductions by continuous functions (Wadge-reducibility). Complexity classes, called Wadge degrees, consist of sets Wadge-reducible to each other, and constitute a hierarchy whose levels, called ranks, can be enumerated with ordinals. We describe a series of operations on automata that preserve unambiguity and lift the Wadge degrees of the recognized languages. We emphasize that this is done without any particular determinacy principle. In particular, we do not require $\mathbf{\Delta}^1_2$-determinacy. These operations help us generate a hierarchy of canonical unambiguous languages of higher and higher topological complexity. This hierarchy has $\varphi_2(0)$ many levels, where $\varphi_2(0)$ stands for the first fixpoint of the ordinal function[1] $x \mapsto \varepsilon_x$ which itself enumerates the fixpoints of the exponentiation $x \mapsto \omega^x$. Compared to the height of the Wadge hierarchy of all deterministic tree languages, which is $(\omega^\omega)^3 + 3$ as established by Filip Murlak in [13], the ordinal $\varphi_2(0)$ is tremendously larger.

The gap between the respective topological complexity of the two considered classes of languages, measured by the difference between the height of their respective Wadge hierarchies, illustrates the discrepancy between these classes. It is nonetheless not the only interest of the descriptive set theoretic framework, as it is shown by the recent results obtained on MSO+U. In [5], a topological complexity result was used to prove that there is no algorithm that decides satisfiability of a formula of MSO+U logic on infinite trees and that has a correctness proof in ZFC (an "almost undecidability" result), partially answering a question that was open for over ten years [4]. The elementary undecidability argument was later given in [6], but the topological result came first and motivated the research towards

[1] not to be mistaken with an ε-move.

the other. Our constructions provide benchmarks for the study of unambiguous languages, and could lead to prominent algorithmic results for this class. It might, for example, help determine if it is decidable whether a given nondeterministic language is unambiguous. The result also could contribute to solving unambiguous index problem as it can help in characterising unambiguous languages of index $(0,2)$.

2 Preliminaries

2.1 The Wadge hierarchy and the Wadge game

The Wadge theory is in essence the theory of *pointclasses* (see [1]). Let X be a topological space. A pointclass is a collection of subsets of X that is closed under continuous preimages. For Γ a pointclass, we denote by $\check{\Gamma}$ its *dual* class containing all the subsets of X whose complements are in Γ, and by $\Delta(\Gamma)$ the ambiguous class $\Gamma \cap \check{\Gamma}$. If $\Gamma = \check{\Gamma}$, we say that Γ is self-dual.

The *Wadge preorder* $\leq_W$ on $\mathscr{P}(X)$ is defined as follows: for $A, B \subseteq X$, $A \leq_W B$ if and only if there exists $f : X \longrightarrow X$ continuous such that $f^{-1}(B) = A$. It is merely by definition a preorder. The Wadge preorder induces an equivalence relation $\equiv_W$ whose equivalence classes are called the *Wadge degrees*, and denoted by $[A]_W$. We say that the set $A \subseteq X$ is *self-dual* if it is Wadge equivalent to its complement, that is if $A \equiv_W A^{\complement}$, and *non self-dual* if it is not. We use the same terminology for the Wadge degrees.

Let Γ be a pointclass of X. There is a strong connection between pointclasses included in Γ and Wadge degrees of sets in Γ, since all non self-dual pointclasses are of the form

$$\{B \subseteq X : B \leq_W A\}$$

for some non self-dual set A, while self-dual pointclasses are all of the form

$$\{B \subseteq X : B \leq_W A \text{ and } A \nleq_W B\},$$

also for some non self-dual set A. We have thus a direct correspondence between $(\mathscr{P}(X), \leq_W)$ restricted to Γ and the pointclasses included in Γ with the inclusion: the pointclasses are exactly the initial segments of the Wadge preorder. In particular, the Wadge hierarchy refines tremendously the Borel and the Projective hierarchies.

A *conciliatory* binary tree over a finite set Σ is a partial function $t : \{0,1\}^* \to \Sigma$ with a prefix closed domain. Those trees can have both infinite and finite branches. A tree is called *full* if $\mathrm{dom}(t) = \{0,1\}^*$. Let $\mathcal{T}_\Sigma^{\leq\omega}$ and T_Σ denote, respectively, the set of all conciliatory binary trees and the set of full binary trees over Σ. Given $x \in \mathrm{dom}(t)$, we denote by t_x the subtree of t rooted in x. Let $\{0,1\}^n$ denote the set of words over $\{0,1\}$ of length n, and let t be a conciliatory tree over Σ. We denote by $t[n]$ the finite initial binary tree of height $n+1$ given by the restriction of t to $\bigcup_{0\leq i\leq n}\{0,1\}^i$.

The space T_Σ equipped with the standard Cantor topology is a Polish space and is in fact homeomorphic to the Cantor space[2]. Let $L, M \subseteq T_\Sigma$, the Wadge game $W(L, M)$ is a two player infinite game that provides a very useful characterization for the Wadge preorder. In this game, each player builds a tree, say t_{I} and t_{II}. At every round, player I plays first, and both players add a finite number of children to the terminal nodes of their tree. Player II is allowed to skip its turn, but has to produce a tree in T_Σ throughout a game. Player II wins the game if and only if $t_I \in L \Leftrightarrow t_{II} \in M$.

[2] See for example [3].

▶ **Lemma 1** ([18]). *Let $L, M \subseteq T_\Sigma$. Then $L \leq_W M$ if and only if player* II *has a winning strategy in the game $W(L, M)$.*

We write $A <_W B$ when II has a winning strategy in $W(A, B)$ *and* I has a winning strategy in $W(B, A)$[3].Given a pointclass Γ of T_Σ with suitable closure properties, the assumption of the determinacy of Γ is sufficient to prove that Γ is semi-linearly ordered by $\leq_W$, denoted SLO(Γ), i.e. that for all $L, M \in \Gamma$,

$$L \leq_W M \qquad \text{or} \qquad M \leq_W L^{\complement},$$

and that $\leq_W$ is well founded when restricted to sets in Γ ([16, 1]). Under these conditions, the Wadge degrees of sets in Γ with the induced order is thus a hierarchy called the *Wadge hierarchy*. Therefore, there exists a unique ordinal, called the height of the Γ-Wadge hierarchy, and a mapping d_W^Γ from the Γ-Wadge hierarchy onto its height, called the *Wadge rank*, such that, for every L, M non-self-dual in Γ, $d_W^\Gamma(L) < d_W^\Gamma(M)$ if and only if $L <_W M$ and $d_W^\Gamma(L) = d_W^\Gamma(M)$ if and only if $L \equiv_W M$ or $L \equiv_W M^{\complement}$. The wellfoundedness of the Γ-Wadge hierarchy ensures that the Wadge rank can be defined by induction as follows:

- $d_W^\Gamma(\varnothing) = d_W^\Gamma(\varnothing^{\complement}) = 1$
- $d_W^\Gamma(L) = \sup\left\{d_W^\Gamma(M) + 1 : M \text{ is non-self-dual}, M <_W L\right\}$ for $L >_W \varnothing$.

Note that given two pointclasses Γ and Γ', for every $L \in \Gamma \cap \Gamma'$, $d_W^\Gamma(L) = d_W^{\Gamma'}(L)$. Under sufficient determinacy assumptions, we can therefore safely speak of *the* Wadge rank of a tree language, denoted by d_W, as its Wadge rank with respect to any topological class including it. However the main result of this article does not provide any Wadge rank for the canonical languages that are constructed, because we do not make use of any determinacy principle.

2.2 The Conciliatory Hierarchy

For conciliatory languages L, M we define the *conciliatory* version of the Wadge game: $C(L, M)$ ([10, 11]). The rules are similar, except for the fact that both players are now allowed to skip and to produce trees with finite branches - or even finite trees. For conciliatory languages L, M we use the notation $L \leq_c M$ if and only if II has a winning strategy in the game $C(L, M)$. If $L \leq_c M$ and $M \leq_c L$, we will write $L \equiv_c M$. The conciliatory hierarchy is thus the partial order induced by $\leq_c$ on the equivalence classes given by $\equiv_c$. We write $A <_c B$ when II has a winning strategy in $C(A, B)$ *and* I has a winning strategy in $C(B, A)$.

From a conciliatory language L over Σ, one defines the corresponding language L^b of full trees over $\Sigma \cup \{b\}$ by:

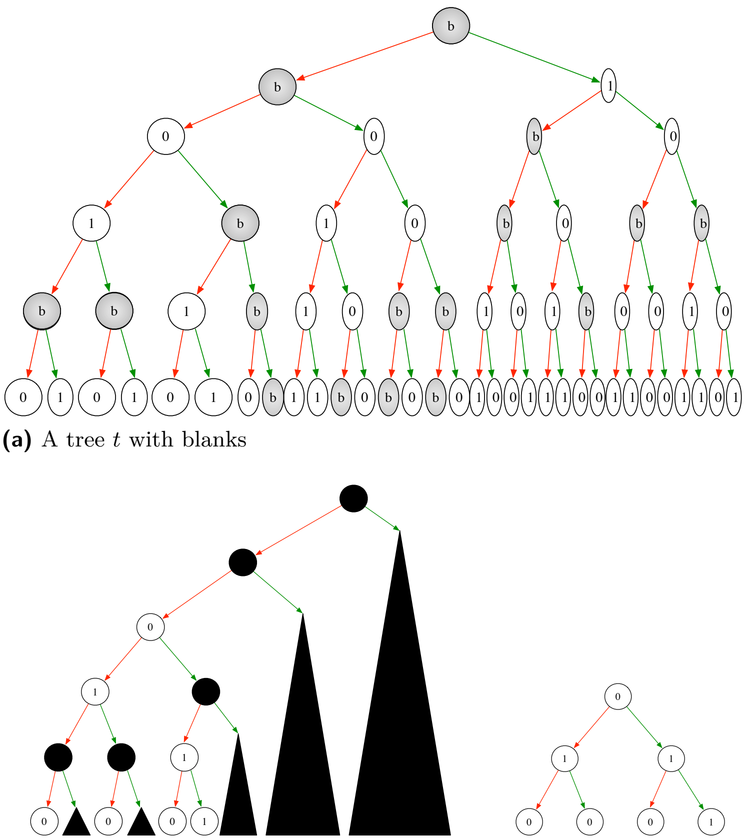
$$L^b = \left\{t \in T_{\Sigma \cup \{b\}} : t_{[\ /b]} \in L\right\},$$

where b is an extra symbol that stands for "blank", and $t_{[\ /b]}$, the *undressing* of t, is informally the conciliatory tree over Σ obtained once all the occurences of b have been removed in a top-down manner. More precisely, if there is a node v such that $t(v) = b$, we ignore this node and replace it with $v0$. If, for every integer n, $t(v0^n) = b$, then $v \notin \mathrm{dom}(t_{[\ /b]})$. This process is illustrated by Figure 1.

Formally, for each $v \in \{0,1\}^*$ we consider two (possibly infinite) sequences (w_i) and (u_i) in $\{0,1\}^*$:

- $w_0 = \varepsilon$, $u_0 = v$,

[3] This is in general stronger than the usual $A <_W B$ if and only if $A \leq_W B$ and $B \nleq_W A$, but the two definitions coincide when the classes considered are determined.



(a) A tree t with blanks

(b) The blanks are deleted in a top-down manner

(c) The resulting tree $t_{[\ /b]}$

Figure 1 The undressing process.

- for $0 \leq i$:
  - if $t(w_i) = b$, we set $u_{i+1} = u_i$ and $w_{i+1} = w_i 0$;
  - if $t(w_i) \neq b$ and $u_i = a u_i'$ for $a \in \{0,1\}$, we set $u_{i+1} = u_i'$ and $w_{i+1} = w_i a$;
  - if $t(w_i) \neq b$ and $u_i = \varepsilon$, we halt the construction at step i.

If the construction is halted at some step i, then $v \in \mathrm{dom}(t_{[\ /b]})$ and $t_{[\ /b]}(v) = t(w_i)$. Otherwise, $v \notin \mathrm{dom}(t_{[\ /b]})$. If Γ is a pointclass of full trees, we say that a conciliatory language L is in Γ if and only if L^b is in Γ.

▶ **Lemma 2.** *Let L and M be conciliatory languages. Then*

$L \leq_c M$ if and only if $L^b \leq_W M^b$.

Proof. A strategy in one game can be translated directly into a strategy in the other game: arbitrary skipping in $C(L, M)$ gives the same power as the b labels in $W(L^b, M^b)$. In particular, in $W(L^b, M^b)$, II does not need to skip at all. ◀

The mapping $L \mapsto L^b$ gives thus a natural embedding of the preorder $\leq_c$ restricted to conciliatory sets in Γ into the Γ-Wadge hierarchy. Hence, for Γ with suitable closure and determinacy properties, the conciliatory degrees of sets in Γ with the induced order constitute a hierarchy called the *conciliatory hierarchy*. We define, by induction, the corresponding *conciliatory rank* of a language:

- $d_c^\Gamma(\varnothing) = d_c^\Gamma(\varnothing^\complement) = 1$
- $d_c^\Gamma(L) = \sup\{d_c^\Gamma(M) + 1 : M <_c L\}$ for $L >_c \varnothing$.

Similarly to the Wadge case, given two pointclasses Γ and Γ', for every conciliatory $L \in \Gamma \cap \Gamma'$, $d_c^\Gamma(L) = d_c^{\Gamma'}(L)$. Under sufficient determinacy assumptions, we can therefore safely speak, of *the* conciliatory rank of a conciliatory tree language, denoted by d_c, as its conciliatory rank with respect to any topological class including it. Observe that the conciliatory hierarchy does not contain self-dual languages: a strategy for I in $C(L, L^\complement)$ is to skip in the first round, and then copy moves of II.

2.3 Automata and conciliatory trees

A *nondeterministic parity tree automaton* $\mathcal{A} = \langle \Sigma, Q, I, \delta, \mathrm{r}\rangle$ consists of a finite input alphabet Σ, a finite set Q of states, a set of initial states $I \subseteq Q$, a transition relation $\delta \subseteq Q \times \Sigma \times Q \times Q$ and a priority function $\mathrm{r} : Q \to \omega$. A run of automaton $\mathcal{A}$ on a binary conciliatory input tree $t \in \mathcal{T}_\Sigma^{\leq\omega}$ is a conciliatory tree $\rho_t \in \mathcal{T}_Q^{\leq\omega}$ with $\mathrm{dom}(\rho_t) = \{\varepsilon\} \cup \{va : v \in \mathrm{dom}(t) \wedge a \in \{0,1\}\}$ such that the root of this tree is labeled with a state $q \in I$, and for each $v \in \mathrm{dom}(t)$, transition $(\rho_t(v), t(v), \rho_t(v_1), \rho_t(v_1)) \in \delta$. The run ρ_t is *accepting* if parity condition is satisfied on each infinite branch of ρ_t, i.e. if the highest rank of a state occurring infinitely often on the branch is even, and if the rank of each leaf node in ρ_t is even. We say that a parity tree automaton A *accepts* a conciliatory tree t if it has an accepting run on t. The language *recognized* by A, denoted $L(A)$ is the set of trees accepted by A. The Rabin-Mostowski index of the automaton is a pair $(\min(r(Q)), \max(r(Q)))$. A language is of index (i,k) if it is recognized by some automaton of index (i,k). An automaton is *unambiguous* if it has at most one accepting run on each input. We denote by $L^\omega(A)$ the set of full trees recognized by A, i.e. $L^\omega(A) = L(A) \cap T_\Sigma$.

▶ **Corollary 3.** *The mapping $L \mapsto L^b$ embeds the conciliatory hierarchy for $\boldsymbol{\Delta}_2^1$-sets restricted to unambiguously recognizable languages into the $\boldsymbol{\Delta}_2^1$-Wadge hierarchy restricted to unambiguously recognizable languages.*

Proof. By Lemma 2 it is enough to prove that each unambiguous automaton A can be transformed into an unambiguous automaton A' such that $L^\omega(A') = L(A)^b$. Given any unambiguous automaton A, this is done by adding an all-accepting state $\top$ to the set of states Q_A, and the set $\{(q, b, q, \top) : q \in Q_A\}$ to the transition relation δ_A. The obtained automaton A' is unambiguous and such that $L^\omega(A') = L(A)^b$. ◀

In the diagrams of automata below, we use ε-transitions, i.e transitions that change state but do not progress run down a tree. This is, however, only a notation shortcut here — we do it to emphasize nondeterministic choice better. The transitions can be easily simulated by adding more transitions of the type as in the above definition to the state they lead from. We also use the following conventions in the diagrams. Nodes represent states of the automaton. Node labels correspond to state ranks. We additionally mark parity of ranks by node colors: nodes corresponding to states with even ranks are green, while nodes corresponding to states with odd ranks are red. A red edge shows the state that is assigned to the left successor node of a transition, a green edge goes to the right successor node. Edge label marks the label of a tree the transition goes through. In order to lighten the notation, transitions that are not depicted on a diagram lead to some definitely all-accepting state.

3 Operations on languages and their automatic counterparts

In this section, we present classical operations ([11]) on conciliatory tree languages that allow us to construct more and more complicated languages, and we prove that they preserve unambiguity, i.e. that if we apply them to unambiguously recognizable languages, the resulting language is equivalent[4] to an unambiguously recognizable one. Without loss of generality, we may choose the alphabet $\Sigma = \{0, 1\}$.

3.1 The sum

For $L, M \subseteq \mathcal{T}_\Sigma^{\leq\omega}$, we define the *sum* of L and M, in symbols $L \oplus M$, as the conciliatory tree language containing all of those trees $t \in \mathcal{T}_\Sigma^{\leq\omega}$ that one of the following conditions holds:

- $t(10^n) = 0$ for each integer n and $t_0 \in M$;
- the node 10^n is the first on the path 10^* labeled with 1 and either $t(10^n0) = 0$ and $t_{10^n00} \in L$, or $t(10^n0) = 1$ and $t_{10^n00} \in L^{\complement}$

This operation behaves well regarding the conciliatory hierarchy.

▶ **Facts 4** ([10, 11]). *Let L, M, and M' be conciliatory tree languages over Σ. Then the following hold.*

1. $(L \oplus M)^{\complement} \equiv_c L \oplus M^{\complement}$.
2. $(L \oplus M) \oplus M' \equiv_c L \oplus (M \oplus M')$.
3. *The operation $\oplus$ preserves the conciliatory ordering: if $M' \leq_c M$, then*

$$L \oplus M' \leq_c L \oplus M.$$

Let $\mathcal{A}$ and $\mathcal{B}$ be two automata that recognize respectively the conciliatory languages M and L. Then the automaton $\mathcal{B} + \mathcal{A}$ depicted in Figure 2 recognizes the sum of L and M. In this picture, $\mathcal{C}$ is any automaton that recognizes a language equivalent to $L^{\complement}$, and the parities i and j are defined as follows:

- $i = 0$ if and only if the empty tree is accepted by $\mathcal{A}$;
- $j = 1$ if and only if $L(\mathcal{A})$ is equivalent to $L(\mathcal{A}) \to \circleddash$.[5]

Note that the operation sum is *in itself* unambiguous, so that if $\mathcal{A}$ and $\mathcal{B}$ are unambiguous, and if there exists an unambiguous $\mathcal{C}$ equivalent to the complement of $\mathcal{B}$, their sum $\mathcal{B} + \mathcal{A}$ is also unambiguous. The core observation here is that only one of the initial ε-transitions can be taken in a root of a given tree, depending on whether there is a node labeled with 1 on branch 10^* in the tree or not. Moreover, if L and M are unambiguously recognizable conciliatory languages, and if the complement of M is equivalent to an unambiguously recognizable language $\check{M}$, the complement of $L \oplus M$ is equivalent to $L \oplus \check{M}$, which is unambiguously recognizable.

For $M \subseteq \mathcal{T}_\Sigma^{\leq\omega}$ and $n \in \omega$, we denote by $M \odot n$ the sum of M with itself n times:

$$M \odot n = \underbrace{M \oplus M \oplus \ldots \oplus M}_{n \text{ times}}.$$

If $\mathcal{A}$ recognizes M, we denote by $\mathcal{A} \bullet n$ the automaton that recognizes $M \odot n$.

---

[4] Relatively to $\equiv_c$.

[5] A player in charge of $L(\mathcal{A}) \to \circleddash$ in a conciliatory game is like a player in charge of $L(\mathcal{A})$, but with the extra possibility at any moment of the play to reach a definitively rejecting position. We denote by $\circleddash$ the automaton that rejects all trees.

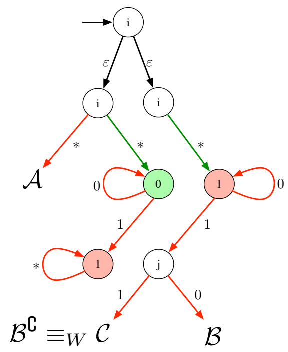


Figure 2 The automaton $\mathcal{B} + \mathcal{A}$ that recognizes $L(\mathcal{B}) \oplus L(\mathcal{A})$. The values of i and j depend on properties of $\mathcal{A}$. The transitions that are not depicted lead to an all-accepting state $\top$.

▶ **Lemma 5.** *Let L, L', M and M' be conciliatory languages such that $L <_c L'$ and $M \leq_c M'$. Then, the following hold.*
1. $M \oplus L <_c M' \oplus L'$;
2. $M <_c M \oplus L$.[6]

Proof.
1. It is clear that $M \oplus L \leq_c M' \oplus L'$, what remains to prove is thus that I has a winning strategy in $C(M' \oplus L', M \oplus L)$. Let τ be the winning strategy for I in $C(L', L)$. Observe that, since $M \leq_c M'$, player I has a winning strategy τ' in $C(M', M^{\complement})$. A strategy σ for I in the game $C(M' \oplus L', M \oplus L)$ is the following. First I plays 0 on the node ε, and then, as long as player II does not play a 1 on the branch 10^*, I follows τ on the left subtree $0\{0,1\}^*$. If ever II plays a 1 on a node 10^n, then I copies II's moves for the branch 10^n0, and then follows τ' on the subtree $10^n0\{0,1\}^*$. Since τ and τ' are winning, σ is a winning strategy for I in $C(M' \oplus L', M \oplus L)$. Thus $M \oplus L <_c M' \oplus L'$.
2. It is clear that $M \leq_c M \oplus L$: a winning strategy for II in $C(M, M \oplus L)$ is indeed to play 0 at ε, 1 at the node 1, 0 at the node 010, and then copy I's moves in the subtree $010\{0,1\}^*$. The winning strategy σ for I in the game $C(M, M \oplus L)$ is similar. First, I plays 0 at ε, 1 at the node 1, 1 at the node 010, and then copy I's moves in the subtree $010\{0,1\}^*$. ◀

3.2 The pseudo-exponentiation

Let $P \subseteq \mathcal{T}_\Sigma^{\leq\omega}$ be a conciliatory tree language. For $t \in \mathcal{T}_\Sigma^{\leq\omega}$, let:

$$i^P(t)(a_1 a_2 \dots a_n) = \begin{cases} t(a_1 0 a_2 0 \dots 0 a_n 0) & \text{if } t_{a_1 0 a_2 0, \dots 0 a_n 1} \in P; \\ b & \text{otherwise.} \end{cases}$$

This process is illustrated in Figure 3. The nodes in red are called the *auxiliary moves*, and the nodes in blue the *main run*. The blue arrows denote the dependency of a node of the main run on a subtree of auxiliary moves. If the auxiliary subtree of a main run node is not in P, then we say that the node is *killed*.

Let $L \subseteq \mathcal{T}_\Sigma^{\leq\omega}$, we define the *action* of P on L, in symbols (P, L), by

$$\left\{t \in \mathcal{T}_\Sigma^{\leq\omega} : i^P(t)_{[\ /b]} \in L\right\}.$$

[6] In particular $M \odot n <_c M \odot (n+1)$ for any $0 < n < \omega$.

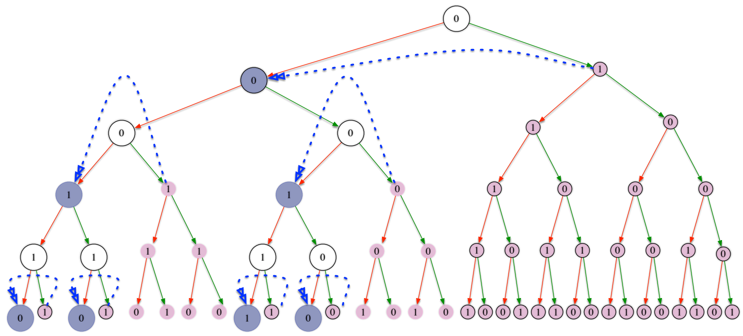


Figure 3

Let $P_{\mathbf{\Pi}^0_1}$ be the complete closed set of all full trees over Σ with all nodes on the leftmost branch 0^* labelled by 0. For $L \subseteq \mathcal{T}_\Sigma^{\leq\omega}$, we denote by $(\mathbf{\Pi}^0_1, L)$ the action of $P_{\mathbf{\Pi}^0_1}$ on L. This operation $(\mathbf{\Pi}^0_1, \cdot)$ behaves well regarding the conciliatory hierarchy.

▶ **Facts 6** ([10, 11]). *Let L and M be conciliatory tree languages over Σ. Then the following hold.*

1. $(\mathbf{\Pi}^0_1, L)^\complement \equiv_c (\mathbf{\Pi}^0_1, L^\complement)$.
2. *If $L \leq_c M$, then $(\mathbf{\Pi}^0_1, L) \leq_c (\mathbf{\Pi}^0_1, M)$.*
3. *If $L <_c M$, then $(\mathbf{\Pi}^0_1, L) <_c (\mathbf{\Pi}^0_1, M)$.*

The sets obtained as results of the operation $(\mathbf{\Pi}^0_1, \cdot)$ are, so to speak, fixed points for $\oplus$.

▶ **Proposition 7.** *Let L, L' and M be conciliatory languages such that $L <_c (\mathbf{\Pi}^0_1, M)$ and $L' <_c (\mathbf{\Pi}^0_1, M)$. Then*

$$L \oplus L' <_c (\mathbf{\Pi}^0_1, M)$$

Proof. The fact that $L \oplus L' \leq_c (\mathbf{\Pi}^0_1, M)$ is clear: if σ_0, σ_1 and σ' are winning strategies respectively in the games $C(L, (\mathbf{\Pi}^0_1, M))$, $C(L^\complement, (\mathbf{\Pi}^0_1, M))$ and $C(L', (\mathbf{\Pi}^0_1, M))$, a winning strategy for II in $C(L \oplus L', (\mathbf{\Pi}^0_1, M))$ is the following. As long as player I does not play a 1 on the branch 10^*, II does not kill any nodes and follows σ' to what I plays in the subtree $0\{0,1\}^*$ to get her main run. If ever II plays a 1 on a node 10^n, then II kills all the nodes of the main run she had already played (by playing 1 on the leftmost branches of appropriate auxiliary subtrees), and begins to play along a tree not in M in her main run, without killing any node. If I plays 0 on the node 10^n0, she kills every node in the main run she had already play, and then follows σ_0 on the subtree $10^n0\{0,1\}^*$. If I plays 1 on the node 10^n0, she kills every node in the main run she had already play, and then she follows σ_1 on the subtree $10^n0\{0,1\}^*$. The proof that I has a winning strategy τ in the game $C((\mathbf{\Pi}^0_1, M), L \oplus L')$ is mutatis mutandis the same, given that I has a winning strategy for each of the games $C((\mathbf{\Pi}^0_1, M), L)$, $C((\mathbf{\Pi}^0_1, M), L^\complement)$ and $C((\mathbf{\Pi}^0_1, M), L')$. ◀

Let $\mathcal{A}$ be an automaton that recognizes $L \subseteq \mathcal{T}_\Sigma^{\leq\omega}$. Then the conciliatory tree language $(\mathbf{\Pi}^0_1, L)$ is recognized by the automaton $\omega^{\mathcal{A}}$ defined from $\mathcal{A}$ by replacing each state of $\mathcal{A}$ by a "gadget", as depicted in Figure 4. By replacing a state by the gadget we mean that all transitions ending in this state should now end in the initial state of the gagdet, and that all the transitions starting from this state should now start from the final state of the gadget. This sort of gadget first appeared in [11].

Observe that if $\mathcal{A}$ is unambiguous, then $\omega^{\mathcal{A}}$ is also unambiguous, so that the operation $(\mathbf{\Pi}^0_1, \cdot)$ preserves the unambiguity of conciliatory tree languages.

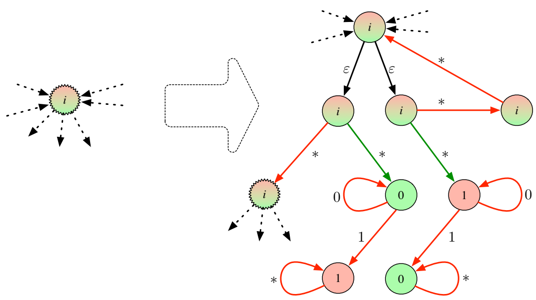
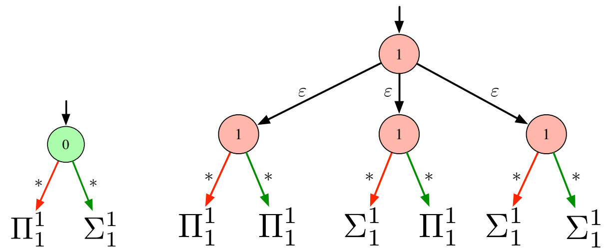


Figure 4 The gadget to replace a state in $\mathcal{A}$.



(a) $\mathcal{A}_1$ **(b)** $\mathcal{A}_2$

Figure 5 Sketch of automata, where $\mathbf{\Pi}^1_1$ and $\mathbf{\Sigma}^1_1$ denote automata that recognize respectively a $\mathbf{\Pi}^1_1$-complete language and the complement of this language.

4 Difference of co-analytic sets

The operations defined in Section 3 are *Borel* in the sense that when we apply them to Borel languages, the resulting language is still Borel. As our purpose is to illustrate the wide discrepancy between deterministic and unambiguously recognizable languages, we need to climb higher in the topological complexity hierarchy. In order to achieve this objective, we will combine a construction due to the third author in [12] with a variant of the pseudo-exponentiation.

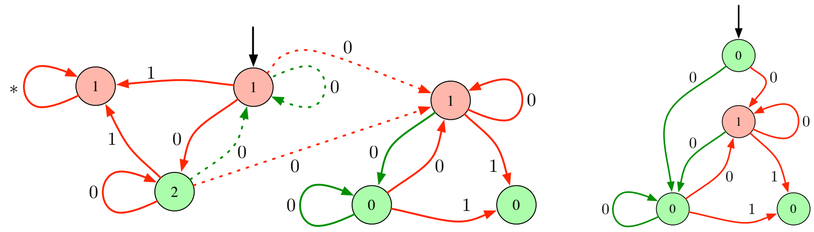
4.1 The $D_2(\mathbf{\Pi}^1_1)$ class

For a topological space X, we denote by $D_2(\mathbf{\Pi}^1_1)(X)$ the class of differences of two coanalytic sets, i.e.

$$D_2(\mathbf{\Pi}^1_1)(X) = \left\{A \cap B : A \in \mathbf{\Pi}^1_1(X) \text{ and } B \in \mathbf{\Sigma}^1_1(X)\right\}.$$

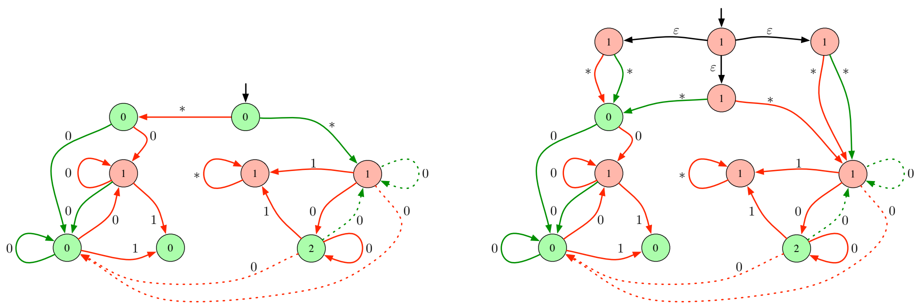
Using the unambiguously recognizable $\mathbf{\Sigma}^1_1$-complete language G (of full trees) from [12], we define an unambiguously recognizable conciliatory language that is $D_2(\mathbf{\Pi}^1_1)$-complete and such that its complement is also unambiguously recognizable. Their definitions are given via the automata that recognize them. The abstract idea behind our construction is depicted by Figure 5 which represents a general form of automata that would recognize languages that are $D_2(\mathbf{\Pi}^1_1)$-complete (Figure 5a), and $\check{D}_2(\mathbf{\Pi}^1_1)$-complete (Figure 5b).

The automaton $\mathcal{A}_1$, indeed, recognizes a tree $t \in \mathcal{T}^{\leq\omega}_{\Sigma}$ if and only if t_0 is in a given conciliatory $\mathbf{\Pi}^1_1$-complete language (say A) and t_1 is in its complement which is $\mathbf{\Sigma}^1_1$-complete. Since the maps $t \mapsto t_0$ and $t \mapsto t_1$ are continuous, the language recognized by the automaton is



(a) $\mathcal{A}_{\mathbf{\Sigma}_1^1}$ **(b)** $\mathcal{A}_{\mathbf{\Pi}_1^1}$

Figure 6 Unambiguous automata that recognize respectively a $\mathbf{\Sigma}_1^1$-complete language and its complement. Line style of the edges is used on the diagram in case of nondeterministic choice: if there are two different transitions from given state over given letter then egdes of one of them are drawn with solid line while the edges of the other are drawn with dashed line.



(a) $\mathcal{A}_{D_2(\mathbf{\Pi}_1^1)}$ **(b)** $\mathcal{A}_{\check{D}_2(\mathbf{\Pi}_1^1)}$

Figure 7 Unambiguous automata that recognize respectively a $D_2(\mathbf{\Pi}_1^1)$-complete and a $\check{D}_2(\mathbf{\Pi}_1^1)$-complete language. The transitions that are not depicted lead to an all-accepting state $\top$.

thus $D_2(\mathbf{\Pi}_1^1)$. Moreover, if $M \in \mathbf{\Pi}_1^1$ and $M' \in \mathbf{\Sigma}_1^1$, $M \cap M' \leq_c L(\mathcal{A}_1)$: a winning strategy for player II in the game $C(M \cap M', L(\mathcal{A}_1))$ is indeed to glue together her winning strategies in the games $C(M, A)$ and $C(M', A^{\complement})$. Hence, the language recognized by $\mathcal{A}_1$ is $D_2(\mathbf{\Pi}_1^1)$-complete. The reasoning for $\mathcal{A}_2$ is similar, observing that

$$(M \cap M')^{\complement} = (M \cap M'^{\complement}) \cup (M^{\complement} \cap M'^{\complement}) \cup (M^{\complement} \cap M').$$

We now define two unambiguous automata: the first one recognizes a $\mathbf{\Sigma}_1^1$-complete language, and the other one recognizes the complement of the first one, i.e. a $\mathbf{\Pi}_1^1$-complete language[7]. They are depicted in Figure 6.

We will denote by $A_{\mathbf{\Sigma}_1^1}$ and $A_{\mathbf{\Pi}_1^1}$ the conciliatory languages recognized respectively by $\mathcal{A}_{\mathbf{\Sigma}_1^1}$ and $\mathcal{A}_{\mathbf{\Pi}_1^1}$. Combining these constructions, we can now define an unambiguously recognizable conciliatory language that is $D_2(\mathbf{\Pi}_1^1)$-complete (Figure 7a) and such that its complement (Figure 7b) is also unambiguously recognizable, via the automata that recognize each of them. We will denote by $A_{D_2(\mathbf{\Pi}_1^1)}$ and $A_{\check{D}_2(\mathbf{\Pi}_1^1)}$ the conciliatory languages recognized respectively by $\mathcal{A}_{D_2(\mathbf{\Pi}_1^1)}$ and $\mathcal{A}_{\check{D}_2(\mathbf{\Pi}_1^1)}$.

[7] See [12] for proofs.

4.2 The operation $(D_2(\boldsymbol{\Pi}^1_1), \cdot)$

For $M \subseteq \mathcal{T}_\Sigma^{\leq\omega}$, we denote by $(D_2(\boldsymbol{\Pi}^1_1), M)$ the action of $A_{D_2(\boldsymbol{\Pi}^1_1)}$ on M. Observe that this operation is highly non-Borel, since if we apply it to a $\boldsymbol{\Sigma}^0_1$-complete conciliatory language, the resulting language will be complete for the pointclass of all the countable unions of $D_2(\boldsymbol{\Pi}^1_1)$ languages. We prove that $(D_2(\boldsymbol{\Pi}^1_1), \cdot)$ behaves well with respect to $\leq_c$.

▶ **Theorem 8.** *Let $M, M' \subseteq \mathcal{T}_\Sigma^{\leq\omega}$. If $M \leq_c M'$, then*
1. $(D_2(\boldsymbol{\Pi}^1_1), M)^\complement \equiv_c (D_2(\boldsymbol{\Pi}^1_1), M^\complement)$;
2. $(D_2(\boldsymbol{\Pi}^1_1), M) \leq_c (D_2(\boldsymbol{\Pi}^1_1), M')$.

Proof. The first point holds merely by the definition of the operation $(D_2(\boldsymbol{\Pi}^1_1), \cdot)$. The proof of the second point relies on a variation of the *remote control* strategy ([10]). Let t be a finite binary tree over $\{0,1,2,3\}$. We say that t is *coherent* if for every node $v \in \text{dom}(t)$, $t(v) \in \{1,2,3\}$ implies that all the nodes in $v1\{0,1\}^* \cap \text{dom}(t)$ have the same label, $t(v)$. Let $(\beta_n)_{n\in\omega}$ be an enumeration of the set of the coherent trees, such that if t_i is an initial segment of t_j, then $i \leq j$. We call β_i the i-th *bet*. A bet encodes information on the auxiliary moves of I in the game $C((D_2(\boldsymbol{\Pi}^1_1), M), (D_2(\boldsymbol{\Pi}^1_1), M'))$: its underlying binary tree determines the part of the main run taken into account, and the values at the nodes whether this node will be killed or not, and how. Suppose I plays a conciliatory tree t. For $v = v_0 \dots v_j \in \text{dom}(\beta_i)$, $\beta_i(v) = 0$ means that the node $0v_00v_1 \dots 0v_j$ stays alive, i.e. that $t_{0v_00v_1\dots 0v_j 1} \in A_{D_2(\boldsymbol{\Pi}^1_1)}$. The value 1 means that the node $0v_00v_1 \dots 0v_j$ is killed because $t_{0v_00v_1\dots 0v_j 10}$ and $t_{0v_00v_1\dots 0v_j 11}$ belong to $A_{\boldsymbol{\Pi}^1_1}$, so that $t_{0v_00v_1\dots 0v_j 1} \in A_{\breve{D}_2(\boldsymbol{\Pi}^1_1)}$. The value 2 means that the node $0v_00v_1 \dots 0v_j$ is killed because $t_{0v_00v_1\dots 0v_j 10} \in A_{\boldsymbol{\Sigma}^1_1}$ and $t_{0v_00v_1\dots 0v_j 11} \in A_{\boldsymbol{\Pi}^1_1}$, and the value 3 means that it is killed because both $t_{0v_00v_1\dots 0v_j 10}$ and $t_{0v_00v_1\dots 0v_j 11}$ belong to $A_{\boldsymbol{\Sigma}^1_1}$. We say that a bet β_i is fulfilled if at the end of the game, for all $v \in \text{dom}(\beta_i)$, $\beta_i(v)$ is true with respect to the conciliatory tree played by I. Notice that it is a $D_2(\boldsymbol{\Pi}^1_1)$ condition (it is a finite intersection of $\boldsymbol{\Sigma}^1_1$ and $\boldsymbol{\Pi}^1_1$ sets), so that II can check if a bet is fulfilled or not with an auxiliary move.

Suppose now that II has a winning strategy σ in $C(M, M')$. We describe a winning strategy σ' for II in the game $C((D_2(\boldsymbol{\Pi}^1_1), M), (D_2(\boldsymbol{\Pi}^1_1), M'))$. Each level of II's main run corresponds to a bet: suppose at some point I has constructed a finite tree t for his main run, and let β_i be a bet such that $\text{dom}(t) = \text{dom}(\beta_i)$. On the level i of her main run, II follows σ modulo β_i, in the sense that she plays along σ as if at all the levels $j < i$ of her main run such that β_j is not a subtree of β_i, the nodes were killed, and she checks with her auxiliary moves for the nodes of the main run at this level whether β_i is fulfilled or not, so that all the nodes of her main run at this level are killed if the bet is not fulfilled. At the end of the game, a unique sequence of bets forming a chain for the inclusion is fulfilled, which contains all information about the way player I used his auxiliary moves, and which nodes he killed. Hence,

$$i^{A_{D_2(\boldsymbol{\Pi}^1_1)}}(\sigma' * t)_{[\ /b]} = \sigma * i^{A_{D_2(\boldsymbol{\Pi}^1_1)}}(t)_{[\ /b]},$$

where $\sigma * t$ denotes the tree resulting from application of strategy σ to tree t. That finishes the proof. ◀

Mutatis mutandis, a winning strategy for I in $C(M, M')$ can also be "remote controlled" to a winning strategy for I in $C((D_2(\boldsymbol{\Pi}^1_1), M), (D_2(\boldsymbol{\Pi}^1_1), M'))$.

▶ **Corollary 9.** *Let M and M' be conciliatory languages such that $M <_c M'$. Then*

$$(D_2(\boldsymbol{\Pi}^1_1), M) <_c (D_2(\boldsymbol{\Pi}^1_1), M')$$

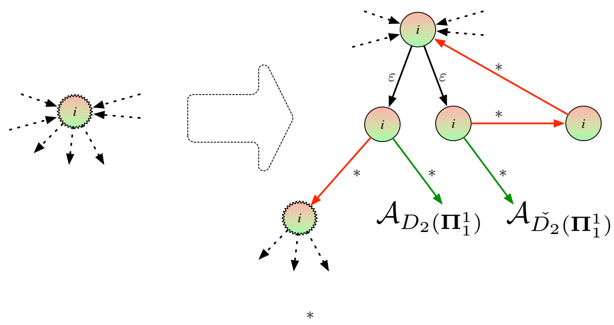


Figure 8 The gadget to replace a state in $\mathcal{A}$. The transitions that are not depicted lead to an all-accepting state $\top$.

The operation $(D_2(\mathbf{\Pi}_1^1), \cdot)$ is much stronger than $(\mathbf{\Pi}_1^0, \cdot)$, and is in fact a fixpoint of it.

▶ **Proposition 10.** *Let $M \subseteq \mathcal{T}_\Sigma^{\leq\omega}$. Then*

$$\left(\mathbf{\Pi}_1^0, (D_2(\mathbf{\Pi}_1^1), M)\right) \equiv_c (D_2(\mathbf{\Pi}_1^1), M).$$

The proof of Proposition 10 is a variant of the remote-control technique and is omitted here.

Let $\mathcal{A}$ be an automaton that recognizes $M \subseteq \mathcal{T}_\Sigma^{\leq\omega}$. Then the conciliatory tree language $(D_2(\mathbf{\Pi}_1^1), M)$ is recognized by the automaton $\boldsymbol{\varepsilon}_{\mathcal{A}}$ defined from $\mathcal{A}$ by replacing each state of $\mathcal{A}$ by a "gadget", as depicted in Figure 8. As in the pseudo-exponentiation case, by replacing a state by the gadget we mean that all transitions ending in this state should now end in the initial state of the gagdet, and that all the transitions starting from this state should now start from the final state of the gadget.

Observe that, since $A_{D_2(\mathbf{\Pi}_1^1)}$ and $A_{\check{D}_2(\mathbf{\Pi}_1^1)}$ are disjoint, if $\mathcal{A}$ is unambiguous, then $\boldsymbol{\varepsilon}_{\mathcal{A}}$ is also unambiguous, so that the operation $(D_2(\mathbf{\Pi}_1^1), \cdot)$ preserves the unambiguity of conciliatory tree languages.

5 A fragment of the unambiguous Wadge hierarchy

Consider the *epsilon function*, the ordinal function that enumerates the fixed-points of the exponentiation of base ω:

$$\varepsilon_0 = \sup_{n<\omega} \underbrace{\omega^{\cdot^{\cdot^{\omega^0}}}}_{n}; \qquad \varepsilon_{\alpha+1} = \sup_{n<\omega} \underbrace{\omega^{\cdot^{\cdot^{\omega^{(\varepsilon_\alpha+1)}}}}}_{n}; \qquad \varepsilon_\lambda = \sup_{\alpha<\lambda} \varepsilon_\alpha \text{ for } \lambda \text{ limit.}$$

We denote by $\varphi_2(0)$ its first fixed-point:

$$\varphi_2(0) = \sup_{n<\omega} \underbrace{\varepsilon_{\cdot_{\cdot_{\varepsilon_0}}}}_{n}.$$

The ordinal $\varphi_2(0)$ is the first value of the second function of the Veblen hierarchy [17]. Another way to characterise $\varphi_2(0)$ is to remember that an ordinal is the set of its predecessors and notice that a non-zero ordinal is of the form respectively ω^α if and only if it is closed under addition, and ε_α if and only if it is closed under $x \longmapsto \omega^x$. Then $\varphi_2(0)$ is the first non-zero ordinal closed under $x \longmapsto \varepsilon_x$ as well as $x \longmapsto \omega^x$ and $x, y \longmapsto x + y$.

We recall that every ordinal $\alpha > 0$ admits a unique Cantor normal form of base ω (CNF) which is an expression of the form

$$\alpha = \omega^{\alpha_k} \cdot n_k + \cdots + \omega^{\alpha_0} \cdot n_0$$

where $k < \omega$, $0 < n_i < \omega$ (any $i \leq k$) and $\alpha_0 < \cdots < \alpha_k < \alpha$.

For every ordinal $0 < \alpha < \varphi_2(0)$, we inductively define a pair of unambiguous automata $(\mathcal{A}_\alpha, \bar{\mathcal{A}}_\alpha)$ whose languages are both non-selfdual and incomparable through the conciliatory ordering. If the CNF of α is $\alpha = \omega^{\alpha_k} \cdot n_k + \cdots + \omega^{\alpha_0} \cdot n_0$ we set

$$\mathcal{A}_\alpha = \mathcal{A}_{\omega^{\alpha_k}} \bullet n_k + \cdots + \mathcal{A}_{\omega^{\alpha_0}} \bullet n_0$$

and

$$\bar{\mathcal{A}}_\alpha = \mathcal{A}_{\omega^{\alpha_k}} \bullet n_k + \cdots + \bar{\mathcal{A}}_{\omega^{\alpha_0}} \bullet n_0,$$

where $\mathcal{A}_{\omega^{\alpha_i}}$ and $\bar{\mathcal{A}}_{\omega^{\alpha_i}}$ are respectively:

- $\ominus$ and $\oplus$ if $\alpha_i = 0$;
- $\boldsymbol{\omega}^{\mathcal{A}_{\alpha_i}}$ and $\boldsymbol{\omega}^{\bar{\mathcal{A}}_{\alpha_i}}$ if $\alpha_i < \omega^{\alpha_i}$;
- $\boldsymbol{\varepsilon}_{\mathcal{A}_{2+\beta}}$ and $\boldsymbol{\varepsilon}_{\bar{\mathcal{A}}_{2+\beta}}$ if $\alpha_i = \omega^{\alpha_i}$ and $\alpha_i = \varepsilon_\beta$ for some $\beta < \alpha_i$.

Here $\oplus$ denotes automaton that accepts all conciliatory trees.

▶ **Lemma 11.** *Let* $0 < \alpha < \beta < \varphi_2(0)$,

1. $\mathcal{A}_\alpha \nleq_c \bar{\mathcal{A}}_\alpha$ *and* $\bar{\mathcal{A}}_\alpha \nleq_c \mathcal{A}_\alpha$.
2. $\mathcal{A}_\alpha <_c \mathcal{A}_\beta$; $\bar{\mathcal{A}}_\alpha <_c \mathcal{A}_\beta$; $\mathcal{A}_\alpha <_c \bar{\mathcal{A}}_\beta$ *and* $\bar{\mathcal{A}}_\alpha <_c \bar{\mathcal{A}}_\beta$.

Proof.

1. The proof, by induction on α, relies on the fact that the operations considered "commute" with each others, see Facts 4, 6 and Theorem 8.
2. The proof, by induction on α and β, relies on the fact that the operations preserve the relation $<_c$ (see Lemma 5, Facts 6 and Corollary 9) on the one hand, and on the fact that they do not "overlap" (see Propositions 7 and 10). ◀

Applying the embedding $L \mapsto L^b$, we have thus generated a family $\left(\mathcal{A}^b_\alpha\right)_{\alpha<\varphi_2(0)}$ of unambiguous automata that respects the strict Wadge ordering: $\alpha < \beta$ if and only if $\mathcal{A}^b_\alpha <_W \mathcal{A}^b_\beta$. Even though the exact Wadge rank of this family is unknown, this fragment of the $\boldsymbol{\Delta}^1_2$-Wadge hierarchy restricted to unambiguously recognizable languages climbs far above the $\boldsymbol{\Sigma}^1_1$ class. Hence the main result follows.

▶ **Theorem 12.** *There exists a family* $\left(\mathcal{A}^b_\alpha\right)_{\alpha<\varphi_2(0)}$ *of unambiguous parity tree automata whose priorities are restricted to* $\{0,1,2\}$ *such that*

1. *they recognize languages of full trees over the alphabet* $\{0,1,b\}$;
2. $\alpha < \beta$ *holds if and only if* $\mathcal{A}^b_\alpha <_W \mathcal{A}^b_\beta$ *holds as well.*

6 Conclusion

In this paper, we have produced a very long chain of unambiguous parity tree automata of different Wadge degrees. Its length, the ordinal $\varphi_2(0)$, is the first fixpoint of the ordinal function that itself enumerates all fixpoints of the ordinal exponentiation $x \mapsto \omega^x$. All these automata share a Rabin-Mostowski index of at most $(0,2)$. This indicates that the whole Wadge hierarchy of unambiguous parity tree automata is even far more complicated than that, not to mention the even higher complexity of the Wadge hierarchy of regular tree languages which, in comparison, seems scary. This illustrates in particular how different the tree-case scenario is from the word-case scenario.

The whole construction is effective. This means that the mapping $\alpha \mapsto \mathcal{A}^b_\alpha$ (for $0 < \alpha < \varphi_2(0)$) is recursive. And also that, for any $0 < \alpha < \beta < \varphi_2(0)$, the relation $\mathcal{A}^b_\alpha <_W \mathcal{A}^b_\beta$ which stipulates that there exists two strategies – one that is winning for player II in the game $W(\mathcal{A}^b_\alpha, \mathcal{A}^b_\beta)$ and another one that is winning for I in the game $W(\mathcal{A}^b_\beta, \mathcal{A}^b_\alpha)$ – can be established by recursively providing such strategies.

However, we did not consider any decidability issue. It thus remains open to show whether one can decide, given any automaton $\mathcal{B}$ and any ordinal $0 < \alpha < \varphi_2(0)$, whether $\mathcal{B} <_W \mathcal{A}^b_\alpha$ holds or not.

References

1 Alessandro Andretta and Alain Louveau. Wadge degrees and pointclasses. In Alexander S. Kechris, Benedikt Löwe, and John R. Steel, editors, *Wadge Degrees and Projective Ordinals: The Cabal Seminar, Volume II*. Cambridge University Press, 2012.

2 André Arnold. Rational omega-languages are non-ambiguous. *Theoretical Computer Science*, 26:221–223, 1983.

3 André Arnold, Jacques Duparc, Filip Murlak, and Damian Niwiński. On the topological complexity of tree languages. *Logic and automata: History and Perspectives*, 2:9–29, 2007.

4 Mikołaj Bojańczyk. A bounding quantifier. In *CSL*, pages 41–55, 2004.

5 Mikołaj Bojańczyk, Tomasz Gogacz, Henryk Michalewski, and Michał Skrzypczak. On the decidability of MSO+U on infinite trees. In *ICALP*, pages 50–61, 2014.

6 Mikołaj Bojańczyk, Paweł Parys, and Szymon Toruńczyk. The MSO+U theory of (N, <) is undecidable. *CoRR*, abs/1502.04578, 2015.

7 Arnaud Carayol and Christof Löding. MSO on the infinite binary tree: Choice and order. In *CSL*, pages 161–176, 2007.

8 Arnaud Carayol, Christof Löding, Damian Niwiński, and Igor Walukiewicz. Choice functions and well-orderings over the infinite binary tree. *Central European Journal of Mathematics*, 8:662–682, 2010.

9 Thomas Colcombet. Forms of determinism for automata (invited talk). In *STACS*, pages 1–23, 2012.

10 Jacques Duparc. Wadge hierarchy and Veblen hierarchy, Part I : Borel sets of finite rank. *The Journal of Symbolic Logic*, 66(1):56–86, 2001.

11 Jacques Duparc and Filip Murlak. On the topological complexity of weakly recognizable tree languages. In *Fundamentals of Computation Theory, 16th International Symposium, FCT 2007, Budapest, Hungary, August 27-30, 2007, Proceedings*, pages 261–273, 2007.

12 Szczepan Hummel. Unambiguous tree languages are topologically harder than deterministic ones. In *GandALF*, pages 247–260, 2012.

13 Filip Murlak. The Wadge hierarchy of deterministic tree languages. *Logical Methods in Computer Science*, 4(4), 2008.

14 Damian Niwiński and Igor Walukiewicz. Ambiguity problem for automata on infinite trees. unpublished note, 1996.

15 Michael O. Rabin. Decidability of second-order theories and automata on infinite trees. *Transactions of the AMS*, 141:1–23, 1969.

16 Robert Van Wesep. Wadge degrees and descriptive set theory. In Alexander S. Kechris, Benedikt Löwe, and John R. Steel, editors, *Wadge Degrees and Projective Ordinals: The Cabal Seminar, Volume II*. Cambridge University Press, 2012.

17 Oswald Veblen. Increasing functions of finite and transfinite ordinals. *Transactions of the American Mathematical Society*, 9(3):280–292, 1908.

18 William W. Wadge. *Reducibility and determinateness on the Baire space*. PhD thesis, University of California, Berkeley, 1984.