Separating Regular Languages with First-Order Logic [*]

Thomas Place Marc Zeitoun
LaBRI, Bordeaux University, France
firstname.lastname@labri.fr

Abstract

Given two languages, a separator is a third language that contains the first one and is disjoint from the second one. We investigate the following decision problem: given two regular input languages of finite words, decide whether there exists a first-order definable separator. We prove that in order to answer this question, sufficient information can be extracted from semigroups recognizing the input languages, using a fixpoint computation. This yields an EXPTIME algorithm for checking first-order separability. Moreover, the correctness proof of this algorithm yields a stronger result, namely a description of a possible separator. Finally, we prove that this technique can be generalized to answer the same question for regular languages of infinite words.

Categories and Subject Descriptors Theory of computation [*Formal languages and automata theory*]: Regular languages; Theory of computation [*Logic*]: Finite Model Theory

Keywords Words, Infinite Words, Regular Languages, Semigroups, First-Order Logic, Expressive Power, Ehrenfeucht-Fraïssé games, Separation.

1. Introduction

In this paper, we investigate a decision problem on word languages: the *separation problem*. The problem is parametrized by a class Sep of *separator languages* and is as follows: given as input two regular word languages, decide whether there exists a third language in Sep containing the first language while being disjoint from the second one.

More than the decision procedure itself, the primary motivation for investigating this problem is the insight it gives on the class Sep. Intuitively, in order to get such a decision procedure, one has to consider *all* instances of separable pairs of languages simultaneously, which requires a strong understanding of the *discriminating power* of Sep. In particular, the separation problem generalizes the *membership problem* whose motivation is also to understand the *expressive power* of the class Sep. In this restricted problem, one only needs to decide whether a single input regular language already belongs to Sep. Since regular languages are closed under complement, testing membership can be achieved by testing whether the input is separable from its complement. Therefore, membership can be reduced to separation.

Solving the membership problem is already known to be a difficult question. However the search for separation algorithms is intrinsically more difficult. In both cases, the problem amounts to finding a language in Sep. However, in the membership case, there is only one candidate which is already known: the input. Therefore, we start with a fixed recognizing device for this unique candidate and powerful tools are available, *viz.* the syntactic monoid of the language, which is now accepted as the natural tool for solving the membership problem for word languages. In the separation case, there can be infinitely many candidates as separators, which means that there is no fixed recognition device that we can use. An even harder question then is to actually construct a separator language in Sep.

First-order logic. In this paper, we choose Sep as the class of *languages definable by first-order sentences* (*i.e.*, sets of words that satisfy some first-order sentence). In this context, the separation problem can be rephrased as follows: given two regular languages as input, decide whether there exists a first-order sentence that is satisfied by all words of the first language, and by no word of the second one. Thus, such a formula witnesses that the input languages are disjoint.

Within the monadic second order logic, which defines on finite words all regular languages, first-order logic is often considered as the yardstick. It is a robust class having several characterizations [5]. It corresponds to star-free languages, and has the same expressive power as linear temporal logic [9]. In particular, it was the first natural class for which the membership problem was proved to be decidable. This result, known as Schützenberger's theorem [12, 18], served as a template and a starting point of a line of research that successfully solved the membership problem for most of the natural classes of regular languages. This makes first-order logic the natural candidate to serve as the example for devising a general approach to the separation problem.

Schützenberger's theorem states that first-order definable languages are exactly those whose syntactic semigroup is aperiodic, *i.e.*, has only trivial subgroups. Since the syntactic semigroup of a language is computable and aperiodicity is a decidable property, this yields a decision procedure for membership. Schützenberger's original proof has been refined over the years. Our own proof for separation by first-order logic actually generalizes a more recent proof by Wilke [23]. Similar results [13, 20] make it possible to decide first-order definability for languages of infinite words, or finite or infinite Mazurkiewicz traces. See [5] for a survey.

[*] Supported by ANR 2010 BLAN 0202 01 FREC

CSL-LICS 2014, July 14–18, 2014, Vienna, Austria.
Copyright © 2014 ACM 978-1-4503-2886-9. . . $15.00.
http://dx.doi.org/10.1145/nnnnnnn.nnnnnnn

Contributions and main ideas. The core of the intuition is to compute the limit of what can be expressed by first-order sentences, with respect to a fixed semigroup. From two regular languages, it is easy to construct a single morphism recognizing them both. We present an algorithm that computes enough information about this morphism to answer the separation question for all pairs of languages that it recognizes. Intuitively, given a morphism from A^+ into a finite semigroup S, we need to compute all pairs $(s, t) \in S \times S$ that cannot be distinguished with first-order logic. By this, we mean that the preimages of s and t in A^+ are not separable by first-order logic: any first-order language containing one preimage has to intersect the other one.

To compute all these pairs, a natural idea is to start with trivial pairs (s, s), and to iteratively compute the missing ones by a fixpoint algorithm. However, for this approach to work, it turns out that one needs to compute even more information than just pairs: we compute FO-indistinguishable sets, i.e. *subsets* of S that cannot be distinguished by first-order logic. Notice that being able to compute these FO-indistinguishable sets also has independent interest from the separation problem. One can view a morphism from A^+ into a finite semigroup as a machine that computes information about input words. The associated FO-indistinguishable sets describe what can and cannot be expressed in first-order logic about these computations.

The connection between the separation problem and the computation of these indistinguishable pairs or subsets has first been observed by Almeida [1]. Rephrased in purely algebraic terms, this amounts to computing the so-called *pointlike sets* for the algebraic variety corresponding to the class of separators under investigation. For the variety corresponding to first-order definable languages, namely that of aperiodic semigroups, pointlike sets have been shown computable by Henckell [6] (see also [7], that answers the problem for even larger classes). Thus, combining this work with Almeida's solves the separation problem by first-order definable languages.

However, this approach does not meet our requirements of understanding how precisely first-order logic can discriminate between two regular languages. Indeed, the motivations and the proofs of [6, 7] are purely algebraic and provide no intuition on the underlying logic. In particular, the techniques only give a yes/no answer to the separation problem, without any insight on a possible separator. Our contributions differ from those of [6, 7] in several ways.

- First, we give a new and self-contained proof that the separation problem by first-order languages is decidable. It is independent from those of [6, 7], and relies on elementary ideas and notions from language theory only, making it accessible to computer scientists. We do not use any involved construction from semigroup theory: we work directly with the logic itself. As mentioned above, the proof refines the algorithm for membership of Wilke [23].
- Second, not only we obtain a yes/no answer, but also an insight of a potential separator, by bounding its *expected quantifier rank*.
- Third, as a consequence of our algorithm, we obtain an EXPTIME upper bound (while complexity is not investigated in [6], a rough analysis yields an EXPSPACE upper bound).
- Fourth, when the input languages are separable, our approach makes it possible to compute a first-order formula that defines a separator, by backtracking the proof of our algorithm.
- Finally, the techniques of [6, 7] are tailored to work with finite words only. We also solve the separation problem for languages of *infinite words* by first-order definable languages, by a smooth extension of our techniques.

Since we do not follow the proofs of [6, 7], it is not surprising that we obtain a different algorithm. However, we are able to derive two variations of it, which allows us to give an alternate and elementary correctness proof of Henckell's original algorithm.

Related work. First-order logic has a number of important fragments. The separation question makes sense when choosing such natural subclasses as classes of separators. It has already been solved for the case of local fragments [17], such as locally testable (LT) and locally threshold testable languages (LTT), although the problem is already NP-hard starting from 2 DFAs as input, while membership is known to be polynomial [2]. It is also decidable for the fragment of first-order logic made of boolean combinations of $\Sigma_1(<)$ sentences, as obtained independently in [4, 16]. Finally, the problem has also been investigated for the fragment FO$^2(<)$ of first-order logic using 2 variables only, and again has been proven to be decidable [16].

Paper outline. We first give the necessary definitions and terminology: languages and semigroups for finite words are defined in Section 2 and first-order logic is defined in Section 3. Section 4 is devoted to the presentation of our algorithm solving first-order separation through the computation of sets that cannot be distinguished by first-order logic. Sections 5 and 6 are then devoted to proving the correctness and completeness of this algorithm, respectively. In Section 7, we present alternate versions of our algorithm. Finally, in Section 8, we generalize the results to infinite words. Due to space limitations, the proofs of Sections 7 and 8 are omitted, and will be made available in the journal version of the paper.

2. Preliminaries

In this section, we provide terminology for words, semigroups and languages. All the definitions are for finite words. We delay the definitions for infinite words to Section 8.

Semigroups. A semigroup is a set S equipped with an associative operation $s \cdot t$ (often written st). A monoid is a semigroup S having an identity element 1_S, *i.e.*, such that $s \cdot 1_S = 1_S \cdot s = s$ for all $s \in S$. Finally, a group is a monoid such that every element s has an inverse s^{-1}, *i.e.*, such that $s \cdot s^{-1} = s^{-1} \cdot s = 1_S$. Given a *finite* semigroup S, it is folklore and easy to see that there is an integer $\omega(S)$ (denoted by ω when S is understood) such that for all s of S, s^ω is idempotent: $s^\omega = s^\omega s^\omega$.

Words, Languages, Morphisms. We fix a finite alphabet A. We denote by A^+ the set of all nonempty finite words and by A^* the set of all finite words over A. If u, v are words, we denote by $u \cdot v$ or by uv the word obtained by the concatenation of u and v. Observe that A^+ (resp. A^*) equipped with the concatenation operation is a semigroup (resp. a monoid).

For convenience, we only consider languages that do not contain the empty word. That is, a language is a subset of A^+ (this does not affect the generality of the argument). We work with regular languages, *i.e.*, languages definable by *nondeterministic finite automata* (NFA).

We shall exclusively work with the algebraic representation of regular languages in terms of semigroups. We say that a language L *is recognized by a semigroup* S if there exists a semigroup morphism $\alpha : A^+ \to S$ and a subset $F \subseteq S$ such that $L = \alpha^{-1}(F)$. It is well known that a language is regular if and only if it can be recognized by a *finite* semigroup.

When working on separation, we consider as input two regular languages L_0, L_1. It will be convenient to have a single semigroup recognizing both of them, rather than having to deal with two objects. Let S_0, S_1 be semigroups recognizing L_0, L_1 together with the associated morphisms α_0, α_1, respectively. Then, $S_0 \times S_1$ equipped with the componentwise multiplication $(s_0, s_1) \cdot (t_0, t_1) =$

(s_0t_0, s_1t_1) is a semigroup that recognizes both L_0 and L_1 with the morphism $\alpha : w \mapsto (\alpha_0(w), \alpha_1(w))$. From now on, we work with such a single semigroup recognizing both languages, and we call α the associated morphism.

Semigroup of Subsets. As explained in the introduction, our separation algorithm works by computing special subsets of a semigroup recognizing both input languages. Intuitively, these subsets are those that cannot be distinguished by first-order logic. More precisely, by *special subset*, we mean that any first-order definable language has an image under α that either contains *all* elements of the subset, or *none* of them. For this reason, we work with the semigroup of subsets.

Let S be a semigroup. Observe that the set 2^S of subsets of S equipped with the operation

$$T \cdot T' = \{s \cdot s' \mid s \in T, \quad s' \in T'\}$$

is a semigroup, that we call the *semigroup of subsets of* S. Note that S can be viewed as a subsemigroup of 2^S, since S is isomorphic to the semigroup $\big\{\{s\} \mid s \in S\big\} \subseteq 2^S$. We denote by $\mathcal{S}, \mathcal{T}, \mathcal{R}, \dots$ subsemigroups of a semigroup of subsets.

Downset $\downarrow \mathcal{S}$, and Expansion $\uparrow \mathcal{S}$. For $\mathcal{S} \subseteq 2^S$ a subsemigroup of 2^S, let us define two sets containing $\mathcal{S}$:

- The *downset* of $\mathcal{S}$ consists of all subsets of sets in $\mathcal{S}$:
$$\downarrow \mathcal{S} = \{T \in 2^S \mid \exists T' \in \mathcal{S}, \quad T \subseteq T'\}.$$
- The *expansion* of $\mathcal{S}$ consists of all unions of sets in $\mathcal{S}$:
$$\uparrow \mathcal{S} = \Big\{ \bigcup_{T \in \mathcal{T}} T \mid \mathcal{T} \subseteq \mathcal{S} \Big\}.$$

Clearly, we have $\mathcal{S} \subseteq \downarrow \mathcal{S}$ and $\mathcal{S} \subseteq \uparrow \mathcal{S}$. It is also easy to check that since $\mathcal{S}$ is a semigroup, so are $\downarrow \mathcal{S}$ and $\uparrow \mathcal{S}$.

Union $[\![\mathcal{S}]\!]$. For $\mathcal{S} \subseteq 2^S$ a subsemigroup of 2^S, we define $[\![\mathcal{S}]\!] \subseteq S$, the *union* of $\mathcal{S}$, as the set

$$[\![\mathcal{S}]\!] = \bigcup_{T \in \mathcal{S}} T \subseteq S$$

We call *index* of $\mathcal{S}$ the size of its union, *i.e.*, $|[\![\mathcal{S}]\!]|$.

3. First-Order Logic and Separation

This section is devoted to the definition of first-order logic on words. See [5, 21] for details on these classical notions.

First-Order Logic. We view words as logical structures composed of a sequence of positions labeled over A. We denote by $<$ the linear order over the positions. We work with first-order logic FO($<$) using unary predicates P_a for all $a \in A$ that select positions labeled with an a, as well as a binary predicate for the linear order $<$. A language L is said to be *first-order definable* if there exists an FO($<$) formula φ such that $L = \{w \in A^+ \mid w \models \varphi\}$. We write FO the class of first-order definable languages.

There are many known characterizations of the class of first-order definable languages. Kamp's Theorem [9] states that it is exactly the class of languages definable in linear temporal logic LTL. It was then also proved that this is also the class of star-free languages [12] (*i.e.*, languages definable by a regular expression that may use complement, but does not use the Kleene star). This result bridged the gap with Schützenberger's Theorem [18], which characterizes star-free languages as those that are recognized by an aperiodic semigroup. These results were later generalized to infinite words [10, 13, 20].

Let φ be an FO($<$) formula. The *quantifier rank* of φ is the length of the largest sequence of nested quantifiers in φ. We denote by FO[k] the class of languages that are definable by FO($<$) formulas of quantifier rank at most k. By definition, we have FO $= \bigcup_{k \in \mathbb{N}}$ FO[k]. For $w, w' \in A^+$ and $k \in \mathbb{N}$, write

$w \equiv_k w'$ if w, w' satisfy the same FO($<$) formulas of quantifier rank at most k.

One can verify that $\equiv_k$ is an equivalence relation of finite index. Therefore, there are finitely many FO[k] languages.

Ehrenfeucht-Fraïssé games. It is well known that the expressive power of logics can be expressed in terms of games. These games are called Ehrenfeucht-Fraïssé games. We define below the specific Ehrenfeucht-Fraïssé game for FO($<$).

The board of the game consists of two words $w, w' \in A^+$ and there are two players called Spoiler and Duplicator. The game is set to last a predefined number k of rounds. When the game starts, both players have k pebbles.

At the start of each round ℓ, Spoiler chooses either w or w'. If he chose w (resp. w') he drops a pebble on some position x_ℓ in w (resp. x'_ℓ in w'). Duplicator must answer by dropping a pebble on some position x'_ℓ in w' (resp. x_ℓ in w). Moreover, Duplicator must ensure that all pebbles that have been placed up to this point verify the following condition: for all $i, j \leqslant \ell$, x_i, x'_i have the same label, and $x_i < x_j$ if and only if $x'_i < x'_j$.

Duplicator wins if she manages to play for all k rounds, while Spoiler wins as soon as Duplicator is unable to play. It is classical that the equivalence $\equiv_k$ can be redefined in terms of Ehrenfeucht-Fraïssé games (see [8, 11, 19] for example).

Lemma 1. *For all $k \in \mathbb{N}$ and $w, w' \in A^+$, we have $w \equiv_k w'$ if and only if Duplicator has a winning strategy for playing k rounds in the Ehrenfeucht-Fraïssé game played on w, w'.*

Lemma 1 has two simple and well-known consequences that will be used to prove our algorithm. First, using Ehrenfeucht-Fraïssé games, it is easy to show that $\equiv_k$ is a congruence for all k.

Lemma 2. *(1) If $u_1 \equiv_k v_1$ and $u_2 \equiv_k v_2$, then $u_1 \cdot u_2 \equiv_k v_1 \cdot v_2$. (2) For all $u \in A^+$ and all $k > 0$, we have $u^{2^k} \equiv_k u^{2^k - 1}$.*

Proof. For item 1, by Lemma 1, Duplicator has a winning strategy in the k-round game played on u_i and v_i for $i = 1, 2$. These strategies can be easily combined into a winning strategy in the k-round game played on $u_1 \cdot u_2$ and $v_1 \cdot v_2$. By Lemma 1, it follows that $u_1 \cdot u_2 \equiv_k v_1 \cdot v_2$.

Property 2 is shown similarly by induction on k, again using Lemma 1. See [19] for details. □

Separation. Given languages L, L_0, L_1, we say that L *separates* L_0 from L_1 if

$$L_0 \subseteq L \text{ and } L_1 \cap L = \varnothing.$$

The pair (L_0, L_1) is said to be *FO-separable* if some language $L \in$ FO separates L_0 from L_1. Since FO is closed under complement, (L_0, L_1) is FO-separable if and only if (L_1, L_0) is. Therefore, we simply say that L_0 and L_1 are FO-separable in this case. We use the same terminology for the class FO[k]. Note that since there are finitely many FO[k] languages for any fixed k, FO[k]-separability is easy: it suffices to test all of these potential separators. In particular, if L_0, L_1 are FO[k]-separable, then there exists a smallest separator: the saturation of L_0 by $\equiv_k$. Note that this is not true for full first-order logic, since removing a single word from an FO language yields again an FO language (however, the formula defining this new language might have a larger quantifier rank). Since languages in FO[k] are unions of $\equiv_k$-classes, we obtain the following useful fact.

Fact 3. Two languages L_0 and L_1 are FO[k]-separable if and only if for all $w_0 \in L_0$ and $w_1 \in L_1$, we have $w_0 \not\equiv_k w_1$.

Example 1. Let $K_0 = (aa)^*$, $K_1 = (aa)^*a$ and

$$L_0 = (bK_0bK_1)^+,$$
$$L_1 = (bK_0bK_1)^*bK_0.$$

It is well known that a^{2^k} and a^{2^k-1} cannot be distinguished by any FO-sentence of quantifier rank k, see *e.g.* [19]. Therefore, K_0 and K_1 are not FO-separable. Reusing this argument then shows that L_0 and L_1 are not FO-separable either. We shall explain below how this is detected by our algorithm.

4. FO-indistinguishable Sets for a Morphism

In this section, we define our main tool for solving the separation problem for first-order logic: FO-indistinguishable sets. The idea behind this notion is the following. Let $\alpha : A^+ \to S$ be a morphism into a finite semigroup. Given a natural k, one associates to each $\equiv_k$-class τ in A^+ the subset $\alpha(\tau)$ of S, which consists of the images under α of all words in τ. This information is exactly what we need to answer the separation question by FO[k]-definable languages for any pair of languages that are both recognized by α. Indeed, two languages L_0, L_1 recognized by α are *not* FO[k]-separable if and only if there exists such a subset intersecting both $\alpha(L_0)$ and $\alpha(L_1)$.

Observe that when k gets larger, these subsets can only get smaller, because $\equiv_k$-classes are unions of $\equiv_{k+1}$-classes. Since S is finite, the refinement stabilizes at some index ℓ: the subsets generated as images of $\equiv_\ell$-classes are the same as those generated as images of $\equiv_k$-classes, for all $k \geqslant \ell$. These stabilized subsets are what we call FO-indistinguishable sets.

The application of the notion of FO-indistinguishable sets to the separation problem is twofold.

- First, being able to compute all FO-indistinguishable sets for α provides a yes-no answer to the separation question for any pair of languages recognized by α.
- Moreover, the stabilization index ℓ is also of particular interest: it is a bound such that if there exists a separator, then it can be chosen with quantifier rank ℓ.

The section is organized as follows. First we give a formal definition of FO-indistinguishable sets and we state a reduction from the separation problem to the computation of these sets. In the second subsection, we give a fixpoint algorithm for computing all FO-indistinguishable sets associated to a given morphism α. Finally, in the last subsection, we run this fixpoint algorithm on the languages of Example 1.

4.1 Definition and reduction from the separation problem

FO-indistinguishable sets. Let $\alpha : A^+ \to S$ be a semigroup morphism. We define the following subsets of 2^S:

- $\mathcal{I}_k[\alpha]$, the set of *FO*[k]*-indistinguishable* sets for α.
- $\mathcal{I}[\alpha]$, the set of *FO-indistinguishable* sets for α.

Let $T = \{s_1, \dots, s_n\} \subseteq S$. We have

- $T \in \mathcal{I}_k[\alpha]$ if there exist $w_1, \dots, w_n \in A^+$ with
  - $w_1 \equiv_k w_2 \equiv_k \cdots \equiv_k w_n$, and
  - $\alpha(w_1) = s_1, \dots, \alpha(w_n) = s_n$.
- $T \in \mathcal{I}[\alpha]$ if for all $k \in \mathbb{N}$, we have $T \in \mathcal{I}_k[\alpha]$.

From the definitions and from the inclusion $\equiv_{k+1} \subseteq \equiv_k$, we obtain the following facts.

Fact 4. (a) $\mathcal{I}_k[\alpha] \supseteq \mathcal{I}_{k+1}[\alpha] \supseteq \mathcal{I}[\alpha]$ for all $k \geqslant 0$.
(b) $\mathcal{I}[\alpha] = \bigcap_k \mathcal{I}_k[\alpha]$.

Fact 5. Both $\mathcal{I}_k[\alpha]$ and $\mathcal{I}[\alpha]$ are closed under taking subsets: $\mathcal{I}_k[\alpha] = \downarrow\mathcal{I}_k[\alpha]$ and $\mathcal{I}[\alpha] = \downarrow\mathcal{I}[\alpha]$.

Conversely however, it may be the case that $\{r, s\}$, $\{s, t\}$ and $\{t, r\}$ are all FO-indistinguishable, while $\{r, s, t\}$ is not.
Lemma 2 entails the following fact.

Fact 6. All $\mathcal{I}_k[\alpha]$ and $\mathcal{I}[\alpha]$ are subsemigroups of 2^S.

As stated in Fact 4 (a), the sets in $\mathcal{I}_k[\alpha]$ can only get refined as k gets larger. Therefore, they stabilize at some index ℓ. However, it may be the case that $\mathcal{I}_k[\alpha] = \mathcal{I}_{k+1}[\alpha]$ even if stabilization is not reached yet. This rules out the naive search for this index. In the following proposition, we give a bound on this stabilization index depending on the size of A and S.

Proposition 7. *For all $k \geqslant |A|2^{|S|^2}$, we have $\mathcal{I}_k[\alpha] = \mathcal{I}[\alpha]$.*

Proposition 7 yields a first algorithm for computing $\mathcal{I}[\alpha]$. Indeed, when k is fixed, one can easily compute $\mathcal{I}_k[\alpha]$ using a brute-force algorithm that enumerates all equivalence classes of $\equiv_k$. However, since the number of such classes is non-elementary in k, this algorithm is very slow. We will present a more efficient fixpoint algorithm at the end of the section. In Section 6, we will obtain the bound $|A|2^{|S|^2}$ as a corollary of the completeness proof of this more efficient algorithm.

From FO-separation to FO-indistinguishable sets. We now make the link with the separation problem. The following theorem shows that computing $\mathcal{I}[\alpha]$ answers the FO-separation problem for input languages recognized by α. Moreover, the second part of the theorem yields a bound on the expected quantifier rank of a separator.

Theorem 8. *Let L_0, L_1 be two regular languages recognized by a morphism $\alpha : A^+ \to S$ into a finite semigroup. Then, L_0 and L_1 are FO-separable if and only if for all $T \in \mathcal{I}[\alpha]$, $\alpha(L_0) \cap T = \varnothing$ or $\alpha(L_1) \cap T = \varnothing$.*

Moreover, if L_0, L_1 are FO-separable, then the actual separator can be chosen with quantifier rank $|A|2^{|S|^2}$.

Proof. Suppose first that L_0 and L_1 are FO-separable, that is, FO[k]-separable for some k. Let $T \in \mathcal{I}[\alpha]$. By contradiction, assume that there exist $s_0 \in \alpha(L_0) \cap T$ and $s_1 \in \alpha(L_1) \cap T$. Then $s_0, s_1 \in T \in \mathcal{I}[\alpha] \subseteq \mathcal{I}_k[\alpha]$. By definition of $\mathcal{I}_k[\alpha]$, there exist $w_0 \in \alpha^{-1}(s_0) \subseteq L_0$ and $w_1 \in \alpha^{-1}(s_1) \subseteq L_1$ such that $w_0 \equiv_k w_1$, which by Fact 3 contradicts FO[k]-separability.

Conversely, assume that for all $T \in \mathcal{I}[\alpha]$, either $\alpha(L_0) \cap T$ or $\alpha(L_1) \cap T$ is empty. Then by Proposition 7, the same property holds for all $T \in \mathcal{I}_\ell[\alpha]$, for $\ell = |A|2^{|S|^2}$. Hence by definition of $\mathcal{I}_\ell[\alpha]$, for all $w_0 \in L_0$ and $w_1 \in L_1$, we have $w_0 \not\equiv_\ell w_1$. So, again by Fact 3, L_0 and L_1 are FO[ℓ]-separable. This proves the equivalence and the last assertion of the statement. □

4.2 An algorithm to compute FO-indistinguishable sets

Let $\alpha : A^+ \to S$ be a morphism into a finite semigroup. We describe a fixpoint algorithm for computing $\mathcal{I}[\alpha]$. We start from sets that are trivially in $\mathcal{I}[\alpha]$ (*i.e.*, singletons $\{\alpha(w)\}$) and then use a saturation procedure to generate more sets, until we reach a fixpoint. Let us first describe this saturation procedure.

Saturation. Let $\mathcal{S}$ be a subsemigroup of 2^S. We define $\mathrm{Sat}(\mathcal{S})$, the *saturation* of $\mathcal{S}$, as the subsemigroup of 2^S generated by

$$\mathcal{S} \cup \{T^\omega \cup T^{\omega+1} \mid T \in \mathcal{S}\}. \qquad (1)$$

The fixpoint algorithm consists in iteratively applying saturation until stabilization, starting from $\mathcal{S} = \alpha(A^+)$, viewed as a subsemigroup of 2^S consisting of singletons. We set $\mathrm{Sat}^0(\mathcal{S}) = \mathcal{S}$, and $\mathrm{Sat}^{i+1}(\mathcal{S}) = \mathrm{Sat}(\mathrm{Sat}^i(\mathcal{S}))$ for all $i \in \mathbb{N}$. By definition, for all $i \in \mathbb{N}$, $\mathrm{Sat}^i(\mathcal{S}) \subseteq \mathrm{Sat}^{i+1}(\mathcal{S}) \subseteq 2^S$. Therefore, there exists i

such that $\mathrm{Sat}^i(\mathcal{S}) = \mathrm{Sat}^{i+1}(\mathcal{S})$. We denote this subsemigroup by $\mathrm{Sat}^*(\mathcal{S})$. Note that computing $\mathrm{Sat}^*(\mathcal{S})$ from $\mathcal{S}$ is straightforward, by repeatedly applying saturation. In the following proposition, we state correctness and completeness of our algorithm: sets in $\mathcal{I}[\alpha]$ are exactly the subsets of elements of $\mathrm{Sat}^*(\alpha(A^+))$.

Proposition 9. *Let $\ell = |A|2^{|S|^2}$. Then we have*

$$\mathcal{I}[\alpha] = \mathcal{I}_\ell[\alpha] = \downarrow \mathrm{Sat}^*(\alpha(A^+)).$$

Since $\mathrm{Sat}^*(\mathcal{S})$ is computable, Proposition 9 immediately implies that so is $\mathcal{I}[\alpha]$. Using Theorem 8, this yields the decidability of the separation problem for first-order logic. Moreover, a simple analysis of the saturation procedure shows an EXPTIME upper bound on the complexity of the problem.

Corollary 10. *Let L_0, L_1 be two regular languages recognized by a morphism $\alpha : A^+ \to S$ into a finite semigroup. Then one can decide in* EXPTIME *with respect to $|S|$ whether L_0, L_1 are FO-separable.*

Proof. By Theorem 8, it suffices to prove that one can compute $\mathcal{I}[\alpha]$ in EXPTIME in the size of S. Indeed, it then suffices to test whether there exists $T \in \mathcal{I}[\alpha]$ such that $\alpha(L_1) \cap T \neq \varnothing$ and $\alpha(L_2) \cap T \neq \varnothing$. This can also be achieved in EXPTIME by testing all possible candidates T. By Proposition 9, we know that computing $\mathcal{I}[\alpha]$ can be done by computing $\mathrm{Sat}^*(\alpha(A^+))$.

By definition, $\mathrm{Sat}^*(\alpha(A^+)) \subseteq 2^S$, therefore $\mathrm{Sat}^*(\alpha(A^+)) = \mathrm{Sat}^{|2^S|}(\alpha(A^+))$. This means that the number of steps the algorithm needs to reach the fixpoint is at most exponential in S. Therefore, it suffices to prove that each step can be done in EXPTIME to conclude that the whole computation can also be done in EXPTIME. Each step requires computing $T^\omega \cup T^{\omega+1}$ for at most $|2^S|$ subsets T. Each computation can be done in EXPTIME, since T^ω is equal to some T^m for $m \leqslant |2^S|$ such that $T^m = T^{2m}$. Finally, computing the subsemigroup of 2^S generated by a subset of 2^S can also be done in EXPTIME. □

Proposition 7 is a simple consequence of Proposition 9. Indeed, for $k \geqslant \ell$, we have $\mathcal{I}[\alpha] \subseteq \mathcal{I}_k[\alpha] \subseteq \mathcal{I}_\ell[\alpha]$ by Fact 4 (a). Since for $\ell = |A|2^{|S|^2}$, Proposition 9 yields $\mathcal{I}_\ell[\alpha] = \mathcal{I}[\alpha]$, we obtain $\mathcal{I}[\alpha] = \mathcal{I}_k[\alpha]$, which is exactly Proposition 7.

An interesting observation about our saturation algorithm is that it can be viewed as a generalization of Schützenberger's Theorem [12, 18]. Indeed, a language is first-order *definable* if and only if it can be recognized by an aperiodic semigroup. One definition of aperiodicity is that a semigroup is aperiodic if and only if it satisfies the identity $s^\omega = s^{\omega+1}$. The counterpart to this definition can be found in the main operation of our saturation procedure, Operation (1). This raises another question: could Operation (1) be replaced to reflect alternate definitions of aperiodicity while retaining Proposition 9? We shall see in Section 7 that this is indeed possible.

It now remains to prove Proposition 9. We show that

$$\mathcal{I}[\alpha] \subseteq \mathcal{I}_\ell[\alpha] \subseteq \downarrow \mathrm{Sat}^*(\alpha(A^+)) \subseteq \mathcal{I}[\alpha].$$

The first inclusion is obvious by Fact 4. In Section 5, we prove that $\downarrow \mathrm{Sat}^*(\alpha(A^+)) \subseteq \mathcal{I}[\alpha]$. This corresponds to correctness of the algorithm: all computed sets indeed belong to $\mathcal{I}[\alpha]$. Finally, in Section 6, we focus on the proof of the most difficult direction, which is the second one: $\mathcal{I}_\ell[\alpha] \subseteq \downarrow \mathrm{Sat}^*(\alpha(A^+))$. It implies completeness of the algorithm, that is, that any FO-indistinguishable set for α – *i.e.*, belonging to $\mathcal{I}[\alpha]$ – is actually contained in some element of the set $\mathrm{Sat}^*(\alpha(A^+))$ computed by the algorithm.

We finish this section by running the algorithm, to show that it detects that the languages of Example 1 are not FO-separable.

4.3 Example 1, contd.

To start our algorithm, we first need a semigroup morphism recognizing both L_0 and L_1. Observe that both languages are recognized by the automaton below, with 4 as final state for L_0, and 2 as final state for L_1. Therefore, its transition semigroup S recognizes both languages[1]. The recognizing morphism $\alpha : A^+ \to S$ thus maps a word to the partial function from states to states that it defines. We still denote the images of letters a, b by $a, b \in S$, respectively. It is then easy to see that $L_0 = \alpha^{-1}(b^2a)$ and $L_1 = \alpha^{-1}(\{b, b^2ab\})$.

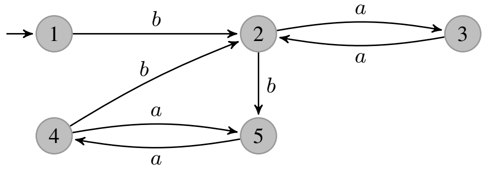


Figure 1. Automaton recognizing both L_0 and L_1

We use Theorem 8 to show that L_0 and L_1 are not FO-separable: we have to find an FO-indistinguishable set $T \in \mathcal{I}[\alpha]$ intersecting both $\alpha(L_0)$ and $\alpha(L_1)$. We claim that $\{b^2a, b^2ab\} \in \alpha(L_0) \times \alpha(L_1)$ is indeed detected as FO-indistinguishable. We actually show that it is computed as an element of $\downarrow \mathrm{Sat}^*(\alpha(A^+))$, which by Proposition 9 implies that it is FO-indistinguishable.

The algorithm starts with $\mathrm{Sat}^0(\mathcal{S})$ consisting of singletons. Then, note that $\{a\}^\omega = \{a^2\}$ and $\{a\}^{\omega+1} = \{a\}$. Therefore, by definition of Operation (1), we have $\{a, a^2\} \in \mathrm{Sat}(\mathcal{S})$. Since $\mathrm{Sat}(\mathcal{S})$ is a subsemigroup, the algorithm also computes $X = \{a, aa\} \cdot \{b\} = \{ab, aab\}$ as an element of $\mathrm{Sat}(\mathcal{S})$. Now by (1), $Y = X^\omega \cup X^{\omega+1} \in \mathrm{Sat}^2(\mathcal{S})$. Computing Y shows that $\{bab, bab^2\} \subseteq Y$. Finally, since $\mathrm{Sat}^2(\mathcal{S})$ is a semigroup, it contains $T = \{b\} \cdot Y \cdot \{a, a^2\}$, which itself contains $\{b^2a, b^2ab\}$, as claimed.

5. Correctness of the Algorithm

In this section we prove correctness of our algorithm computing FO-indistinguishable sets, that is the inclusion $\downarrow \mathrm{Sat}^*(\alpha(A^+)) \subseteq \mathcal{I}[\alpha]$ in Proposition 9. Recall that we work with a morphism $\alpha : A^+ \to S$ into a finite semigroup S. We prove the following proposition.

Proposition 11. *For every $k \in \mathbb{N}$, $\mathrm{Sat}^*(\alpha(A^+)) \subseteq \mathcal{I}_k[\alpha]$.*

By Fact 4, we have $\mathcal{I}[\alpha] = \bigcap_k \mathcal{I}_k[\alpha]$. Therefore, it is immediate from Proposition 11 that $\mathrm{Sat}^*(\alpha(A^+)) \subseteq \mathcal{I}[\alpha]$. Since $\downarrow \mathcal{I}[\alpha] = \mathcal{I}[\alpha]$ by Fact 5, it follows that $\downarrow \mathrm{Sat}^*(\alpha(A^+)) \subseteq \mathcal{I}[\alpha]$. It remains to prove Proposition 11, which we do in the rest of the section.

We show by structural induction that $\downarrow \mathrm{Sat}^*(\alpha(A^+))$ consists of FO-indistinguishable sets only. We start from singletons, which are obviously FO-indistinguishable. Then, we apply:

- Operation (1), which can be seen, using Ehrenfeucht-Fraïssé games, to preserve FO-indistinguishability.
- Closure under subsemigroup, which preserves it by Fact 6.
- Finally, closure under $\downarrow$, which also preserves it by Fact 5.

Formally, let $k \in \mathbb{N}$ and $T \in \mathrm{Sat}^*(\alpha(A^+))$, and let us prove that $T \in \mathcal{I}_k[\alpha]$. By definition $T \in \mathrm{Sat}^i(\alpha(A^+))$ for some $i \in \mathbb{N}$. We proceed by induction on i. For $i = 0$, this is obvious since

[1] Recall that the transition semigroup consists of partial mappings induced by words from the state set to itself. It is easy to see that it recognizes the language accepted by the automaton, see [15, Sec. 3.1].

$\mathrm{Sat}^0(\alpha(A^+)) = \big\{\{\alpha(w)\} \mid w \in A^+\big\}$, and by definition, any singleton $\{\alpha(w)\}$ is in $\mathcal{I}_k[\alpha]$.

Assume now that $i \geqslant 1$. Recall that $\mathrm{Sat}^i(\alpha(A^+))$ is the semigroup generated by

$$\mathcal{R} = \mathrm{Sat}^{i-1}(\alpha(A^+)) \cup \{T^\omega \cup T^{\omega+1} \mid T \in \mathrm{Sat}^{i-1}(\alpha)\}.$$

Assume first that the result is proved for every set in $\mathcal{R}$ and set $T \in \mathrm{Sat}^i(\alpha(A^+))$. Then $T = T_1 \cdots T_n$ with $T_1, \dots, T_n \in \mathcal{R}$. By assumption $T_1, \dots, T_n \in \mathcal{I}_k[\alpha]$. By Fact 6, $\mathcal{I}_k[\alpha]$ is a semigroup. Therefore, $T = T_1 \cdots T_n \in \mathcal{I}_k[\alpha]$.

It remains to prove that all sets in $\mathcal{R}$ belong to $\mathcal{I}_k[\alpha]$. Let $R \in \mathcal{R}$. If $R \in \mathrm{Sat}^{i-1}(\alpha(A^+))$, this is by induction hypothesis. Therefore, assume that $R = T^\omega \cup T^{\omega+1}$ for a set $T \in \mathrm{Sat}^{i-1}(\alpha(A^+))$. By induction hypothesis, $T \in \mathcal{I}_k[\alpha]$. By definition, this means that there exists a set of words $W \subseteq A^+$ such that $\alpha(W) = T$ and for all $w, w' \in W$, we have $w \equiv_k w'$. Consider the set of words $W' = W^{2^k\omega} \cup W^{2^k\omega+1}$. By definition, $\alpha(W') = R$. Therefore, it suffices to prove that for any two words $w, w' \in W'$, we have $w \equiv_k w'$ to conclude that $R \in \mathcal{I}_k[\alpha]$.

Let $w \in W$ be some arbitrary chosen word. By Lemma 2 (1), it is immediate that any word of W' is $\equiv_k$-equivalent to either $u_0 = w^{2^k\omega} \in W^{2^k\omega}$ or to $u_1 = w^{2^k\omega+1} \in W^{2^k\omega+1}$. To conclude that all words of W' are $\equiv_k$-equivalent, it remains to prove that $u_0 \equiv_k u_1$, which follows directly from Lemma 2 (2).

6. Completeness of the Algorithm

In this section, we prove the most interesting inclusion from Proposition 9: $\mathcal{I}_\ell[\alpha] \subseteq {\downarrow}\mathrm{Sat}^*(\alpha(A^+))$ for $\ell = |A|2^{|S|^2}$.

For the rest of the section, we fix a morphism $\alpha : A^+ \to S$ into a finite semigroup. Recall that we identify $\alpha(A^+)$ with the subsemigroup $\big\{\{\alpha(w)\} \mid w \in A^+\big\}$, so we view α as a morphism into this subsemigroup of 2^S. We prove our result in a proposition that is itself proved by induction. In order to state this proposition, we need additional terminology.

Set generated by an $\equiv_k$-class. Let B be an alphabet, $\mathcal{S}$ be a subsemigroup of 2^S, and $\beta : B^+ \to \mathcal{S}$ be a morphism. For $k \in \mathbb{N}$ and τ an $\equiv_k$-class of words in B^+, the *β-generated set by τ* is $\llbracket \beta(\tau) \rrbracket = \bigcup_{w \in \tau} \beta(w)$.

The main idea behind the proof is that for k large enough, the α-generated sets by $\equiv_k$-classes are all computed by Sat^*. Let us formalize this result as an inductive property.

Proposition 12. *Let $\mathcal{S}$ be a subsemigroup of 2^S and $\beta : B^+ \to \mathcal{S}$ be a surjective morphism. Set $k \geqslant |B| \cdot 2^{|\llbracket \mathcal{S} \rrbracket|^2}$. Then for every $\equiv_k$-class τ, we have $\llbracket \beta(\tau) \rrbracket \in {\downarrow}\mathrm{Sat}^*(\mathcal{S})$.*

Before proving Proposition 12, we explain how to use it to prove the inclusion $\mathcal{I}_\ell[\alpha] \subseteq {\downarrow}\mathrm{Sat}^*(\alpha(A^+))$ of Proposition 9.

Proof of Completeness in Proposition 9. Put $\mathcal{S} = \alpha(A^+)$, viewed as a subsemigroup of 2^S. We define a surjective morphism $\beta : A^+ \to \mathcal{S}$ by $\beta(w) = \{\alpha(w)\}$. Recall that $\ell = |A|2^{|S|^2}$ and let $T \in \mathcal{I}_\ell[\alpha]$.

By definition of $\mathcal{I}_\ell[\alpha]$, there exists an $\equiv_\ell$-class τ such that $T \subseteq \llbracket \beta(\tau) \rrbracket$. By definition, $\llbracket \mathcal{S} \rrbracket \subseteq S$, hence $|\llbracket \mathcal{S} \rrbracket| \leqslant |S|$. Therefore, $\ell \geqslant |A| \cdot 2^{|\llbracket \mathcal{S} \rrbracket|^2}$ and we can apply Proposition 12, so that $\llbracket \beta(\tau) \rrbracket \in {\downarrow}\mathrm{Sat}^*(\mathcal{S})$. Since $T \subseteq \llbracket \beta(\tau) \rrbracket$ and ${\downarrow}\mathrm{Sat}^*(\mathcal{S})$ is closed under taking subsets, we get $T \in {\downarrow}\mathrm{Sat}^*(\mathcal{S})$. We conclude that $\mathcal{I}_\ell[\alpha] \subseteq {\downarrow}\mathrm{Sat}^*(\mathcal{S})$. □

It remains to prove Proposition 12. We set β, $k \geqslant |B| \cdot 2^{|\llbracket \mathcal{S} \rrbracket|^2}$ and τ a $\equiv_k$-class as in the statement of the proposition. We need to prove that $\llbracket \beta(\tau) \rrbracket \in {\downarrow}\mathrm{Sat}^*(\mathcal{S})$. The proof is a generalization of Wilke's argument [23] for deciding first-order definability. We proceed by induction on the following parameters listed by order of importance:

1. the index $|\llbracket \mathcal{S} \rrbracket|$ of $\mathcal{S}$,
2. the size of B.

The proof is divided in three main parts:

- first, we consider the case when $|B| = 1$.
- otherwise, we distinguish two subcases, depending on a property of β called *tameness*.

6.1 Special Case: $|B| = 1$.

In that case, B is a singleton $\{b\}$. With this hypothesis, we actually prove a slightly stronger result than Proposition 12, which will be useful later in the induction. Recall that τ is a $\equiv_k$-class over B.

Lemma 13. $\llbracket \beta(\tau) \rrbracket \in \mathrm{Sat}^*(\mathcal{S})$.

Note that by definition, $\mathrm{Sat}^*(\mathcal{S}) \subseteq {\downarrow}\mathrm{Sat}^*(\mathcal{S})$. Therefore, Proposition 12 is indeed a consequence of Lemma 13 when $|B| = 1$.

Proof of Lemma 13. Using Lemma 2, it is easy to see that any $\equiv_k$-class over the singleton alphabet $\{b\}$ is either a singleton $\{b^n\}$, or of the form $\{b^k \mid k \geqslant K\}$ for some $K \in \mathbb{N}$.

Let $w \in \tau$ be a word of minimal length. By hypothesis, $w = b^n$ for some $n \geqslant 1$. A standard semigroup theory argument shows that there exists $m \leqslant |\mathcal{S}| \leqslant 2^{|\llbracket \mathcal{S} \rrbracket|}$ such that $\beta(b^m) = \beta(b^\omega)$.

If $n \leqslant m$, it is simple to see that $k > n$, whence we deduce that $\tau = \{w\}$. Hence $\llbracket \beta(\tau) \rrbracket = \beta(w) \in \mathrm{Sat}^*(\mathcal{S})$ and we are done.

Otherwise, $n > m$. If $\llbracket \beta(\tau) \rrbracket \neq \beta(w)$, then by choice of k and by the preliminary remark, we have $\llbracket \beta(\tau) \rrbracket = \bigcup_{i \geqslant 0} \beta(b^{\omega+i})$. To conclude, we prove that $\bigcup_{i \geqslant 0} \beta(b^{\omega+i}) \in \mathrm{Sat}^*(\mathcal{S})$. Note that

$$\bigcup_{i \geqslant 0} \beta(b^{\omega+i}) = (\beta(b)^\omega \cup \beta(b)^{\omega+1}) \cdots (\beta(b)^\omega \cup \beta(b)^{2\omega-1}).$$

Therefore, it suffices to prove that for all i, we have

$$\beta(b)^\omega \cup \beta(b)^{\omega+i} \in \mathrm{Sat}^*(\mathcal{S}).$$

By definition, for any $i \geqslant 0$, $\beta(b)^{\omega+i} = \beta(b^{\omega+i}) \in \mathcal{S} \subseteq \mathrm{Sat}^*(\mathcal{S})$. Moreover, observe that

$$\beta(b)^\omega \cup \beta(b)^{\omega+i} = (\beta(b)^{\omega+i})^\omega \cup (\beta(b)^{\omega+i})^{\omega+1}.$$

Therefore the result is immediate by Operation (1). □

This terminates the case $|B| = 1$. For the remainder of the proof, we now assume that $|B| \geqslant 2$. As explained above, we distinguish two cases depending on a property of β.

Tameness. We say that β is *tame* if for all $b \in B$, all $t \in \llbracket \mathcal{S} \rrbracket$, there exist $R_\ell, R_r \in \mathcal{S}$ such that $t \in \beta(b) \cdot R_r$ and $t \in R_\ell \cdot \beta(b)$.

6.2 Case 1: β is tame

This is the base case: we don't use induction. We use tameness to prove that $\llbracket \mathcal{S} \rrbracket \in \mathrm{Sat}^*(\mathcal{S})$ and hence that *all* subsets of $\llbracket \mathcal{S} \rrbracket$ are computed as elements of ${\downarrow}\mathrm{Sat}^*(\mathcal{S})$. Since by definition, $\llbracket \beta(\tau) \rrbracket \subseteq \llbracket \mathcal{S} \rrbracket$ for any FO-class τ, this terminates the proof of this case. The fact that $\llbracket \mathcal{S} \rrbracket \in \mathrm{Sat}^*(\mathcal{S})$ is a consequence of the following lemma:

Lemma 14. *There exists a group $\mathcal{G} \subseteq \mathcal{S}$ such that $\llbracket \mathcal{G} \rrbracket = \llbracket \mathcal{S} \rrbracket$.*

We first use Lemma 14 to finish the proof of this case. Let $\mathcal{G} = \{T_1, \dots, T_n\}$ be a group as given by the lemma. To obtain $\llbracket \mathcal{S} \rrbracket \in \mathrm{Sat}^*(\mathcal{S})$, it suffices to prove that $\llbracket \mathcal{G} \rrbracket \in \mathrm{Sat}^*(\mathcal{S})$. Since $\mathcal{G}$ is a group, we get $T_i^\omega = 1_{\mathcal{G}}$, so $T_i = T_1^\omega \cdots T_{i-1}^\omega T_i^{\omega+1} T_{i+1}^\omega \cdots T_n^\omega$ for all i. Combining these equalities gives us the inclusion

$$\llbracket \mathcal{G} \rrbracket \subseteq (T_1^\omega \cup T_1^{\omega+1}) \cdots (T_n^\omega \cup T_n^{\omega+1}).$$

By definition (1) of Sat, it follows that $\llbracket \mathcal{G} \rrbracket \in \mathrm{Sat}(\mathcal{S}) \subseteq \mathrm{Sat}^*(\mathcal{S})$, and we are done with the proof in Case 1.

It remains to prove Lemma 14. We first prove that while $\mathcal{S}$ might not be a group itself, it is what we call a *pseudo-group*.

Pseudo-groups. Let $\mathcal{T}$ be a subsemigroup of 2^S. We say that $\mathcal{T}$ is a *pseudo-group* if for all $T \in \mathcal{T}$ and $t \in \llbracket \mathcal{T} \rrbracket$, there exist $R_\ell, R_r \in \mathcal{T}$ such that $T \cdot R_r \ni t$ and $R_\ell \cdot T \ni t$.

Lemma 15. *$\mathcal{S}$ is a pseudo-group.*

Proof. We only do the existence proof for R_ℓ. The proof for R_r is symmetrical. Since β is surjective, there exists $w \in B^+$ such that $T = \beta(w)$. We proceed by induction on the length of w. If w is of length 1, it is immediate by tameness that there exists $R_\ell \in \mathcal{S}$ such that $R_\ell \cdot T \ni t$ and we are finished.

Assume now that the result holds for words of length m and that w is of length $m + 1$. This means that $w = bu$ with u a word of length m. By induction hypothesis, there exists $R'_\ell \in \mathcal{S}$ such that such that $R'_\ell \cdot \beta(u) \ni t$. This means that there exists at least one $r' \in R'_\ell$ such that $\{r'\} \cdot \beta(u) \ni t$. Using tameness again, we get $R_\ell \in \mathcal{S}$ such that $R_\ell \cdot \beta(b) \ni r'$. It follows that $R_\ell \cdot \beta(w) \ni t$, which concludes the proof. □

We now finish the proof of Lemma 14. We prove that any pseudo-group $\mathcal{T} \subseteq \mathcal{S}$ that is not already a group contains a strict subsemigroup $\mathcal{R}$ that remains a pseudo-group, and such that $\llbracket \mathcal{R} \rrbracket = \llbracket \mathcal{T} \rrbracket$. Applying this result iteratively to $\mathcal{S}$ yields the desired group $\mathcal{G}$.

Let $\mathcal{T} \subseteq \mathcal{S}$ be a pseudo-group that is not already a group. An easy and standard argument implies that there must exist $R \in \mathcal{T}$ such that $R \cdot \mathcal{T} \subsetneq \mathcal{T}$ or $\mathcal{T} \cdot R \subsetneq \mathcal{T}$. By symmetry assume that it is the former and set $\mathcal{R} = R \cdot \mathcal{T}$. By definition, $\mathcal{R}$ is closed under product and is therefore a semigroup. It remains to prove that $\mathcal{R}$ is a pseudo-group and that $\llbracket \mathcal{R} \rrbracket = \llbracket \mathcal{T} \rrbracket$.

$\mathcal{R}$ is a pseudo-group. Set $RT \in \mathcal{R}$ and $r \in \llbracket \mathcal{R} \rrbracket$. We want to construct $R_r, R_\ell \in \mathcal{R}$ such that $r \in RT \cdot R_r$ and $r \in R_\ell \cdot RT$. We begin with R_r. Since $\mathcal{T}$ is a pseudo-group, there exists $T_r \in \mathcal{T}$ such that $r \in RTR \cdot T_r$, therefore it suffices to set $R_r = RT_r \in \mathcal{R}$. It remains to construct R_ℓ. Using again the fact that $\mathcal{T}$ is a pseudo-group, we get $T_\ell \in \mathcal{T}$ such that $r \in T_\ell \cdot RT$. In particular, this means that there exists $t_\ell \in T_\ell$ such that $r \in \{t_\ell\} \cdot RT$. Using our pseudo-group hypothesis once again, we obtain $T' \in \mathcal{T}$ such that $t_\ell \in R \cdot T'$. It follows that $r \in RT' \cdot RT$, and it suffices to set $R_\ell = RT' \in \mathcal{R}$.

$\llbracket \mathcal{R} \rrbracket = \llbracket \mathcal{T} \rrbracket$. By definition, we have $\llbracket \mathcal{R} \rrbracket \subseteq \llbracket \mathcal{T} \rrbracket$. We prove the reverse inclusion. Set $t \in \llbracket \mathcal{T} \rrbracket$. Since $\mathcal{T}$ is a pseudo-group, there exists $T \in \mathcal{T}$ such that $t \in RT$. By definition, $RT \in \mathcal{R}$, hence $t \in \llbracket \mathcal{R} \rrbracket$, which ends the proof.

6.3 Case 2: β is not tame.

This is the case where we use induction. By hypothesis on β, there exist $b \in B$ and $t \in \llbracket \mathcal{S} \rrbracket$ such that there exists no $R_r \in \mathcal{S}$ verifying $t \in \beta(b) \cdot R_r$ or no $R_\ell \in \mathcal{S}$ verifying $t \in R_\ell \cdot \beta(b)$. By symmetry, we assume the former, *i.e.*, there exists no $R_r \in \mathcal{S}$ verifying $t \in \beta(b) \cdot R_r$. We set t and b as these objects for the rest of this proof.

Recall that we have $k \geqslant |B| \cdot 2^{|\llbracket \mathcal{S} \rrbracket|^2}$ as in the statement of Proposition 12. Set τ a $\equiv_k$-class. Our goal is to construct $R_\tau \in \mathrm{Sat}^*(\mathcal{S})$ such that $\llbracket \beta(\tau) \rrbracket \subseteq R_\tau$. To use induction, we set

$$B' = B \setminus \{b\}, \qquad k' = |B'| \cdot 2^{|\llbracket \mathcal{S} \rrbracket|^2},$$
$$\widetilde{B} = \{b\} \qquad \widetilde{k} = |\widetilde{B}| \cdot 2^{|\llbracket \mathcal{S} \rrbracket|^2} = 2^{|\llbracket \mathcal{S} \rrbracket|^2}$$

We define Δ as the set of $\equiv_{k'}$ -classes of words over the alphabet B' and Λ as the set of $\equiv_{\widetilde{k}}$ -classes of words over the alphabet $\{b\}$.

The morphism β can be restricted to the alphabet B'. It follows by choice of k' that we can apply the induction hypothesis on the second parameter (the size of the alphabet). This yields the following result.

Fact 16. For all $\delta \in \Delta$, there exists $R_\delta \in \mathrm{Sat}^*(\mathcal{S})$ such that $\llbracket \beta(\delta) \rrbracket \subseteq R_\delta$.

Similarly, β can be restricted to the alphabet $\{b\}$. Moreover, since $\{b\}$ is of size one, by choice of $\widetilde{k}$ we can apply Lemma 13 to every $\lambda \in \Lambda$ and get the following stronger result.

Fact 17. For all $\lambda \in \Lambda$, we have $\llbracket \beta(\lambda) \rrbracket \in \mathrm{Sat}^*(\mathcal{S})$.

We now give an overview of the proof. Set $C = \{\lambda \cdot \delta \mid \lambda \in \Lambda \text{ and } \delta \in \Delta\}$ as a new alphabet. Modulo some prefix in B'^* and suffix in b^* (both possibly empty), any word w in τ can be viewed as a sequence of factors in $b^+ B'^+$. Therefore, by looking at all pairs of classes in Λ, Δ induced by these factors, w can be seen as a word $\overline{w} \in C^+$. For this sketch, assume that the prefix and suffix are both empty. Moreover, β can be adapted over C as a new morphism γ by setting $\gamma : \lambda \cdot \delta \mapsto \llbracket \beta(\lambda) \rrbracket \cdot R_\delta$ and we get $\beta(w) \subseteq \gamma(\overline{w})$. We then prove three results.

1. by choice of k, one can construct a $\equiv_{\overline{k}}$ -class $\overline{\tau}$ over C for a well-chosen $\overline{k}$ such that $\llbracket \beta(\tau) \rrbracket \subseteq \llbracket \gamma(\overline{\tau}) \rrbracket$.
2. by choice of b and t, the index of $\mathcal{T} = \gamma(C^+)$ is strictly smaller than the index of $\mathcal{S}$. Therefore, we can apply induction to γ and get $R \in \mathrm{Sat}^*(\mathcal{T})$ such that $\llbracket \gamma(\overline{\tau}) \rrbracket \subseteq R$.
3. by definition of γ, we get $\mathrm{Sat}^*(\mathcal{T}) \subseteq \mathrm{Sat}^*(\mathcal{S})$ and therefore $R \in \mathrm{Sat}^*(\mathcal{S})$.

By combining the three items, it suffices to take $R_\tau = R \supseteq \llbracket \gamma(\widetilde{\tau}) \rrbracket \supseteq \llbracket \beta(\tau) \rrbracket$ to end the proof. Intuitively, this is what we do. However, there is a slight difference: observe that with the definitions of this sketch, C is a set of pairs of $\equiv_{k'}$ and $\equiv_{\widetilde{k}}$ -classes, and is non-elementary large. Therefore, this definition would yield a much larger bound on k than what claimed. To overcome this problem, we shall use $\gamma(C^+) \subseteq 2^S$ as alphabet instead of C. We now turn to the actual proof.

Set $\mathcal{R}$ as the semigroup $\beta(b) \cdot {\uparrow}\mathcal{S}$. Observe that by definition, for all $\lambda \in \Lambda$, all words in λ have alphabet $\{b\}$. Therefore, $\llbracket \beta(\lambda) \rrbracket \in \beta(b) \cup \mathcal{R}$. It follows that for all $\lambda \in \Lambda$ and $\delta \in \Delta$, $\llbracket \beta(\lambda) \rrbracket \cdot R_\delta \in \mathcal{R}$. We set $\mathcal{T}$ as the subsemigroup of $\mathcal{R}$ generated by

$$\{\llbracket \beta(\lambda) \rrbracket \cdot R_\delta \mid \lambda \in \Lambda \text{ and } \delta \in \Delta\}$$

Note that by definition $\mathcal{T} \subseteq \mathrm{Sat}^*(\mathcal{S})$ and ${\uparrow}\mathcal{T} \subseteq {\uparrow}\mathcal{S}$. We set $C = \mathcal{T}$ and $\gamma : C^+ \to \mathcal{T}$ as the semigroup morphism defined by simply evaluating in $\mathcal{T}$ the product of the letters of a word in C. Finally, set $\overline{k} = |\mathcal{T}| \cdot 2^{|\llbracket \mathcal{T} \rrbracket|^2}$.

Lemma 18. *The index of $\mathcal{T}$ is strictly smaller than the index of $\mathcal{S}$.*

Proof. This is where we use our hypothesis on b and t. We prove that $\mathcal{R}$ has strictly smaller index than $\mathcal{S}$. Since $\mathcal{T}$ is a subsemigroup of $\mathcal{R}$, the desired result will follow. By definition of $\mathcal{R}$, we have $\llbracket \mathcal{R} \rrbracket \subseteq \llbracket \mathcal{S} \rrbracket$. Therefore, it suffices to prove that this inclusion is strict. We prove that $t \notin \llbracket \mathcal{R} \rrbracket$, which concludes the proof. We proceed by contradiction: assume that $t \in \llbracket \mathcal{R} \rrbracket$. This means that there exists $T \in \mathcal{R}$ such $t \in T$. By definition of $\mathcal{R}$, we obtain sets $S_1, \dots, S_n \in {\uparrow}\mathcal{S}$ such that $T = \beta(b)S_1 \cup \cdots \cup \beta(b)S_n$. Since $t \in T$, at least one of these sets, say $\beta(b)S_j$, contains t. Since $S_j \in {\uparrow}\mathcal{S}$, we get $R_1, \dots, R_m \in \mathcal{S}$ such that $t \in \beta(b)R_1 \cup \cdots \cup \beta(b)R_m$. In particular, $t \in \beta(b)R_i$ for some i, which contradicts the choice of t and b. □

Lemma 18 means that we can apply induction on $\equiv_{\overline{k}}$ -classes for the morphism γ. This was exactly point 2 in our sketch. Moreover, since $\mathcal{T} \subseteq \mathrm{Sat}^*(\mathcal{S})$ we get point 3.

Lemma 19. $\mathrm{Sat}^*(\mathcal{T}) \subseteq \mathrm{Sat}^*(\mathcal{S})$.

It remains to define the $\equiv_{\overline{k}}$ -class $\overline{\tau}$ over C. Let $w \in B^+$ be an arbitrary word in τ. There exist $n \geqslant 0$ and $m_1, \dots, m_n \geqslant 1$ such that w can be uniquely decomposed as:

$$w = w' \cdot b^{m_1} \cdot w_1 \cdot b^{m_2} \cdot w_2 \cdots b^{m_n} \cdot w_n \cdot \widetilde{w}$$

where $w_1, \dots, w_n$ are *non-empty* words containing no b, *i.e.*, words of B'^+, w' is a *possibly empty* prefix containing no b, *i.e.*, a word of B'^* and $\widetilde{w}$ a *possible empty* suffix in b^*. This divides w in three parts: the prefix w', the infix $b^{m_1} \cdot w_1 \cdot b^{m_2} \cdot w_2 \cdots b^{m_n} \cdot w_n$ and the suffix $\widetilde{w}$. We assume in this proof that we are in the most complicated case, *i.e.*, none of these parts are empty (the other cases are handled similarly).

We set $\overline{w}$ as the word $c_1 \cdots c_n \in C^+$ defined as follows. For all $i \geqslant 1$, set λ_i as the $\equiv_{\widetilde{k}}$ -class of b^{m_i} and δ_i as the $\equiv_{k'}$ -class of w_i. For all i, let us set $c_i = \llbracket \beta(\lambda_i) \rrbracket \cdot R_{\delta_i} \in C$. By construction, and definition of the sets $\llbracket \beta(\lambda) \rrbracket$, R_δ, we have the following result:

Fact 20. $\beta(b^{m_1} \cdot w_1 \cdot b^{m_2} \cdot w_2 \cdots b^{m_n} \cdot w_n) \subseteq \gamma(\overline{w})$.

Finally, let $\overline{\tau}$ be the $\equiv_{\overline{k}}$ -class of $\overline{w}$. In the sketch, we assumed that the prefix w' and the suffix $\widetilde{w}$ were empty. Here, we have to take them into account. Therefore, we also set $\delta' \in \Delta$ as the $\equiv_{k'}$ -class of w' and $\widetilde{\lambda} \in \Lambda$ as the $\equiv_{\widetilde{k}}$ -class of $\widetilde{w}$.

Using Ehrenfeucht-Fraïssé games, we prove that $\overline{\tau}$, $\widetilde{\lambda}$ and δ' are well-defined, *i.e.*, that the definition depends only on the $\equiv_k$-class τ and not on the choice of w. This is where the choices for the values of $\overline{k}$, $\widetilde{k}$ and k' matter.

Lemma 21. *Let u, v be words in τ, and define $u', \overline{u}, \widetilde{u}, v', \overline{v}, \widetilde{v}$ as above. Then $\overline{u} \equiv_{\overline{k}} \overline{v}$, $u' \equiv_{k'} v'$ and $\widetilde{u} \equiv_{\widetilde{k}} \widetilde{v}$.*

Proof. This is an Ehrenfeucht-Fraïssé argument. By Lemma 18, $|\llbracket \mathcal{T} \rrbracket| < |\llbracket \mathcal{S} \rrbracket|$. Using twice this inequality, we observe that $k' + \overline{k} = (|B|-1) \cdot 2^{|\llbracket \mathcal{S} \rrbracket|^2} + |\mathcal{T}| \cdot 2^{|\llbracket \mathcal{T} \rrbracket|^2} \leqslant (|B|-1) \cdot 2^{|\llbracket \mathcal{S} \rrbracket|^2} + 2^{|\llbracket \mathcal{S} \rrbracket|-1} \cdot 2^{(|\llbracket \mathcal{S} \rrbracket|-1)^2} < |B| \cdot 2^{|\llbracket \mathcal{S} \rrbracket|^2}$, whence $k' + \overline{k} + 1 \leqslant k$. Moreover, by hypothesis $u \equiv_k v$. Hence, by Lemma 1, Duplicator has a winning strategy in the k-round game played on u and v.

It is straightforward to see that if Spoiler places his first pebble on the first b of u, Duplicator has to answer by placing her pebble on the first b of v. Then, every move of Spoiler that is made to the left of these pebbles (*i.e.*, in u', v') must be answered by Duplicator to the left of these pebbles (*i.e.*, in u', v'). It follows that $u' \equiv_{k-1} v'$ and hence that $u' \equiv_{k'} v'$. Similarly, we get that $\widetilde{u} \equiv_{k-1} \widetilde{v}$ and hence that $\widetilde{u} \equiv_{\widetilde{k}} \widetilde{v}$

For $\overline{u}, \overline{v}$, this is slightly more complicated. We describe a winning strategy for Duplicator in the $\overline{k}$-round game played on $\overline{u}$ and $\overline{v}$. To obtain this strategy, Duplicator plays at the same time a shadow game on u and v. In this game all pebbles are placed on positions labeled with a b and such that the next position is labeled by a letter that is not a b. We explain how to play one round.

Assume that Spoiler puts his pebble in $\overline{u}$ (the dual case is answered in the same way) on some position labeled with $c \in C$. By definition, this position corresponds to an infix $b^i u'$ in u with $u' \in B'^+$, and such that $\llbracket \beta(\lambda_{b^i}) \rrbracket \cdot R_{\delta_{u'}} = c$ (with $\lambda_{b^i}, \delta_{u'}$ the $\equiv_{\widetilde{k}} -$ and $\equiv_{k'}$-classes of b^i, u', respectively). Duplicator simulates a move of Spoiler in her shadow game, putting a pebble in u on the last b of this infix $b^i u'$. One can verify that this gives her an answer in v on the last b of an infix $b^j v'$ with $v' \in B'^+$. Moreover, recall that $k' + \overline{k} + 1 \leqslant k$. Hence, since the game on $\overline{u}, \overline{v}$ lasts only $\overline{k}$ rounds and $u \equiv_k v$, at least $k' + 1$ rounds can still be played in the shadow game. It is then straightforward to see that this means that $u' \equiv_{k'} v'$ and $b^i \equiv_{k'} b^j$. In particular, since $k' \geqslant \widetilde{k}$ we get that $b^i \equiv_{\widetilde{k}} b^j$. Therefore, the $\equiv_{k'}$ -class of v' is $\delta_{u'}$, the $\equiv_{\widetilde{k}}$ -class of b^j is λ_{b^i}, hence the position corresponding to $b^j v'$ in $\widetilde{v}$ is labeled with a c. This is Duplicator's answer. □

It remains to prove point 1 in our sketch. In order to take $\delta', \widetilde{\lambda}$ into account, we prove here a slightly generalized version.

Lemma 22. *We have $\llbracket \beta(\tau) \rrbracket \subseteq \llbracket \beta(\delta') \rrbracket \cdot \llbracket \gamma(\overline{\tau}) \rrbracket \cdot \llbracket \beta(\widetilde{\lambda}) \rrbracket$.*

Proof. Let $s \in \llbracket \beta(\tau) \rrbracket$. By definition there exists $u \in \tau$ such that $s \in \beta(u)$. Recalling the construction of $\overline{u}$, there exists $n \in \mathbb{N}$ such that u can be uniquely decomposed as:

$$u = u' \cdot b^{m_1} \cdot u_1 \cdot b^{m_2} \cdot u_2 \cdots b^{m_n} \cdot u_n \cdot \widetilde{u}.$$

Set $S' = \beta(u')$, $T = \beta(b^{m_1} \cdot u_1 \cdot b^{m_2} \cdot u_2 \cdots b^{m_n} \cdot u_n)$ and $\widetilde{S} = \beta(\widetilde{u})$. By definition, $s \in S' \cdot T \cdot \widetilde{S}$. We prove $S' \subseteq \llbracket \beta(\delta') \rrbracket$, $T \subseteq \llbracket \gamma(\overline{\tau}) \rrbracket$ and $\widetilde{S} \subseteq \llbracket \beta(\widetilde{\lambda}) \rrbracket$, which will finish the proof.

By Lemma 21, it is immediate that $u' \in \delta'$ and $\widetilde{u} \in \widetilde{\lambda}$. Therefore, $S' \subseteq \llbracket \beta(\delta') \rrbracket$ and $\widetilde{S} \subseteq \llbracket \beta(\widetilde{\lambda}) \rrbracket$. For T, by Fact 20, $T \subseteq \gamma(\overline{u})$ and by Lemma 21, $\overline{u} \in \overline{\tau}$. Therefore $T \subseteq \llbracket \gamma(\overline{\tau}) \rrbracket$. □

We can now finish the proof by combining the results. Observe that words in δ' are by definition words of B'^+ and words in $\widetilde{\lambda}$ are in b^+. Therefore, by Fact 16, there exists $R_{\delta'} \in \mathrm{Sat}^*(\mathcal{S})$ such that $\llbracket \beta(\delta') \rrbracket \subseteq R_{\delta'}$ and by Fact 17, $\llbracket \beta(\widetilde{\lambda}) \rrbracket \in \mathrm{Sat}^*(\mathcal{S})$. Moreover, by Lemma 18 the index of $\mathcal{T}$ is strictly smaller than the index of $\mathcal{S}$. Therefore, by choice of $\overline{k}$, we can apply the induction hypothesis on $\overline{\tau}$. This yields a set $P \in \mathrm{Sat}^*(\mathcal{T})$ such that $\llbracket \gamma(\overline{\tau}) \rrbracket \subseteq P$. By Lemma 19, $P \in \mathrm{Sat}^*(\mathcal{S})$. Finally, set $R_\tau = R_{\delta'} \cdot P \cdot \llbracket \beta(\widetilde{\lambda}) \rrbracket$. By definition, $\mathrm{Sat}^*(\mathcal{S})$ is a semigroup, therefore, $R_\tau \in \mathrm{Sat}^*(\mathcal{S})$. Furthermore, $\llbracket \beta(\delta') \rrbracket \cdot \llbracket \gamma(\overline{\tau}) \rrbracket \cdot \llbracket \beta(\widetilde{\lambda}) \rrbracket \subseteq R_{\delta'} \cdot P \cdot \llbracket \beta(\widetilde{\lambda}) \rrbracket = R_\tau$. It then follows from Lemma 22 that $\llbracket \beta(\tau) \rrbracket \subseteq R_\tau$.

7. Alternate Algorithms

In the well-known decidable characterization of first-order logic by Schützenberger [12, 18], it is stated that a language is first-order *definable* if and only if its syntactic semigroup is *aperiodic*. In the literature, there are many equivalent definitions of aperiodicity. In this paper, we consider three of them: one is equational, the second considers subgroups and the third considers the $\mathscr{H}$-classes. The relation '$\mathscr{H}$' is one of Green's relations which are well known in semigroup theory. Two elements s, s' of a semigroup S are $\mathscr{H}$-equivalent if $s = s'$ or there exist $t_\ell, t'_\ell, t_r, t'_r \in S$ such that $st_r = s'$, $s't'_r = s$, $t_\ell s = s'$ and $t'_\ell s' = s$. We state the three equivalent definitions.

Lemma 23 (Folklore, see [15])**.** *A finite semigroup S is aperiodic if and only if it satisfies one of the following equivalent statements:*

1. *for all $s \in S$, $s^\omega = s^{\omega+1}$.*
2. *all subgroups in S are trivial.*
3. *all $\mathscr{H}$-classes in S are trivial.*

Our saturation procedure Sat can be viewed as a generalization of the first definition of aperiodicity. Indeed, Operation (1) reflects the equation $s^\omega = s^{\omega+1}$. In this section, we present two alternate and equivalent saturation procedures that reflect the two other definitions. Let $\alpha : A^+ \to S$ be a morphism into a finite semigroup.

Let $\mathcal{S}$ be a subsemigroup of 2^S. We set $\mathrm{Sat}_G(\mathcal{S})$ as the subsemigroup of 2^S generated by

$$\mathcal{S} \cup \{\llbracket \mathcal{G} \rrbracket \mid \mathcal{G} \subseteq \mathcal{S} \text{ and } \mathcal{G} \text{ is a group in } \mathcal{S}\}. \qquad (2)$$

Similarly, $\mathrm{Sat}_H(\mathcal{S})$ is the subsemigroup of 2^S generated by

$$\mathcal{S} \cup \{\llbracket \mathcal{H} \rrbracket \mid \mathcal{H} \subseteq \mathcal{S} \text{ and } \mathcal{H} \text{ is an } \mathscr{H}\text{-class in } \mathcal{S}\}. \qquad (3)$$

The operator Sat_G reflects the second definition of aperiodicity and Sat_H the third. In the following proposition (whose proof is omitted), we state that the three saturation procedures are equivalent and can therefore all be used to compute $\mathcal{I}[\alpha]$ by Proposition 9.

Proposition 24. *Let $\mathcal{S}$ be a subsemigroup of 2^S. Then*

$$\downarrow \mathrm{Sat}^*(\mathcal{S}) = \downarrow \mathrm{Sat}^*_G(\mathcal{S}) = \downarrow \mathrm{Sat}^*_H(\mathcal{S}).$$

Note that the saturation procedure Sat_H can be viewed as a simplification of Henckell's original algorithm [6].

8. Separation for Infinite Words

In this section we generalize FO-indistinguishable sets to ω-words and explain how our fixpoint algorithm can be generalized in order to compute them. In this case as well, we are able to apply the notion to separation and obtain both a bound on the size of a potential separator and an EXPTIME upper bound on the complexity of the problem. It turns out that once the right tools are defined, our proof generalizes smoothly to the case of ω-words. In particular several arguments in this proof are replaced by using the finite word case as a subresult.

The section is organized as follows. We first generalize our terminology to the setting of ω-words. In the second part, we generalize FO-indistinguishable sets, and we state the link with separation. Finally, we explain how to generalize our fixpoint algorithm to compute these new FO-indistinguishable sets.

8.1 Preliminary Definitions

ω-words and ω-languages. Recall that A is a finite alphabet. We denote by A^∞ the set of ω-words over A. Note that we still use the term "word" to mean an element of A^+. If u is a word and v an ω-word, we denote by $u \cdot v$ the ω-word obtained by concatenating u to the left of v, and by u^∞ the ω-word obtained by infinite concatenation of u with itself[2]. An ω-language is a subset of A^∞. Regular ω-languages are those that are accepted by *nondeterministic Büchi automata* (NBA). Again, we will only work with the algebraic representation of ω-languages that we recall below.

ω-semigroups. We briefly recall the definition of ω-semigroups, which play the role of semigroups in the setting of ω-words. For more details, we refer the reader to [14].

An *ω-semigroup* is a pair $\mathbf{S} = (S_+, S_\infty)$ where S_+ is a semigroup and S_∞ is a set. Moreover, $\mathbf{S}$ is equipped with two additional products: a *mixed product* $S_+ \times S_\infty \to S_\infty$ that maps $s, t \in S_+, S_\infty$ to an element denoted st, and an *infinite product* $(S_+)^\infty \to S_\infty$ that maps an infinite sequence $s_1, s_2, \dots \in (S_+)^\infty$ to an element of S_∞ denoted by $s_1 s_2 \cdots$. We require these products as well as the semigroup product of S_+ to satisfy all possible forms of associativity, cf. [14] for details. Finally, we denote by s^∞ the element $sss\cdots$. Observe that $\mathbf{A} = (A^+, A^\infty)$ is an ω-semigroup.

The notions of subsemigroups and morphisms can be adapted to ω-semigroups. In particular, if T_+ is a subsemigroup of S_+ and T_∞ is the set obtained by applying the infinite product to all sequences of T_+, then $\mathbf{T} = (T_+, T_\infty)$ is a sub-ω-semigroup of $\mathbf{S}$ called the *sub-ω-semigroup generated by* T_+.

An ω-semigroup is said to be *finite* if both S_+ and S_∞ are finite. Note that even if an ω-semigroup is finite, it is not obvious that a finite representation of the infinite product exists. However, it was proven by Wilke [22] that the infinite product is fully determined by the mapping $s \mapsto s^\infty$, yielding a finite representation for finite ω-semigroups. An ω-language L is said to be *recognized* by an ω-semigroup $\mathbf{S} = (S_+, S_\infty)$ if there exists $F \subseteq S_\infty$ as well as a morphism $\alpha : \mathbf{A} \to \mathbf{S}$ such that $L = \alpha^{-1}(F)$. It is well known that an ω-language is regular if and only if it is recognized by a *finite* ω-semigroup. Moreover [22], from any NBA recognizing L, one can compute a canonical smallest ω-semigroup recognizing L, called the *syntactic ω-semigroup.*

As for finite words, when working on separation, it is convenient to consider a single recognizing object for both input languages rather than two separate objects. Again, this is not restrictive, given two ω-languages and two associated recognizing ω-semigroups, one can define (and compute) a single ω-semigroup that recognizes both languages by taking the cartesian product of the two original ω-semigroups.

Semigroup of Subsets. For an ω-semigroup $\mathbf{S}$, note that $2^{\mathbf{S}} = (2^{S_+}, 2^{S_\infty})$ is an ω-semigroup with the products defined in the natural way. Moreover, $\mathbf{S}$ can be viewed as a sub-ω-semigroup of $2^{\mathbf{S}}$. Indeed, $\mathbf{S}$ is isomorphic to the ω-semigroup $(\{\{s\} \mid s \in S_+\}, \{\{s\} \mid s \in S_\infty\})$, which is a sub-$\omega$-semigroup of $2^{\mathbf{S}}$.

First-order logic for ω-words. First-order logic is defined in the same way on ω-words as on words. Therefore, for the sake of simplifying the notations, we keep the same terminology. One remark is of importance, however: in the proof, we will manipulate at the same time $\equiv_k$-classes of words and $\equiv_k$-classes of ω-words. To avoid confusion, we call the latter $\equiv_k$-ω-classes, and devote the terminology '$\equiv_k$-classes' to words.

8.2 FO-indistinguishable sets for ω-languages.

FO-indistinguishable sets. Let $\alpha : \mathbf{A} \to \mathbf{S}$ be an ω-semigroup morphism and set $(S_+, S_\infty) = \mathbf{S}$. Observe that α can be restricted as a classical semigroup morphism $\alpha_+ : A^+ \to S_+$. Therefore, FO-indistinguishable subsets of S_+ are already defined, and it suffices to generalize the notion to S_∞. We give the full definition (recalling the definition for S_+) below. We define the two following pairs of sets:

- $\mathbf{I}_k[\alpha] = (\mathcal{I}^+_k[\alpha], \mathcal{I}^\infty_k[\alpha])$ with $\mathcal{I}^+_k[\alpha] \subseteq 2^{S_+}$ and $\mathcal{I}^\infty_k[\alpha] \subseteq 2^{S_\infty}$, the pair of sets of FO$[k]$-indistinguishable sets for α.
- $\mathbf{I}[\alpha] = (\mathcal{I}^+[\alpha], \mathcal{I}^\infty[\alpha])$ with $\mathcal{I}^+[\alpha] \subseteq 2^{S_+}$ and $\mathcal{I}^\infty[\alpha] \subseteq 2^{S_\infty}$, the pair of sets of FO-indistinguishable sets for α.

Let $T = \{s_1, \dots, s_m\} \subseteq S_+$ (resp. $\in S_\infty$). We have

- $T \in \mathcal{I}^+_k[\alpha]$ (resp. $\in \mathcal{I}^\infty_k[\alpha]$) if there exist $w_1, \dots, w_n \in A^+$ (resp. $\in A^\infty$) with
  - $w_1 \equiv_k w_2 \equiv_k \cdots \equiv_k w_n$, and
  - $\alpha(w_1) = s_1, \dots, \alpha(w_n) = s_n$.
- $T \in \mathcal{I}^+[\alpha]$ (resp. $\in \mathcal{I}^\infty[\alpha]$) if for all $k \in \mathbb{N}$, we have $T \in \mathcal{I}_k[\alpha]$ (resp. $\in \mathcal{I}^\infty_k[\alpha]$).

As for finite words the two following facts are by definition.

Fact 25. (a) $\mathcal{I}^+_k[\alpha] \supseteq \mathcal{I}^+_{k+1}[\alpha] \supseteq \mathcal{I}^+[\alpha]$ and $\mathcal{I}^\infty_k[\alpha] \supseteq \mathcal{I}^\infty_{k+1}[\alpha] \supseteq \mathcal{I}^\infty[\alpha]$ for all $k \geqslant 0$.
(b) $\mathcal{I}^+[\alpha] = \bigcap_k \mathcal{I}^+_k[\alpha]$ and $\mathcal{I}^\infty[\alpha] = \bigcap_k \mathcal{I}^\infty_k[\alpha]$.

Fact 26. $\mathbf{I}_k[\alpha]$ and $\mathbf{I}[\alpha]$ are sub-ω-semigroups of $2^{\mathbf{S}}$.

We finish the definition by generalizing Proposition 7, *i.e.*, our bound on the stabilization index, to the setting of ω-words.

Proposition 27. *For all $k > |A|2^{|\mathbf{S}|^2}$, we have $\mathbf{I}[\alpha] = \mathbf{I}_k[\alpha]$.*

As for finite words, Proposition 27 yields a brute-force algorithm for computing $\mathbf{I}[\alpha]$. Again, this algorithm is non-elementary in k. We generalize below our fixpoint algorithm for the setting of ω-languages, and get an EXPTIME procedure. As before, the bound is proven as a corollary of the completeness proof of the algorithm.

From FO-separation to FO-indistinguishable sets. By definition, the generalization of Theorem 8 to ω-words is immediate. This yields the following theorem.

[2] In the literature, the ω-word u^∞ is usually denoted by u^ω. Here, we use this non standard notation in order to avoid confusion with the idempotent power ω in semigroups.

Theorem 28. *Let L_0, L_1 be two regular ω-languages recognized by a morphism $\alpha : \mathbf{A} \to \mathbf{S}$ into a finite ω-semigroup. Then, L_0 and L_1 are FO-separable if and only if for all $T \in \mathcal{I}^{\infty}[\alpha]$, either $\alpha(L_0) \cap T = \varnothing$ or $\alpha(L_1) \cap T = \varnothing$.*

Moreover, if L_0, L_1 are FO-separable, then the actual separator can be chosen with quantifier rank $|A|2^{|\mathbf{S}|^2} + 1$.

It follows from Theorem 28 and Proposition 27 that one can decide whether two ω-languages are separable by a first-order formula. Moreover, we also get an upper bound on the quantifier rank of the potential separator.

8.3 An algorithm to compute $\mathbf{I}[\alpha]$

Let $\alpha : \mathbf{A} \to \mathbf{S}$ be a morphism into a finite ω-semigroup $\mathbf{S} = (S_+, S_\infty)$. We give an algorithm for computing $\mathbf{I}[\alpha]$.

Observe that α can be restricted as a semigroup morphism $\alpha_+ : A^+ \to S_+$ and that by definition the sets $\mathcal{I}^+_k[\alpha], \mathcal{I}^+[\alpha]$ are exactly the sets $\mathcal{I}_k[\alpha_+], \mathcal{I}[\alpha_+]$. Therefore, one can compute the set $\mathcal{I}^+[\alpha]$ in EXPTIME by reusing our fixpoint algorithm for the setting of finite words (see Section 4). Thus, it suffices to explain how to compute $\mathcal{I}^{\infty}[\alpha]$. The following proposition (whose proof is omitted), shows that one can compute it directly from $\mathcal{I}^+[\alpha]$.

Proposition 29. *Let $\ell = |A|2^{|\mathbf{S}|^2} + 1$ and $\mathbf{R} = (\mathcal{R}_+, \mathcal{R}_\infty)$ the sub-ω-semigroup of $2^{\mathbf{S}}$ generated by $\mathcal{I}^+[\alpha]$. Then*

$$\mathcal{I}^{\infty}[\alpha] = \mathcal{I}^{\infty}_{\ell}[\alpha] = {\downarrow} \mathcal{R}_\infty.$$

Since we already know how to compute $\mathcal{I}^+[\alpha]$ in EXPTIME, it follows from Proposition 29 that one can compute $\mathbf{I}[\alpha]$ in EXPTIME as well. This generalizes our upper bound on the complexity of the separation problem to ω-languages.

Corollary 30. *Let L_0, L_1 be two regular ω-languages recognized by a morphism $\alpha : \mathbf{A} \to \mathbf{S}$ into a finite ω-semigroup. Then one can decide in* EXPTIME *with respect to $|\mathbf{S}|$ whether L_0, L_1 are FO-separable.*

Proof. Let L_0, L_1 be two regular ω-languages recognized by a morphism $\alpha : \mathbf{A} \to \mathbf{S}$ into a finite ω-semigroup. We prove that one can decide in EXPTIME with respect to $|\mathbf{S}|$ whether L_0, L_1 are FO-separable. By Theorem 28, it suffices to prove that $\mathbf{I}[\alpha]$ can be computed in EXPTIME in the size of $\mathbf{S}$. We already know that $\mathcal{I}^+[\alpha]$ can be computed in EXPTIME (see above). Therefore, it suffices to prove that the sub-ω-semigroup of $2^{\mathbf{S}}$ generated by $\mathcal{I}^+(\alpha)$ can be computed in EXPTIME. This is also easily done since one can prove that this sub-ω-semigroup is $(\mathcal{I}^+[\alpha], \mathcal{R}_\infty)$ with $\mathcal{R}_\infty = \{ST^\infty \mid S, T \in \mathcal{I}^+[\alpha]\}$ (see [14]). □

9. Conclusion

We gave combinatorial and self-contained proofs that one can decide in EXPTIME whether two regular languages of finite or infinite words are separable by first-order logic. Further, we obtained an upper bound on the quantifier rank of an expected separator. There are several open questions left in this line of research. First, we do not know if the bounds are tight. We conjecture that the problem is EXPTIME-complete starting from semigroups. A related question is the complexity, starting from NFAs. Our results imply a 2-EXPTIME upper bound (for DFAs, checking first-order definability is PSPACE-complete [3]).

A more interesting problem is the efficient computation of an actual separator. The computation itself is possible by enumerating all the FO[ℓ]-languages for our upper bound $\ell = |A|2^{|S|^2}$ on the expected quantifier rank of a separator. However, as the number of such formulas is non-elementary in ℓ, this is not a satisfying solution. It turns out that our completeness proof (Proposition 12) can be rephrased as an algorithm that computes separators. Let $\alpha : A^+ \to S$ be a morphism into a finite semigroup and $T \in \mathcal{I}[\alpha]$. One can actually use the induction in Proposition 12 to construct an FO[ℓ]-formula φ_T, such that

- φ_T accepts any word whose $\equiv_\ell$-class has image T: for any $\equiv_\ell$-class τ s.t. $[\![\alpha(\tau)]\!] = T$, we have $\tau \subseteq \{w \mid w \models \varphi_T\}$.
- Any word accepted by φ_T has an image that is indistinguishable from T: if $w \models \varphi_T$, then $\alpha(w) \cup T \in \mathcal{I}[\alpha]$.

By definition of $\mathcal{I}[\alpha]$, any two languages L_0, L_1 which are at the same time both recognized by α and FO-separable can be separated by the union of formulas φ_T such that $T \in \mathcal{I}[\alpha]$ and $T \cap \alpha(L_0) = \varnothing$. A rough analysis of the procedure yields a 2-EXPTIME complexity in the size of $|S|$. We leave the detailed presentation of this procedure for further work.

References

[1] J. Almeida. Some algorithmic problems for pseudovarieties. *Publ. Math. Debrecen*, 54:531–552, 1999. Proc. of Automata and Formal Languages, VIII.

[2] D. Beauquier and J. E. Pin. Languages and scanners. *Theoret. Comput. Sci.*, 84(1):3–21, 1991.

[3] S. Cho and D. T. Huynh. Finite-automaton aperiodicity is PSPACE-complete. *Theoret. Comp. Sci.*, 88(1):99–116, 1991.

[4] W. Czerwiński, W. Martens, and T. s. Masopust. Efficient separability of regular languages by subsequences and suffixes. In *ICALP'13*, 2013.

[5] V. Diekert and P. Gastin. First-order definable languages. In *Logic and Automata: History and Perspectives*, volume 2, pages 261–306. Amsterdam Univ. Press, 2008.

[6] K. Henckell. Pointlike sets: the finest aperiodic cover of a finite semigroup. *J. Pure Appl. Algebra*, 55(1-2):85–126, 1988.

[7] K. Henckell, J. Rhodes, and B. Steinberg. Aperiodic pointlikes and beyond. *Internat. J. Algebra Comput.*, 20(2):287–305, 2010.

[8] N. Immerman. *Descriptive Complexity*. Springer, 1999.

[9] H. W. Kamp. *Tense Logic and the Theory of Linear Order*. Phd thesis, CS Department, University of California at Los Angeles, USA, 1968.

[10] R. E. Ladner. Application of model theoretic games to discrete linear orders and finite automata. *Inform. Control*, 33(4):281–303, 1977.

[11] L. Libkin. *Elements Of Finite Model Theory*. Springer, 2004.

[12] R. McNaughton and S. Papert. *Counter-Free Automata*. MIT Press, 1971.

[13] D. Perrin. Recent results on automata and infinite words. In *MFCS'84*, 1984.

[14] D. Perrin and J. E. Pin. *Infinite Words*. Elsevier, 2004.

[15] J. E. Pin. Mathematical foundations of automata theory. `http://www.liafa.jussieu.fr/~jep/PDF/MPRI/MPRI.pdf`, 2014.

[16] T. Place, L. van Rooijen, and M. Zeitoun. Separating regular languages by piecewise testable and unambiguous languages. In *MFCS' 13*, 2013.

[17] T. Place, L. van Rooijen, and M. Zeitoun. Separating regular languages by locally testable and locally threshold testable languages. In *FSTTCS'13*, LIPIcs, 2013.

[18] M. P. Schützenberger. On finite monoids having only trivial subgroups. *Inform. Control*, 8:190–194, 1965.

[19] H. Straubing. *Finite Automata, Formal Logic and Circuit Complexity*. Birkhauser, 1994.

[20] W. Thomas. Star-free regular sets of omega-sequences. *Inform. and Control*, 42(2):148–156, 1979.

[21] W. Thomas. Languages, automata, and logic. In *Handbook of formal languages*. Springer, 1997.

[22] T. Wilke. An Eilenberg theorem for infinity-languages. In *ICALP'91*, 1991.

[23] T. Wilke. Classifying discrete temporal properties. In *STACS' 99*, 1999.

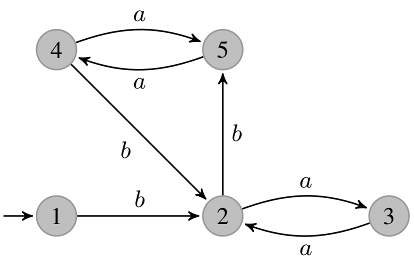
A. Running the Example in Section 4

In this section, we give details on the computation of FO-indistinguishable sets for the example presented in Section 4. Recall that we consider the languages $K_0 = (aa)^*$, $K_1 = (aa)^*a$ and

$$L_0 = (bK_0bK_1)^+$$
$$L_1 = (bK_0bK_1)^*bK_0$$

A simple Ehrenfeucht-Fraïssé argument shows that L_0 and L_1 are not FO-separable. We consider a morphism $\alpha : A^+ \to S$ recognizing both languages, and we show that there exists an FO-indistinguishable set for α, say $T \in \mathcal{I}[\alpha]$, that intersects both $\alpha(L_0)$ and $\alpha(L_1)$.

Both languages are recognized by the following automaton,

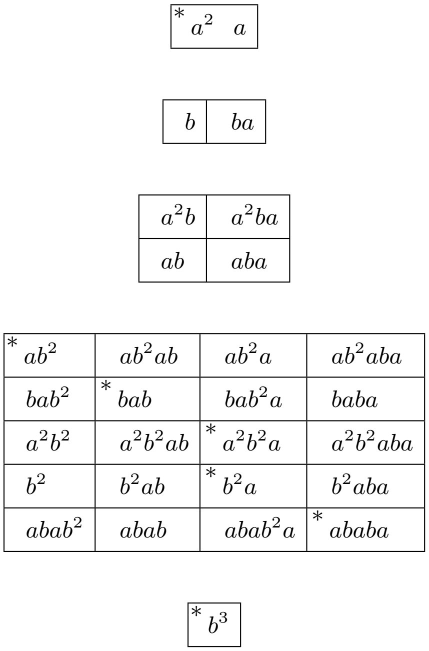


with 4 as final state for L_0, and 2 as final state for L_1. The transition semigroup of this automaton, which recognizes both languages, is generated by the following partial functions on states:

	1	2	3	4	5
a	–	3	2	5	4
b	2	5	–	2	–

A presentation of this semigroup is given by the relations $a^3 = a$, $ba^2 = b$, $b^3 = babab = 0$, $a^2bab = bab$, $b^2ab^2 = b^2$, $(bab)^2 = bab$.

The structure in $\mathcal{J}$-classes is shown in Figure 2.



* a^2	a

b	ba

a^2b	a^2ba
ab	aba

* ab^2	ab^2ab	ab^2a	ab^2aba
bab^2	* bab	bab^2a	$baba$
a^2b^2	a^2b^2ab	* a^2b^2a	a^2b^2aba
b^2	b^2ab	* b^2a	b^2aba
$abab^2$	$abab$	$abab^2a$	* $ababa$

* b^3

Figure 2. $\mathcal{J}$-classes of the transition semigroup

The recognizing morphism $\alpha : A^+ \to S$ thus maps a word to the partial function from states to states that it defines. To simplify notation, we still write a and b for the images of the letters a and b, respectively.

It is easy to see that $ba^2 = b$. Since the image of a word of L_0 is always equal to $ba^2ba = b^2a$, and since α recognizes L_0, we obtain $L_0 = \alpha^{-1}(b^2a)$. Similarly $L_1 = \alpha^{-1}(b^2ab)$.

To show that L_0 and L_1 are not FO-separable, we use Proposition 8: we have to find an FO-indistinguishable set $T \in \mathcal{I}[\alpha]$ containing both $b^2a = \alpha(L_0)$ and $b^2ab = \alpha(L_1)$.

Let us explain how such a subset T of S is produced by the algorithm. The algorithm starts with $\mathrm{Sat}^0(\mathcal{S})$ consisting of singletons. Then, notice that $\{a\}^\omega = \{a^2\}$ and $\{a\}^{\omega+1} = \{a\}$. Therefore, during the first saturation phase of the algorithm (1), we add the set $\{a, a^2\}$ to the list of FO-indistinguishable sets.

Since $\mathrm{Sat}(\mathcal{S})$ is a subsemigroup, the algorithm also computes $X = \{a, aa\} \cdot \{b\} = \{ab, aab\}$ as an element of $\mathrm{Sat}(\mathcal{S})$. One can[3] then compute $X^2 = \{(ab)^2, ab^2, bab, a^2b^2\}$, $X^3 = \{b^3, (ab)^2b, ab^2ab, bab^2, a^2b^2ab\}$, and $X^4 = \{b^3\} \cup X^2$, so that $X^8 = X^4$. Set $Y = X^\omega \cup X^{\omega+1} = X^4 \cup X^5 = X^2 \cup X^3$. By (1), $Y \in \mathrm{Sat}^2(\mathcal{S})$. Observe that $\{bab, bab^2\} \subset Y$. Now, $\{b^2a, b^2ab\} \subset \{b\} \cdot \{bab, bab^2\} \cdot \{a, a^2\} \subset \{b\} \cdot Y \cdot \{a, a^2\}$ which belongs to $\mathrm{Sat}^2(\mathcal{S})$ since it is a subsemigroup of 2^S.

B. Proof of Proposition 24

In this appendix, we prove Proposition 24 in Section 7. We recall the proposition below.

Proposition 24. *Let $\mathcal{S}$ be a subsemigroup of 2^S, then $\downarrow \mathrm{Sat}^*(\mathcal{S}) = \downarrow \mathrm{Sat}^*_G(\mathcal{S}) = \downarrow \mathrm{Sat}^*_H(\mathcal{S})$.*

We prove that $\downarrow \mathrm{Sat}^*(\mathcal{S}) \subseteq \downarrow \mathrm{Sat}^*_H(\mathcal{S}) \subseteq \downarrow \mathrm{Sat}^*_G(\mathcal{S}) \subseteq \downarrow \mathrm{Sat}^*(\mathcal{S})$. Let us first prove that $\downarrow \mathrm{Sat}^*(\mathcal{S}) \subseteq \downarrow \mathrm{Sat}^*_H(\mathcal{S})$.

$\boldsymbol{\downarrow \mathrm{Sat}^*(\mathcal{S}) \subseteq \downarrow \mathrm{Sat}^*_H(\mathcal{S})}$. We prove that for any $T \in \mathrm{Sat}^*(\mathcal{S})$ there exists $T' \in \mathrm{Sat}^*_H(\mathcal{S})$ such that $T \subseteq T'$. It is then immediate by definition of $\downarrow$ that $\downarrow \mathrm{Sat}^*(\mathcal{S}) \subseteq \downarrow \mathrm{Sat}^*_H(\mathcal{S})$.

Set $T \in \mathrm{Sat}^*(\mathcal{S})$. Then $T \in \mathrm{Sat}^i(\mathcal{S})$ for some $i \in \mathbb{N}$. We proceed by induction on i. If $i = 0$, then $T \in \mathrm{Sat}^0(\mathcal{S}) = \mathcal{S} = \mathrm{Sat}^0_H(\mathcal{S})$. Assume now that $i > 0$. By definition, T belongs to the semigroup generated by

$$\mathrm{Sat}^{i-1}(\mathcal{S}) \cup \{R^\omega \cup R^{\omega+1} \mid R \in \mathrm{Sat}^{i-1}(\mathcal{S})\}.$$

The only nontrivial case is when $T = R^\omega \cup R^{\omega+1}$ for some $R \in \mathrm{Sat}^{i-1}(\mathcal{S})$. By induction hypothesis, there exists $R' \in \mathrm{Sat}^*_H(\mathcal{S})$ such that $R \subseteq R'$. Observe that $(R')^{\omega+1}$ and $(R')^\omega$ are $\mathscr{H}$-equivalent elements in the semigroup $\mathrm{Sat}^*_H(\mathcal{S})$, and are therefore both contained in some $\mathscr{H}$-class $\mathcal{H} \subseteq \mathrm{Sat}^*_H(\mathcal{S})$. By definition of $\mathrm{Sat}^*_H(\mathcal{S})$, we then have $\llbracket \mathcal{H} \rrbracket \in \mathrm{Sat}^*_H(\mathcal{S})$. Hence, $T = R^\omega \cup R^{\omega+1} \subseteq (R')^\omega \cup (R')^{\omega+1} \subseteq \llbracket \mathcal{H} \rrbracket \in \mathrm{Sat}^*_H(\mathcal{S})$, which ends the proof for this inclusion.

$\boldsymbol{\downarrow \mathrm{Sat}^*_H(\mathcal{S}) \subseteq \downarrow \mathrm{Sat}^*_G(\mathcal{S})}$. We prove that $\mathrm{Sat}^*_H(\mathcal{S}) \subseteq \mathrm{Sat}^*_G(\mathcal{S})$. Let $T \in \mathrm{Sat}^*_H(\mathcal{S})$. Then $T \in \mathrm{Sat}^i_H(\mathcal{S})$ for some $i \in \mathbb{N}$. To prove that $T \in \mathrm{Sat}^*_G(\mathcal{S})$, we proceed by induction on i. If $i = 0$, then the result is clear since $\mathrm{Sat}^0_H(\mathcal{S}) = \mathcal{S} = \mathrm{Sat}^0_G(\mathcal{S})$. Assume now that $i > 0$. By definition, T belongs to the semigroup generated by

$$\mathrm{Sat}^{i-1}_H(\mathcal{S}) \cup \{\llbracket \mathcal{H} \rrbracket \mid \mathcal{H} \text{ is an } \mathscr{H}\text{-class in } \mathrm{Sat}^{i-1}_H(\mathcal{S})\}.$$

The only nontrivial case is when $T = \llbracket \mathcal{H} \rrbracket$ for some $\mathscr{H}$-class $\mathcal{H}$ of $\mathrm{Sat}^{i-1}_H(\mathcal{S})$. We claim that either $\mathcal{H}$ is a singleton, or there exists a group $\mathcal{G}$ in $\mathrm{Sat}^{i-1}_H(\mathcal{S})$ and $R \in \mathrm{Sat}^{i-1}_H(\mathcal{S})$ such that $\mathcal{H} = R \cdot \mathcal{G}$. Let us first show how to use this claim to deduce

[3] These computations were checked using the `semigroup` program available at `http://www.liafa.jussieu.fr/~jep/semigroupes.html`

that $T = \llbracket \mathcal{H} \rrbracket$ belongs to $\mathrm{Sat}^*_G(\mathcal{S})$. If $\mathcal{H}$ is a singleton $\{H\}$ with $H \in \mathrm{Sat}^{i-1}_H(\mathcal{S})$, by induction $H \in \mathrm{Sat}^{i-1}_G(\mathcal{S})$, and so $T = \llbracket \mathcal{H} \rrbracket = H \in \mathrm{Sat}^*_G(\mathcal{S})$. Otherwise, again by induction, R belongs to $\mathrm{Sat}^*_G(\mathcal{S})$ and $\mathcal{G}$ is a group in $\mathrm{Sat}^*_G(\mathcal{S})$. By definition of $\mathrm{Sat}^*_G(\mathcal{S})$, we have $\llbracket \mathcal{G} \rrbracket \in \mathrm{Sat}^*_G(\mathcal{S})$. Hence, since $\mathrm{Sat}^*_G(\mathcal{S})$ is a semigroup, we have $T = \llbracket \mathcal{H} \rrbracket = R \cdot \llbracket \mathcal{G} \rrbracket \in \mathrm{Sat}^*_G(\mathcal{S})$.

It remains to prove the claim (which actually is not specific to subsemigroups of a semigroup of subset): every $\mathscr{H}$-class $\mathcal{H}$ of a semigroup $\mathcal{T}$ is either a singleton, or of the form $R \cdot \mathcal{G}$, for $R \in \mathcal{T}$ and $\mathcal{G}$ a group in $\mathcal{T}$. Let Stab $= \{T \in \mathcal{T} \mid \mathcal{H} \cdot T = \mathcal{H}\}$. If $\mathcal{H}$ is not a singleton, then Green's Lemma implies that Stab is nonempty, and therefore it is a subsemigroup of $\mathcal{T}$. Let $\mathcal{G}$ be an $\mathscr{H}$-class of its minimal ideal. By standard results in semigroup theory, $\mathcal{G}$ is a group. Let us check that $\mathcal{H} = H \cdot \mathcal{G}$, for any $H \in \mathcal{H}$. Indeed, let $H \in \mathcal{H}$ and let E be the identity of $\mathcal{G}$. Since $E \in$ Stab, we have $H = H'E$ for some $H' \in \mathcal{H}$, and so $HE = H$. Let now $H_1 \in \mathcal{H}$. By definition, we have $H_1 = H \cdot X$ for some $X \in \mathcal{T}$. Note that since $\mathcal{G}$ is in the minimal ideal, we have $EXE \in \mathcal{G}$. Hence $H_1 = H_1E = HXE = H(EXE) \in H\mathcal{G}$.

$\boldsymbol{\downarrow \mathrm{Sat}^*_G(\mathcal{S}) \subseteq \downarrow \mathrm{Sat}^*(\mathcal{S})}$. Assume now that $T \in \mathrm{Sat}^*_G(\mathcal{S})$, we construct $T' \in \mathrm{Sat}^*(\mathcal{S})$ such that $T \subseteq T'$. By definition, $T \in \mathrm{Sat}^i_G(\mathcal{S})$ for some i. We proceed by induction on i. When $i = 0$ the result is immediate as for the previous inclusions. Assume now that $i > 0$. Then T is in the subsemigroup generated by

$$\mathrm{Sat}^{i-1}_G(\mathcal{S}) \cup \{\llbracket \mathcal{G} \rrbracket \mid \mathcal{G} \subseteq \mathrm{Sat}^{i-1}_G(\mathcal{S}) \text{ and } \mathcal{G} \text{ is a group}\}$$

Again, the only non-trivial case is when $T = \llbracket \mathcal{G} \rrbracket$ for $\mathcal{G}$ a group in $\mathrm{Sat}^{i-1}_G(\mathcal{S})$. Set $\mathcal{G} = \{T_1, \dots, T_n\}$ with $T_i \in \mathrm{Sat}^{i-1}_G(\mathcal{S})$ and let $1_\mathcal{G}$ be the identity element of $\mathcal{G}$. Since $\mathcal{G}$ is a group, for all i, $T_i^\omega = 1_\mathcal{G}$. In particular this means that for all i, $T_i = T_1^\omega \cdots T_{i-1}^\omega T_i^{\omega+1} T_{i+1}^\omega \cdots T_n^\omega$. By combining these equalities, we get

$$\llbracket \mathcal{G} \rrbracket = T_1 \cup \cdots \cup T_n \subseteq (T_1^\omega \cup T_1^{\omega+1}) \cdots (T_n^\omega \cup T_n^{\omega+1})$$

By induction hypothesis, we know that for all i, there exists $T_i' \in \mathrm{Sat}^*(S)$ such that $T_i \subseteq T_i'$. By definition of Sat this means that for all i, $R_i = T_i'^\omega \cup T_i'^{\omega+1} \in \mathrm{Sat}^*(S)$. Moreover, since $\mathrm{Sat}^*(S)$ is a semigroup, it also contains $T' = R_1 \cdots R_n$. By definition, we have $\llbracket \mathcal{G} \rrbracket \subseteq T' \in \mathrm{Sat}^*(S)$.

C. Proof of Proposition 29

In this appendix, we prove Proposition 29. Recall that we work with a morphism $\alpha : \mathbf{A} \to \mathbf{S}$ into a finite ω-semigroup $\mathbf{S} = (S_+, S_\infty)$.

Proposition 29. *Let $\ell = |A|2^{|\mathbf{S}|^2} + 1$ and $\mathbf{R} = (\mathcal{R}_+, \mathcal{R}_\infty)$ the sub-ω-semigroup of $2^\mathbf{S}$ generated by $\mathcal{I}^+[\alpha]$. Then $\mathcal{I}^\infty[\alpha] = \mathcal{I}^\infty_\ell[\alpha] = \downarrow \mathcal{R}_\infty$.*

As for the finite setting, we prove that: $\mathcal{I}^\infty[\alpha] \subseteq \mathcal{I}^\infty_\ell[\alpha] \subseteq \downarrow \mathcal{R}_\infty \subseteq \mathcal{I}^\infty[\alpha]$. By Fact 25, we already know that $\mathcal{I}^\infty[\alpha] \subseteq \mathcal{I}^\infty_\ell[\alpha]$. Moreover, we know from Fact 6 that $\mathbf{I}[\alpha]$ is a sub-ω-semigroup. Therefore, by definition, $\mathcal{R}_\infty \subseteq \mathcal{I}^\infty[\alpha]$ whence $\downarrow \mathcal{R}_\infty \subseteq \mathcal{I}^\infty[\alpha]$. It remains to prove the more difficult $\mathcal{I}^\infty_\ell[\alpha] \subseteq \downarrow \mathcal{R}_\infty$ inclusion. We devote the remainder of this appendix to this proof.

The proof is done by generalizing Proposition 12 and its proof to the ω-language setting. We state this generalization now:

Proposition 31. *Let $(\mathcal{S}_+, \mathcal{S}_\infty)$ be a sub-ω-semigroup of $2^\mathbf{S}$ and let $\beta : \mathbf{B} \to (\mathcal{S}_+, \mathcal{S}_\infty)$ be a surjective morphism. Set $(\mathcal{U}_+, \mathcal{U}_\infty)$ as the sub-ω-semigroup of $2^\mathbf{S}$ generated by $Sat^*(\mathcal{S}_+)$.*

Set $k \geqslant |B|2^{|\llbracket \mathcal{S}_+ \rrbracket|^2} + 1$, then for any $\equiv_k$-ω-class τ, $\llbracket \beta(\tau) \rrbracket \in \downarrow \mathcal{U}_\infty$.

Before proving Proposition 31, we first explain how to use it to prove that $\mathcal{I}^\infty_\ell[\alpha] \subseteq \downarrow \mathcal{R}_\infty$. Set $T \in \mathcal{I}^\infty_\ell[\alpha]$, by definition there exists an $\equiv_\ell$-ω-class τ such that $T \subseteq \llbracket \alpha(\tau) \rrbracket$.

Recall that $(\alpha(A^+), \alpha(A^\infty))$ can be viewed as sub-ω-semigroup of $2^\mathbf{S}$. Set $\beta : \mathbf{A} \to (\alpha(A^+), \alpha(A^\infty))$ such that $\beta(w) = \{\alpha(w)\}$. Let $(\mathcal{U}_+, \mathcal{U}_\infty)$ be the sub-ω-semigroup of $2^\mathbf{S}$ generated by $Sat^*(\alpha(A^+))$. By choice of ℓ we can apply Proposition 31 to β, τ and we get $\llbracket \beta(\tau) \rrbracket \in \downarrow \mathcal{U}_\infty$. Moreover, observe that by Proposition 9, we already know that $Sat^*(\alpha(A^+)) \subseteq \mathcal{I}^+[\alpha]$. In particular, this means that $\downarrow \mathcal{U}_\infty \subseteq \downarrow \mathcal{R}_\infty$. Hence we have $\llbracket \beta(\tau) \rrbracket \in \downarrow \mathcal{R}_\infty$ and therefore $T \in \downarrow \mathcal{R}_\infty$ which terminates the proof.

It remains to prove Proposition 31, this is done by generalizing the argument for Proposition 12.

Proof. As for Proposition 12, we work by induction on the index of $\mathcal{S}_+$ and the size of B. However, in this case, several inductive arguments are now replaced by an application of Proposition 12 as a subresult. Again, we divide the proof in three parts. We first investigate the case $|B| = 1$ and then distinguish two cases depending on the tameness of β.

Observe that when $|B| = 1$, B^∞ is a singleton: the word b^∞. By surjectivity, this means that $\mathcal{S}_\infty$ is also a singleton and the result is immediate. Assume now that $|B| \geqslant 2$ and recall that we say that β is *tame* if for all $b \in B$ and all $t \in \llbracket \mathcal{S}_+ \rrbracket$, there exists $R_\ell, R_r \in \mathcal{S}_+$ such that $t \in \beta(b) \cdot R_r$ and $t \in R_\ell \cdot \beta(b)$. Notice that the tameness property remains unchanged from the finite case and is therefore only a property of the mapping $\beta : B^+ \to \mathcal{S}_+$. As for Proposition 12 we distinguish two cases depending on whether β is tame or not.

C.1 Case 1: β is tame

In the same case, we saw in the proof of Proposition 12 that $\llbracket \mathcal{S}_+ \rrbracket \in \mathrm{Sat}^*(\mathcal{S}_+)$. By surjectivity of β it is immediate that $(\llbracket \mathcal{S}_+ \rrbracket)^\infty = \llbracket \mathcal{S}_\infty \rrbracket$. Therefore, $\llbracket \mathcal{S}_\infty \rrbracket \in \mathcal{U}_\infty$. We conclude that for any ω-class τ of ω-words, $\llbracket \beta(\tau) \rrbracket \subseteq \llbracket \mathcal{S}_\infty \rrbracket \in \mathcal{U}_\infty$ which terminates this case.

C.2 Case 2: β is not tame

By hypothesis on β, there exists $b \in B$, and $t \in \llbracket \mathcal{S}_+ \rrbracket$, such that there exists no $R_r \in \mathcal{S}_+$ verifying $t \in \beta(b) \cdot R_r$ or no $R_\ell \in \mathcal{S}_+$ verifying $t \in R_\ell \cdot \beta(b)$. By symmetry, we assume that we are in the first case, *i.e.*, there exists no $R_r \in \mathcal{S}_+$ verifying $t \in \beta(b) \cdot R_r$. We fix t and b as these objects for the rest of this proof.

Set τ a $\equiv_k$-ω-class. We need to construct $R_\tau \in \mathcal{U}_\infty$ such that $\llbracket \beta(\tau) \rrbracket \subseteq R_\tau$. Set w some arbitrary word in τ. We distinguish three subcases depending on the letter 'b' occuring in w.

Subcase a: w contains finitely many b. We assume that w contains at least one letter b (otherwise we can immediately conclude by induction). We treat this case by splitting w into a finite prefix that ends with the last occurrence of the finitely many b and an infinite suffix that contains no b. The prefix can then be treated using Proposition 12 and the suffix by induction on the size of B.

We set $w = w_1 \cdot b \cdot w_2$ where w_2 is a ω-word that contains no b. Set δ_1 as the $\equiv_{k-1}$-class of w_1b and δ_2 as the $\equiv_{k-1}$-ω-class of w_2. Observe that by choice of k we can apply Proposition 12 to the $\equiv_{k-1}$-class δ_1. This yields $T_1 \in \mathrm{Sat}^*(\mathcal{S}_+)$ such that $\llbracket \beta(\delta_1) \rrbracket \subseteq T_1$. Moreover, since words in δ_2 contain no b, we can apply the induction hypothesis to the $\equiv_{k-1}$-ω-class δ_2. We obtain $T_2 \in \mathcal{U}_\infty$ such that $\llbracket \beta(\delta_2) \rrbracket \subseteq T_2$. Set $T = T_1 \cdot T_2 \in \mathcal{U}_\infty$. A simple Ehrenfeucht-Fraïssé argument shows that any word u in τ can be decomposed as $u = u_1bv_2$ with $v_1b \in \delta_1$ and $v_2 \in \delta_2$. Therefore, $\llbracket \beta(\tau) \rrbracket \subseteq \llbracket \beta(\delta_1) \rrbracket \cdot \llbracket \beta(\delta_2) \rrbracket \subseteq T$ and we are done with this case.

Subcase b: $w = w_0 b^{\infty}$. We assume that w_0 is the shortest prefix such that $w = w_0 b^{\infty}$, i.e. w_0 does not end with a b (if w_0 can be chosen empty the result is again immediate by induction).

Set δ_1 as the $\equiv_{k-1}$-class of w_0 and δ_2 as the $\equiv_{k-1}$-ω-class of b^{∞}. Observe that by choice of k we can apply Proposition 12 to the $\equiv_{k-1}$-class δ_1. This yields $T_1 \in \text{Sat}^*(\mathcal{S}_+)$ such that $\lfloor\!\lfloor \beta(\delta_1) \rfloor\!\rfloor \subseteq T_1$. Moreover, since words in δ_2 have an alphabet of size 1, we can apply the induction hypothesis to the $\equiv_{k-1}$-ω-class δ_2. We obtain $T_2 \in \mathcal{U}_{\infty}$ such that $\lfloor\!\lfloor \beta(\delta_2) \rfloor\!\rfloor \subseteq T_2$. Set $T = T_1 \cdot T_2 \in \mathcal{U}_{\infty}$. A simple Ehrenfeucht-Fraïssé argument shows that any word u in τ can be decomposed as $u = u_0 b^{\infty}$ with $u_0 \in \delta_1$ and $b^{\infty} \in \delta_2$. Therefore, $\lfloor\!\lfloor \beta(\tau) \rfloor\!\rfloor \subseteq \lfloor\!\lfloor \beta(\delta_1) \rfloor\!\rfloor \cdot \lfloor\!\lfloor \beta(\delta_2) \rfloor\!\rfloor \subseteq T$ and we are done with this case.

Subcase c: w contains infinitely many b and does not end with a suffix b^{∞}. This case is a generalization of the argument we used in the non-tame case of the proof of Proposition 12. The main difference is that we replace induction on the size of B by an application of Proposition 12. In particular, this means that we only use induction on the index of $\mathcal{S}_+$.

Let us first observe that the assumption on w is common to all words in τ. Indeed, having infinitely many b and ending by a suffix b^{∞} can both be tested by FO($<$) formulas of quantifier rank 2 and $k \geqslant 2$ by definition.

Set $B' = B \setminus \{b\}$, $\widetilde{k} = |B'| \cdot 2^{|\lfloor\!\lfloor \mathcal{S} \rfloor\!\rfloor|^2} + 1$ and $\bar{k} = |\{b\}| \cdot 2^{|\lfloor\!\lfloor \mathcal{S} \rfloor\!\rfloor|^2} = 2^{|\lfloor\!\lfloor \mathcal{S} \rfloor\!\rfloor|^2}$. We define Δ as the set of $\equiv_{\widetilde{k}}$ -classes of words over the alphabet B' and Λ as the set of $\equiv_{\bar{k}}$ -classes of words over the alphabet $\{b\}$.

The morphism β can be restricted on the alphabet B'. It follows by choice of $\widetilde{k}$ that we can apply Proposition 12 to every $\delta \in \Delta$. This yields the following result.

Fact 32. For all $\delta \in \Delta$, there exists $R_{\delta} \in \text{Sat}^*(\mathcal{S})$ such that $\lfloor\!\lfloor \beta(\delta) \rfloor\!\rfloor \subseteq R_{\delta}$.

Similarly, β can be restricted on the alphabet $\{b\}$. Moreover, since $\{b\}$ is of size one, by choice of $\bar{k}$ we can apply Lemma 13 to every $\lambda \in \Lambda$ and get the following stronger result.

Fact 33. For all $\lambda \in \Lambda$, $\lfloor\!\lfloor \beta(\lambda) \rfloor\!\rfloor \in \text{Sat}^*(\mathcal{S})$.

We follow the same outline as for Proposition 12. Set $\mathcal{R}_+$ as the semigroup $\beta(b) \cdot {\uparrow}\mathcal{S}_+$. By definition for all $\delta \in \Delta$ and $\lambda \in \Lambda$, $\lfloor\!\lfloor \beta(\lambda) \rfloor\!\rfloor \cdot R_{\delta} \in \mathcal{R}_+$. We set $\mathcal{T}_+$ as the subsemigroup of $\mathcal{R}_+$ generated by

$$\{\lfloor\!\lfloor \beta(\lambda) \rfloor\!\rfloor \cdot R_{\delta} \mid \lambda \in \Lambda \text{ and } \delta \in \Delta\}$$

Set $(\mathcal{T}_+, \mathcal{T}_{\infty})$ as the sub-ω-semigroup generated by $\mathcal{T}_+$, $C = \mathcal{T}_+$, $\mathbf{C} = (C^+, C^{\infty})$ and $\gamma : \mathbf{C} \to (\mathcal{T}_+, \mathcal{T}_{\infty})$ as the evaluation ω-semigroup morphism defined in the obvious way. Moreover, set $(\mathcal{V}_+, \mathcal{V}_{\infty})$ as the sub-ω-semigroup of $2^{\mathbf{S}}$ generated by $Sat^*(\mathcal{T}_+)$. Set $\widehat{k} = |\mathcal{T}_+| \cdot 2^{|\lfloor\!\lfloor \mathcal{T}_+ \rfloor\!\rfloor|^2} + 1$. Observe that Lemma 18 in the proof of Proposition 12 can be reused in order to get the following result (note that this is where the hypothesis on b and t is used).

Fact 34. $\mathcal{T}_+$ has strictly smaller index than $\mathcal{S}_+$.

By choice of $\widehat{k}$, this means that we can apply induction on $\equiv_{\widehat{k}}$ -ω-classes for the morphism γ. Moreover, by Lemma 19, $\mathcal{V}_+ = \text{Sat}^*(\mathcal{T}_+) \subseteq \text{Sat}^*(\mathcal{S}_+) = \mathcal{U}_+$. By definition, this generalizes to $\mathcal{U}_{\infty}$ and $\mathcal{V}_{\infty}$.

Lemma 35. $\mathcal{V}_{\infty} \subseteq \mathcal{U}_{\infty}$.

We now define a $\equiv_{\widehat{k}}$-ω-class $\widehat{\tau}$ over C. By hypothesis, w, there exists and infinite sequence of naturals $n \geqslant 0$ and $m_1, \dots, m_n, \dots \geqslant 1$ such that w can be uniquely decomposed as:

$$w = w_0 \cdot b^{m_1} \cdot w_1 \cdot b^{m_2} \cdot w_2 \cdot b^{m_3} \cdot w_3 \cdots$$

Such that w_0 is a (possibly empty) word in B'^* and $w_1, w_2, \dots$ are (non-empty) words in B'^+. We set $\hat{w}$ as the ω-word $c_1 c_2 c_3 \cdots \in C^{\infty}$ defined as follows. For all $i \geqslant 1$, set δ_i as the $\equiv_{\widetilde{k}}$ -class of the word w_i and λ_i as the $\equiv_{\bar{k}}$ -class of the words b^{m_i}. For all i, we set $c_i = \lfloor\!\lfloor \beta(\lambda_i) \rfloor\!\rfloor \cdot R_{\delta_i} \in C$. By construction, we have the following result

Fact 36. $\beta(b^{m_1} \cdot w_1 \cdot b^{m_2} \cdot w_2 \cdot b^{m_3} \cdot w_3 \cdots) \subseteq \gamma(\widehat{w})$.

We finish by setting $\widehat{\tau}$ as the $\equiv_{\widehat{k}}$ -ω-class of the ω-word $\widehat{w}$. Observe that $\widehat{\tau}$ does not take into account the prefix w_0. Therefore, we also fix $\delta \in \Delta$ as the $\equiv_{\widetilde{k}}$ -class of the word w_0.

As in the finite setting, we can prove that $\widehat{\tau}$ and δ do not depend on the choice of w.

Lemma 37. *Let u, v be ω-words in τ, then $\widehat{u} \equiv_{\widehat{k}} \widehat{v}$ and $u_0 \equiv_{\widetilde{k}} v_0$.*

Proof. This is proved using an Ehrenfeucht-Fraïssé argument that is identical to the one used in Lemma 21. □

It now remains to generalize Lemma 22.

Lemma 38. $\lfloor\!\lfloor \beta(\tau) \rfloor\!\rfloor \subseteq \lfloor\!\lfloor \beta(\delta) \rfloor\!\rfloor \cdot \lfloor\!\lfloor \gamma(\widehat{\tau}) \rfloor\!\rfloor$.

Proof. Let $s \in \lfloor\!\lfloor \beta(\tau) \rfloor\!\rfloor$. By definition there exists $u \in \tau$ such that $s \in \beta(u)$. Recall that by construction of $\widehat{u}$, the word u can be uniquely decomposed as

$$u = u_0 \cdot b^{k_1} \cdot u_1 \cdot b^{k_2} \cdot u_2 b^{k_3} \cdot u_3 \cdots$$

Set $S_0 = \beta(u_0)$ and $T = \beta(b^{k_1} \cdot u_1 \cdot b^{k_2} \cdot u_2 \cdots)$. As before, we prove that $S_0 \subseteq \lfloor\!\lfloor \beta(\delta) \rfloor\!\rfloor$ and $T \subseteq \lfloor\!\lfloor \gamma(\widehat{\tau}) \rfloor\!\rfloor$, which ends the proof since $s \in S_0 \cdot T$.

For $S_0 \in \lfloor\!\lfloor \beta(\delta) \rfloor\!\rfloor$, this is immediate from Lemma 37. For T, it follows from Fact 36 that $T \subseteq \gamma(\widehat{u})$. Moreover, by Lemma 37, $\widehat{u} \in \widehat{\tau}$, therefore, $T \subseteq \lfloor\!\lfloor \gamma(\widehat{\tau}) \rfloor\!\rfloor$. □

We can now finish the proof. By Fact 32, there exists $R_{\delta} \in \text{Sat}^*(\mathcal{S}_+)$ such that $\lfloor\!\lfloor \beta(\delta) \rfloor\!\rfloor \subseteq R_{\delta}$. By Fact 34 the index of $\mathcal{T}_+$ is strictly smaller than the index of $\mathcal{S}_+$, therefore, by choice of $\widehat{k}$, we can apply the induction hypothesis on $\widehat{\tau}$. This yields a set $P \in \mathcal{V}_{\infty}$ such that $\lfloor\!\lfloor \gamma(\widehat{\tau}) \rfloor\!\rfloor \subseteq P$.

By Lemma 35, we have $P \in \mathcal{U}_{\infty}$. Set $R_{\tau} = R_{\delta} \cdot P$. Observe that by definition of $R_{\tau} \in \mathcal{V}_{\infty}$. Furthermore, $\lfloor\!\lfloor \beta(\delta) \rfloor\!\rfloor \cdot \lfloor\!\lfloor \gamma(\widehat{\tau}) \rfloor\!\rfloor \subseteq R_{\delta} \cdot P = R_{\tau}$. It then follows from Lemma 38 that $\lfloor\!\lfloor \beta(\tau) \rfloor\!\rfloor \subseteq R_{\tau}$. □