Theoretical Computer Science 262 (2001) 669–697

Theoretical Computer Science

www.elsevier.com/locate/tcs

Topological properties of omega context-free languages

Olivier Finkel

Equipe de Logique Mathématique, CNRS URA 753 et Université Paris 7, UFR de Mathématiques, 2 Place Jussieu 75251 Paris cedex 05, France

Received March 2000; revised October 2000; accepted October 2000
Communicated by M. Nivat

Abstract

This paper is a study of topological properties of omega context-free languages (ω-*CFL*). We first extend some decidability results for the deterministic ones (ω-*DCFL*), proving that one can decide whether an ω-*DCFL* is in a given Borel class, or in the Wadge class of a given ω-regular language. We prove that ω-*CFL* exhaust the hierarchy of Borel sets of finite rank, and that one cannot decide the borel class of an ω-*CFL*, giving an answer to a question of Lescow and Thomas (A Decade of Concurrency, Springer Lecture Notes in Computer Science, vol. 803, Springer, Berlin, 1994, pp. 583–621). We give also a (partial) answer to a question of Simmonet (Automates et théorie descriptive, Ph.D. Thesis, Université Paris 7, March 1992) about omega powers of finitary languages. We show that Büchi–Landweber's Theorem cannot be extended to even closed ω-*CFL*: in a Gale–Stewart game with a (closed) ω-*CFL* winning set, one cannot decide which player has a winning strategy. From the proof of topological properties we derive some arithmetical properties of ω-*CFL*.

Keywords: Omega context-free languages; Topological properties; Wadge class; Borel class; Decide the Borel class; Omega powers; Gale–Stewart game; Arithmetical properties

1. Introduction

Since Büchi studied the ω-languages recognized by finite automata to prove the decidability of the monadic second order theory of one successor over the integers [5] the so-called ω-regular languages have been intensively studied. See [44, 37] for much results and references.

E-mail address: finkel@logique.jussieu.fr (O. Finkel).

PII: S0304-3975(00)00405-9

As Pushdown automata are a natural extension of finite automata, Cohen and Gold [9, 10] and Linna [30] studied the ω-languages accepted by omega pushdown automata, considering various acceptance conditions for omega words. It turned out that the omega languages accepted by omega pushdown automata were also those generated by context-free grammars where infinite derivations are considered, also studied by Nivat [34, 35] and Boasson and Nivat [4]. These languages were then called the omega context-free languages (ω-*CFL*). See also Staiger's paper [41] for a survey of general theory of ω-languages, including more powerful accepting devices, like Turing machines.

MacNaughton's Theorem implies that ω-regular languages are boolean combination of G_δ sets [32]. Topological properties of ω-regular languages were first studied by Landweber in [28] where he showed that one can decide whether a given ω-regular language is in a given Borel class.

The hierarchy induced on ω-regular languages by the Borel Hierarchy was refined in [2, 26] but Wagner had found the most refined one, now called the Wagner hierarchy and which is the hierarchy induced on ω-regular languages by the Wadge Hierarchy of Borel sets [47].

This paper is mainly a study of similar results for ω-context free languages.

In Sections 2 and 3, we first review some above definitions and results about ω-regular, ω-context free languages, and topology.

In Section 4, we study the ω-languages accepted by deterministic omega pushdown automata, called the omega deterministic context-free languages (ω-*DCFL*). Walukiewicz proved in [48] that in a Gale–Stewart game with an ω-*DCFL* as winning set, one can decide which player has a winning strategy. We give a new proof, based on this result, that one can decide whether an ω-*DCFL* is in a given Borel class, which leads to a much stronger result: one can decide whether an effectively given ω-*DCFL* is in the Wadge class of an effectively given ω-regular language.

In Sections 5–7, we next study the class of ω-*CFL*. We first restate some previous undecidability results. Then we prove that there are ω-*CFL* in each Borel class of finite order. And that, for any class $\boldsymbol{\Sigma}_n^0$ or $\boldsymbol{\Pi}_n^0$, n being an integer, one cannot decide whether an ω-*CFL* is in $\boldsymbol{\Sigma}_n^0$ or $\boldsymbol{\Pi}_n^0$. Our proofs rely on the recent work of Duparc about the Wadge hierarchy of Borel sets [15]. We use the Wadge game [46], and the operation of exponentiation of sets defined by Duparc [15].

These results give partial answers to questions of Thomas and Lescow [29].

In Section 8, we study ω-powers of finitary languages. The ω-power of a language $W \subseteq X^\star$ is a fundamental operation over finitary languages which leads to ω-languages. Whenever W is a regular language (respectively a context-free language), then W^ω is an ω-regular language, (respectively an ω-*CFL*). Then the question of the topological complexity of W^ω naturally arises and it is posed in [36, 39, 41, 42]. When W is a regular language, W^ω is a boolean combination of G_δ sets because it is an ω-regular set. We prove results on omega powers of finitary context-free languages, giving examples of context free languages (L_n) such that $(L_n)^\omega$ is a Borel set of finite rank n for every integer $n \geqslant 1$.

In Section 9, we consider Gale–Stewart games and we prove that Büchi–Landweber Theorem cannot extend to ω-*CFL*: in Gale–Stewart games with closed ω-*CFL* winning set, one cannot decide which player has a winning stategy.

In Section 10, we derive some arithmetical properties of omega context free languages from the preceding topological properties. We prove that one cannot decide whether an ω-*CFL* is in the arithmetical class Σ_n or Π_n, for an integer $n \geqslant 1$. Then we show that one cannot decide whether an ω-*CFL* is accepted by a deterministic Turing machine (or more generally by a deterministic **X**-automaton as defined in [19]) with Büchi (respectively Muller) acceptance condition.

2. ω-regular and ω-context-free languages

We assume the reader to be familiar with the theory of formal languages and of ω-regular languages, see for example [25, 44]. We first recall some of the definitions and results concerning ω-regular and ω-context-free languages and omega pushdown automata as presented in [44, 9, 10].

When Σ is a finite alphabet, a finite string (word) over Σ is any sequence $x = x_1 \ldots x_k$, where $x_i \in \Sigma$ for $i = 1, \ldots, k$, and k is an integer $\geqslant 1$. The length of x is k, denoted by $|x|$.

If $|x| = 0$, x is the empty word denoted by λ.

We write $x(i) = x_i$ and $x[i] = x(1) \ldots x(i)$ for $i \leqslant k$ and $x[0] = \lambda$.

$\Sigma^\star$ is the set of finite words over Σ.

The first infinite ordinal is ω.

An ω-word over Σ is an ω-sequence $a_1 \ldots a_n \ldots$, where $a_i \in \Sigma$, $\forall i \geqslant 1$.

When σ is an ω-word over Σ, we write $\sigma = \sigma(1)\sigma(2) \ldots \sigma(n) \ldots$

and $\sigma[n] = \sigma(1)\sigma(2) \ldots \sigma(n)$ the finite word of length n, prefix of σ.

The set of ω-words over the alphabet Σ is denoted by Σ^ω.

An ω-language over an alphabet Σ is a subset of Σ^ω.

The usual concatenation product of two finite words u and v is denoted $u.v$ (and sometimes just uv). This product is extended to the product of a finite word u and an ω-word v: the infinite word $u.v$ is then the ω-word such that:

$$(u.v)(k) = u(k) \text{ if } k \leqslant |u|,$$

and

$$(u.v)(k) = v(k - |u|) \text{ if } k > |u|.$$

For $V \subseteq \Sigma^\star$, $V^\omega = \{\sigma = u_1 \ldots u_n \ldots \in \Sigma^\omega / u_i \in V, \ \forall i \geqslant 1\}$ is the ω-power of V.

For $V \subseteq \Sigma^\star$, the complement of V (in $\Sigma^\star$) is $\Sigma^\star - V$ denoted V^-.

For a subset $A \subseteq \Sigma^\omega$, the complement of A is $\Sigma^\omega - A$ denoted A^-.

The prefix relation is denoted $\sqsubseteq$: the finite word u is a prefix of the finite word v (denoted $u \sqsubseteq v$) if and only if there exists a (finite) word w such that $v = u.w$.

This definition is extended to finite words which are prefixes of ω-words:

the finite word u is a prefix of the ω-word v (denoted $u \sqsubseteq v$) iff there exists an ω-word w such that $v = u.w$.

Definition 2.1. A finite state machine (*FSM*) is a quadruple $M = (K, \Sigma, \delta, q_0)$, where K is a finite set of states, Σ is a finite input alphabet, $q_0 \in K$ is the initial state and δ is a mapping from $K \times \Sigma$ into 2^K. A *FSM* is called deterministic (*DFSM*) iff: $\delta : K \times \Sigma \to K$.

A Büchi automaton (*BA*) is a 5-tuple $M = (K, \Sigma, \delta, q_0, F)$ where $M' = (K, \Sigma, \delta, q_0)$ is a finite state machine and $F \subseteq K$ is the set of final states. A Muller automaton (*MA*) is a 5-tuple $M = (K, \Sigma, \delta, q_0, F)$ where $M' = (K, \Sigma, \delta, q_0)$ is a *FSM* and $F \subseteq 2^K$ is the collection of designated state sets. A Büchi or Muller automaton is said deterministic if the associated *FSM* is deterministic.

Let $\sigma = a_1 a_2 \ldots a_n \ldots$ be an ω-word over Σ.

A sequence of states $r = q_1 q_2 \ldots q_n \ldots$ is called an (infinite) run of $M = (K, \Sigma, \delta, q_0)$ on σ, starting in state p, iff: (1) $q_1 = p$ and (2) for each $i \geqslant 1$, $q_{i+1} \in \delta(q_i, a_i)$.

In case a run r of M on σ starts in state q_0, we call it simply "a run of M on σ".

For every (infinite) run $r = q_1 q_2 \ldots q_n \ldots$ of M, $In(r)$ is the set of states in K entered by M infinitely many times during run r:

$In(r) = \{q \in K / \{i \geqslant 1 / q_i = q\}$ is infinite$\}$.

For $M = (K, \Sigma, \delta, q_0, F)$ a *BA*, the ω-language accepted by M is $L(M) = \{\sigma \in \Sigma^\omega /$there exists a run r of M on σ such that $In(r) \cap F \neq \emptyset\}$.

For $M = (K, \Sigma, \delta, q_0, F)$ a *MA*, the ω-language accepted by M is $L(M) = \{\sigma \in \Sigma^\omega /$ there exists a run r of M on σ such that $In(r) \in F\}$.

The classical result of MacNaughton [32] established that the expressive power of deterministic *MA* (*DMA*) is equal to the expressive power of non-deterministic *MA* (*NDMA*) which is also equal to the expressive power of non-deterministic *BA* (*NDBA*).

There is also a characterization of the languages accepted by *MA* by means of the "ω-Kleene closure" which we give now the definition:

Definition 2.2. For any family L of finitary languages over the alphabet Σ, the ω-Kleene closure of L, is

$$\omega\text{-}KC(L) = \left\{ \bigcup_{i=1}^{n} U_i . V_i^\omega / U_i, V_i \in L, \forall i \in [1, n] \right\}.$$

Theorem 2.3. *For any ω-language L, the following conditions are equivalent*:

(1) *L belongs to ω-$KC(REG)$, where REG is the class of (finitary) regular languages.*

(2) *There exists a DMA that accepts L.*

(3) *There exists a MA that accepts L.*

(4) *There exists a BA that accepts L.*

An ω-language L satisfying one of the conditions of the above theorem is called an ω-regular language. The class of ω-regular languages will be denoted by REG_ω.

We now define the pushdown machines and the classes of ω-context-free languages.

Definition 2.4. A pushdown machine (PDM) is a 6-tuple $M=(K,\Sigma,\Gamma,\delta,q_0,Z_0)$, where K is a finite set of states, Σ is a finite input alphabet, Γ is a finite pushdown alphabet, $q_0\in K$ is the initial state, $Z_0\in\Gamma$ is the start symbol, and δ is a mapping from $K\times(\Sigma\cup\{\lambda\})\times\Gamma$ to finite subsets of $K\times\Gamma^\star$.

If $\gamma\in\Gamma^+$ describes the pushdown store content, the leftmost symbol will be assumed to be on "top" of the store. A configuration of a PDM is a pair (q,γ) where $q\in K$ and $\gamma\in\Gamma^\star$.

For $a\in\Sigma\cup\{\lambda\}$, $\gamma,\beta\in\Gamma^\star$ and $Z\in\Gamma$, if (p,β) is in $\delta(q,a,Z)$, then we write $a:(q,Z\gamma)\mapsto_M (p,\beta\gamma)$.

$\mapsto_M^\star$ is the transitive and reflexive closure of $\mapsto_M$. (The subscript M will be omitted whenever the meaning remains clear).

Let $\sigma=a_1a_2\ldots a_n\ldots$ be an ω-word over Σ. An infinite sequence of configurations $r=(q_i,\gamma_i)_{i\geqslant 1}$ is called a run of M on σ, starting in configuration (p,γ), iff:

(1) $(q_1,\gamma_1)=(p,\gamma)$.

(2) For each $i\geqslant 1$, there exists $b_i\in\Sigma\cup\{\lambda\}$ satisfying $b_i:(q_i,\gamma_i)\mapsto_M(q_{i+1},\gamma_{i+1})$ such that either $a_1a_2\ldots a_n\ldots=b_1b_2\ldots b_n\ldots$ or $b_1b_2\ldots b_n\ldots$ is a finite prefix of $a_1a_2\ldots a_n\ldots$

The run r is said to be complete when $a_1a_2\ldots a_n\ldots=b_1b_2\ldots b_n\ldots$

As for FSM, for every such run, $In(r)$ is the set of all states entered infinitely often during run r.

A complete run r of M on σ, starting in configuration (q_0,Z_0), will be simply called "a run of M on σ".

Definition 2.5. A Büchi pushdown automaton ($BPDA$) is a 7-tuple $M=(K,\Sigma,\Gamma,\delta,q_0,Z_0,F)$ where $M'=(K,\Sigma,\Gamma,\delta,q_0,Z_0)$ is a PDM and $F\subseteq K$ is the set of final states.

The ω-language accepted by M is $L(M)=\{\sigma\in\Sigma^\omega/$there exists a complete run r of M on σ such that $In(r)\cap F\neq\emptyset\}$.

Definition 2.6. A Muller pushdown automaton ($MPDA$) is a 7-tuple $M=(K,\Sigma,\Gamma,\delta,q_0,Z_0,F)$ where $M'=(K,\Sigma,\Gamma,\delta,q_0,Z_0)$ is a PDM and $F\subseteq 2^K$ is the collection of designated state sets.

The ω-language accepted by M is $L(M)=\{\sigma\in\Sigma^\omega/$there exists a complete run r of M on σ such that $In(r)\in F\}$.

Remark 2.7. We consider here two acceptance conditions for ω-words, the Büchi and the Muller acceptance conditions, respectively, denoted 2-acceptance and 3-acceptance in [28] and in [10] and $(\inf,\sqcap)$ and $(\inf,=)$ in [41].

Cohen and Gold, and independently Linna, established a characterization theorem for ω-CFL:

Theorem 2.8. *Let CFL be the class of context-free* (*finitary*) *languages. Then for any* ω*-language L the following three conditions are equivalent*:
(1) $L \in \omega\text{-}KC(CFL)$.
(2) *There exists a BPDA that accepts L.*
(3) *There exists a MPDA that accepts L.*

In [9] are also studied the ω-languages generated by ω-context free grammars and it is shown that each of the conditions (1)–(3) of the above theorem is also equivalent to: (4) L is generated by a context free grammar G by leftmost derivations. These grammars are also studied in [34, 35].

Then we can let the following definition:

Definition 2.9. An ω-language is an ω-context free language (ω-*CFL*) iff it satisfies one of the conditions of the above theorem.

Unlike the case of finite automata, deterministic *MPDA* do not define the same class of ω-languages as non deterministic *MPDA*. Let us now define deterministic pushdown machines.

Definition 2.10. A *PDM* $M = (K, \Sigma, \Gamma, \delta, q_0, Z_0)$ is said deterministic (*DPDM*) iff for each $q \in K$, $Z \in \Gamma$, and $a \in \Sigma$:
1. $\delta(q, a, Z)$ contains at most one element,
2. $\delta(q, \lambda, Z)$ contains at most one element, and
3. if $\delta(q, \lambda, Z)$ is non-empty, then $\delta(q, a, Z)$ is empty for all $a \in \Sigma$.

It turned out that the class of ω-languages accepted by deterministic *BPDA* is strictly included into the class of ω-languages accepted by deterministic *MPDA*. Let us denote $DCFL_\omega$ this latest class, the class of omega deterministic context free languages (ω-*DCFL*), and *DCFL* the class of deterministic context free (finitary) languages. Then recall the following:

Proposition 2.11. 1. $DCFL_\omega$ *is closed under complementation, but not under union, neither under intersection.*
2. $DCFL_\omega \subsetneq \omega\text{-}KC(DCFL) \subsetneq CFL_\omega$ (*these inclusions are strict*).

Remark 2.12. If M is a deterministic pushdown machine, then for every $\sigma \in \Sigma^\omega$, there exists at most one run r of M on σ determined by the starting configuration. The *DPDM* has the continuity property iff for every $\sigma \in \Sigma^\omega$, there exists a unique run of M on σ and this run is complete. It is shown in [10] that each ω-language accepted by a *DBPDA* (respectively *DMPDA*) can be accepted by a *DBPDA* (respectively *DMPDA*) with the continuity property. So we shall assume now that all *DPDA* have the continuity property.

3. Topology

We assume the reader to be familiar with basic notions of topology which may be found in [29, 27].

Topology is an important tool for the study of ω-languages, and leads to characterization of several classes of ω-languages.

For a finite alphabet X, we consider X^ω as a topological space with the Cantor topology. The open sets of X^ω are the sets in the form $W.X^\omega$, where $W \subseteq X^\star$. A set $L \subseteq X^\omega$ is a closed set iff its complement $X^\omega - L$ is an open set. The class of open sets of X^ω will be denoted by $\mathbf{G}$ or by $\boldsymbol{\Sigma}^0_1$. The class of closed sets will be denoted by $\mathbf{F}$ or by $\boldsymbol{\Pi}^0_1$. Closed sets are characterized by the following:

Proposition 3.1. *A set $L \subseteq X^\omega$ is a closed set of X^ω iff for every $\sigma \in X^\omega$, $[\forall n \geqslant 1, \exists u \in X^\omega$ such that $\sigma(1)\ldots\sigma(n).u \in L]$ implies that $\sigma \in L$.*

Define now the next classes of the Borel Hierarchy:

Definition 3.2. The classes $\boldsymbol{\Sigma}^0_n$ and $\boldsymbol{\Pi}^0_n$ of the Borel Hierarchy on the topological space X^ω are defined as follows:

$\boldsymbol{\Sigma}^0_1$ is the class of open sets of X^ω.
$\boldsymbol{\Pi}^0_1$ is the class of closed sets of X^ω.
$\boldsymbol{\Pi}^0_2$ or $\mathbf{G}_\delta$ is the class of countable intersections of open sets of X^ω.
$\boldsymbol{\Sigma}^0_2$ or $\mathbf{F}_\sigma$ is the class of countable unions of closed sets of X^ω.
$\boldsymbol{\Sigma}^0_3$ or $\mathbf{G}_{\delta\sigma}$ is the class of countable unions of $\boldsymbol{\Pi}^0_2$-subsets of X^ω.
$\boldsymbol{\Pi}^0_3$ or $\mathbf{F}_{\sigma\delta}$ is the class of countable intersections of $\boldsymbol{\Sigma}^0_2$ subsets of X^ω.
And for any integer $n \geqslant 1$:
$\boldsymbol{\Sigma}^0_{n+1}$ is the class of countable unions of $\boldsymbol{\Pi}^0_n$-subsets of X^ω.
$\boldsymbol{\Pi}^0_{n+1}$ is the class of countable intersections of $\boldsymbol{\Sigma}^0_n$-subsets of X^ω.

Recall some basic results about these classes:

Proposition 3.3 (Moschovakis [33]). (a) *$\boldsymbol{\Sigma}^0_n \cup \boldsymbol{\Pi}^0_n \subsetneq \boldsymbol{\Sigma}^0_{n+1} \cap \boldsymbol{\Pi}^0_{n+1}$, for each integer $n \geqslant 1$.*

(b) *A set $W \subseteq X^\omega$ is in the class $\boldsymbol{\Sigma}^0_n$ if and only if its complement W^- is in the class $\boldsymbol{\Pi}^0_n$.*

(c) *$\boldsymbol{\Sigma}^0_n - \boldsymbol{\Pi}^0_n \neq \emptyset$ and $\boldsymbol{\Pi}^0_n - \boldsymbol{\Sigma}^0_n \neq \emptyset$ hold for each integer $n \geqslant 1$.*

We shall say that a subset of X^ω is a Borel set of rank 1 iff it is in $\boldsymbol{\Sigma}^0_1 \cup \boldsymbol{\Pi}^0_1$ and that it is a Borel set of rank $n \geqslant 2$ iff it is in $\boldsymbol{\Sigma}^0_n \cup \boldsymbol{\Pi}^0_n$ but not in $\boldsymbol{\Sigma}^0_{n-1} \cup \boldsymbol{\Pi}^0_{n-1}$.

Remark 3.4. The hierarchy defined above is the hierarchy of Borel sets of finite rank. The Borel Hierarchy is also defined for transfinite levels but this will not be useful in the sequel. It may be found in [33, 29].

There is a nice characterization of $\boldsymbol{\Pi}^0_2$-subsets of X^ω. First define the notion of W^δ:

Definition 3.5. For $W \subseteq X^\star$, let: $W^\delta = \{\sigma \in X^\omega / \exists^\omega i$ such that $\sigma[i] \in W\}$. ($\sigma \in W^\delta$ iff σ has infinitely many prefixes in W).

Then we can state the following proposition:

Proposition 3.6. *A subset L of X^ω is a $\boldsymbol{\Pi}_2^0$-subset of X^ω iff there exists a set $W \subseteq X^\star$ such that $L = W^\delta$.*

Landweber studied first the topological properties of ω-regular languages. He proved that every ω-regular language is a boolean combination of G_δ-sets and he also characterized the ω-regular languages in each of the Borel classes $\mathbf{F}, \mathbf{G}, \mathbf{F}_\sigma, \mathbf{G}_\delta$, and showed that one can decide, for an effectively given ω-regular language L, whether L is in $\mathbf{F}, \mathbf{G}, \mathbf{F}_\sigma$, or $\mathbf{G}_\delta$.

It turned out that an ω-regular language is in the class $\mathbf{G}_\delta$ iff it is accepted by a *DBA*.

Introduce now the Wadge Hierarchy:

Definition 3.7. For $E \subseteq X^\omega$ and $F \subseteq Y^\omega$, E is said Wadge reducible to F ($E \leqslant_W F$) iff there exists a continuous function $f : X^\omega \to Y^\omega$, such that $E = f^{-1}(F)$.

E and F are Wadge equivalent iff $E \leqslant_W F$ and $F \leqslant_W E$. This will be denoted by $E \equiv_W F$. And we shall say that $E <_W F$ iff $E \leqslant_W F$ but not $F \leqslant_W E$.

The relation $\leqslant_W$ is reflexive and transitive, and $\equiv_W$ is an equivalence relation.

The equivalence classes of $\equiv_W$ are called wadge degrees.

WH is the class of Borel subsets of finite rank of a set X^ω, where X is a finite set, equipped with $\leqslant_W$ and with $\equiv_W$.

For $E \subseteq X^\omega$ and $F \subseteq Y^\omega$, if $E \leqslant_W F$ and $E = f^{-1}(F)$ where f is a continuous function from X^ω into Y^ω, then f is called a continuous reduction of E to F. Intuitively it means that E is less complicated than F because to check whether $x \in E$ it suffices to check whether $f(x) \in F$ where f is a continuous function. Hence the Wadge degree of an ω-language, which will be precisely defined below, is a measure of its topological complexity.

Remark 3.8. In the above definition, we consider that a subset $E \subseteq X^\omega$ is given together with the alphabet X.

Then, we can define the Wadge class of a set F:

Definition 3.9. Let F be a subset of X^ω. The wadge class of F is $[F]$ defined by: $[F] = \{E / E \subseteq Y^\omega$ for a finite alphabet Y and $E \leqslant_W F\}$.

Recall that each Borel class $\boldsymbol{\Sigma}_n^0$ and $\boldsymbol{\Pi}_n^0$ is a Wadge class.

And that a set $F \subseteq X^\omega$ is a $\boldsymbol{\Sigma}_n^0$ (respectively $\boldsymbol{\Pi}_n^0$)-complete set iff for any set $E \subseteq Y^\omega$, E is in $\boldsymbol{\Sigma}_n^0$ (respectively $\boldsymbol{\Pi}_n^0$) iff $E \leqslant_W F$.

$\boldsymbol{\Sigma}_n^0$ (respectively $\boldsymbol{\Pi}_n^0$)-complete sets are thoroughly characterized in [40].

There is a close relationship between Wadge reducibility and games which we now introduce. Define first the Wadge game $W(A,B)$ for $A \subseteq X_A^\omega$ and $B \subseteq X_B^\omega$:

Definition 3.10. The Wadge game $W(A,B)$ is a game with perfect information between two players, player 1 who is in charge of A and player 2 who is in charge of B.

Player 1 first writes a letter $a_1 \in X_A$, then player 2 writes a letter $b_1 \in X_B$, then player 1 writes a letter $a_2 \in X_A$, and so on ...

The two players alternatively write letters a_n of X_A for player 1 and b_n of X_B for player 2.

After ω steps, the player 1 has written an ω-word $a \in X_A^\omega$ and player 2 has written an ω-word $b \in X_B^\omega$.

Player 2 is allowed to skip, even infinitely often, provided he really write an ω-word in ω steps.

The player 2 wins the play iff $[a \in A \leftrightarrow b \in B]$, i.e. iff $[(a \in A$ *and* $b \in B)$ or $(a \notin A$ *and* $b \notin B$ *and* b *is infinite*$)]$.

Recall that a strategy for player 1 is a function $\sigma : (X_B \cup \{s\})^\star \to X_A$. And a strategy for player 2 is a function $f : X_A^+ \to X_B \cup \{s\}$.

σ is a winning strategy (w.s.) for player 1 iff he always wins a play when he uses the strategy σ, i.e. when the nth letter he writes is given by $a_n = \sigma(b_1 \dots b_{n-1})$, where b_i is the letter written by player 2 at step i and $b_i = s$ if player 2 skips at step i.

A winning strategy for player 2 is defined in a similar manner.

Martin's Theorem states that every Gale–Stewart Game $G(X)$ (see Section 9 below for more details), with X a borel set, is determined and this implies the following:

Theorem 3.11 (Wadge). *Let $A \subseteq X_A^\omega$ and $B \subseteq X_B^\omega$ be two Borel sets, where X_A and X_B are at most countable alphabets. Then the Wadge game $W(A,B)$ is determined: one of the two players has a winning strategy. And $A \leqslant_W B$ iff the player 2 has a winning strategy in the game $W(A,B)$.*

Recall that a set X is well ordered by a binary relation $<$ iff $<$ is a linear order on X and there is not any strictly decreasing (for $<$) infinite sequence of elements in X.

Theorem 3.12 (Wadge). *Up to the complement and $\equiv_W$, the class of Borel subsets of finite rank of X^ω, for X a finite alphabet, is a well ordered hierarchy. There is an ordinal $|WH|$, called the length of the hierarchy, and a map d_W^0 from WH onto $|WH|$, such that for all $A, B \in WH$:*

$$d_W^0 A < d_W^0 B \leftrightarrow A <_W B$$

and

$$d_W^0 A = d_W^0 B \leftrightarrow [A \equiv_W B \text{ or } A \equiv_W B^-].$$

Remark 3.13. We do not give here the ordinal $|WH|$. Details may be found in [15].

It is natural to ask for the restriction of the Wadge Hierarchy to ω-regular languages. In fact there is an effective version of the Wadge Hierarchy restricted to ω-regular languages:

Theorem 3.14 (Corollary of Büchi–Landweber's Theorem [7]). *For A and B some ω-regular sets, one can effectively decide which player has a w.s. in the game $W(A,B)$ and the winner has a w.s. given by a transducer.*

The hierarchy obtained on ω-regular languages is now called the Wagner hierarchy and has length ω^ω. Wagner [47] gave an automata structure characterization, based on notion of chain and superchain, for an automaton to be in a given class. And one can also compute the Wadge degree of any ω-regular language. Wilke and Yoo proved in [49] that this can be done in polynomial time. The Wagner hierarchy is also recently studied in [11, 12, 38].

4. Deterministic omega context-free languages

We now study topological properties of the languages in $DCFL_\omega$. These are boolean combination of G_δ sets. Cohen and Gold proved that one can decide whether an effectively given ω-$DCFL$ in an open or a closed set [9]. Linna characterized the ω-languages accepted by $DBPDA$ as the G_δ languages in $DCFL_\omega$ and proved in [31] that one can decide whether an effectively given ω-$DCFL$ is a G_δ or a F_σ set.

We give an essentially different proof of these results, which is heavily based on a recent result of Walukiewicz [48], and which leads to a much finer result:

Not only one can decide whether an effectively given ω-$DCFL$ A (given by a $DMPDA$ accepting A) is in the Borel class $\mathbf{F}, \mathbf{G}, \mathbf{G}_\delta$ or $\mathbf{F}_\sigma$, but one can decide whether A is in the wadge class of any ω-regular language B.

Let $A \subseteq X_A^\omega$ be an ω-$DCFL$ given by a $DMPDA$ $\mathscr{A} = (Q_A, X_A, \Gamma_A, \delta_A, q_{A,0}, Z_{A,0}, F_A)$ and $B \subseteq X_B^\omega$ be an ω-regular language accepted by a DMA $\mathscr{B} = (Q_B, X_B, \delta_B, q_{B,0}, F_B)$.

We define the set $C \subseteq (X_A \cup X_B \cup \{s\})^\omega$ by:

For $\sigma = a_1 b_1 a_2 b_2 a_3 b_3 \ldots \in (X_A \cup X_B \cup \{s\})^\omega$, let $a = a_1 a_2 a_3 \ldots$ and $b = b_1 b_2 b_3 \ldots$ and $(b/s) = b$ where letters s are removed.

Then $\sigma \in C$ iff $[(a \notin A$ or $(b/s) \notin B)$ and $(a \in A$ or (b/s) is a finite word or $(b/s) \in B)]$.

And $\sigma \notin C$ iff $[(a \in A$ and $(b/s) \in B)$ or $(a \notin A$ and (b/s) is an infinite word and $(b/s) \notin B)]$.

The Gale–Stewart game $G(C)$ is defined as followed:

Player 1 writes a letter $a_1 \in X_A$, then player 2 writes a letter $b_1 \in X_B \cup \{s\}$ (s for skip), then player 1 writes a letter $a_2 \in X_A$, and player 2 writes a letter $b_2 \in X_B \cup \{s\} \ldots$

After ω steps, the two players have composed an infinite word:

$\sigma \in (X_A \cup X_B \cup \{s\})^\omega$, and player 2 wins the play iff $\sigma \notin C$.

We easily see that player 2 has a w.s. in the Wadge game $W(A,B)$ iff he has a w.s. in the game $G(C)$.

It is now easy to show that C is accepted by a deterministic Muller pushdown automaton $\mathscr{C}$: $\mathscr{C}$ is essentially the product of the two machines $\mathscr{A}$ and $\mathscr{B}$, where suitable accepting conditions are chosen. (The exact definition of these conditions is left to the reader).

In a recent paper, Walukiewicz proved that in a pushdown game, one of the two players has a w.s., given by a pushdown transducer which is effectively constructible [48]. He considered pushdown games where each of the two players alternatively plays a move in the graph of configurations of a deterministic pushdown automaton.

This result implies that in our game $G(C)$, one of the two players has a w.s. and this strategy is effectively constructible.

So we can decide which player has a w.s. in the game $W(A,B)$, i.e. whether $A \leqslant_W B$.

With a similar construction, we see that one can decide whether $B \leqslant_W A$.

When this result is applied with B a $\boldsymbol{\Pi}_2^0$ complete ω-regular set, for example, $B = \{\sigma \in \{0,1\}^\omega /$ there are infinitely many letters 1 in $\sigma\}$ [40], one can decide whether an effectively given ω-*DCFL* is a $\mathbf{G}_\delta$ set.

The same result holds for the classes $\mathbf{F}, \mathbf{G}, \mathbf{F}_\sigma$, because there are some $\boldsymbol{\Pi}_1^0$ (respectively $\boldsymbol{\Sigma}_1^0, \boldsymbol{\Sigma}_2^0$)-complete ω-regular sets [40].

But when we apply this result to an ω-regular set B, we can decide whether A is in the *wadge class* of B. And we can decide whether $B \equiv_W A$, because we can decide whether $A \leqslant_W B$ and $B \leqslant_W A$.

5. Decision problems for ω-*CFL*

We shall say that an ω-*CFL* A is effectively given when a *MPDA* accepting A is given.

We shall say that an ω-*DCFL* A is effectively given when a *DMPDA* accepting A is given.

We now state some undecidability results. Remark that some of these results are not new, but we shall reprove them in order to rely on this proof in the sequel.

Theorem 5.1. *Let Σ be an alphabet having at least two letters. It is undecidable, for some effectively given ω-context-free languages A and B in Σ^ω, whether*:

(1) *$A \cap B$ is empty.*

(2) *$A \cap B$ is infinite.*

(3) *$A \cap B$ has the continuum power.*

(4) *A^- is infinite.*

(5) *A^- has the continuum power.*

(6) *$A = \Sigma^\omega$.*

Proof. Let us first return to the *Post correspondence Theorem*:

Theorem 5.2. *Let Γ be an alphabet having at least two elements. Then it is undecidable to determine, for arbitrary n-tuples $(x_1,\ldots,x_n)$ and $(y_1,\ldots,y_n)$ of non-empty words in $\Gamma^\star$, whether there exists a non-empty sequence of indices $i_1,\ldots,i_k$ such that $x_{i_1}\ldots x_{i_k}=y_{i_1}\ldots y_{i_k}$.*

Let then $\Gamma=\{a,b\}$ and x,y some n-tuples of non-empty words of $\Gamma^\star$.

Let $L(x).c=L_X=\{ba^{i_k}\ldots ba^{i_1}cx_{i_1}\ldots x_{i_k}c/k\geqslant 1, 1\leqslant i_j\leqslant n$ for $j\in[1,k]\}$, where $c\notin\Gamma$ and $\Sigma=\Gamma\cup\{c\}$.

It is well known that $L(x)$ is a deterministic context free language [24], and so is L_X. Then L_X^ω is an ω-*CFL*, and L_Y^ω is also an ω-*CFL*.

Remark that L_X^ω is a G_δ-subset of Σ^ω, because $L_X^\omega=\bigcap_{n\geqslant 1}L_X^n.\Sigma^\omega$.

$L_X^\omega\cap L_Y^\omega$ is non-empty iff $[\exists i_1,\ldots,i_k$, with $k\geqslant 1$, such that $x_{i_1}\ldots x_{i_k}=y_{i_1}\ldots y_{i_k}]$ iff $[L_X^\omega\cap L_Y^\omega$ is infinite$]$ iff $[L_X^\omega\cap L_Y^\omega$ has the continuum power$]$.

These assertions are undecidable, because of the Post Theorem.

Let τ be the morphism $\{a,b,c\}^\star\to\{a,b\}^\star$ defined by: $a\to bab$, $b\to ba^2b$, $c\to ba^3b$.

τ transforms L_X^ω into an ω-*CFL* $\tau(L_X^\omega)=[\tau(L_X)]^\omega$, because the family *CFL* is closed under morphism.

$\tau(L_X^\omega)$ codes with two letters the language L_X^ω. And the same holds for L_Y^ω. Then (1)–(3) of the theorem are proved.

To prove (4), now consider the ω-language $\Sigma^\omega-(L_X^\omega\cap L_Y^\omega)=(L_X^\omega)^-\cup(L_Y^\omega)^-$. We shall first prove that $(L_X^\omega)^-=\Sigma^\omega-(L_X^\omega)$ is an ω-*CFL*. For that prove the following:

Lemma 5.3 (Ginsburg [24]). *With the above notations $L=L(x)^-\cap\{a,b\}^\star c\{a,b\}^\star$ is a context free language.*

This language L is the union of L_1 and L_2, where $L_1=L_1'.c.\{a,b\}^\star$, L_1' being the complement in $\{a,b\}^\star$ of $\{ba^{i_k}\ldots ba^{i_1}/k\geqslant 1, 1\leqslant i_j\leqslant n\}$.

This set is rational, therefore L_1' is also rational and by concatenation product, L_1 is also rational.

Now define L_2, following [24]:

For each non-empty word $w\in\{a,b\}^\star$, we define:

$D(w)=\{u\neq\lambda/u\in\{a,b\}^\star, |u|<|w|\}$, and

$J(w)=\{u\neq\lambda/u\in\{a,b\}^\star, u\neq w$ and $|u|=|w|\}$

Then for each n-tuple $w=(w_1,\ldots,w_n)$ of non-empty words, we let:

$M(w)=c\{a,b\}^\star\{a,b\}\cup b\{a,b\}^\star c\cup\bigcup_{1\leqslant i\leqslant n}\bigcup_{u\in D(w_i)}(ba^icu\cup ba^ib\{a,b\}^\star cu)\cup\bigcup_{1\leqslant i\leqslant n}\bigcup_{u\in J(w_i)}(ba^ic\{a,b\}^\star u\cup ba^ib\{a,b\}^\star c\{a,b\}^\star u)$.

$M(x)$ is a rational language, because each set $D(w_i)$ and $J(w_i)$ is finite.

Let the substitution $h:\{a,b,c\}^\star\to P(\{a,b,c\}^\star)$ defined by $a\to\{a\}$, $b\to\{b\}$, and $c\to M(x)$, and let $L_2=h(L(x))$.

The class *CFL* being closed under substitution, $L(x)$ and $M(x)$, (and also $\{a\}$ and $\{b\}$) being *CFL*, the language L_2 is context free.

L_2 contains $\{ba^{i_k}\dots ba^{i_1}cw/w\neq x_{i_1}\dots x_{i_k},\ \ w\in\{a,b\}^\star\}$, and $L_2\subseteq L(x)^-\cap\{a,b\}^\star c\{a,b\}^\star$. Then $L_1\cup L_2=L(x)^-\cap\{a,b\}^\star c\{a,b\}^\star$, and this language, union of two *CFL*, is a *CFL*.

Now consider $(L_X^\omega)^-=[L(x).c]^\star.L.c.\Sigma^\omega\cup C_{\text{fin}}$, where $C_{\text{fin}}=\{\sigma\in\Sigma^\omega/\sigma$ contains only a finite number of $c\}$.

C_{fin} is an ω-regular language then it is an ω-*CFL*.

$[L(x).c]^\star.L.c$ is a context-free language because the class *CFL* is closed under star operation and concatenation product, so $[L(x).c]^\star.L.c.\Sigma^\omega$ is an ω-*CFL*. The class CFL_ω is closed under union [9], then $(L_X^\omega)^-$ is an ω-*CFL*, and so is $(L_Y^\omega)^-$.

Therefore the union $(L_X^\omega)^-\cup(L_Y^\omega)^-\ =(L_X^\omega\cap L_Y^\omega)^-$ is also an ω-*CFL*. Denote it by $L_{X,Y}$. Then it holds that

$$(L_{X,Y})^-=(L_X^\omega\cap L_Y^\omega).$$

But we have proved that one cannot decide whether $(L_X^\omega\cap L_Y^\omega)$ is empty, infinite, or has the continuum power. Hence, one cannot decide whether the ω-language $(L_{X,Y})^-$ is empty (i.e. $L_{X,Y}=\Sigma^\omega$), infinite or has the continuum power. Then (4)–(6) of the theorem are proved for an alphabet having three letters.

To prove these results for an alphabet having two letters, consider the above morphism τ. Then

$$\begin{aligned}\{a,b\}^\omega-(\tau[L_X^\omega]\cap\tau[L_Y^\omega])=&(\{a,b\}^\omega-\tau(\{a,b,c\}^\omega))\\&\cup(\tau(\{a,b,c\}^\omega)-(\tau[L_X^\omega]\cap\tau[L_Y^\omega])).\end{aligned}$$

But $\{a,b\}^\omega-\tau(\{a,b,c\}^\omega)$ is an ω-regular language therefore it is an ω-*CFL*. And $\tau(\{a,b,c\}^\omega)-(\tau[L_X^\omega]\cap\tau[L_Y^\omega])=\tau[\{a,b,c\}^\omega-(L_X^\omega\cap L_Y^\omega)]=\tau[L_{X,Y}]$. Then, it is also an ω-*CFL* because the family CFL_ω is closed under λ-free morphism [9]. Then, the union $\{a,b\}^\omega-(\tau[L_X^\omega]\cap\tau[L_Y^\omega])$ is an ω-*CFL*. Denote

$$T_{X,Y}=\{a,b\}^\omega-(\tau[L_X^\omega]\cap\tau[L_Y^\omega]).$$

Items (4)–(6) of the theorem then follow from (1)–(3) already proved, because $(T_{X,Y})^-=(\tau[L_X^\omega]\cap\tau[L_Y^\omega])$ and then one cannot decide whether the complement of the ω-*CFL* $T_{X,Y}$ over the alphabet $\{a,b\}$ is empty (i.e. $T_{X,Y}=\{a,b\}^\omega$), infinite or has the continuum power.

Remark 5.4. We have defined $L_{X,Y}=(L_X^\omega\cap L_Y^\omega)^-$. This ω-language is an ω-*CFL*. And one cannot decide whether $L_{X,Y}=\Sigma^\omega$. Two cases may happen. In the first case $L_{X,Y}=\Sigma^\omega$ and then it is an open and closed set. In the second case, we shall prove in Section 7 that $L_{X,Y}$ is neither open nor closed in Σ^ω and we shall deduce further undecidability results about the topological complexity of omega context-free languages.

6. Operation "exponentiation of sets"

Wadge gave first a description of the Wadge hierarchy of Borel sets [46]. Duparc recently got a new proof of Wadge's results and he gave a normal form of Borel sets, i.e. an inductive construction of a Borel set of every given degree [13, 15]. His proof relies on set theoretic operations which are the counterpart of arithmetical operations over ordinals needed to compute the Wadge degrees.

In fact, Duparc studied the Wadge hierarchy via the study of the conciliating hierarchy. He introduced in [13, 15] conciliating sets which are sets of finite *or* infinite words over an alphabet X, i.e. subsets of $X^\star \cup X^\omega = X^{\leqslant\omega}$. It turned out that the conciliating hierarchy is isomorphic to the Wadge hierarchy of non-self dual Borel sets, via the following correspondence:

For $A \subseteq X_A^{\leqslant\omega}$ and d a letter not in X_A, define

$$A^d = \{x \in (X_A \cup \{d\})^\omega / x(/d) \in A\},$$

where $x(/d)$ is the sequence obtained from x when removing every occurrence of the letter d.

The set theoretic operations are then defined over concilating sets. We shall only need in this paper the operation of exponentiation. We first recall the following:

Definition 6.1. Let X_A be a finite alphabet and $\twoheadleftarrow \notin X_A$, let $X = X_A \cup \{\twoheadleftarrow\}$. Let x be a finite or infinite word over the alphabet $X = X_A \cup \{\twoheadleftarrow\}$.

Then $x^\twoheadleftarrow$ is inductively defined by
$\lambda^\twoheadleftarrow = \lambda$,
For a finite word $u \in (X_A \cup \{\twoheadleftarrow\})^\star$:
$(u.a)^\twoheadleftarrow = u^\twoheadleftarrow.a$, if $a \in X_A$,
$(u.\twoheadleftarrow)^\twoheadleftarrow = u^\twoheadleftarrow$ with its last letter removed if $|u^\twoheadleftarrow| > 0$,
$(u.\twoheadleftarrow)^\twoheadleftarrow = \lambda$ if $|u^\twoheadleftarrow| = 0$,
and for u infinite:
$(u)^\twoheadleftarrow = \lim_{n\in\omega}(u[n])^\twoheadleftarrow$, where, given β_n and u in $X_A^\star$,
$u \sqsubseteq \lim_{n\in\omega}\beta_n \leftrightarrow \exists n \forall p \geqslant n \;\; \beta_p[|u|] = u$.

Remark 6.2. For $x \in X^\omega$, $x^\twoheadleftarrow$ denotes the string x, once every $\twoheadleftarrow$ occurring in x has been "evaluated" to the back space operation (the one familiar to your computer!), proceeding from left to right inside x. In other words, $x^\twoheadleftarrow = x$ from which every interval of the form "$a \twoheadleftarrow$" ($a \in X_A$) is removed.

For example, if $u = (a \twoheadleftarrow)^n$, for $n \geqslant 1$, $u = (a \twoheadleftarrow)^\omega$ or $u = (a \twoheadleftarrow\twoheadleftarrow)^\omega$ then $(u)^\twoheadleftarrow = \lambda$,
if $u = (ab \twoheadleftarrow)^\omega$ then $(u)^\twoheadleftarrow = a^\omega$,
if $u = bb(\twoheadleftarrow a)^\omega$ then $(u)^\twoheadleftarrow = b$.

We can now define the operation $A \to A^\sim$ of exponentiation of conciliating sets:

Definition 6.3. For $A \subseteq X_A^{\leqslant\omega}$ and $\twoheadleftarrow \notin X_A$, let $X = X_A \cup \{\twoheadleftarrow\}$ and $A^{\sim} = \{x \in (X_A \cup \{\twoheadleftarrow\})^{\leqslant\omega} / x^{\twoheadleftarrow} \in A\}$.

The operation $\sim$ is monotone with regard to the Wadge ordering and produce some sets of higher complexity, in the following sense:

Theorem 6.4 (Duparc [15]). (a) *For $A \subseteq X_A^{\leqslant\omega}$ and $B \subseteq X_B^{\leqslant\omega}$, A^d and B^d borel sets, $A^d \leqslant_W B^d \leftrightarrow (A^{\sim})^d \leqslant_W (B^{\sim})^d$.*

(b) *If $A^d \subseteq (X_A \cup \{d\})^\omega$ is a $\boldsymbol{\Sigma}_n^0$-complete (respectively $\boldsymbol{\Pi}_n^0$-complete) set (for an integer $n \geqslant 1$), then $(A^{\sim})^d$ is a $\boldsymbol{\Sigma}_{n+1}^0$-complete (respectively $\boldsymbol{\Pi}_{n+1}^0$-complete) set.*

(c) *If $A \subseteq X_A^\omega$ is a $\boldsymbol{\Pi}_n^0$-complete set (for an integer $n \geqslant 2$), then $(A^{\sim})$ is a $\boldsymbol{\Pi}_{n+1}^0$-complete set.*

Remark 6.5. (c) of preceding theorem follows (b) because whenever $A \subseteq X_A^\omega$ is a $\boldsymbol{\Pi}_n^0$-complete set (for an integer $n \geqslant 2$), then A^d is also a $\boldsymbol{\Pi}_n^0$-complete set.

And because whenever, for $A \subseteq X_A^\omega$, $A^d \subseteq (X_A \cup \{d\})^\omega$ is a $\boldsymbol{\Pi}_n^0$-complete set (for an integer $n \geqslant 3$), then A is also a $\boldsymbol{\Pi}_n^0$-complete set.

This property will be useful only in Section 8 when we study the ω-powers of finitary languages.

We now prove that the class CFL_ω is closed under this operation $\sim$.

Theorem 6.6. *Whenever $A \subseteq X_A^\omega$ is an ω-CFL, then $A^{\sim} \subseteq (X_A \cup \{\twoheadleftarrow\})^\omega$ is an ω-CFL.*

Proof. An ω-word $\sigma \in A^{\sim}$ may be considered as an ω-word $\sigma^{\twoheadleftarrow} \in A$ to which we possibly add, before the first letter $\sigma^{\twoheadleftarrow}(1)$ of $\sigma^{\twoheadleftarrow}$ (respectively between two consecutive letters $\sigma^{\twoheadleftarrow}(n)$ and $\sigma^{\twoheadleftarrow}(n+1)$ of $\sigma^{\twoheadleftarrow}$), a finite word v_1 (respectively v_{n+1}) where:

v_{n+1} belongs to the context free (finitary) language L_3 generated by the context free grammar with the following production rules:

$S \to aS \twoheadleftarrow S$ with $a \in X_A$,

$S \to a \twoheadleftarrow S$ with $a \in X_A$,

$S \to \lambda$ (λ being the empty word).

this language L_3 corresponds to words where every letter of X_A has been removed after using the back space operation.

And v_1 belongs to the finitary language $L_4 = (\twoheadleftarrow)^\star.(L_3.(\twoheadleftarrow)^\star)^\star$. This language corresponds to words where every letter of X_A has been removed after using the back space operation and this operation maybe has been used also when there was not any letter to erase. L_4 is a context-free language because the class CFL is closed under star operation and concatenation product. $\square$

Remark 6.7. Recall that a one counter automaton is a pushdown automaton with a pushdown alphabet in the form $\Gamma = \{Z_0, z\}$ where Z_0 is the bottom symbol and always remains at the bottom of the pushdown store. And a one counter language is a (finitary) language which is accepted by a one counter automaton by final states. It is easy to see that in fact L_3 and L_4 are deterministic one-counter languages, i.e. L_3 and L_4 are accepted by deterministic one-counter automata. And for $a \in X_A$, the language $L_3.a$ is also accepted by a deterministic one-counter automaton.

Then, we can state the following:

Lemma 6.8. *Whenever $A \subseteq X_A^\omega$, the ω-language $A^\sim \subseteq (X_A \cup \{\twoheadleftarrow\})^\omega$ is obtained by substituting in A the language $L_3.a$ for each letter $a \in X_A$, where L_3 is the CFL defined above, and then making a left concatenation by the language L_4.*

Let now A be an ω-*CFL* given by $A = \bigcup_{i=1}^{n} U_i.V_i^\omega$ where U_i and V_i are context free languages. Then $A^\sim = \bigcup_{i=1}^{n} (L_4.U_i').V_i'^\omega$, where U_i' (respectively V_i') is obtained by substituting the language $L_3.a$ to each letter $a \in X_A$ in U_i (respectively V_i).

The class *CFL* is closed under substitution, so U_i' and V_i' are *CFL*, and so is the language $(L_4.U_i')$ by concatenation product. Hence the ω-language $A^\sim$ is an ω-*CFL* because ω-$KC(CFL) \subseteq CFL_\omega$.

Proposition 6.9. *From a MPDA accepting the ω-language $A \subseteq X_A^\omega$, one can effectively construct a MPDA which accepts the ω-language $A^\sim \subseteq (X_A \cup \{\twoheadleftarrow\})^\omega$.*

Proof. Let A be an ω-*CFL* which is accepted by a Muller pushdown automaton $M = (K, X_A, \Gamma, \delta, q_0, Z_0, F)$. The ω-language accepted by M is $L(M) = A = \{\sigma \in X_A^\omega /$there exists a complete run r of M on σ such that $In(r) \in F\}$.

We shall construct another *MPDA* $M^\sim$ which accepts the ω-language $A^\sim$ over the alphabet $X = X_A \cup \{\twoheadleftarrow\}$.

Describe first informally the behaviour of the machine $M^\sim$ when it reads an ω-word $\sigma \in A^\sim$. Recall that this word may be considered as an ω-word $\sigma^{\twoheadleftarrow} \in A$ to which we possibly add, before the first letter $\sigma^{\twoheadleftarrow}(1)$ of $\sigma^{\twoheadleftarrow}$ (respectively between two consecutive letters $\sigma^{\twoheadleftarrow}(n)$ and $\sigma^{\twoheadleftarrow}(n+1)$ of $\sigma^{\twoheadleftarrow}$), a finite word v_1 (respectively v_{n+1}) where v_1 belongs to the context-free language L_4 and v_{n+1} belongs to the context-free language L_3.

$M^\sim$ starts the reading as a pushdown automaton accepting the language L_4. Then $M^\sim$ begins to read as M, but at any moment of the computation it may guess (using the non-determinism) that it reads a finite segment v of L_3 which will be erased (using the eraser $\twoheadleftarrow$). It reads v using an additional stack letter E which permits to simulate a one counter automaton at the top of the stack while keeping the memory of the stack of M. Then, after the reading of v, $M^\sim$ simulates again the machine M and so on.

More formally, $M^\sim = (K^\sim, X_A \cup \{\twoheadleftarrow\}, \Gamma \cup \{E\}, \delta^\sim, q_0', Z_0, F^\sim)$, where

$$K^\sim = K \cup \{q_0'\} \cup \{q^1 / q \in K\}$$

q_0' is a new state not in K

E is a new letter not in Γ

$F^{\sim} = \{P \subseteq K^{\sim} / P \cap K \in F\}$

and the transition relation $\delta^{\sim}$ is defined by the following cases (where the transition rules (a)–(d) are used to simulate a pushdown automaton accepting L_4 and the MPDA $M^{\sim}$ enters in a state q^1, for $q \in K$, when it simulates a one counter automaton accepting L_3):

(a) $\delta^{\sim}(q'_0, \twoheadleftarrow, Z_0) = (q'_0, Z_0)$.

(b) $(q'_0, EZ_0) \in \delta^{\sim}(q'_0, a, Z_0)$ for each $a \in X_A$.

(c) $\delta^{\sim}(q'_0, \twoheadleftarrow, E) = (q'_0, \lambda)$.

(d) $\delta^{\sim}(q'_0, a, E) = (q'_0, EE)$ for each $a \in X_A$.

(e) $(q, v) \in \delta^{\sim}(q'_0, a, Z_0)$ iff $(q, v) \in \delta(q_0, a, Z_0)$ for each $a \in X_A \cup \{\lambda\}$ and $v \in \Gamma^\star$ and $q \in K$.

(f) $(q', v) \in \delta^{\sim}(q, a, \gamma)$ iff $(q', v) \in \delta(q, a, \gamma)$ for each $a \in X_A \cup \{\lambda\}$ and $\gamma \in \Gamma$ and $v \in \Gamma^\star$ and $q, q' \in K$.

(g) $(q^1, E\gamma) \in \delta^{\sim}(q, a, \gamma)$ for each $a \in X_A$ and $\gamma \in \Gamma$ and $q \in K$.

(h) $\delta^{\sim}(q^1, a, E) = (q^1, EE)$ for each $a \in X_A$ and $q \in K$.

(i) $\delta^{\sim}(q^1, \twoheadleftarrow, E) = (q^1, \lambda)$.

(j) $(q', v) \in \delta^{\sim}(q^1, a, \gamma)$ iff $(q', v) \in \delta(q, a, \gamma)$ for each $a \in X_A$ and $\gamma \in \Gamma$ and $v \in \Gamma^\star$ and $q, q' \in K$.

(k) $(q^1, E\gamma) \in \delta^{\sim}(q^1, a, \gamma)$ for each $a \in X_A$ and $\gamma \in \Gamma$ and $q \in K$. □

Consider now subsets of $X^{\leqslant \omega}$ in the form $A \cup B$, where A is a finitary context free language and B is an ω-*CFL*. Remark that A and B should not be accepted by the same pushdown automaton (but it may be). Prove then the following.

Proposition 6.10. *If $C = A \cup B$, where A is a finitary context free language and B is an ω-CFL over the same alphabet $X_A = X_B$, then $C^{\sim}$ is also the union of a finitary context free language and an ω-CFL over the alphabet $X_A \cup \{\twoheadleftarrow\}$.*

Proof. It is easy to see from the definition of the operation of exponentiation of sets that if $C = A \cup B$ then: $C^{\sim} = A^{\sim} \cup B^{\sim}$.

But if B is an ω-*CFL* over $X_B = X_A$, then by Theorem 6.6 $B^{\sim}$ is an ω-*CFL* D_1.

Consider now the set $A^{\sim}$: This subset of $(X_A \cup \{\twoheadleftarrow\})^{\leqslant \omega}$ is constituted of finite *and* infinite words. Let h be the substitution: $X \to P((X_A \cup \{\twoheadleftarrow\})^\star)$ defined by $a \to a.L_3$ where L_3 is the context free language defined above. Then it is easy to see that the finite words are obtained by substituting in A the language $a.L_3$ for each letter $a \in X_A$ and concatenating on the left by the language L_4.

But *CFL* is closed under substitution and concatenation [3], then this language is a context-free finitary language D_2.

The infinite words in $A^{\sim}$ constitutes the ω-language

$D_2.(L_3 - \{\lambda\})^\omega$ if $\lambda \notin A$, and

$D_2.(L_3-\{\lambda\})^\omega \cup (L_4-\{\lambda\})^\omega$ if $\lambda \in A$,

The languages $L_4-\{\lambda\}$ and $L_3-\{\lambda\}$ are context free, thus the set of infinite words in $A^\sim$ is an ω-CFL D_3 because ω-$KC(CFL) \subseteq CFL_\omega$ by Theorem 2.8. Then:

$$A^\sim = D_1 \cup D_2 \cup D_3.$$

But CFL_ω is closed under union hence $D_1 \cup D_3$ is an ω-CFL. This ends the proof.

Proposition 6.11. (a) *If $A \subseteq \Sigma^\star$ is a context free language, then A^d is an ω-CFL.*

(b) *If $A \subseteq \Sigma^\omega$ is an ω-CFL, then A^d is an ω-CFL.*

(c) *If A is the union of a finitary context free language and of an ω-CFL over the same alphabet Σ, then A^d is an ω-CFL over the alphabet $\Sigma \cup \{d\}$.*

Proof of (a). Let $A \subseteq \Sigma^\star$ be a finitary context-free language. Substitute first the language $(d^\star).a$ for each letter $a \in \Sigma$. In such a way we obtain another finitary context-free language A' because CFL is closed under substitution and the languages $(d^\star).a$ are context free. Indeed $A^d = A'.d^\omega$ hence A^d is an ω-CFL because ω-$KC(CFL) \subseteq CFL_\omega$ by Theorem 2.8. □

Proof of (b). Let $A \subseteq \Sigma^\omega$ be an ω-CFL. The ω-language A^d is obtained from A by substituting the language $(d^\star).a$ for each letter $a \in \Sigma$ in the words of A. But the class CFL_ω is closed under λ-free context-free substitution [9], hence A^d is an ω-CFL. □

Proof of (c). Let A and B be subsets of $\Sigma^{\leqslant\omega}$ for a finite alphabet Σ. Then, we easily see that if $C = A \cup B$, $C^d = A^d \cup B^d$ holds. (c) is now an easy consequence of (a) and (b) because CFL_ω is closed under union. □

7. Topological properties of ω-CFL

From preceding theorems we first deduce that the ω-CFL exhaust the hierarchy of Borel sets of finite rank.

Theorem 7.1. *For each non-negative integer $n \geqslant 1$, there exist $\boldsymbol{\Sigma}_n^0$-complete ω-CFL A_n and $\boldsymbol{\Pi}_n^0$-complete ω-CFL B_n.*

Proof. For $n=1$ consider the $\boldsymbol{\Sigma}_1^0$-complete ω-regular language

$A_1 = \{\alpha \in \{0,1\}^\omega / \exists i \ \alpha(i)=1\}$

and the $\boldsymbol{\Pi}_1^0$-complete ω-regular language

$B_1 = \{\alpha \in \{0,1\}^\omega / \forall i \ \alpha(i)=0\}$.

These languages are omega context-free languages because $REG_\omega \subseteq CFL_\omega$.

Now consider the $\boldsymbol{\Sigma}_2^0$-complete ω-regular language

$A_2 = \{\alpha \in \{0,1\}^\omega / \exists^{<\omega} i \ \alpha(i)=1\}$

and the $\boldsymbol{\Pi}_2^0$-complete ω-regular language

$B_2 = \{\alpha \in \{0,1\}^\omega / \exists^\omega i \quad \alpha(i)=0\}$,

where $\exists^{<\omega} i$ means: "there exist only finitely many i such that ...", and
$\exists^{\omega} i$ means: "there exist infinitely many i such that ...".
A_2 and B_2 are omega context-free languages because they are ω-regular languages.

To obtain omega context-free languages further in the Borel hierarchy, consider now O_1 (respectively C_1) subsets of $\{0,1\}^{\leqslant\omega}$ such that $(O_1)^d$ (respectively $(C_1)^d$) are $\boldsymbol{\Sigma}^0_1$-complete (respectively $\boldsymbol{\Pi}^0_1$-complete).

For example $O_1 = \{x \in \{0,1\}^{\leqslant\omega} / \exists i \ x(i) = 1\}$ and $C_1 = \{\lambda\}$.

We shall have to apply $n \geqslant 1$ times the operation of exponentiation of sets.

More precisely, we define, for a set $A \subseteq X_A^{\leqslant\omega}$:
$A^{\sim.0} = A$
$A^{\sim.1} = A^{\sim}$ and
$A^{\sim.(n+1)} = (A^{\sim.n})^{\sim}$.

Now apply n times (for an integer $n \geqslant 1$) the operation $\sim$ (with different new letters $\twoheadleftarrow_1, \twoheadleftarrow_2, \twoheadleftarrow_3, \ldots, \twoheadleftarrow_n$) to O_1 and C_1.

By Theorem 6.4, it holds that for an integer $n \geqslant 1$:
$(O_1^{\sim.n})^d$ is a $\boldsymbol{\Sigma}^0_{n+1}$-complete subset of $\{0,1,\twoheadleftarrow_1,\ldots,\twoheadleftarrow_n,d\}^{\omega}$.
$(C_1^{\sim.n})^d$ is a $\boldsymbol{\Pi}^0_{n+1}$-complete subset of $\{0,1,\twoheadleftarrow_1,\ldots,\twoheadleftarrow_n,d\}^{\omega}$
and it is easy to see that O_1 and C_1 are in the form $E \cup F$ where E is a finitary context-free language and F is an omega context-free language. Then it follows from Propositions 6.10 and 6.11 that the ω-languages $(O_1^{\sim.n})^d$ and $(C_1^{\sim.n})^d$ are context-free. Hence the class CFL_{ω} exhausts the hierarchy of Borel sets of finite rank: we obtain the omega context-free languages $A_n = (O_1^{\sim.(n-1)})^d$ and $B_n = (C_1^{\sim.(n-1)})^d$, for $n \geqslant 3$. $\square$

This gives a partial answer to questions of Thomas and Lescow [29] about the hierarchy of ω-*CFL*: This hierarchy exhausts the hierarchy of Borel sets of finite rank.

A natural question now arises: Do the results of [28] extend to ω-*CFL*? Unfortunately, the answer is no. Cohen and Gold proved that one cannot decide whether an ω-*CFL* is in the class $\mathbf{F}, \mathbf{G}$, or $\mathbf{G}_{\delta}$. We extend this result to all classes $\boldsymbol{\Sigma}^0_n$ and $\boldsymbol{\Pi}^0_n$, for n an integer $\geqslant 1$.

Theorem 7.2. *Let n be an integer $\geqslant 1$. Then it is undecidable whether an effectively given ω-CFL is in the class $\boldsymbol{\Sigma}^0_n$ (respectively $\boldsymbol{\Pi}^0_n$).*

Proof. Recall that, by Theorem 5.1, it is undecidable, for an effectively given ω-*CFL* A over the alphabet Σ, whether $A = \Sigma^{\omega}$.

More precisely, one cannot decide whether $L_{X,Y} = (L_X^{\omega} \cap L_Y^{\omega})^{-}$ is equal to Σ^{ω}.

Whenever $L_{X,Y} = \Sigma^{\omega}$, $L_{X,Y}$ is an open and closed subset of Σ^{ω}. We shall prove the following:

Lemma 7.3. *Whenever $L_{X,Y} \neq \Sigma^\omega$, $L_{X,Y}$ is neither open nor closed in Σ^ω.*

Proof. Suppose $L_{X,Y} \neq \Sigma^\omega$. Then $(L_X^\omega \cap L_Y^\omega) = (L_X \cap L_Y)^\omega$ is non-empty, so there exists a non-empty sequence of indices $i_1, \dots, i_k$ such that $x_{i_1} \dots x_{i_k} = y_{i_1} \dots y_{i_k}$, and $(ba^{i_k} \dots ba^{i_1} cx_{i_1} \dots x_{i_k} c)^\omega \in (L_X \cap L_Y)^\omega$.

Then each sequence $(i_1 \dots i_k)^n$, where n is an integer $\geqslant 1$, gives another solution of Post correspondence problem. For each $n \geqslant 1$, $(ba^{i_k} \dots ba^{i_1})^n$ is a prefix of an ω-word of $(L_X^\omega \cap L_Y^\omega)$.

So if $(L_X^\omega \cap L_Y^\omega)$ was closed, the infinite word $(ba^{i_k} \dots ba^{i_1})^\omega$ should be in $(L_X^\omega \cap L_Y^\omega)$. This cannot happen because every word in $(L_X^\omega \cap L_Y^\omega)$ contains an infinite number of letters “c”.

Therefore $(L_X^\omega \cap L_Y^\omega)$ is not a closed set and $L_{X,Y} = (L_X^\omega \cap L_Y^\omega)^-$ is not an open set.

Let us now prove that $(L_X^\omega \cap L_Y^\omega)$ is not an open subset of Σ^ω. Otherwise, this set should be $V.\Sigma^\omega$, for a set $V \subseteq \Sigma^\star$.

$(L_X^\omega \cap L_Y^\omega)$ is supposed to be non-empty, so $V \neq \emptyset$, and for $v \in V$, the word $v.a^\omega$ should be in $(L_X^\omega \cap L_Y^\omega)$. This is again not possible because every word in $(L_X^\omega \cap L_Y^\omega)$ contains an infinite number of letters “c”.

Then $(L_X^\omega \cap L_Y^\omega)$ is not open and $L_{X,Y} = (L_X^\omega \cap L_Y^\omega)^-$ is not closed. $\square$

In order to apply precisely the results of Duparc, we have here to consider conciliating sets, i.e. subsets of $\Sigma^{\leqslant \omega}$, for some alphabet Σ. Then, we shall prove the following:

Lemma 7.4. (a) *For each integer $n \geqslant 1$, the language $((L_{X,Y} \cup \Sigma^\star)^{\sim.n})^d$ is an omega context free language.*

(b) *Whenever $L_{X,Y} \neq \Sigma^\omega$, the language $(L_{X,Y} \cup \Sigma^\star)^d$ is neither open nor closed in $(\Sigma \cup \{d\})^\omega$.*

(c) *Whenever $L_{X,Y} = \Sigma^\omega$, it holds that $(L_{X,Y} \cup \Sigma^\star)^d = (\Sigma \cup \{d\})^\omega$.*

Proof of (a). The ω-language $L_{X,Y}$ is context free (see the proof of Theorem 5.1) and $\Sigma^\star$ is also a context-free language. Then (a) follows from the Propositions 6.10 and 6.11. $\square$

Proof of (b). To prove that whenever $L_{X,Y} \neq \Sigma^\omega$, the language $(L_{X,Y} \cup \Sigma^\star)^d$ is neither open nor closed in $(\Sigma \cup \{d\})^\omega$, we use a similar method as for Lemma 7.3 with minor modifications. $\square$

Proof of (c). $L_{X,Y} = \Sigma^\omega$ implies $(L_{X,Y} \cup \Sigma^\star)^d = (\Sigma \cup \{d\})^\omega$ because of the definition of $A \to A^d$ and then $(L_{X,Y} \cup \Sigma^\star)^d$ is open and closed in $(\Sigma \cup \{d\})^\omega$. $\square$

Consider now as above O_1 (respectively C_1) subsets of $\{0,1\}^{\leqslant \omega}$ such that $(O_1)^d$ (respectively $(C_1)^d$) are $\boldsymbol{\Sigma}_1^0$-complete (respectively $\boldsymbol{\Pi}_1^0$-complete):

For example, $O_1 = \{x \in \{0,1\}^{\leqslant \omega} / \exists i \ x(i) = 1\}$ and $C_1 = \{\lambda\}$.

We denote $(L_{X,Y} \cup \Sigma^\star) = A_{X,Y}$.

We cannot decide whether $(A_{X,Y})^d \leqslant_W (O_1)^d$ neither whether $(A_{X,Y})^d \leqslant_W (C_1)^d$, because we cannot decide whether $(A_{X,Y})^d$ is open neither whether $(A_{X,Y})^d$ is closed.

Now apply the operation $\sim$, n times, to the sets $A_{X,Y}$ and O_1 and C_1.

Then by Theorem 6.4:

(a) $(A_{X,Y}^{\sim.n})^d \leqslant_W (O_1^{\sim.n})^d \leftrightarrow (A_{X,Y})^d \leqslant_W (O_1)^d$.
(b) $(A_{X,Y}^{\sim.n})^d \leqslant_W (C_1^{\sim.n})^d \leftrightarrow (A_{X,Y})^d \leqslant_W (C_1)^d$.
(c) $(O_1^{\sim.n})^d$ is $\boldsymbol{\Sigma}^0_{n+1}$-complete.
(d) $(C_1^{\sim.n})^d$ is $\boldsymbol{\Pi}^0_{n+1}$-complete.

Therefore, for each $n \geqslant 0$, one cannot decide whether:

(a) $(A_{X,Y}^{\sim.n})^d \leqslant_W (O_1^{\sim.n})^d$ i.e. $(A_{X,Y}^{\sim.n})^d$ is in the class $\boldsymbol{\Sigma}^0_{n+1}$.
(b) $(A_{X,Y}^{\sim.n})^d \leqslant_W (C_1^{\sim.n})^d$ i.e. $(A_{X,Y}^{\sim.n})^d$ is in the class $\boldsymbol{\Pi}^0_{n+1}$.

Remark 7.5. We can use standard construction methods to find a (non-deterministic) Muller pushdown automaton $\mathbb{A}^n_{X,Y}$ which accepts the language $(A_{X,Y}^{\sim.n})^d$, because from a *MPDA* which accepts an ω-language A, we can effectively construct a *MPDA* which accepts $A^\sim$.

And it is clear from the proof of Proposition 6.10 that if $C = A \cup B$ where A is a finitary language accepted by a pushdown automaton $\mathbb{A}$ and B is an ω-language accepted by a *MPDA* $\mathbb{B}$, then we can construct a pushdown automaton $\mathbb{A}'$ accepting finite words and a *MPDA* $\mathbb{B}'$ such that $C^\sim = L(\mathbb{A}') \cup L(\mathbb{B}')$. And the operation $A \to A^d$ is also effective in the same manner from the proof of Proposition 6.11.

From the above proof, we can deduce that one cannot decide whether the intersection of the two deterministic ω-*CFL* L^ω_X and L^ω_Y is open (respectively closed):

Theorem 7.6. *One cannot decide whether the intersection of two ω-DCFL is open (respectively closed).*

Remark 7.7. The languages L^ω_X and L^ω_Y are in fact accepted by Büchi deterministic *PDA*.

In terms of [41], we can write the above result as:

It is undecidable whether the intersection of two effectively given ω-languages in *DPDA*$(inf, \sqcap)$ is in *DPDA*$(ran, \sqcap)$ (respectively *DPDA*$(ran, \subseteq)$).

8. ω-powers of finitary languages

The ω-power of a language $W \subseteq X^\star$ is a fundamental operation over finitary languages which leads to ω-languages.

Whenever W is a regular language (respectively a *CFL*), then W^ω is an ω-regular language, (respectively an ω-*CFL*).

The question of the topological complexity of W^ω naturally arises.

When W is a regular language, W^ω is a boolean combination of G_δ sets because it is an ω-regular set.

In order to study the topological complexity of W^ω, when W is a context free language, we first introduce a variant of the definition of $A^\sim$:

Definition 8.1. For $A \subseteq X_A^{\leqslant\omega}$ and $\twoheadleftarrow \notin X_A$, let $X = X_A \cup \{\twoheadleftarrow\}$ and
$A^\approx = \{x \in (X_A \cup \{\twoheadleftarrow\})^{\leqslant\omega} / x^\twoheadleftarrow \in A\}$,
where $x^\twoheadleftarrow$ is inductively defined by
$\lambda^\twoheadleftarrow = \lambda$.
For a finite word $u \in (X_A \cup \{\twoheadleftarrow\})^\star$:
$(u.a)^\twoheadleftarrow = u^\twoheadleftarrow.a$, if $a \in X_A$,
$(u.\twoheadleftarrow)^\twoheadleftarrow = u^\twoheadleftarrow$ with its last letter removed if $|u^\twoheadleftarrow| > 0$,
$(u.\twoheadleftarrow)^\twoheadleftarrow$ is undefined if $|u^\twoheadleftarrow| = 0$,
and for u infinite:
$(u)^\twoheadleftarrow = \lim_{n\in\omega} (u[n])^\twoheadleftarrow$, where, given β_n and u in $X_A^\star$,
$u \sqsubseteq \lim_{n\in\omega} \beta_n \leftrightarrow \exists n \forall p \geqslant n \; \beta_p[|u|] = u$.

The only difference is that here: $(u.\twoheadleftarrow)^\twoheadleftarrow$ is undefined if $|u^\twoheadleftarrow| = 0$. It is easy to see (from Duparc's proof [15]) that if $A \subseteq X_A^\omega$ is a Borel set such that $A \neq X_A^\omega$, i.e. $A^- \neq \emptyset$, then $A^\approx$ is wadge equivalent to $A^\sim$ because:

(a) In the Wadge game $W(A^\sim, A^\approx)$ the player in charge of $A^\approx$ has clearly a winning strategy which consists in copying the play of the other player except if player 1 writes the eraser $\twoheadleftarrow$ but he has nothing to erase. In this case player 2 writes for example a letter $a \in X_A$ and the eraser $\twoheadleftarrow$ at the next step of the play. Now if, in ω steps, player 1 has written the ω-word α and player 2 has written the ω-word β, it is easy to see that $[\alpha^\twoheadleftarrow = \beta^\twoheadleftarrow]$ and then $\alpha \in A^\sim$ iff $\beta \in A^\approx$. Thus, player 2 has a winning strategy in the Wadge game $W(A^\sim, A^\approx)$.

(b) Consider now the Wadge game $W(A^\approx, A^\sim)$. The only extra possibility for the player in charge of $A^\approx$ is to get out of the set $A^\approx$ by writing the eraser $\twoheadleftarrow$ when in fact there is not any letter of his previous play to erase. But then his final play is surely outside $A^\approx$.
If $A \neq X_A^\omega$, i.e. $A^- \neq \emptyset$, then the player in charge of the set $A^\sim$ may write a word in $(A^\sim)^-$, (by playing the letter $\twoheadleftarrow$ to erase his previous play and then writing a word in A^-) then player 2 has a winning strategy in the Wadge game $W(A^\approx, A^\sim)$.

Lemma 8.2. *Whenever $A \subseteq X_A^\omega$ is an ω-power of a language L_A, i.e. $A = L_A^\omega$, then $A^\approx$ is also an ω-power, i.e. there exists a (finitary) language E_A such that $A^\approx = E_A^\omega$.*

Proof. Let h be the substitution: $X_A \to P((X_A \cup \{\twoheadleftarrow\})^\star)$ defined by $a \to L_3.a$ where L_3 is the context-free language defined above. Then it is easy to see that now $A^\approx$ is obtained by substituting in A the language $L_3.a$ for each letter $a \in X_A$. (We have not here to consider the language L_4 which appeared in the expression of $A^\sim$). Then $E_A = h(L_A)$.

Recall now the definition of one counter automata and one counter (and iterated counter) languages: A one counter automaton is a pushdown automaton with a pushdown alphabet in the form $\Gamma=\{Z_0,z\}$ where Z_0 is the bottom symbol and always remains at the bottom of the pushdown store. A one counter language is a (finitary) language which is accepted by a one counter automaton by final states. Let *OCL* be the family of one counter languages. The family *ICL* of iterated counter languages is the closure under substitution of the family *OCL*. It is also the class of (finitary) languages which are accepted by a pushdown automaton such that, during any computation, the words in the pushdown store remain in a bounded language in the form $(z_k)^\star,\dots,(z_2)^\star(z_1)^\star Z_0$, where $\{Z_0,z_1,\dots,z_k\}$ is the pushdown alphabet, [1]. We can now state the following:

Theorem 8.3. *For each integer $n\geqslant 1$, there exists a context free language P_n such that P_n^ω is a $\boldsymbol{\Pi}_n^0$-complete set.*

In fact there is such a language in the subclass of iterated counter languages.

Proof. Let $B_1=\{\sigma\in\{0,1\}^\omega/\forall i\geqslant 1\ \sigma(i)=0\}=\{0^\omega\}$.

It is a $\boldsymbol{\Pi}_1^0$-complete set of the form P_1^ω where P_1 is the singleton containing the only word 0.

Remark that $P_1=\{0\}$ is a regular set thus a context free language.

Let then $B_2=\{\alpha\in\{0,1\}^\omega/\exists^\omega i\ \alpha(i)=1\}$ be the well known $\boldsymbol{\Pi}_2^0$-complete regular ω-language. It holds that $B_2=(0^\star.1.0^\star)^\omega$. Let $P_2=(0^\star.1.0^\star)$. Then P_2 is rational hence context free and then $h(P_2)\in CFL$ and $(h(P_2))^\omega=(P_2^\omega)^\approx$ is a $\boldsymbol{\Pi}_3^0$-complete set by Theorem 6.4.

Iterating this method n times, we easily obtain a context free language P_{n+2} such that $(P_{n+2})^\omega$ is a $\boldsymbol{\Pi}_{n+2}^0$-complete set.

P_1 and P_2 are one counter languages because they are rational.

The languages L_3a, for $a\in X_A$, are one counter languages. Then for each integer $n\geqslant 1$ the language P_n is an iterated counter language.

Remark 8.4. It is undecidable whether $(L_X\cap L_Y)^\omega$ is open (respectively closed). Then, it is undecidable, for effectively given *CFL* L_1 and L_2, whether $(L_1\cap L_2)^\omega$ is open (respectively closed).

9. Gale–Stewart games

Recall the following:

Definition 9.1 (*Gale and Stewart* [23]). Let $A\subseteq X^\omega$, where X is a finite alphabet. The game $G(A)$ is a game with perfect information between two players, player 1 first writes a letter $a_1\in X$, then player 2 writes a letter $b_1\in X$, then player 1 writes $a_2\in X$, and so on … After ω steps, the two players have composed a word $\sigma=a_1b_1a_2b_2\dots$ of X^ω. Player 1 wins the play iff $\sigma\in A$, otherwise, player 2 wins the play.

It follows from Martin's Theorem that every Gale–Stewart game $G(A)$, where A is a Borel set, is determined, i.e. that one of the two players has a winning strategy.

And Büchi–Landweber Theorem [7] states that whenever A is an ω-regular language, one can decide which player has a w.s. and one can effectively construct a w.s. given by a transducer.

Walukiewicz's Theorem extends this result to deterministic pushdown automata [48].

The problem of the synthesis of winning strategies is of practical interest in computer science, because the conditions of a Gale Stewart game may be seen as a specification, while the two players are respectively a non-terminating reactive program and the "environment".

The question of the effective construction of w.s. is asked in [45, 29, 17].

We show here that for non deterministic ω-*CFL* A, we cannot even decide which player has a w.s.:

Theorem 9.2. *For an effectively given closed* ω-*CFL* A, *it is undecidable to determine which player has a winning strategy in the Gale–Stewart game* $G(A)$.

Proof. We have shown in section 5 that one cannot decide whether the ω-*CFL* $L_{X,Y}$, over the alphabet Σ, is equal to Σ^ω.

Now define the set $B_{X,Y}$ which is composed of the ω-words $\sigma = a_1b_1a_2b_2\ldots$ such that $b_1b_2\ldots$ is in $L_{X,Y}$.

Consider the game $G(B_{X,Y})$.

If $L_{X,Y} = \Sigma^\omega$, player 1 always wins the play, then he has an obvious w.s.

If $L_{X,Y} \neq \Sigma^\omega$. Player 2 has a w.s. which consists in writing a word $b_1b_2\ldots$ which is not in $L_{X,Y}$.

Then, we cannot decide which player has a w.s., because we cannot decide whether $L_{X,Y} = \Sigma^\omega$ and it is easy to construct, from a *MPDA* accepting $L_{X,Y}$, a *MPDA* accepting $B_{X,Y}$.

The set $L_{X,Y}$ is a $\mathbf{F}_\sigma$-set, because its complement $L_X^\omega \cap L_Y^\omega$ is a $\mathbf{G}_\delta$-set, and we can easily deduce that $B_{X,Y}$ is also a $\mathbf{F}_\sigma$-set.

With a slight modification, we can show that this result remains true where we consider only closed ω-*CFL*.

For that we can replace $L_X^\omega \cap L_Y^\omega$ by $(L_X \cap L_Y).\Sigma^\omega$ and call its complement $L'_{X,Y}$. Then we can show, as in Section 5, that $L'_{X,Y}$ is an ω-*CFL*, and that one cannot decide whether $L'_{X,Y} = \Sigma^\omega$.

But now $L'_{X,Y}$ is a closed ω-*CFL*, and we can associate a Gale–Stewart game $G(B'_{X,Y})$, where $B'_{X,Y}$ is another closed ω-*CFL*, and such that one cannot decide which player has a w.s. in the game $G(B'_{X,Y})$. $\square$

10. Arithmetical properties

In this section, we shall deduce from the preceding proofs some results about ω-context free languages and the Arithmetical hierarchy.

First recall the definition of the Arithmetical hierarchy of ω-languages, [41].

Let X be a finite alphabet. An ω-language $L \subseteq X^\omega$ belongs to the class Σ_n if and only if there exists a recursive relation $R_L \subseteq (\mathbb{N})^{n-1} \times X^\star$ such that

$$L = \{\sigma \in X^\omega / \exists a_1 \ldots Q_n a_n \ (a_1, \ldots, a_{n-1}, \sigma[a_n + 1]) \in R_L\},$$

where Q_i is one of the quantifiers $\forall$ or $\exists$ (not necessarily in an alternating order). An ω-language $L \subseteq X^\omega$ belongs to the class Π_n if and only if its complement $X^\omega - L$ belongs to the class Σ_n.

The inclusion relations that hold between the classes Σ_n and Π_n are the same as for the corresponding classes of the Borel hierarchy.

Proposition 10.1 (See Staiger [42]). (a) $\Sigma_n \cup \Pi_n \subsetneq \Sigma_{n+1} \cap \Pi_{n+1}$, *for each integer* $n \geqslant 1$.

(b) *A set* $W \subseteq X^\omega$ *is in the class* Σ_n *if and only if its complement* W^- *is in the class* Π_n.

(c) $\Sigma_n - \Pi_n \neq \emptyset$ *and* $\Pi_n - \Sigma_n \neq \emptyset$ *hold for each integer* $n \geqslant 1$.

The classes Σ_n and Π_n are strictly included in the respective classes $\boldsymbol{\Sigma}_n^0$ and $\boldsymbol{\Sigma}_n^0$ of the Borel hierarchy:

Theorem 10.2 (See Staiger [42]). *For each integer* $n \geqslant 1$, $\Sigma_n \subsetneq \boldsymbol{\Sigma}_n^0$ *and* $\Pi_n \subsetneq \boldsymbol{\Pi}_n^0$.

Remark that cardinality arguments suffice to show that the inclusions are strict.

We are now able to prove the following result:

Theorem 10.3. *Let n be an integer* $\geqslant 1$. *Then it is undecidable whether an effectively given* ω-*CFL is in the class* Σ_n (*respectively* Π_n).

Proof. Return to the proof of Theorem 7.2. Let n be an integer $\geqslant 1$. We had found a family of omega context free languages

$$(A_{X,Y}^{\sim.n})^d = ((L_{X,Y} \cup \Sigma^\star)^{\sim.n})^d$$

over the alphabet $\{a, b, c, \twoheadleftarrow_1, \twoheadleftarrow_2, \ldots, \twoheadleftarrow_n, d\}$ such that $(A_{X,Y}^{\sim.n})^d$ is either $\{a, b, c, \twoheadleftarrow_1, \twoheadleftarrow_2, \ldots, \twoheadleftarrow_n, d\}^\omega$ or an ω-language which is neither a $\boldsymbol{\Pi}_{n+1}^0$-subset nor a $\boldsymbol{\Sigma}_{n+1}^0$-subset of $\{a, b, c, \twoheadleftarrow_1, \twoheadleftarrow_2, \ldots, \twoheadleftarrow_n, d\}^\omega$.

In the first case $\{a, b, c, \twoheadleftarrow_1, \twoheadleftarrow_2, \ldots, \twoheadleftarrow_n, d\}^\omega$ is in $\Sigma_1 \cap \Pi_1$ hence also in the class Σ_n (respectively Π_n) for each integer $n \geqslant 1$.

And in the second case it follows from Theorem 10.2 that $(A_{X,Y}^{\sim.n})^d$ is neither in the class Σ_{n+1} nor in the class Π_{n+1}. But one cannot decide which case holds.

Recall that the ω-languages accepted by deterministic Turing machines with a Büchi (respectively Muller) acceptance condition are exactly the languages which are Π_2-languages (respectively boolean combinations of Π_2-languages) [41].

Thus, in the above proof we have seen that $(A_{X,Y}^{\sim.2})^d$ is either $\{a, b, c, \twoheadleftarrow_1, \twoheadleftarrow_2, d\}^\omega$ (and in that case it is accepted by a deterministic Büchi or Muller automaton hence also by a Büchi deterministic Turing machine) or an ω-language which is neither a $\boldsymbol{\Pi}_3^0$-subset nor a $\boldsymbol{\Sigma}_3^0$-subset of $\{a, b, c, \twoheadleftarrow_1, \twoheadleftarrow_2, d\}^\omega$. Hence in this latter case $(A_{\tilde{X},\tilde{Y}}^{\cdot 2})^d$ is not a boolean combination of Π_2-languages (because $\Pi_2 \subseteq \boldsymbol{\Pi}_2^0$ and boolean combinations of $\boldsymbol{\Pi}_2^0$-sets are $\boldsymbol{\Pi}_3^0 \cap \boldsymbol{\Sigma}_3^0$-sets [33]).

As it was proved above, one cannot decide which case holds, so we can deduce the following:

Theorem 10.4. *It is undecidable to determine whether an effectively given ω-CFL is accepted by a deterministic Turing machine with Büchi* (*respectively Muller*) *acceptance condition.*

In fact this result can be extended to other deterministic machines. Consider **X**-automata as defined in [19] which are automata equipped with a storage type **X**. Then the ω-languages accepted by deterministic **X**-automata with a Büchi (respectively Muller) acceptance condition are languages which are $\boldsymbol{\Pi}_2^0$-languages (respectively boolean combinations of $\boldsymbol{\Pi}_2^0$-languages) [19].

But if Γ is a finite alphabet and **X** is a storage type, the ω-language Γ^ω is accepted by an **X**-automaton. Hence this provides the following generalization:

Theorem 10.5. *Let* **X** *be a storage type as defined in* [19]. *Then it is undecidable to determine whether an effectively given ω-CFL is accepted by a deterministic* **X**-*automaton with Büchi* (*respectively Muller*) *acceptance condition.*

11. Concluding remarks and further work

This paper is the first of several papers about topological properties of ω-*CFL*:

11.1. Omega deterministic CFL

We have proved that, for any effectively given ω-regular language A and ω-*DCFL* B, we can decide whether B is in the Wadge class of A, or in the Wadge degree of A.

A natural question now arises. Are the Wadge degrees of ω-*DCFL* also Wadge degrees of ω-regular languages? And can we decide whether $A \equiv_W B$, for ω-*DCFL* A and B?

The answer to the first question is in fact an emphatic no: there are many more wadge classes in $DCFL_\omega$ than in REG_ω. Considering the first classes of the Wadge hierarchies of REG_ω and of $DCFL_\omega$, one get:

The restriction of the Wadge hierarchy to $(G_\delta \cap F_\sigma)$-sets in REG_ω has only length ω and it is formed by boolean combinations of (regular) closed sets, as it is proved in [43, 47].

The restriction of the Wadge hierarchy to $(G_\delta \cap F_\sigma)$-sets in $DCFL_\omega$, (defined by $DBPDA$) has length ω^ω.

Duparc gives a proof using descriptive set theory methods [16].

We shall present in future papers a study of the Wadge hierarchy of ω-*DCFL* which is analogous to Wagner's study of ω-regular languages, using notions of chains and superchains, [22].

This will give an (effective) extension of the Wagner Hierarchy, as announced in [17], although included in the set of boolean combinations of G_δ-sets.

We just indicate here how one can generate many more Wadge degrees in $DCFL_\omega$ than in REG_ω.

In his study of the Wadge hierarchy of Borel sets, Duparc defined also the operation of multiplication of an ω-language by a countable ordinal. The operation of multiplication by ω is well adapted to the context of ω-*DCFL*, and it may be defined as follows:

Definition 11.1 (*Duparc*). Let $A \subseteq \Sigma^\omega$ be an ω-language over the alphabet Σ and O_+, O_- be two new letters not in Σ, then $A.\omega$ is defined over the alphabet $\Sigma \cup \{O_+, O_-\}$ by:

$$A.\omega = \bigcup_{n \geqslant 1} (O_+)^n.\Sigma.(\Sigma^\star.\{O_+, O_-\})^{\leqslant (n-1)}.\Sigma^\star.(O_+.A \cup O_-.A^-).$$

Thus, an (infinite) word of $A.\omega$ has an initial prefix in the form $(O_+)^n.a$ for an integer $n \geqslant 1$ and $a \in \Sigma$. Then there are at most n more letters from $\{O_+, O_-\}$ in the word and the last such letter determines whether the suffix following this last letter O_+ or O_- is in A or in A^-.

It is not very difficult to show that whenever A is in $DCFL_\omega$, (and then A^- is also in $DCFL_\omega$ because $DCFL_\omega$ is closed under complementation), the ω-language $A.\omega$ is also in $DCFL_\omega$. But with regard to the Wadge degrees, $d_W^0(A.\omega) = d_W^0(A).\omega$. Starting with the ω-language Σ^ω over the alphabet Σ, of Wadge degree 1, one get languages in $DCFL_\omega$ which have Wadge degrees $\omega, \omega^2, \ldots, \omega^n, \ldots$

These languages are $(G_\delta \cap F_\sigma)$-sets because their Wadge degrees are countable ordinals, but they are not boolean combinations of closed sets because their Wadge degrees are $\geqslant \omega$ (see [15]).

For instance, the ω-*DCFL*

$$\bigcup_{n \geqslant 1} (O_+)^n.\Sigma.(\Sigma^\star.\{O_+, O_-\})^{\leqslant (n-1)}.\Sigma^\star.(O_+).\Sigma^\omega$$

over the alphabet Σ is not Wadge equivalent to any ω-regular language.

11.2. Omega CFL

We have given an answer to a question of [29]: CFL_ω exhausts the Hierarchy of Borel sets of finite rank.

We have shown that the Wadge hierarchy of ω-*CFL* is not effective: we cannot decide the Wadge class of an ω-*CFL*, neither its Borel class.

But a lot of questions are still opened:

Are all omega context free languages Borel sets of finite rank?

Since this paper was written, we have answered to this question, showing that there exist some ω-*CFL* which are non Borel sets, [21].

What is the length of the Wadge hierarchy of Borel ω-*CFL*?

In another paper, we show that it is an ordinal $\geqslant \varepsilon_0$, where ε_0 is the limit of the ordinals α_n defined by $\alpha_0 = \omega$ and $\alpha_{n+1} = \omega^{\alpha_n}$, [20].

Acknowledgements

Thanks to Jean-Pierre Ressayre, Jacques Duparc and Gilles Amiot for many useful and stimulating discussions about Wadge and Wagner Hierarchies.

Thanks also to the anonymous referees for useful comments on a previous version of this paper.

References

[1] J-M. Autebert, J. Berstel, L. Boasson, Context free languages and pushdown automata, in: G. Rozenberg and A. Salomaa Handbook of Formal Languages, vol. 1, Springer, Berlin, 1996.
[2] R. Barua, The Hausdorff–Kuratowski hierarchy of ω-regular languages and a hierarchy of Muller automata, Theoret. Comput. Sci. 96 (1992) 345–360.
[3] J. Berstel, Transductions and Context Free Languages, Teubner Studienbücher Informatik, 1979.
[4] L. Boasson, M. Nivat, Adherences of Languages, J. Comput. System Sci. 20 (3) (1980) 285–309.
[5] J.R. Büchi, Weak second order arithmetic and finite automata, Z. Math. Logik Grundlag. Math. 6 (1960) 66–92.
[6] J.R. Büchi, On a Decision Method in Restricted Second Order Arithmetic, Logic Methodology and Philosophy of Science (Proc. 1960 Internat. Congr.). Stanford University Press, Stanford, CA, 1962, pp. 1–11.
[7] J.R. Büchi, L.H. Landweber, Solving sequential conditions by finite state strategies, Trans. Amer. Math. Soc. 138 (1969) 295–311.
[8] J.R. Büchi, D. Siefkes, in: The Monadic Second Order Theory of All Countable ordinals, Decidable Theories 2, Springer Lecture Notes in Mathematics, vol. 328, Springer, Berlin, 1973.
[9] R.S. Cohen, A.Y. Gold, Theory of ω-languages, Parts 1 and 2, J. Comput. System Sci. 15(2) (1977) 169–184 and 185–208.
[10] R.S. Cohen, A.Y. Gold, ω-computations on deterministic pushdown machines, J. Comput. System Sci. (3) (1978) 257–300.
[11] O. Carton, D. Perrin, Chains and superchains for ω-rational sets, automata and semigroups, Internat. J. Algebra Comput. 7 (6) (1997) 673–695.
[12] O. Carton, D. Perrin, The Wagner hierarchy of ω-rational sets, Internat. J. Algebra Comput. 9 (1999) 597–620.
[13] J. Duparc, La Forme Normale des Boréliens de Rang Fini, Ph.D. Thesis, Université Paris 7, 1995.
[14] J. Duparc, The Normal form of Borel sets, Part 1: Borel sets of finite rank, t.320, Série 1, C.R.A.S., Paris, 1995, pp. 651–656.
[15] J. Duparc, Wadge Hierarchy and Veblen hierarchy: part 1: Borel sets of finite rank, J. Symbolic Logic, (2001). Available from http://www.logigue.jussieu.fr/www.duparc.
[16] J. Duparc, A hierarchy of context free Omega languages, Theoret. Comput. Sci., submitted for publication.
[17] J. Duparc, O. Finkel, J-P. Ressayre, Computer Science and the Fine Structure of Borel Sets, Theoret. Comput. Sci., in preparation, available from http://www.logigue.jussieu.fr/www.duparc.
[18] S. Eilenberg, Automata, Languages and Machines, vol. A, Academic Press, New York, 1974.
[19] J. Engelfriet, H.J. Hoogeboom, X-automata on ω-words, Theoret. Comput. Sci. 110 (1) (1993) 1–51.
[20] O. Finkel, Wadge hierarchy of omega context free languages, Theoret. Comput. Sci., in preparation.

[21] O. Finkel, Borel hierarchy and omega context free languages, Theoret. Comput. Sci., submitted for publication.

[22] O. Finkel, Wadge hierarchy and deterministic omega context free languages, in preparation.

[23] D. Gale, F.M. Stewart, Infinite games with perfect information, Ann. Math. Studies 28 (1953) 245–266.

[24] S. Ginsburg, The Mathematical Theory of Context Free Languages, McGraw-Hill, New York, 1966.

[25] J.E. Hopcroft, J.D. Ullman, Formal Languages and their Relation to Automata, Addison-Wesley, Reading, MA, 1969.

[26] M. Kaminsky, A classification of ω-regular languages, Theoret. Comput. Sci. 36 (1985) 217–229.

[27] K. Kuratowski, Topology, Academic Press, New York, 1966.

[28] L.H. Landweber, Decision problems for ω-automata, Math. Systems Theory 3 (4) (1969) 376–384.

[29] H. Lescow, W. Thomas, Logical specifications of infinite computations, in: J.W. de Bakker, et al., (Eds.), A Decade of Concurrency, Springer Lecture Notes in Computer Science, vol. 803, Springer, Berlin, 1994, pp. 583–621.

[30] M. Linna, On ω-sets associated with context-free languages, Inform. and Control 31 (3) (1976) 272–293.

[31] M. Linna, A decidability result for deterministic ω-context-free languages, Theoret. Comput. Sci. 4 (1977) 83–98.

[32] R. MacNaughton, Testing and Generating infinite sequences by a finite automaton, Inform. and Control 9 (1966) 521–530.

[33] Y.N. Moschovakis, Descriptive Set Theory, North-Holland, Amsterdam, 1980.

[34] M. Nivat, Mots infinis engendrés par une grammaire algébrique, RAIRO Infor. Théor. 11 (1977) 311–327.

[35] M. Nivat, Sur les ensembles de mots infinis engendrés par une grammaire algébrique, RAIRO Infor. Théor. 12 (1978) 259–278.

[36] D. Niwinski, Problem on ω-powers Posed in the Proceedings of the 1990 Workshop, Logics and Recognizable Sets, University of Kiel.

[37] D. Perrin, J.-E. Pin, Infinite Words, Book in preparation, available from http://www.liafa.jussieu.fr/jep/Resumes/InfiniteWords.html.

[38] V. Selivanov, Fine hierarchy of regular ω-languages, Theoret. Comput. Sci. 191 (1998) 37–59.

[39] P. Simonnet, Automates et théorie descriptive, Ph.D. Thesis, Université Paris 7, March 1992.

[40] L. Staiger, Hierarchies of Recursive ω-Languages, J. Inform. Process. Cybernet. EIK 22 (5/6) (1986) 219–241.

[41] L. Staiger, ω-languages, in: G. Rozenberg, A. Salomaa (Eds.), Handbook of Formal Languages, vol. 3, Springer, Berlin, 1997.

[42] L. Staiger, On ω-power languages, in: G. Paun, A. Salomaa (Ed.), New Trends in Formal Languages, Control, Cooperation, and Combinatorics, Lecture Notes in Computer Science, vol. 1218, Springer, Berlin, 1997, pp. 377–393.

[43] L. Staiger, K. Wagner, Automatentheoretische und Automatenfreie Charakterisierungen Topologischer Klassen Regulärer Folgenmengen., Elektron. Informationsverarb. Kybernetik EIK 10 (7) (1974) 379–392.

[44] W. Thomas, Automata on Infinite Objects, in: J. Van Leeuwen (Ed.), Handbook of Theoretical Computer Science, vol. B, Elsevier, Amsterdam, 1990, pp. 133–191.

[45] W. Thomas, On the synthesis of strategies in infinite games, STACS'95, Lecture Notes in Computer Science, vol. 900, Springer, Berlin, 1995, pp. 1–13.

[46] W.W. Wadge, Ph.D. Thesis, Berkeley, 1984.

[47] K. Wagner, On omega regular sets, Inform. and Control 43 (1979) 123–177.

[48] I. Walukiewicz, Pushdown processes: games and model checking, published in CAV'96, LNCS 1102, p. 62–75, full version in Inform. Comput., to appear.

[49] Th. Wilke, H. Yoo, Computing the Wadge degree, the Lifschitz degree and the Rabin index of a regular language of infinite words in polynomial time, in: P.D. Mosses, M. Nielsen, M.I. Schwartzbach (Eds.), TAPSOFT' 95: Theory and Practice of Software Development, Lecture Notes in Computer Science, vol. 915, Springer, Berlin, 1995, pp. 288–302.