Decision Problems on Orders of Words

Lucian Ilie

To be presented, with the permission of the Faculty of Mathematics and Natural Sciences of the University of Turku, for public criticism in the Auditorium of the Computer Science Department on November 28th, 1998, at 12 noon

University of Turku
Dapartment of Mathematics
FIN-20014 Turku, Finland
1998

Supervised by
Professor Arto Salomaa

Academy of Finland
and
Turku Centre for Computer Science TUCS
FIN-20520 Turku, Finland

Reviewed by
Professor Grzegorz Rozenberg

Department of Computer Science
Leiden University, P.O.Box 9512
2300 RA Leiden, The Netherlands
and
Department of Computer Science
University of Colorado at Boulder
Boulder, CO 80309, U.S.A.

Professor Gheorghe Păun

Institute of Mathematics of the Romanian Academy
P.O.Box 1–764, R-70700 Bucharest, Romania

ISBN 952-12-0280-7
ISSN 1239-1883
Painosalama Oy

Abstract

In this thesis we study several interconnected decision problems and effective constructions on words and languages, as well as some related problems, all being connected with different orders of words. Most important ones are mentioned in what follows. Decision problems: the slenderness problem for context-free languages, the problem of whether or not a language (regular or context-free) is the set of prefixes (factors, subwords) of an infinite word, the confluence problem with respect to a quasi order. Effective constructions: effective representations of slender context-free languages, effective regularity of the sets of subwords or superwords of a given language. Related problems: the characterization of slender context-free languages, a generalization of Haines' theorem, generalized factor relations and a generalization of Higman's theorem.

Keywords: language theory, combinatorics on words, decidability, effective construction, quasi order, slender language, prefix, factor, confluence, downset, well order.

Acknowledgements

First of all, I would like to thank Professor Arto Salomaa for granting me the honour of doing my Ph.D. under his supervision. His friendly guidance has been invaluable throughout my work and, besides this, it was an excellent opportunity for me to learn many things from him. Arto's personality has influenced my life far beyond mathematics.

It is my pleasure to thank Professor Grzegorz Rozenberg for reviewing my thesis and making suggestions that improved it significantly. I would like also to thank Professor Gheorghe Păun for his detailed comments on my thesis and for his great help during my first steps in the field of language theory.

I have learned a lot from Professor Tero Harju and Professor Juhani Karhumäki and I would like to thank both for their continuous encouragement and advice.

Special thanks are due to Professor Alexandru Mateescu for our interesting discussions and to Elisa Mikkola for her help during my study. Also, I would like to thank my colleagues at the Faculty of Mathematics, University of Bucharest, and especially Professor Virgil Căzănescu, for their support.

Excellent working conditions and financial support provided by the Turku Centre for Computer Science, one of the top research centers in Finland, are greatfully acknowledged.

Finally, I thank my wife, Silvana, for her great support for my work.

Turku
September 1998 Lucian Ilie

Contents

Chapter 1

Introduction

The concept of decidability plays a major role in mathematics, in general, and in language theory, in particular. This is due to the fact that the existence of undecidable problems, i.e., problems for which no effective procedure for solving them exists, shows some of the limitations of mathematics.

In order to show that a problem can be solved, it is sufficient to give an algorithm for it. On the other hand, to prove that a certain problem cannot be solved, i.e., is undecidable, one has to prove that no algorithm solves it. Hence, a formalization of the notion of algorithm is needed. However, it took quite a lot of effort to make things clear.

Everything started in 1931 when Kurt Gödel published his famous incompleteness theorem, destroying Hilbert's dream of finding an effective procedure for determining the truth or falsity of any mathematical proposition. Subsequent work, especially by Church, Kleene, Post, and Turing, clarified and formalized our intuitive notion of an effective procedure, this being considered as one of the great intellectual achievements of this century.

Once the notion of an effective procedure was formalized, it was shown that there was no effective procedure for solving many specific problems. The essential reason for this goes back to the Paradox of the Liar and the diagonalization argument of Cantor. Surprising was the fact that some simple, very natural, and clearly formulated problems turned out to be undecidable. In particular, many examples of such problems can be found in language theory.

However, the questions of proving that a particular problem is decidable and of proving that a particular problem is undecidable depend essentially on the considered problem and either case may be very difficult. To give just one famous example of each type, we mention the theorem of Matijasevitch, cf. [Mat], proving that Hilbert's tenth problem is undecidable and Makanin's algorithm, cf. [Mak], for the solvability of word equations.

A most interesting issue is to look for the borderline between decidability and undecidability, as it approaches the frontier of our possibilities. Since it is clear that any subproblem of a decidable problem is still decidable and that the undecidability of a subproblem implies the undecidability of the whole problem, this borderline amounts to extending a decidable problem until it becomes undecidable or, conversely, restricting a problem to become decidable.

In this thesis, we consider several interconnected problems on words and languages and are concerned with both proving that some of them are decidable and some are undecidable, as well as with investigating the borderline between the two. We also investigate problems which are not decision problems, that is, they require more than yes-no answers, and connections between effective constructions and decision problems. Here and there, also related issues are considered.

A brief description of the problems we consider is given in the sequel, where the structuring of the thesis is outlined.

Chapter 2 surveys briefly the main concepts and notations needed in the other chapters. Very basic results on words, languages, and decidability are given.

Chapter 3 considers the *slender* languages, that is, languages for which the number of words of a same length is bounded from above by a constant. The main result of the chapter is the characterization of the slender context-free languages as *unions of paired loops*, that is, finite unions of sets of the form $\{uv^nwx^ny \mid n \geq 0\}$. This confirms a conjecture of Păun and Salomaa and generalizes the corresponding result for the regular case which was solved by the same authors and, independently, by Shallit. As consequences of this result, some problems concerning linearity, ambiguity, and closure properties of the slender context-free languages are solved.

A deeper analysis of the structure of the slender context-free languages is performed in Chapter 4. The most important result gives some upper bounds on the lengths of words in the paired loops of such a language. By means of these bounds, two new and simple proofs for the decidability of the slenderness problem for context-free languages are given. Moreover, we prove that a representation of a slender context-free language as a (disjoint) union of paired loops can be effectively given and that the maximal number of words of the same length is computable.

Chapter 5 investigates languages obtained as prefixes or factors of infinite words and settles the following two problems: it is decidable whether an arbitrary context-free language can be written as the set of prefixes of an infinite word as well as whether an arbitrary regular language is the set of factors of an infinite word. A special case, of factors which appear infinitely often, is considered at the end. The existence of infinite words having this set context-free non-regular is proved.

We introduce in Chapter 6 a new notion, of *confluence*, arising from our considerations in the previous chapter. A language is said to be confluent with respect to a given quasi order if, for any two words in the language, it contains one which is bigger, with respect to the given quasi order, than either of the former two. This property is investigated mainly for regular and context-free languages with respect to prefixes and factors; the basic decidability results are established. Some generalizations are also considered. The results in the previous chapters are very useful for the proofs here.

In Chapter 7, we deal with well quasi orders, in general, and with subwords and Parikh subwords, in particular. The problems in the previous two chapters are investigated here for these two relations, that is, the problem of whether a language, regular or context-free, is the set of (Parikh) subwords of an infinite word and the confluence problem. Then, some connections between generalized confluence and well quasi orders are established. Next, some results concerning the effective regularity of the sets of subwords and superwords are presented. A related argument gives a generalization of Haines' theorem. Finally, we consider some relations which generalize the factor relation and lie in between factors and subwords. A condition equivalent with the finite basis property of these relations is given. It generalizes a particular form of Higman's theorem.

At the end of most chapters, some open problems or directions for further research are discussed.

The material of the thesis comes mostly from my papers [Il1], [Il2], [Il3], [Il4], [Il5], [Il6] and my joint papers [HaIl1], [HaIl2], and [IlSa]. However, some new results are introduced and some of the proofs are improved.

Chapter 2

Basics on words

This preliminary chapter surveys briefly the main concepts used in the thesis. It also presents some very basic results about words, languages, and decidability that are of use in the sequel. However, specific definitions are given at appropriate places in the next chapters. This chapter is intended to be consulted whenever needed.

For more details of combinatorics on words we refer to [ChKa], [Lo] and for formal language theory to [Har], [HoUl], [Sa1], [Sa2].

2.1 Words

This section contains some basic facts about words such as catenation, free monoid, lengths of words, several quasi orders, primitive words, conjugacy, infinite words, etc.

The free monoid

An **alphabet** is a finite non-empty set. The elements of an alphabet Σ are called **letters**. A **word** is a finite sequence of zero or more letters of Σ; the word with zero letters is called the **empty** word and is denoted by λ. For instance, $\lambda, a, bb, ababb$ are words over the alphabet $\Sigma = \{a, b\}$.

The set of all words over Σ is denoted by Σ^*, whereas the set of all non-empty words over Σ is denoted by $\Sigma^+ = \Sigma^* - \{\lambda\}$. The **concatenation** (or **catenation**) of two words u, v, denoted uv, is obtained by juxtaposition, that is, writing v at the end of u. The catenation is an associative operation with λ as the identity: $w\lambda = \lambda w = w$, for any word w. The sets Σ^* and Σ^+ are the **free monoid** and **free semigroup**, respectively, generated by Σ with respect to the operation of catenation. (λ is the unit of Σ^*.)

The nth **power** of a word w, for $n \geq 0$, is denoted by w^n and is defined as usual in the monoid Σ^*: $w^0 = \lambda, w^n = w^{n-1}w$.

The **length** of a word w, denoted $|w|$, is the number of letters appearing in w; each letter is counted as many times as it occurs; by definition, $|\lambda| = 0$. (Thus, $|\cdot| : \Sigma^* \longrightarrow \mathbb{N}$ is a monoid morphism.) For a letter $a \in \Sigma$ and a word $w \in \Sigma^*$, the number of occurrences of a in w is denoted by $|w|_a$. We have then $|w| = \sum_{a\in\Sigma} |w|_a$. The set of all letters which appear in w is $\mathrm{alph}(w) = \{a \in \Sigma \mid |w|_a \geq 1\}$.

The **reversal** of a word w, denoted w^R, is the word having the letters of w in the reverse order, that is, if $w = a_1a_2\cdots a_{|w|}$, then $w^R = a_{|w|}a_{|w|-1}\cdots a_1$.

Quasi orders of words

We define next some of the most important quasi orders on Σ^*. For two words $u, v \in \Sigma^*$, we say that u is a **factor** of v if there are two words w, z such that $v = wuz$. If $w = \lambda$, then u is a **prefix** of v and if $z = \lambda$, then u is a **suffix** of v. We say that u is a **subword**[1] of v if all letters of u appear in the same order in v; if the order is not important, then u is called a **Parikh subword** of v. The notations[2] for these relations are given below.

prefix: $u \leq_p v$ iff there is $w \in \Sigma^*$ such that $v = uw$,
factor: $u \leq_f v$ iff there are $w, z \in \Sigma^*$ such that $v = wuz$,
subword: $u \leq_s v$ iff $u = a_1a_2\ldots a_n, n \geq 0, a_i \in \Sigma, 1 \leq i \leq n,$
$v = v_1a_1v_2a_2\ldots v_na_nv_{n+1}, v_i \in \Sigma^*, 1 \leq i \leq n+1.$
Parikh subword: $u \leq_\Psi v$ iff $|u|_a \leq |v|_a$, for any $a \in \Sigma$.

Notice that $\leq_p, \leq_f$, and $\leq_s$ are partial orders but $\leq_\Psi$ is not. (It is not antisymmetric; for instance, $ab \leq_\Psi ba, ba \leq_\Psi ab$, but $ab \neq ba$.)

For a word $w \in \Sigma^*$, we denote by $Pref(w), Suf(w)$, and $Fact(w)$ the sets of all prefixes, suffixes, and factors of w, respectively.

For two words u and v such that $u \leq_p v$, the **left quotient** of v by u, denoted $u^{-1}v$, is the unique word w such that $v = uw$. The **right quotient** is defined similarly.

Primitive words and conjugacy

A word w is **primitive** if it is not a non-trivial power of another word, that is, $w = z^n$ implies $n = 1, z = w$. (So λ is not primitive since $\lambda^2 = \lambda$.) Two words u and v are **conjugates**, denoted $u \sim v$, if $u = pq, v = qp$, for some words p and q. The next theorem summarizes several very important properties concerning primitive words and conjugacy.

[1]What we call here "factor" and "subword" are sometimes called "subword" and "scattered subword", respectively. The subword partial order is also called "division" partial order in [Lo].

[2]The subword partial order is denoted in [ChKa] by $\leq_d$.

Theorem 2.1.1. *(i) Any non-empty word w is a power of a unique primitive word called the* **primitive root** *of w and denoted $\rho(w)$.*

(ii) Two words u, v are conjugates iff their primitive roots are conjugates. Thus, if u is primitive and $u \sim v$, then v is also primitive.

(iii) u, v **commute**, *i.e., $uv = vu$, iff u and v are powers of a same word.*

(iv) u, v are conjugates iff there is w such that $uw = wv$; this equality holds iff there are p, q such that $u = pq, v = qp$, and $w = (pq)^n p, n \geq 0$.

(v) If w is primitive, then w has exactly two occurrences in ww, as a prefix and as a suffix, i.e., $ww = uwv$ implies either $u = \lambda$ or $v = \lambda$.

We next recall the periodicity theorem of Fine and Wilf, cf. [FiWi]. We give also an original proof by induction of this important result, the shortest according to our knowledge. Other proofs can be found in [Lo], [ChKa] (a very illustrative proof), [CrRy] (another proof by induction).

For its statement, we need one more definition. A word w has a **period** $p \geq 0$ if $w_i = w_{i+p}$, for all $1 \leq i \leq |w| - p$, where w_i stands for the ith letter of w.

Theorem 2.1.2 (Fine and Wilf [FiWi]). *If a word w has periods p and q and $|w| = p + q - \gcd(p, q)$, then w has also period $d = \gcd(p, q)$.*

Proof. By induction on $n = |w|$. The first steps are trivial. Suppose the statement true for all words shorter than w. Assume $p \geq q$ and put $w = uv$, $|u| = p - d$. For any $1 \leq i \leq q - d$, $u_i = w_i = w_{i+p} = w_{i+p-q} = u_{i+p-q}$, so u has period $p - q$. Since u has also period q and $\gcd(p - q, q) = d$, the inductive hypothesis shows that u has period d. Thus w has period d, too. ■

We shall use this result in the following form which follows immediately from the theorem above.

Corollary 2.1.3. *If u^n and v^m, for some $u, v \in \Sigma^*, n, m > 0$, have a common factor of length at least $|u| + |v|$, then $\rho(u) \sim \rho(v)$.*

Infinite words

A **right-infinite**[3] word over Σ is a word which is unbounded from the right. It can be viewed as a function $\alpha : \mathbb{Z}_+ \longrightarrow \Sigma$ from the set of positive integers into the alphabet Σ. Analogously, a **left-infinite** word is defined as a function $\alpha : \mathbb{Z}_- \longrightarrow \Sigma$. The set of all right-(left-)infinite words is denoted by Σ^ω (${}^\omega\Sigma$). For a finite word $w \in \Sigma^*$, we denote $w^\omega = www\ldots \in \Sigma^\omega$ and ${}^\omega w = \ldots www \in {}^\omega\Sigma$.

[3] An infinite word is sometimes called "ω-word"; we use the prefixes "right-", "left-", and "bi-" because this precise distinction will be very important later on.

A **bi-infinite** (two-sided) word is an infinite word without any end. We can define a bi-infinite word as a function $\alpha : \mathbb{Z} \longrightarrow \Sigma$ or, in fact, as an equivalence class of the set $\Sigma^{\mathbb{Z}}$ with respect to the equivalence relation ρ defined for $\alpha, \beta \in \Sigma^{\mathbb{Z}}$ by $\alpha\rho\beta$ if and only if there is an integer k such that for any $n \in \mathbb{Z}, \alpha(n) = \beta(n+k)$. We denote by ${}^{\omega}\Sigma^{\omega}$ the set of all bi-infinite words over Σ.

For an infinite word $\alpha \in \Sigma^{\omega} \cup {}^{\omega}\Sigma \cup {}^{\omega}\Sigma^{\omega}$, we denote by $Pref(\alpha), Suf(\alpha)$, and $Fact(\alpha)$ the set of all finite prefixes, suffixes, and factors of α, respectively. Notice that, for $\alpha \in {}^{\omega}\Sigma$, $Pref(\alpha) = \emptyset$ and, for $\alpha \in \Sigma^{\omega}$, $Suf(\alpha) = \emptyset$.

2.2 Languages

In this section, we define languages and some of their operations, as well as (very briefly) regular and context-free languages together with some of their very basic properties.

Languages and operations

A **language** over an alphabet Σ is any subset of Σ^*. For a language L, we denote its cardinality by $\mathrm{card}(L)$ and the set of all letters appearing in the words of L by $\mathrm{alph}(L) = \bigcup_{w \in L} \mathrm{alph}(w)$. The empty language is denoted $\emptyset$.

The operations of **union**, **intersection**, **difference**, and **complement** are defined in the usual way for sets; we notice the notations only: $L_1 \cup L_2, L_1 \cap L_2, L_1 - L_2, \overline{L} = \Sigma^* - L$. Some operations specific for languages are given below:

– **taking prefixes** or **factors**: $Pref(L) = \bigcup\limits_{w \in L} Pref(w), Fact(L) = \bigcup\limits_{w \in L} Fact(w)$;

– **reversal**: $L^R = \{w^R \mid w \in L\}$;

– **catenation**: $L_1 L_2 = \{w_1 w_2 \mid w_1 \in L_1, w_2 \in L_2\}$;

– **power**: $L^0 = \{\lambda\}, L^n = L^{n-1}L$, for any $n \geq 1$;

– **catenation closure** or **Kleene star**: $L^* = \bigcup\limits_{i=0}^{\infty} L^i$;

– **λ-free catenation closure** or **Kleene plus**: $L^+ = \bigcup\limits_{i=1}^{\infty} L^i$;

When L has only one element, say $L = \{w\}$, we write w^* and w^+ instead of $\{w\}^*$ and $\{w\}^+$, respectively. Notice also that $\emptyset^* = \{\lambda\}, \emptyset^+ = \emptyset$.

A **morphism** is a mapping $h : \Sigma^* \longrightarrow \Delta^*$, where Σ and Δ are alphabets, such that $h(uv) = h(u)h(v)$, for any $u, v \in \Sigma^*$. An **inverse morphism** is $h : \Delta^* \longrightarrow 2^{\Sigma^*}$ given by $h^{-1}(w) = \{x \in \Sigma^* \mid h(x) = w\}$, for any $w \in \Delta^*$.

– **morphism**: $h(L) = \{h(w) \mid w \in L\}$, for $L \subseteq \Sigma^*$;

– **inverse morphism**: $h^{-1}(L) = \bigcup_{w \in L} h^{-1}(w)$, for $L \subseteq \Delta^*$;

– **quotient**; left quotient: $L_1^{-1}L_2 = \{w \mid v = uw, \text{ for some } u \in L_1, v \in L_2\}$ and right quotient, $L_2L_1^{-1}$, similarly defined.

A family $\mathcal{F}$ of languages is said to be **closed** under an n-ary operation $\mathcal{O}$ if $\mathcal{O}(L_1, L_2, \ldots, L_n) \in \mathcal{F}$, for any $L_1, L_2, \ldots, L_n \in \mathcal{F}$.

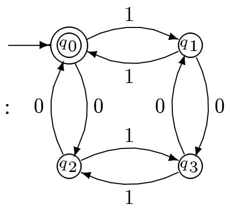
Regular languages

A **deterministic finite automaton** (DFA, for short) is a 5-tuple $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ where Σ is an alphabet, Q is a finite set of **states**, $q_0 \in Q$ is the **initial** state, $F \subseteq Q$ is the set of **final** states, and $\delta : Q \times \Sigma \longrightarrow Q$ is the **transition function**. The mapping δ is extended to $\overline{\delta} : Q \times \Sigma^* \longrightarrow Q$ by $\overline{\delta}(q, \lambda) = q, \overline{\delta}(q, wa) = \delta(\overline{\delta}(q, w), a)$, for $q \in Q, w \in \Sigma^*, a \in \Sigma$. By $\overline{\delta}(q, w) = p$ is meant that w takes $\mathcal{A}$ from the state q to the state p.

The **language accepted** by $\mathcal{A}$ is $L(\mathcal{A}) = \{w \in \Sigma^* \mid \overline{\delta}(q_0, w) \in F\}$.

The DFAs are often represented as graphs. We give only an example which is self-explained.

$\mathcal{A}$:



$\Sigma = \{0, 1\}$

$Q = \{q_0, q_1, q_2, q_3\}$

$F = \{q_0\}$

δ :

	0	1
q_0	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2

$L(\mathcal{A}) = \{w \in \{0, 1\}^* \mid |w|_0 \text{ and } |w|_1 \text{ are both even}\}$

A language is **regular** if it is accepted by some DFA.

Another way of defining the regular languages is by structural induction: any regular language is obtained starting with $\emptyset$ and some of $\{a\}, a \in \Sigma$, and applying finitely many times the operations of union, catenation, and Kleene star. This is the definition by *regular expressions*. The equivalence of the two definitions is known as *Kleene's theorem*.

A **generalized sequential machine** (gsm, for short) is defined similarly as a finite automaton except that at each transition a word is emitted at output.

A **full trio** is a family closed under morphisms, inverse morphisms, and intersections with regular sets. Any full trio is closed under gsms and contains all regular sets.

Theorem 2.2.1. *The family of regular languages is effectively closed under union, intersection, complement, catenation, Kleene star, morphisms, inverse morphisms, gsms, reversal, taking prefixes, and taking factors.*

By effectively closed is meant that, for instance, given two DFAs accepting two regular languages L_1 and L_2, one can effectively find a DFA accepting $L_1 \cup L_2$.

Context-free languages

A **context-free grammar** is a 4-tuple $G = (N, \Sigma, S, P)$, where N and Σ are alphabets, of **nonterminals** and **terminals**, respectively, $S \in N$ is the **start symbol**, and $P \subseteq N \times (N \cup \Sigma)^*$ is a finite set of **productions**. If $(A, \alpha) \in P$ is a production, this is denoted also $A \longrightarrow \alpha$ and, for any $\beta, \gamma \in (N \cup \Sigma)^*$, we say that $\beta A \gamma$ **derives** directly $\beta\alpha\gamma$, denoted $\beta A \gamma \Longrightarrow_G \beta\alpha\gamma$. The reflexive and transitive closure of $\Longrightarrow_G$ is denoted $\Longrightarrow_G^*$.

The **language generated** by G is $L(G) = \{w \in \Sigma^* \mid S \Longrightarrow_G^* w\}$.

A self-explained example is given below.

$$
\begin{array}{lrl}
G = (N, \Sigma, S, P) & P: & S \longrightarrow (S + S) \\
N = \{S\} & & S \longrightarrow (S * S) \\
\Sigma = \{+, *, (,), a\} & & S \longrightarrow a
\end{array}
$$

The language generated by G is the set of all (correct) arithmetic operations with operators $+$ and $*$ and operands represented by the letter a.

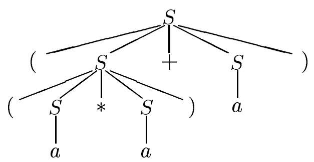
A language is **context-free** if it is generated by a context-free grammar.

An equivalent way of defining the context-free languages is via push-down automata but their rather long definition is omitted.

Theorem 2.2.2. *The family of context-free languages is effectively closed under union, catenation, Kleene star, morphisms, inverse morphisms, gsms, reversal, intersections with regular sets, quotient with regular sets, taking prefixes, and taking factors. It is not closed under intersection and complement.*

A context-free grammar $G = (N, \Sigma, S, P)$ is said to be in the **Chomsky normal form** if all productions are of the form $A \longrightarrow BC$ or $A \longrightarrow a$, where $A, B, C \in N, a \in \Sigma$. For any context-free grammar G, there is a grammar in the Chomsky normal form G' such that $L(G') = L(G) - \{\lambda\}$.

We shall also use the notion of a **derivation tree**. It is used to describe graphically the way in which a word is derived in a context-free grammar. Instead of defining it formally we give an example of a derivation tree below.



A derivation tree of the grammar in the above example for the word $((a * a) + a)$

Theorem 2.2.3 (Pumping lemma). *If $L \subseteq \Sigma^*$ is a context-free language, then there is a constant $p \in \mathbb{N}$ such that every $z \in L$ with $|z| > p$ can be* *written as $z = uvwxy$, for some $u, v, w, x, y \in \Sigma^*$ such that (i) $|vwx| \leq p$, (ii) $vx \neq \lambda$, and (iii) $uv^n wx^n y \in L$, for all $n \geq 0$.*

A set $M \subseteq \mathbb{N}^k$ is **linear** if it has the form $M = \{v_0 + \sum_{i=1}^{k} n_i v_i \mid n_i \in \mathbb{N}\}$, for some $v_i \in \mathbb{N}^k, 0 \leq i \leq k$. A **semilinear** set is a finite union of linear sets.

For an alphabet $\Sigma = \{a_1, a_2, \ldots, a_k\}$, the **Parikh mapping** is a mapping $\Psi : \Sigma^* \longrightarrow \mathbb{N}^k$ defined, for any $w \in \Sigma^*$, by $\Psi(w) = (|w|_{a_1}, |w|_{a_2}, \ldots, |w|_{a_k})$. The **Parikh set** of a language L is $\Psi(L) = \{\Psi(w) \mid w \in L\}$. A language L is **semilinear** if the set $\Psi(L)$ is semilinear.

Theorem 2.2.4 (Parikh [Pa]). *Any context-free language is effectively semilinear.*

2.3 Decidability

In the last section of this preliminary chapter, we define the decidable and undecidable problems and reduction, discuss a bit about Church's thesis, define the Post Correspondence Problem, and, finally, mention a few basic problems decidable or undecidable for regular or context-free languages.

Decidable and undecidable problems

By a **problem**, we mean a question such as: "Is a given context-free language infinite?", or "Is the intersection of two given context-free languages empty?", or "Compute the set of factors of a given regular language". We remark that the first two problems require yes-no answers – such problems are called **decision** problems – whereas the answer to the third problem has to be a language. An **instance** of each of the above problems is a particular context-free language, two particular context-free languages, or a particular regular language, respectively. In general, an instance of a problem is a list of arguments, one for each parameter of the problem.

Formally, a decision problem is a language, of the encodings over some finite alphabet of those instances for which the answer is affirmative; in the first example, this may consist of the (encodings of the) context-free grammars generating finite languages.

As it is intuitively clear, a **decidable** or **solvable** problem is a problem for which an algorithm or effective procedure solving it exists. (The first term applies to decision problems and the second to any problem. Formally, a decision problem is decidable iff the associated language is recursive.)

Correspondingly, an **undecidable** or **unsolvable** problem is a problem for which no algorithm exists. Therefore, the need of formally defining the notion of an algorithm arises. A formal proof for the equivalence between the intuitive and a formal notion is impossible for the following reason. Assume we want to show that a certain formal notion is equivalent with the intuitive notion of the algorithm. In order to prove this, a formal definition of the notion of algorithm is required, so we are back at the beginning. An answer to this dilemma is provided by **Church's thesis** which says that the Turing machine is an adequate formal model for the intuitive notion of an **algorithm**. As shown above, it cannot be proved. For (the overwhelming) arguments supporting Church's thesis, as well as for the definitions of Turing machines and recursive languages, we refer to [Rog] and [RoSa].

The reason why we do not need here these formal definitions is that the customary way of establishing the undecidability (or unsolvability) of a problem P_1 is to reduce to it a problem P_2 known to be undecidable (or unsolvable). We say that a problem P_2 **reduces** to a problem P_1 if there is an algorithm A such that, for any instance p_2 of P_2, $A(p_2)$ is an instance of P_1 and the solution of p_2 (as an instance of P_2) is the same as the solution of $A(p_2)$ (as an instance of P_1). (For decision problems, this means that the answer for p_2 is "yes" if and only if the answer for $A(p_2)$ is "yes".) Now, if P_1 were solvable, then, assuming that A_1 is an algorithm solving P_1, an algorithm for solving P_2 would be the following: take an instance p_2 of P_2, construct $A(p_2)$, and apply the algorithm A_1 to the instance $A(p_2)$ (of P_1). The obtained solution is, by definition, the solution of p_2 as an instance of P_2, hence the algorithm for P_2 works correctly.

Post Correspondence Problem

One of the most suitable reference points for undecidability proofs in language theory is the **Post Correspondence Problem** (PCP, for short). Its original definition[4] is as follows. Given a 4-tuple $(\Sigma, n, \alpha, \beta)$, where Σ is an alphabet, $n \geq 1$, and $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$, $\beta = (\beta_1, \beta_2, \ldots, \beta_n)$ are ordered n-tuples of non-empty words over Σ, the problem is whether or not there exists a finite non-empty sequence of indices $(i_1, i_2, \ldots, i_k)$, for some $k \geq 1, 1 \leq i_j \leq n$, for all $1 \leq j \leq k$, such that $\alpha_{i_1}\alpha_{i_2}\cdots\alpha_{i_k} = \beta_{i_1}\beta_{i_2}\cdots\beta_{i_k}$. The sequence $(i_1, i_2, \ldots, i_k)$ is called a **solution** of the above instance of the PCP.

Theorem 2.3.1 (Post [Po]). *The Post Correspondence Problem is undecidable.*

Other problems

We next define some problems for languages. The following basic problems are defined for arbitrary languages $L_1 \subseteq \Sigma^*$, $L_2 \subseteq \Sigma^*$ which are given by grammars G_1, G_2: $L(G_i) = L_i, i = 1, 2$.

[4] For other equivalent definitons and other aspects of PCP, we refer to [RoSa] or to the recent survey [HaKa].

– **membership**: for an arbitrary $w \in \Sigma^*$, is w in $L(G_1)$?
– **emptiness**: is $L(G_1)$ empty?
– **finiteness**: is $L(G_1)$ finite?
– **inclusion**: is $L(G_1)$ a subset of $L(G_2)$?
– **equivalence**: are the sets $L(G_1)$ and $L(G_2)$ equal?
– **universality**: is it true that $L(G_1) = \Sigma^*$?
– **emptiness of intersection**: is $L(G_1) \cap L(G_2)$ empty?

Theorem 2.3.2. *All the above problems are decidable for regular languages.*

Theorem 2.3.3. *The membership, emptiness, and finiteness problems are decidable for context-free language, whereas the other problems mentioned are undecidable.*

Notice also that the effective closure of regular or context-free languages metioned in the previous section are solvable problems.

Chapter 3

Lengths of words

We consider in this chapter *slender* languages, that is, languages for which the number of words of the same length is bounded from above by a constant. We prove that the slender context-free languages are precisely the *unions of paired loops*, that is, finite unions of sets of the form $\{uv^n wx^n y \mid n \geq 0\}$. This confirms a conjecture of Păun and Salomaa in [PaSa1]. As consequences, we settle some other open problems in the area, concerning linearity, ambiguity, and closure properties of slender context-free languages.

3.1 Slender languages

Considerations concerning lengths of words are an important part of language theory. An infinite sequence $(l_n)_{n\geq 0}$ can be associated in a natural way to a language L: l_n is the number of words of length n in L. The investigation of the case when all terms of the sequence $(l_n)_{n\geq 0}$ are bounded from above by a fixed constant was started by Păun and Salomaa in a series of papers [PaSa1], [PaSa2], [PaSa3]. Such languages have been termed *slender*. As shown in [ADPS], these languages are not only important from theoretical point of view, but also they have applications in cryptography. Some applications can be found also in [CaYu]. The same notion has been investigated for regular languages by Shallit, cf. [Sh], in connection with numeration systems and for L languages in [DPS], [NiSa]. The case when l_n is bounded by a polynomial in n has been considered in [Ra2], [SYZS]. Honkala has introduced and studied a generalization of the notion of slenderness, called Parikh slenderness, by considering languages for which the number of words with a same Parikh vector is bounded from above by a constant, see [Ho1], [Ho2], [Ho3], [Ho4]. As an application of his study, he gave a new very nice method for ambiguity proofs of context-free languages.

One of the most important problems is to find characterizations of such languages. As proved by Păun and Salomaa, cf. [PaSa1], and Shallit, cf. [Sh], the slender regular languages are exactly the *unions of single loops*, that is, finite unions of sets of the form uv^*w.

We prove in this chapter that a characterization similar with the one above holds in the context-free case. More precisely, we show that the slender context-free languages are precisely the *unions of paired loops*, that is, finite unions of sets of the form $\{uv^nwx^ny \mid n \geq 0\}$. This confirms a conjecture of Păun and Salomaa, cf. [PaSa1].

As consequences of this result, we solve several open problems in this area.

3.1.1 Definions and examples

Let k be a positive integer. A language $L \subseteq \Sigma^*$ is called $\boldsymbol{k}$**-slender**[1] if

$$\mathrm{card}\{w \in L \mid |w| = n\} \leq k,$$

for any $n \geq 0$. L is **slender** if it is k-slender for some $k \geq 1$.

A **single loop** is a set of the form

$$\{uv^nw \mid n \geq 0\},$$

for some $u, v, w \in \Sigma^*$. A **paired loop** is a set of the form

$$\{uv^nwx^ny \mid n \geq 0\},$$

where $u, v, w, x, y \in \Sigma^*$. Notice that a single loop is a particular case of a paired loop. In the sequel, by a *loop*[2] we shall understand a single or paired loop.

We call a **union of single (paired) loops** any finite union of single (paired) loops, respectively. A **disjoint union of paired loops** is a finite union of pairwise disjoint paired loops.

Examples

1. The empty language is the only 0-slender language.
2. The language $\{a^nba^n \mid n \geq 0\}$ is 1-slender.
3. The language $L_1 = \{a^nba^m \mid n, m \geq 0\}$ is not slender since

$$\mathrm{card}\{w \in L_1 \mid |w| = n\} = n.$$

[1] A 1-slender language is also called "thin" in [PaSa1]. We shall avoid this term since it is already used with different meanings by [Bea1] and [BePe].

[2] The notion of "loop" has been generalized to the notion of "m-tuple loop", that is, a set of the form $\{x_0y_1^nx_1y_2^nx_2\cdots x_{m-1}y_m^nx_m \mid n \geq 0\}$, in [Ko], where some classes of slender context-sensitive languages are considered.

4. For the language $L_2 = \{(a^n b)^m \mid n, m \geq 0\}$, we have

$$\mathrm{card}\{w \in L_2 \mid |w| = n\} = \mathcal{D}(n),$$

where $\mathcal{D}(n)$ is the number of divisors of n. Hence, L_2 is not slender.

5. For any $k \geq 1$, the language $L_3 = \{a^i b a^{n-i} \mid 1 \leq i \leq k, n \geq 0\}$ is k-slender but not $(k-1)$-slender.

3.2 Loops and slenderness

We prove in this section two technical lemmas which will be useful for the characterization theorem in the next section. The second one will be also used in the next chapter.

Lemma 3.2.1. *Consider the words $u_i, v_i, w_i, x_i, y_i \in \Sigma^*$, for all $i \geq 0$, such that, for any $i \geq 0$, $v_i x_i \neq \lambda$ and $|v_i x_i| \leq p$, where p is a fixed integer. Consider also the sets*

$$A_i = \{u_i v_i^n w_i x_i^n y_i \mid n \geq 0\},$$

for all $i \geq 0$, such that the language

$$L = \bigcup_{i=0}^{\infty} A_i$$

is slender. Then, for any set $I \subseteq \mathbb{N}$ such that, for any $i, j \in I$ with $i \neq j$, $A_i \cap A_j$ is finite, we have that I is finite.

Proof. Suppose that L is k-slender, for some $k \geq 1$. In order to prove the lemma, it is enough to show that

$$\mathrm{card}(I) \leq kp. \tag{3.2.1}$$

We argue by contradiction. Suppose that $\mathrm{card}(I) \geq kp + 1$ and take a subset $I_0 \subseteq I$ such that $\mathrm{card}(I_0) = kp + 1$.

Since $A_i \cap A_j$ is finite for any $i, j \in I_0, i \neq j$, we can find a positive integer N such that

$$N \geq |u_i| + |w_i| + |y_i|, \tag{3.2.2}$$

for any $i \in I_0$, and any word of length at least N in some $A_i, i \in I_0$, does not belong to any $A_j, j \in I_0 - \{i\}$, that is,

$$u_i v_i^n w_i x_i^n y_i \neq u_j v_j^m w_j x_j^m y_j, \tag{3.2.3}$$

for all $i, j \in I_0, i \neq j$, and all $n, m \geq 0$ such that

$$|u_i v_i^n w_i x_i^n y_i| \geq N \text{ and } |u_j v_j^m w_j x_j^m y_j| \geq N.$$

Consider now the set

$$S = \{N, N+1, N+2, \ldots, N+p-1\}.$$

Since, by hypothesis, $0 < |v_i x_i| \leq p$, for $i \in I_0$, it follows by (3.2.2) that, for any $i \in I_0$, the set A_i has at least one word with the length in the set S. But any two words with the lengths in S which belong to different sets A_i are different by (3.2.3). It follows that there are at least $\mathrm{card}(I_0) = kp + 1$ different words having the lengths in the set S. By the pigeonhole principle, there is $n \in S$ such that the set

$$\bigcup_{i \in I_0} A_i \subseteq L$$

contains at least $k+1$ different words of length n. But this contradicts the k-slenderness of L. It follows that (3.2.1) holds and so I is finite, as claimed.

■

Lemma 3.2.2. *If, for some words $u_i, v, w, x, y_i \in \Sigma^*, 1 \leq i \leq 2$, the two sets*

$$A_1 = \{u_1 v^n w x^n y_1 \mid n \geq 0\} \text{ and}$$
$$A_2 = \{u_2 v^n w x^n y_2 \mid n \geq 0\}$$

have infinitely many common words and $|u_1 y_1| \leq |u_2 y_2|$, then $A_2 \subseteq A_1$.

Proof. Since $A_1 \cap A_2$ is infinite, we have $vx \neq \lambda$. Assume that both v and x are non-empty. The case when some of them is empty is treated similarly.

Also, the equation

$$u_1 v^n w x^n y_1 = u_2 v^m w x^m y_2 \tag{3.2.4}$$

has infinitely many solutions $(n, m) \in \mathbb{N} \times \mathbb{N}$. Put $v = z^r, x = t^s$, for primitive words $z, t \in \Sigma^+, r \geq 1, s \geq 1$. Consider n_0, m_0 satisfying (3.2.4) such that

(i) $\min(|u_1 v^{n_0}|, |u_2 v^{m_0}|) \geq \max(|u_1|, |u_2|) + |z|$,
(ii) $\min(|x^{n_0} y_1|, |x^{m_0} y_2|) \geq \max(|y_1|, |y_2|) + |t|$.

As, by hypothesis, $|u_1 y_1| \leq |u_2 y_2|$, we have, by (3.2.4), $n_0 \geq m_0$.

Consider two cases, depending on whether or not $|u_1 v^{n_0}| = |u_2 v^{m_0}|$.

Case 1. $|u_1 v^{n_0}| = |u_2 v^{m_0}|$. Then also $|x^{n_0} y_1| = |x^{m_0} y_2|$ and we have

$$u_1 v^{n_0} = u_2 v^{m_0},$$
$$x^{n_0} y_1 = x^{m_0} y_2.$$

As $n_0 \geq m_0$, we must have $|u_2| \geq |u_1|$ and $|y_2| \geq |y_1|$. It follows by Theorem 2.1.1 (v) that

$$\begin{aligned} u_2 &= u_1 z^k, \quad k \geq 0, \\ y_2 &= t^l y_1, \quad l \geq 0, \end{aligned}$$

and by (3.2.4) for $n = m_0, m = m_0$ we get that $z^{rn_0} = z^{rm_0+k}$ and $t^{sn_0} = t^{sm_0+l}$, hence $r(n_0 - m_0) = k$ and $s(n_0 - m_0) = l$.

We may write now

$$\begin{aligned} A_2 &= \{u_2 v^n w x^n y_2 \mid n \geq 0\} \\ &= \{u_1 z^k z^{rn} w t^{sn} t^l y_1 \mid n \geq 0\} \\ &= \{u_1 z^{r(n+n_0-m_0)} w t^{s(n+n_0-m_0)} y_1 \mid n \geq 0\} \\ &\subseteq \{u_1 z^{rn} w t^{sn} y_1 \mid n \geq 0\} \\ &= A_1. \end{aligned}$$

Case 2. $|u_1 v^{n_0}| \neq |u_2 v^{m_0}|$. We assume $|u_1 v^{n_0}| > |u_2 v^{m_0}|$. The other case is similar.

This implies $|x^{n_0} y_1| < |x^{m_0} y_2|$ and, because $n_0 \geq m_0$, we must have $|y_1| < |y_2|$.

Assume now $|u_1| \geq |u_2|$. (The case when $|u_1| \leq |u_2|$ is similar; the only difference will be pointed out at the very end of the proof.) By Theorem 2.1.1 (v), it follows that

$$\begin{aligned} u_1 &= u_2 z^k, \quad k \geq 0, \\ y_2 &= t^l y_1, \quad l > 0, \end{aligned}$$

and (3.2.4) with $n = n_0, m = m_0$ becomes

$$z^{r(n_0-m_0)+k} w = w t^{s(m_0-n_0)+l}. \tag{3.2.5}$$

Then, by Theorem 2.1.1 (iv),

$$z^{r(n_0-m_0)+k} \sim t^{s(m_0-n_0)+l},$$

hence, by Theorem 2.1.1 (ii), $z \sim t$ and we have the equality $r(n_0 - m_0) + k = s(m_0 - n_0) + l$. As $n_0 \geq m_0$, it follows that $k \leq l$ and

$$l = k + (r+s)(n_0 - m_0). \tag{3.2.6}$$

Claim. There are $z_1, z_2 \in \Sigma^*$ such that

$$\begin{aligned} z &= z_1 z_2, \\ t &= z_2 z_1, \\ w &= (z_1 z_2)^c z_1, c \geq 0. \end{aligned}$$

Proof of Claim. Since $z \sim t$, there are $z_1, z_2 \in \Sigma^*$ such that $z = z_1 z_2, t = z_2 z_1$. If $z_1 = \lambda$ or $z_2 = \lambda$, then $z = t$ and (3.2.5) becomes

$$z^{r(n_0-m_0)+k}w = wz^{r(n_0-m_0)+k},$$

hence, by Theorem 2.1.1 (iii), z and w are powers of the same word. As z is primitive, it follows that $w \in z^*$ and we are done.

Assume now $z_1 \neq \lambda$ and $z_2 \neq \lambda$. Denote $n = r(n_0 - m_0) + k$; $n \geq 1$ because $|u_1 v^{n_0}| > |u_2 v^{m_0}|$. We write (3.2.5) as

$$(z_1 z_2)^n w = w(z_2 z_1)^n,$$

so, by Theorem 2.1.1 (iv), there are $p, q \in \Sigma^*$ such that

$$\begin{aligned}
&(z_1 z_2)^n = pq,\\
&(z_2 z_1)^n = qp,\\
&w = (pq)^i p, i \geq 0.
\end{aligned}$$

Put $p = (z_1 z_2)^j z_1'$, for some $z_1' \in \Sigma^*, |z_1'| < |z|, j \geq 0$.

Assume that $|z_1'| \neq |z_1|$ and consider $|z_1'| < |z_1|$. (The other case is similar.) Then $z_1 = z_1' z_1''$, for some $z_1'' \in \Sigma^+$. It follows that $q = z_1'' z_2 (z_1 z_2)^{n-j-1}$ and so

$$(z_2 z_1)^n = qp = z_1'' z_2 (z_1 z_2)^{n-1} z_1'.$$

If $n \geq 2$, then $z_2 z_1$ is a proper factor (neither a prefix nor a suffix) of $(z_2 z_1)^2$, a contradiction with the fact that $z_2 z_1$ is primitive. (z primitive implies, by Theorem 2.1.1 (ii), that $z_2 z_1$ is also primitive.)

If $n = 1$, then

$$z_2 z_1' z_1'' = z_1'' z_2 z_1'$$

hence, by Theorem 2.1.1 (iii), there is $z_3 \in \Sigma^+$ such that $z_1'', z_2 z_1' \in z_3^+$. Then $z \in z_3 z_3^+$, again a contradiction.

Consequently, $|z'| \neq |z_1|$ is impossible hence $|z_1'| = |z_1|$. Then $z_1' = z_1$ and $p = (z_1 z_2)^j z_1, q = z_2 (z_1 z_2)^{n-j-1}$. Finally

$$w = (z_1 z_2)^{ni+j} z_1,$$

and we can take $c = ni + j \geq 0$. The claim is proved. ∎

Using now the statement of the claim above and the equality (3.2.6) we may write

$$\begin{aligned}
A_2 &= \{u_2 v^n w x^n y_2 \mid n \geq 0\}\\
&= \{u_2 z^{(r+s)n+c+l} z_1 y_1 \mid n \geq 0\}\\
&= \{u_2 z^{(r+s)(n+n_0-m_0)+c+k} z_1 y_1 \mid n \geq 0\}\\
&\subseteq \{u_2 z^{(r+s)n+c+k} z_1 y_1 \mid n \geq 0\}\\
&= \{u_1 v^n w x^n y_1 \mid n \geq 0\}\\
&= A_1.
\end{aligned}$$

In the case when $|u_1| \leq |u_2|$, we obtain, as above,

$$\begin{aligned} u_2 &= u_1 z^k, \quad k \geq 0, \\ y_2 &= t^l y_1, \quad\ l > 0, \end{aligned}$$

and $z^{r(n_0-m_0)-k}w = wt^{s(m_0-n_0)+l}$. Also, we have the following equality:

$$l + k = (r+s)(n_0 - m_0).$$

Using this and the statement of the claim proved above (which is still valid here; the proof is the same), we write

$$\begin{aligned} A_2 &= \{u_2 v^n w x^n y_2 \mid n \geq 0\} \\ &= \{u_1 z^{(r+s)n+c+l+k} z_1 y_1 \mid n \geq 0\} \\ &= \{u_1 z^{(r+s)(n+n_0-m_0)+c} z_1 y_1 \mid n \geq 0\} \\ &\subseteq \{u_1 z^{(r+s)n+c} z_1 y_1 \mid n \geq 0\} \\ &= \{u_1 v^n w x^n y_1 \mid n \geq 0\} \\ &= A_1. \end{aligned}$$

Consequently, in all cases $A_2 \subseteq A_1$, as claimed. ■

3.3 The characterization theorem

We prove in this section the main result of this chapter, namely the characterization of the slender context-free languages as unions of paired loops.

Theorem 3.3.1. *Every slender context-free language is a union of paired loops.*

Proof. Let $L \subseteq \Sigma^*$ be a k-slender context-free language. Assume that L is infinite. (If it is finite, then, obviously, it is a union of paired loops.)

According to the pumping lemma for context-free languages (see Theorem 2.2.3) there exists $p \in \mathbb{N}$ such that every $z \in L$ with $|z| > p$ can be written in the form $z = uvwxy, u, v, w, x, y \in \Sigma^*$, where

$$|vwx| \leq p, \tag{3.3.7}$$

$$vx \neq \lambda, \tag{3.3.8}$$

$$uv^n wx^n y \in L, \text{ for all } n \geq 0. \tag{3.3.9}$$

Denote

$$L_1 = \{w \in L \mid |w| \leq p\},$$

and put

$$L - L_1 = \{z_0, z_1, z_2, \dots\},$$

where the order is arbitrary. Applying now the pumping lemma for each $z_i, i \geq 0$, we get $z_i = u_i v_i w_i x_i y_i$, for $u_i, v_i, w_i, x_i, y_i \in \Sigma^*$ verifying the properties corresponding to (3.3.7), (3.3.8), and (3.3.9). For any $i \geq 0$, denote

$$A_i = \{u_i v_i^n w_i x_i^n y_i \mid n \geq 0\}.$$

Consider a subset $I \subseteq \mathbb{N}$ such that

$$\bigcup_{i \in I} A_i = \bigcup_{i \in \mathbb{N}} A_i$$

and

$$A_i \not\subseteq A_j, \text{ for any } i, j \in I, i \neq j. \tag{3.3.10}$$

(In this way, all redundant A_i's are eliminated.) We have then

$$L = L_1 \cup L_2,$$

where

$$L_2 = \bigcup_{i \in I} A_i.$$

Because L_1 is finite, it is a union of paired loops. Therefore it is enough to prove that L_2 is a union of paired loops.

Consider now the set of triples

$$C = \{(v_i, w_i, x_i) \mid i \in I\}.$$

From (3.3.7) it follows that C is finite. Let d be its cardinality and write

$$C = \{(v_1, w_1, x_1), (v_2, w_2, x_2), \dots, (v_d, w_d, x_d)\}.$$

For every $r, 1 \leq r \leq d$, we denote

$$B_r = \{i \in I \mid (v_i, w_i, x_i) = (v_r, w_r, x_r)\}.$$

It follows that

$$I = \bigcup_{r=1}^{d} B_r$$

(In fact, $B_r, 1 \leq r \leq d$, constitute a partition of the set I).

In order to prove I finite, it is enough to show that all $B_r, 1 \leq r \leq d$, are finite.

Consider some fixed r. For any $i, j \in B_r, i \neq j$, we have that $A_i \cap A_j$ is finite. Indeed, if $A_i \cap A_j$ is infinite, then, by Lemma 3.2.2, either A_i is a subset of A_j or conversely, in contradiction with the assumption (3.3.10).

We now apply Lemma 3.2.1 for the sets $(A_i)_{i \in B_r}$. By (3.3.9), we have the inclusion

$$\bigcup_{i \in B_r} A_i \subseteq L,$$

so $\bigcup_{i \in B_r} A_i$ is slender. The finiteness of any intersection $A_i \cap A_j$, $i, j \in B_r, i \neq j$, has been proved above and the other conditions are fulfilled from (3.3.7) and (3.3.8). It follows that B_r is finite. Consequently, L is a finite union of paired loops and the proof is completed. ∎

We notice that the characterization of slender regular languages as unions of single loops by Păun and Salomaa [PaSa1] and Shallit [Sh] follows immediately from our characterization Theorem 3.3.1. Indeed, an infinite paired loop is a regular language if and only if it is, in fact, a single loop.

Corollary 3.3.2. *Every slender regular language is a union of single loops.*

3.3.1 A stronger version

From the proof of Theorem 3.3.1 we obtain a result which is actually stronger because it says more about the structure of the loops. This will be very useful in the next chapter for a closer analysis of the structure of the slender context-free languages.

Before stating the result, we need to recall briefly how the pumping lemma for context-free languages (Theorem 2.2.3) is proved. Given a context-free language $L \subseteq \Sigma^*$, consider first a context-free grammar in the Chomsky normal form $G = (N, \Sigma, S, P)$ generating L and put[3]

$$p = 2^{\mathrm{card}(N)-1}$$

where N stands for the set of nonterminals of G. Then, for any word $z \in L$ such that $|z| > p$, in any derivation tree of G for z, there is a path from the root to a leaf, the length of which is at least $\mathrm{card}(N) + 1$. Among the $\mathrm{card}(N) + 1$ vertices closest to the frontier which are labeled by nonterminals, there must

[3]Usually, this constant is taken to be $2^{\mathrm{card}(N)}$ but $p = 2^{\mathrm{card}(N)-1}$ is still good; in fact, this is the least value for which the pumping lemma works with the same proof. We take this value in order to improve the upper bounds given by Theorem 4.2.3 in the next chapter.

be two labeled by the same nonterminal. This repeated nonterminal gives the decomposition of z as in the statement of the pumping lemma. (See also the beginning of the proof of Theorem 3.3.1.) In what follows, we shall denote this constant p by p_L and it will be understood that it comes from a grammar in Chomsky normal form for L.

The next result, stronger than the one in Theorem 3.3.1, should be now clear.

Theorem 3.3.3. *If $L \subseteq \Sigma^*$ is a context-free language, then*
(i) L is slender if and only if it is a union of paired loops;
(ii) if L is slender, then it is given by a union of paired loops

$$L = \bigcup_{i=1}^{m} \{u_i v_i^n w_i x_i^n y_i \mid n \geq 0\},$$

for some $m \geq 0, u_i, v_i, w_i, x_i, y_i \in \Sigma^*, 1 \leq i \leq m$, such that, for each $i, 1 \leq i \leq m$, with $|u_i v_i w_i x_i y_i| > p_L$ there is a derivation tree for G as in Figure 3.3.1 below:

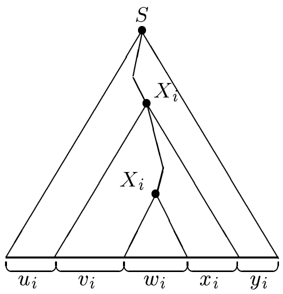


Figure 3.3.1: the derivation tree for the loop i

where $X_i \in N$ and $|v_i w_i x_i| \leq p_L$.

3.4 Consequences

Using the characterization result in Theorem 3.3.1, we settle several open problems about slender context-free languages, mostly concerning closure properties. Actually, only the closure under morphisms and intersections were left open in [PaSa1], but we prove also the closure under difference.

First, we have the following corollary of Theorem 3.3.1.

Corollary 3.4.1. *Any slender context-free language is linear and non-ambiguous.*

Proof. The linearity is obvious. For the other part, a slender context-free language is a union of paired loops and, as proved in [PaSa1], it is a disjoint union of paired loops, hence non-ambiguous. ■

Remark. We notice that the proof in [PaSa1] that any union of paired loops is a disjoint union of paired loops is not constructive. A constructive solution will be given in Chapter 4, cf. Theorem 4.4.5.

We now start considering closure properties of slender context-free languages. The closure under morphisms comes first.

Theorem 3.4.2. *The family of slender context-free languages is closed under morphisms.*

Proof. Immediate from Theorem 3.3.1 since any morphic image of a union of paired loops is again a union of paired loops. ■

The closure of the family of slender context-free language under intersection and difference will follow easily from the following lemma. This lemma is actually much stronger and it will be used to prove some effective constructions in the next chapter.

We shall denote, for a positive integer n, the set of words of length at most n over Σ by $\Sigma^{\leq n}$.

Lemma 3.4.3. *The intersection and the difference of two given paired loops are effectively disjoint unions of paired loops.*

Proof. Consider two paired loops

$$A = \{uv^n wx^n y \mid n \geq 0\}, u, v, w, x, y \in \Sigma^*,$$
$$B = \{u'v'^m w'x'^m y' \mid m \geq 0\}, u', v', w', x', y' \in \Sigma^*.$$

We suppose that all words v, x, v', x' are non-empty. The cases when some of them are empty are treated similarly.

We consider three cases.

Case 1. $\rho(v) \not\sim \rho(v')$ or $\rho(x) \not\sim \rho(x')$. We prove that in this case $A \cap B$ is finite.

More precisely, consider n_1 and m_1 to be the least non-negative integers n and m such that the following two conditions are fulfilled

(i) $\min(|uv^n|, |u'v'^m|) \geq \max(|u|, |u'|) + |v| + |v'|$,
(ii) $\min(|x^n y|, |x'^m y'|) \geq \max(|y|, |y'|) + |x| + |x'|$,

and put

$$M_1 = \max(|uv^{n_1} wx^{n_1} y|, |u'v'^{m_1} w'x'^{m_1} y'|).$$

Then we claim that $A \cap B \subseteq \Sigma^{\leq M_1 - 1}$.

Suppose that there is $z \in A \cap B$ such that $|z| \geq M_1$ and put $z = uv^n wx^n y = u'v'^m w'x'^m y'$. By the choice of M_1, it follows that $n \geq n_1, m \geq m_1$ and so the words v^n and v'^m have a common factor of length at least $|v| + |v'|$. Therefore, by Fine and Wilf's theorem, (see Corollary 2.1.3) $\rho(v) \sim \rho(v')$. The same applies for x^n and x'^m, so also $\rho(x) \sim \rho(x')$, a contradiction.

Case 2. $\rho(v) \sim \rho(v'), \rho(x) \sim \rho(x')$, and $\dfrac{|v|}{|x|} = \dfrac{|v'|}{|x'|}$. We prove first that $A \cap B$ is infinite if and only if there is $z \in A \cap B$ such that $M_1 \leq |z| < M_1 + \mathrm{lcm}(|vx|, |v'x'|)$ and when $A \cap B$ is finite, then $A \cap B \subseteq \Sigma^{\leq M_1 - 1}$.

Denote

$$n_0 = \frac{\mathrm{lcm}(|vx|, |v'x'|)}{|vx|}, m_0 = \frac{\mathrm{lcm}(|vx|, |v'x'|)}{|v'x'|}.$$

Assume first that there is $z \in A \cap B$ such that $|z| \geq M_1$ and put $z = uv^n wx^n y = u'v'^m w'x'^m y'$. Since $n_0|v| = m_0|v'|$ and $n_0|x| = m_0|x'|$, it follows that

$$uv^{n+rn_0} wx^{n+rn_0} y = u'v'^{m+rm_0} w'x'^{m+rm_0} y',$$

for any $r \geq 0$, so $A \cap B$ is infinite.

Assume now $A \cap B$ infinite. There is $z \in A \cap B, |z| \geq M_1$. If $|z| < M_1 + \mathrm{lcm}(|vx|, |v'x'|)$, then there is nothing to prove. Otherwise, if $z = uv^n wx^n y = u'v'^m w'x'^m y'$, then, as above,

$$uv^{n-n_0} wx^{n-n_0} y = u'v'^{m-m_0} w'x'^{m-m_0} y'.$$

Therefore, we have obtained a word $z' \in A \cap B$ such that $|z'| = |z| - \mathrm{lcm}(|vx|, |v'x'|)$. The procedure continues until $z \in A \cap B$ as required is found.

Suppose now that $A \cap B$ is infinite and the shortest word in the set $\{z \in A \cap B \mid |z| \geq M_1\}$ is $z_0 = uv^p wx^p y = u'v'^q w'x'^q y'$. As above, we have

$$uv^{p+sn_0} wx^{p+sn_0} y = u'v'^{q+sm_0} w'x'^{q+sm_0} y',$$

for any $s \geq 0$, and, because of length considerations,

$$uv^{p+n} wx^{p+n} y \neq u'v'^{q+m} w'x'^{q+m} y',$$

for any $n, m \geq 0$ such that n is not divisible by n_0 or m is not divisible by m_0. Consequently, we have

$$\begin{aligned} A \cap B = &\{z \in A \cap B \mid |z| < M_1\} \\ &\cup \{uv^p (v^{n_0})^n w (x^{n_0})^n x^p y \mid n \geq 0\}, \\ A - B = &\{z \in A - B \mid |z| < |z_0|\} \\ &\cup \bigcup_{i=1}^{n_0 - 1} \{uv^{p+i} (v^{n_0})^n w (x^{n_0})^n x^{p+i} y \mid n \geq 0\}. \end{aligned}$$

Case 3. $\rho(v) \sim \rho(v'), \rho(x) \sim \rho(x')$, and $\dfrac{|v|}{|x|} \neq \dfrac{|v'|}{|x'|}$. Assume that $\dfrac{|v|}{|x|} < \dfrac{|v'|}{|x'|}$. The other case is similar.

Take n_2 and m_2 the least non-negative integers n and m such that the following three conditions are fulfilled

(i) $\min(|uv^n|, |u'v'^m|) \geq \max(|u|, |u'|) + |v| + |v'|$,
(ii) $\min(|x^n y|, |x'^m y'|) \geq \max(|y|, |y'|) + |x| + |x'|$,
(iii) $|uwxv'| - |u'| + n|v| \leq n\dfrac{|vx|}{|v'x'|}|v'| - \dfrac{|u'w'y'| - |uwy|}{|v'x'|}|v'|$.

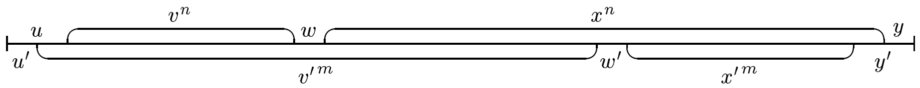
Notice that $|v| < \dfrac{|vx|}{|v'x'|}|v'|$, so there exist such integers n and m, hence n_2 and m_2 are well defined. Put also

$$M_2 = \max(|uv^{n_2}wx^{n_2}y|, |u'v'^{m_2}w'x'^{m_2}y'|).$$

Assume now that there is $z \in A \cap B$ such that $|z| \geq M_2$ and put

$$z = uv^n wx^n y = u'v'^m w'x'^m y'. \tag{3.4.11}$$

Due to the choice of M_2, n_2, and m_2, the equality (3.4.11) is graphically represented as in the following picture.



where we assumed $|u| \geq |u'|, |y'| \geq |y|$. (All the other cases are similar.) We are in the conditions of Fine and Wilf's theorem and we get that the primitive roots of v, x, v', and x' are conjugates. More precisely, using also Theorem 2.1.1, we have the following relations:

$$\begin{array}{ll}
v = z^{r_0}, & z \in \Sigma^+, z \text{ primitive}, r_0 \geq 1, \\
v' = (z_1'' z_1')^{r_1}, & z = z_1' z_1'', r_1 \geq 1, \\
x = (z_2'' z_2')^{r_2}, & z = z_2' z_2'', r_2 \geq 1, \\
x' = (z_3'' z_3')^{r_3}, & z = z_3' z_3'', r_3 \geq 1, \\
w = z^{q_1} z_2', & q_1 \geq 0, \\
w' = z_1'' z^{q_2} z_3', & q_2 \geq 0, \\
u = u' z_1'' z^{q_3}, & q_3 \geq 0, \\
y' = z_3'' z^{q_4} z_2' y, & q_4 \geq 0.
\end{array}$$

(Notice that it may be that $w' = z_1''(z_3'')^{-1}$, in the case that w' is very short and z_3'' is a suffix of z_1'', but the reasoning in this case is analogous. A similar remark holds for y' as well.)

Therefore, we have

$$uv^nwx^ny = u'z_1''z^{q_1+q_3}z^{(r_0+r_2)n}z_2'y, \tag{3.4.12}$$
$$u'v'^mw'x'^my' = u'z_1''z^{q_2+q_4+1}z^{(r_1+r_3)m}z_2'y. \tag{3.4.13}$$

We now obtain, as in Case 2, that $A \cap B$ is infinite if and only if there is $z \in A \cap B$ such that $M_2 \leq |z| < M_2 + |z| \operatorname{lcm}(r_0 + r_2, r_1 + r_3)$ and when $A \cap B$ is finite, then $A \cap B \subseteq \Sigma^{\leq M_2 - 1}$.

Also, if $A \cap B$ is infinite and the shortest word in the set $\{t \in A \cap B \mid |t| \geq M_2\}$ is $t_0 = u'z_1''z^{q_1+q_3}z^{(r_0+r_2)p}z_2'y$, then, due to length considerations, we get from (3.4.11), (3.4.12), and (3.4.13) that

$$\begin{aligned} A \cap B = {} & \{t \in A \cap B \mid |t| < M_2\} \\ & \cup \{u'z_1''z^{q_1+q_3+(r_0+r_2)p}(z^{(r_0+r_2)n_0})^n z_2'y \mid n \geq 0\}, \\ A - B = {} & \{t \in A - B \mid |t| < |t_0|\} \\ & \cup \bigcup_{i=1}^{n_0-1} \{u'z_1''z^{q_1+q_3+(r_0+r_2)p+i}(z^{(r_0+r_2)n_0})^n z_2'y \mid n \geq 0\}. \end{aligned}$$

The lemma is proved. ∎

The following result is an immediate consequence of Theorem 3.3.1 and Lemma 3.4.3.

Corollary 3.4.4. *The family of slender context-free languages is closed under difference and intersection.*

This result will be strengthened in the next chapter by proving that the closure under difference and intersection is also effective. We cannot conclude the effectivity of this closure from the fact that the result in Lemma 3.4.3 is effective, because we need an algorithm for writing a slender context-free language as a union of paired loops. Such an algorithm will be given in the next chapter.

Chapter 4

Decidability and slenderness

We investigate in this chapter the structure of the slender context-free languages. We find some effective representations of such languages as unions of paired loops of a special form. This allows us to prove that the slenderness problem is decidable for context-free languages. Some other decidability problems and effective constructions are considered.

4.1 Slenderness problem

Besides the characterization of the slender context-free languages given in Chapter 3, another important problem is the decidability of slenderness. This problem was shown to be decidable for unambiguous context-free languages in [ADPS]. (In virtue of the fact that all slender context-free languages are unambiguous, one may be tempted to say that this implies the decidability of the slenderness problem for arbitrary context-free languages which, however, is not true, since ambiguity is undecidable for context-free languages.) In the much more difficult case of arbitrary context-free languages, the problem was also proved to be decidable by Raz, cf. [Ra2]. (See [Ra1] for an extended abstract.) Recently, another proof of this result was given by Honkala, cf. [Ho4], as a consequence of a more general result; he proved that the slenderness and the Parikh slenderness problems are both decidable for bounded semilinear languages.

In this chapter, we perform a deeper analysis of the structure of slender context-free languages. As proved in Chapter 3, the slender context-free languages are precisely the unions of paired loops. As our essential tool, we prove that the lengths of words in the paired loops composing a slender context-free language L can be bounded from above by some constants depending on L only. This is done by a close analysis of the structure of the derivation trees corresponding to the paired loops.

In this way, we are able to give two new and simple proofs for the decidability of the slenderness problem for context-free languages. Moreover, we show that a representation as a union of paired loops can be effectively found for an arbitrary slender context-free language; even a representation as a disjoint such union. As a consequence, we prove that the maximal number of words of the same length is computable. Finally, some undecidability results are considered.

4.2 Bounds on loops

We show in this section that, if L is a slender context-free language, then it can be written as a union of paired loops

$$L = \bigcup_{i=1}^{m} \{u_i v_i^n w_i x_i^n y_i \mid n \geq 0\}, \tag{4.2.1}$$

such that the lengths of the words $u_i, v_i, w_i, x_i, y_i, 1 \leq i \leq n$, are bounded by some a priori computable constants depending on L only. This result is our most important tool in this chapter; the proofs of the other results are based essentially on it.

4.2.1 Combined loops

We start with two technical lemmas concerning paired loops of a slender language. The essential idea contained in these lemmas, which will become clear in the proof of Theorem 4.2.3, is that, given a word belonging to a slender context-free language, we cannot pump independently at different places. If such a situation appears, then the pumped positions collapse.

Lemma 4.2.1. *Consider the words $w_i, u, v, x, y \in \Sigma^*, 1 \leq i \leq 5$, such that $xy \neq \lambda$ and the sets*

$$R_m = \{w_1 x^n w_2 y^n w_3 u^m w_4 v^m w_5 \mid n \geq 0\},$$

for all $m \geq 0$. If the language

$$R = \bigcup_{m=0}^{\infty} R_m$$

is slender, then there exists $m_0 \geq 0$ such that

$$\bigcup_{m=0}^{\infty} R_m = \bigcup_{m=0}^{m_0} R_m.$$

Proof. By contradiction. Assume that, for any $m' \geq 0$,

$$\bigcup_{m=0}^{\infty} R_m \neq \bigcup_{m=0}^{m'} R_m.$$

So, there is $m'' \geq m' + 1$ such that $R_{m''} \not\subseteq \bigcup_{m=0}^{m'} R_m$. We may assume that $R_{m''} \not\subseteq \bigcup_{m=0}^{m''-1} R_m$ because if this is not the case, then there is $m''', m' + 1 \leq m''' \leq m'' - 1$, such that $R_{m'''} \not\subseteq \bigcup_{m=0}^{m'} R_m$. Therefore, we may suppose that there is an infinite sequence $(m_i)_{i \geq 1}$, $0 \leq m_1 < m_2 < \cdots$, such that, for any $i \geq 1$,

$$R_{m_i} \not\subseteq \bigcup_{m=0}^{m_i - 1} R_m.$$

By Lemma 3.2.2, it follows that, for any $1 \leq i < j$, R_{m_i} and R_{m_j} have finitely many common words. Therefore, if $k \geq 1$ is an arbitrary fixed integer, we can find a positive integer M such that

$$M \geq \max_{1 \leq i \leq k|xy|+1} (|w_1 w_2 w_3 w_4 w_5| + m_i |uv|)$$

and all words of length at least M in the set

$$\bigcup_{i=1}^{k|xy|+1} R_{m_i}$$

belong exactly to one set R_{m_i}. (The integer M is chosen such that, for any $1 \leq i \leq k|xy| + 1$, the shortest word in R_{m_i} is shorter than M, and any word $z \in R_{m_i}$ such that $|z| \geq M$ does not belong to any R_{m_j} for $1 \leq j \leq k|xy| + 1, j \neq i$.)

Consider the interval $[M, M + |xy| - 1]$ of integers. For any $1 \leq i \leq k|xy| + 1$, the lengths of words in R_{m_i} form an arithmetical progression of ratio $|xy| \neq 0$, having the first element smaller than M. Thus, each set $\{|z| \mid z \in R_{m_i}\}, 1 \leq i \leq k|xy| + 1$, has one element in the interval. Since there are $k|xy| + 1$ sets, we can find $k + 1$ words of the same length. As they have the lengths at least M, they are all different. Consequently, there are $k + 1$ different words of the same length in R. As k has been arbitrarily chosen, this contradicts the slenderness of R. ■

Lemma 4.2.2. *Consider the words $w_i, u, v, x, y \in \Sigma^*, 1 \leq i \leq 5$, such that $uv \neq \lambda$ and the sets*

$$R_m = \{w_1 x^m w_2 u^n w_3 v^n w_4 y^m w_5 \mid n \geq 0\},$$

for all $m \geq 0$. *If the language*

$$R = \bigcup_{m=0}^{\infty} R_m$$

is slender, then there exists $m_0 \geq 0$ *such that*

$$\bigcup_{m=0}^{\infty} R_m = \bigcup_{m=0}^{m_0} R_m.$$

Proof. Similar with the proof of Lemma 4.2.1. ■

We are now ready to state the main result of this section. It will be proved in the remaining part of this section. Remember that the notation p_L was defined in Chapter 3 (see page 34).

Theorem 4.2.3. *If* L *is a slender context-free language, then* L *can be written as a union of paired loops in (4.2.1):*

$$L = \bigcup_{i=1}^{m} \{u_i v_i^n w_i x_i^n y_i \mid n \geq 0\}$$

with the following conditions:
(i) $|v_i x_i| \leq p_L$ *and*
(ii) $|u_i w_i y_i| \leq p_L^2 + p_L$.

Proof. Consider the representation of L from Theorem 3.3.3 and consider an arbitrary loop of L, say $\{u_i v_i^n w_i x_i^n y_i \mid n \geq 0\}$, such that $|u_i v_i w_i x_i y_i| > p_L$. (If there is no such loop, then L is already in the required form.) Corresponding to this loop, we have, by Theorem 3.3.3 (ii), the derivation tree of G in Figure 4.2.1.

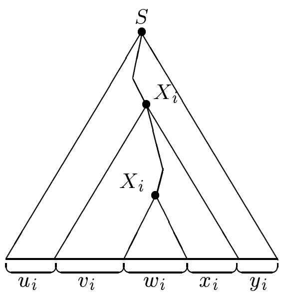


Figure 4.2.1: the derivation tree for the loop i

4.2.2 Reduction of the tree

We now give a procedure for an iterative reduction of the tree in Figure 4.2.1. The basic idea is to apply the pumping lemma and to remove the pumping words (and, accordingly, to modify the derivation tree), these steps being applied as long as possible. The procedure is clearly terminating since the length of the word is strictly reduced at each iteration.

Algorithm 4.2.4 (Tree reduction algorithm).

(1) In the tree in Figure 4.2.1, replace the subtree rooted at the vertex labeled X_i which is closer to the root by the subtree rooted at the vertex labeled X_i which is closer to the frontier; we obtain the derivation tree of G depicted below, which we call *current tree* in the algorithm.

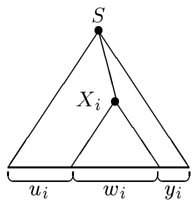


Figure 4.2.2: the removing of the pumping words

(2) As long as the length of the frontier of the current tree is at least $p_L + 1$, perform the steps (3)–(5) below.

(3) Find a longest path from the root to a leaf in the current tree. (Break a tie arbitrarily.) Because the length of the frontier is at least $p_L + 1 = 2^{\mathrm{card}(N)-1} + 1$, this path contains at least $\mathrm{card}(N) + 1$ vertices labeled by nonterminals.

(4) Find the nonterminal, say X, which labels two vertices on this path in such a way that the vertex labeled X which is closer to the root is as close to the frontier as possible.

(Clearly, the frontier of the subtree rooted at the vertex labeled X which is closer to the root has the length at most p_L.)

(5) Replace the subtree rooted at the vertex labeled X which is closer to the root by the subtree rooted at the vertex labeled X which is closer to the frontier. (As we did at step (1) with X_i instead of X.) The new current derivation tree is obtained.

The essential idea of the proof, which will become clear later on, is as follows. By applying the reduction algorithm above, we may pump independently at different positions in a word and then, using Lemmas 4.2.1 and 4.2.2, we show that those positions collapse (see also the remark before Lemma 4.2.1).

In this way we prove that the loop we have started with is included in a bigger set which is still included in L and which can be decomposed as a union of paired loops as required.

4.2.3 Application of the algorithm

Consider next the possible situations that may appear when steps (3)–(5) in the reduction algorithm are applied twice consecutively and also when the second considered application of (3)–(5) is the last one, that is, after it is performed, the frontier of the current tree becomes at most p_L.

Denote by X (Y) the repeated nonterminal found at step (4) at the first (second) application, respectively. Also, denote by u, v the words pumped by X and by x, y the ones pumped by Y. We have $0 < |uv| \leq p_L, 0 < |xy| \leq p_L$. We assume that all words u, v, x, y are nonempty. The cases when some of them are empty are similarly treated.

Depending on the relative position of the considered vertices in the tree at the moment when step (4) is performed the second time (one labeled X and two labeled Y), there are four main cases. (The others are analogues of some of these.) We now consider all of them in details.

Case 1. The relative position of the vertices in the tree is depicted in Figure 4.2.3 (a).

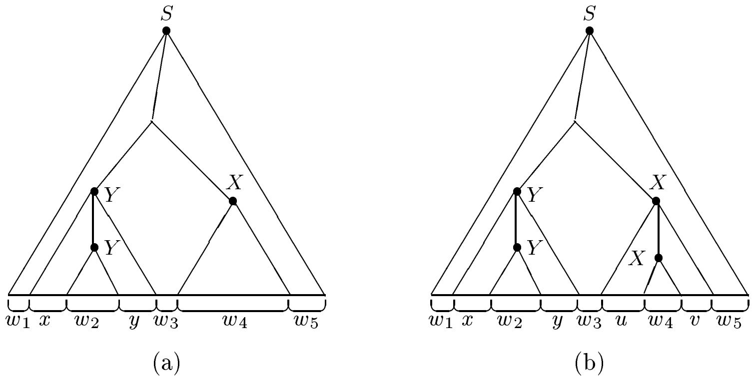


Figure 4.2.3: (a) relative position in Case 1; (b) combined pumpings in Case 1

When pumping by both X and Y we have the situation in Figure 4.2.3 (b). Thus

$$w_1 x^n w_2 y^n w_3 u^m w_4 v^m w_5 \in L, \tag{4.2.2}$$

for all $n, m \geq 0$. Denote, for any $m \geq 0$,

$$R_m = \{w_1 x^n w_2 y^n w_3 u^m w_4 v^m w_5 \mid n \geq 0\}.$$

From (4.2.2) we have

$$\bigcup_{m=0}^{\infty} R_m \subseteq L$$

hence $\bigcup_{m=0}^{\infty} R_m$ is slender. By Lemma 4.2.1, there is $m_0 \geq 0$ such that

$$\bigcup_{m=0}^{\infty} R_m = \bigcup_{m=0}^{m_0} R_m. \tag{4.2.3}$$

Chose now m and n such that the following conditions are fulfilled:

(i) $|v^m| \geq |w_3 u^{m_0} w_4 v^{m_0}| + |v| + |y|$,
(ii) $|y^n| \geq |x| + |y|$,
(iii) $m\left(\dfrac{|uv|}{|xy|}|x| - |u|\right) + |yu| - |w_3| \leq n|y| \leq (m - m_0)\dfrac{|uv|}{|xy|}|x| - |uxw_2w_3|$.

Clearly, there are such m and n. By (4.2.3), there are m', n' with $0 \leq m' \leq m_0, n' \geq 0$, such that

$$w_1 x^n w_2 y^n w_3 u^m w_4 v^m w_5 = w_1 x^{n'} w_2 y^{n'} w_3 u^{m'} w_4 v^{m'} w_5. \tag{4.2.4}$$

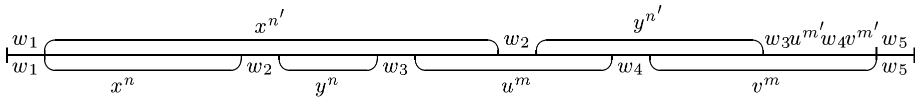
It is easy to see that the first inequality of (iii) implies

$$|x^{n'} w_2| + |y| + |u| \leq |x^n w_2 y^n w_3 u^m| \tag{4.2.5}$$

and that the second inequality of (iii) gives

$$|x^{n'}| \geq |x^n w_2 y^n w_3| + |u| + |x|. \tag{4.2.6}$$

From (i), (ii), (4.2.5), and (4.2.6), it follows that the equality (4.2.4) is depicted as in the figure below



where $x^{n'}$ and y^n, $x^{n'}$ and u^m, u^m and $y^{n'}$, and $y^{n'}$ and v^m overlap each other on a part longer than $|xy|, |xu|, |uy|$, and $|yv|$, respectively. Using now the periodicity theorem of Fine and Wilf we obtain that the primitive roots of x, y, u, v are all conjugated. More precisely, we have, using Theorem 2.1.1,

$$\begin{array}{ll}
x = z^{r_1}, & z \in \Sigma^+, z \text{ primitive}, r_1 \geq 1, \\
y = (z_1'' z_1')^{r_2}, & z = z_1' z_1'', r_2 \geq 1, \\
u = (z_2'' z_2')^{r_3}, & z = z_2' z_2'', r_3 \geq 1, \\
v = (z_3'' z_3')^{r_4}, & z = z_3' z_3'', r_4 \geq 1.
\end{array}$$

For the other words, which are not pumping words, we have

$$\begin{aligned} w_2 &= z^{s_1} z_1', & s_1 &\geq 0, \\ w_3 &= z_1'' z^{s_2} z_2', & s_2 &\geq 0, \\ w_4 &= z_2'' z^{s_3} z_3', & s_3 &\geq 0. \end{aligned}$$

Remark. Notice that it may also be that $w_3 = z_1''(z_2'')^{-1}$, when w_3 is very short and z_2'' is a sufix of z_1'', but the reasoning in this case is similar. The same holds for w_4. Such a remark can be made as well at several places in the sequel, but we shall omit it without mentioning. In all omitted cases, the reasoning is completely unchanged.

Using the above equalities, we now have, by (4.2.2),

$$w_1 x^n w_2 y^n w_3 u^m w_4 v^m w_5 = w_1 z^{s_1+s_2+s_3+2} z^{(r_1+r_2)n+(r_3+r_4)m} z_3' w_5 \in L, \quad (4.2.7)$$

for any $n, m \geq 0$. Thus, the independent pumpings in (4.2.2) collapse into the form of (4.2.7).

Assume now that the second application of (3)–(5) is the last one. This means that $|w_1 w_2 w_3 w_4 w_5| \leq p_L$ which implies

$$|w_1 z^{s_1+s_2+s_3+2} z_3' w_5| \leq p_L. \quad (4.2.7')$$

By construction, we know also that $|x w_2 y| \leq p_L$ and $|u w_4 v| \leq p_L$, thus

$$|z^{r_1+r_2}| \leq p_L, \quad (4.2.7'')$$
$$|z^{r_3+r_4}| \leq p_L. \quad (4.2.7''')$$

Case 2. The relative position of the vertices here is shown in Figure 4.2.4 (a). Notice that here $y = w_3 w_4 w_5$.

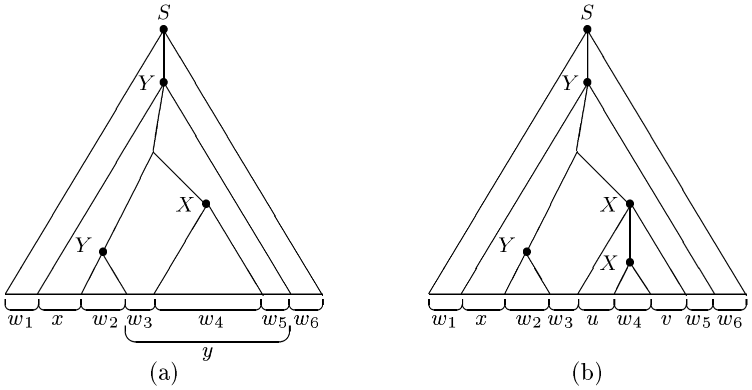


Figure 4.2.4: (a) relative position in Case 2; (b) combined pumpings in Case 2

The situation is shown in Figure 4.2.4 (b) when we pump by both X and Y. We get in this case that

$$w_1 x^n w_2 w_3 u^{m_1} w_4 v^{m_1} w_5 w_3 u^{m_2} w_4 v^{m_2} w_5 \ldots w_3 u^{m_n} w_4 v^{m_n} w_5 w_6 \in L, \quad (4.2.8)$$

for any $n \geq 0$ and $m_i \geq 0, 1 \leq i \leq n$. Consider the particular case $n = 2$, that is

$$w_1 x^2 w_2 w_3 u^n w_4 v^n w_5 w_3 u^m w_4 v^m w_5 w_6 \in L, \tag{4.2.9}$$

for any $m \geq 0, n \geq 0$. Put, for any $m \geq 0$,

$$R_m = \{w_1 x^2 w_2 w_3 u^n w_4 v^n w_5 w_3 u^m w_4 v^m w_5 w_6 \mid n \geq 0\}.$$

Because of (4.2.9), $\bigcup_{m=0}^{\infty} R_m$ is slender and thus, by Lemma 4.2.1,

$$\bigcup_{m=0}^{\infty} R_m = \bigcup_{m=0}^{m_0} R_m,$$

for some $m_0 \geq 0$. Take now m and n such that:

(i) $|v^n| \geq |v| + |u|$ and

(ii) $|u| + |v| - |w_5 w_3| \leq n|v| \leq (m - m_0)|u| - |w_4 w_5 w_3| - 2|u|$.

There are $0 \leq m' \leq m_0, n' \geq 0$ such that

$$w_1 x^2 w_2 w_3 u^n w_4 v^n w_5 w_3 u^m w_4 v^m w_5 w_6 = \\ w_1 x^2 w_2 w_3 u^{n'} w_4 v^{n'} w_5 w_3 u^{m'} w_4 v^{m'} w_5 w_6. \tag{4.2.10}$$

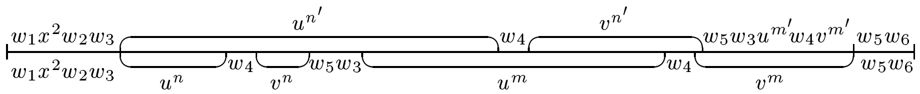
From the two inequalities of (ii), we get, respectively,

$$|u^n w_4 v^n w_5 w_3 u^m| \geq |u^{n'} w_4| + |u| + |v|$$

and

$$|u^{n'}| \geq |u^n w_4 v^n w_5 w_3| + 2|u|.$$

Hence, the equality (4.2.10) looks as in the picture below.



and the overlappings between $u^{n'}$ and v^n, $u^{n'}$ and u^m, and u^m and $v^{n'}$ are long enough to apply again Fine and Wilf's theorem. We get, using also Theorem 2.1.1,

$$\begin{array}{ll} u = z^{r_1}, & z \in \Sigma^+, z \text{ primitive}, r_1 \geq 1, \\ v = (z_1'' z_1')^{r_2}, & z = z_1' z_1'', r_2 \geq 1, \\ w_4 = z^{s_1} z_1', & s_1 \geq 0, \\ w_5 = z_1'' z^{s_2} z_2', & z = z_2' z_2'', s_2 \geq 0, \\ w_3 = z_2'' z^{s_3}, & s_3 \geq 0. \end{array}$$

We now compute the words of (4.2.8) in terms of z:

$$\begin{aligned}w_1 x^n w_2 w_3 u^{m_1} w_4 v^{m_1} w_5 w_3 u^{m_2} w_4 v^{m_2} w_5 \ldots w_3 u^{m_n} w_4 v^{m_n} w_5 w_6 &= \\ w_1 x^n w_2 (z_2'' z_2')^{(s_1+s_2+s_3+2)n} (z_2'' z_2')^{(r_1+r_2)\sum_{i=1}^n m_i} w_6.\end{aligned}$$

Replacing $\sum_{i=1}^n m_i$ by $m \geq 0$, we have that

$$w_1 x^n w_2 (z_2'' z_2')^{(s_1+s_2+s_3+2)n} (z_2'' z_2')^{(r_1+r_2)m} w_6 \in L, \tag{4.2.11}$$

for any $m, n \geq 0$.

Applying Lemma 4.2.1 for

$$R_m = \{w_1 x^n w_2 (z_2'' z_2')^{(s_1+s_2+s_3+2)n} (z_2'' z_2')^{(r_1+r_2)m} w_6 \mid n \geq 0\},$$

and taking m and n such that

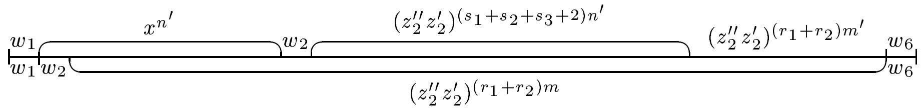
(i) $n = 0$ and

(ii) $(m - m_0)(r_1 + r_2)|z| \geq ((s_1 + s_2 + s_3 + 2)\dfrac{|z|}{|x|} + 1)(|w_2| + |x| + |z|)$

(m_0 comes from Lemma 4.2.1) we find $0 \leq m' \leq m_0, n' \geq 0$ such that

$$w_1 w_2 (z_2'' z_2')^{(r_1+r_2)m} w_6 = w_1 x^{n'} w_2 (z_2'' z_2')^{(s_1+s_2+s_3+2)n'} (z_2'' z_2')^{(r_1+r_2)m'} w_6 \tag{4.2.12}$$

and in (4.2.12) $x^{n'}$ and $(z_2'' z_2')^{(r_1+r_2)m}$ overlap each other long enough to apply Fine and Wilf's theorem. See also the picture below.



We get

$$\begin{aligned} &x = (z_3'' z_3')^{r_3}, \quad z = z_3' z_3'', r_3 \geq 1, \\ &w_2 = z_3'' z^{s_4} z_2', \quad s_4 \geq 0, \end{aligned}$$

and (4.2.11) becomes

$$w_1 z_3'' z^{s_4} z^{(r_3+s_1+s_2+s_3+2)n+(r_1+r_2)m} z_2' w_6 \in L, \tag{4.2.13}$$

for any $n, m \geq 0$. Consequently, we have obtained in this case a result similar to the one in Case 1. (Compare (4.2.7) and (4.2.13).)

Consider now the end of the application of the algorithm. We have then $|w_1 w_2 w_6| \leq p_L$, so

$$|w_1 z_3'' z^{s_4} z_2' w_6| \leq p_L. \tag{4.2.13'}$$

Also, because $|x w_2 w_3 w_4 w_5| \leq p_L$ and $|u w_4 v| \leq p_L$ (by construction) we get

$$|z^{r_3+s_1+s_2+s_3+2}| \leq p_L, \tag{4.2.13''}$$
$$|z^{r_1+r_2}| \leq p_L. \tag{4.2.13'''}$$

Case 3. The relative position in the tree is depicted for this case in Figure 4.2.5 (a).

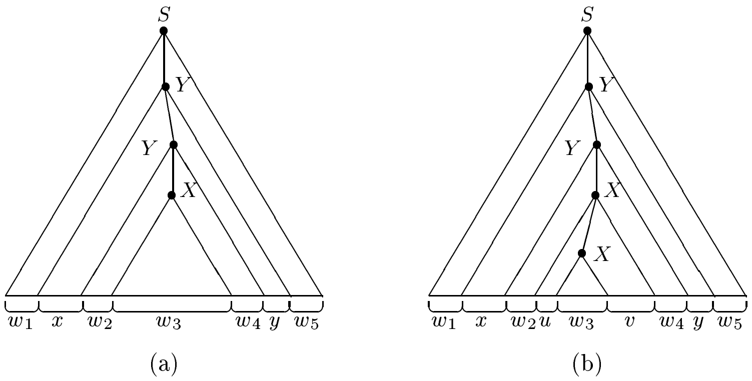


Figure 4.2.5: (a) relative position in Case 3; (b) combined pumpings in Case 3

As seen in Figure 4.2.5 (b), we have in this case

$$w_1 x^m w_2 u^n w_3 v^n w_4 y^m w_5 \in L, \tag{4.2.14}$$

for any $m, n \geq 0$.

At the beginning, the reasoning is similar with the one for Case 1, just that we apply here Lemma 4.2.2 for

$$R_m = \{w_1 x^m w_2 u^n w_3 v^n w_4 y^m w_5 \mid n \geq 0\},$$

obtaining $m_0 \geq 0$ with

$$\bigcup_{m=0}^{\infty} R_m = \bigcup_{m=0}^{m_0} R_m.$$

We distinguish two subcases here, depending on whether or not $\dfrac{|x|}{|y|} = \dfrac{|u|}{|v|}$.

Subcase 3.1. $\dfrac{|x|}{|y|} = \dfrac{|u|}{|v|}$. We show that in this case the pumpings in (4.2.14) collapse to two which pump simultaneously. (In the sense of (4.2.16) below.)

Choose m, n such that

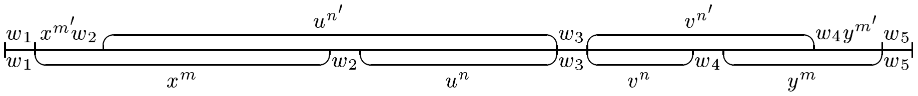
(i) $|x^m| \geq |x^{m_0} w_2| + |x| + |u|$ and

(ii) $|y^m| \geq |w_4 y^{m_0}| + |y| + |v|$.

Thus, as in Case 1, there are $n' \geq 0$ and $0 \leq m' \leq m_0$ such that

$$w_1 x^m w_2 u^n w_3 v^n w_4 y^m w_5 = w_1 x^{m'} w_2 u^{n'} w_3 v^{n'} w_4 y^{m'} w_5. \tag{4.2.15}$$

The equality (4.2.15) is picturally represented below.



As in Case 1, we apply Fine and Wilf's theorem and Theorem 2.1.1 and get

$$\begin{array}{ll} x = z^{r_1}, & z \in \Sigma^+, z \text{ primitive}, r_1 \geq 1, \\ u = (z_1'' z_1')^{r_2}, & z = z_1' z_1'', r_2 \geq 1, \\ w_2 = z^{s_1} z_1', & s_1 \geq 0, \end{array}$$

and

$$\begin{array}{ll} y = t^{p_1}, & t \in \Sigma^+, t \text{ primitive}, p_1 \geq 1, \\ v = (t'' t')^{p_2}, & t = t' t'', p_2 \geq 1, \\ w_4 = t'' t^{q_1}, & q_1 \geq 0. \end{array}$$

We write the words in (4.2.14) as

$$w_1 z^{s_1} z^{r_2 n + r_1 m} z_1' w_3 t'' t^{p_2 n + p_1 m} t^{q_1} w_5 \in L, \tag{4.2.16}$$

for any $n, m \geq 0$. Moreover, from $\dfrac{|x|}{|y|} = \dfrac{|u|}{|v|}$, it follows that whenever (4.2.15) holds for some m, n, m', n', then

$$|x^m w_2 u^n| = |x^{m'} w_2 u^{n'}|$$

and hence we cannot obtain further overlappings. (The two occurrences of the word w_3 in the two sides of (4.2.15) fit exactly; see also the picture.) Therefore, the best we can get is (4.2.16).

When (3)–(5) are applied for the last time, we have $|w_1 w_2 w_3 w_4 w_5| \leq p_L$, thus

$$|w_1 z^{s_1} z_1' w_3 t'' t^{q_1} w_5| \leq p_L. \tag{4.2.16$'$}$$

From $|x w_2 w_3 w_4 y| \leq p_L$ and $|u w_3 v| \leq p_L$ we obtain

$$|z^{r_1} t^{p_1}| \leq p_L, \tag{4.2.16$''$}$$

$$|z^{r_2} t^{p_2}| \leq p_L. \tag{4.2.16$'''$}$$

Subcase 3.2. $\dfrac{|x|}{|y|} \neq \dfrac{|u|}{|v|}$. In this case, the situation changes completely and all pumpings in (4.2.14) collapse again to a single one.

Without loss of generality, we may assume that $\dfrac{|x|}{|y|} < \dfrac{|u|}{|v|}$. The other case is analogous.

Choose now m and n such that

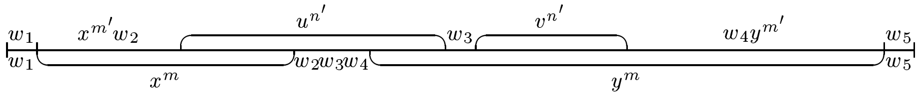
(i) $n = 0$,
(ii) $|x^m| \geq |x^{m_0} w_2| + |x| + |u|$, and
(iii) $(m - m_0)\left(\frac{|xy|}{|uv|}|u| - |x|\right) \geq |w_3 w_4| + |u| + |y|$.

Notice that $\frac{|x|}{|y|} < \frac{|u|}{|v|}$ implies $\frac{|xy|}{|uv|}|u| - |x| > 0$, so there exist m and n fulfilling (i)–(iii) above.

Now, there are $n' \geq 0, 0 \leq m' \leq m_0$ such that

$$w_1 x^m w_2 w_3 w_4 y^m w_5 = w_1 x^{m'} w_2 u^{n'} w_3 v^{n'} w_4 y^{m'} w_5. \tag{4.2.17}$$

The equality (4.2.17) is depicted below.



Following the same idea used so far, we apply Fine and Wilf's theorem and get that z and t are conjugates. More precisely, we have, by Theorem 2.1.1,

$$z = z_2' z_2'', \quad t = z_2'' z_2',$$

and

$$\begin{aligned} y &= (z_2'' z_2')^{p_1}, & p_1 &\geq 1, \\ v &= (z_3'' z_3')^{p_2}, & z &= z_3' z_3'', p_2 \geq 1, \\ w_3 &= z_1'' z^{s_2} z_3', & s_2 &\geq 0, \\ w_4 &= z_3'' z^{s_3} z_2', & s_3 &\geq 0. \end{aligned}$$

Therefore, (4.2.14) becomes

$$w_1 z^{s_1+s_2+s_3+2} z^{(r_2+p_2)n+(r_1+p_1)m} z_2' w_5 \in L, \tag{4.2.18}$$

for any $n, m \geq 0$.

At the end of the application of the algorithm, we have $|w_1 w_2 w_3 w_4 w_5| \leq p_L$ so

$$|w_1 z^{s_1+s_2+s_3+2} z_2' w_5| \leq p_L. \tag{4.2.18$'$}$$

Since $|x w_2 w_3 w_4 y| \leq p_L$ and $|u w_3 v| \leq p_L$, we get also that

$$|z^{r_1+p_1}| \leq p_L, \tag{4.2.18$''$}$$
$$|z^{r_2+p_2}| \leq p_L. \tag{4.2.18$'''$}$$

Case 4. We have now the relative position in Figure 4.2.6 (a) and the pumpings in Figure 4.2.6 (b). Here we have $x = w_2 w_3, y = w_5 w_6$.

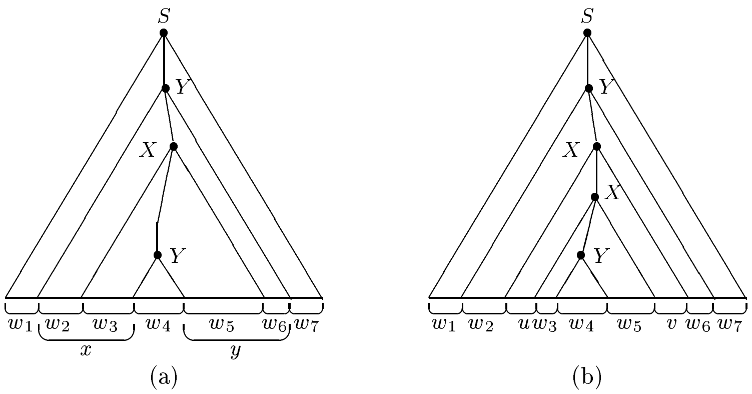


Figure 4.2.6: (a) relative position in Case 4; (b) combined pumpings in Case 4

With the notations in Figure 4.2.6 (b), we have

$$w_1 w_2 u^{n_1} w_3 w_2 u^{n_2} w_3 \dots w_2 u^{n_m} w_3 w_4 w_5 v^{n_m} w_6 \dots w_5 v^{n_2} w_6 w_5 v^{n_1} w_6 w_7 \in L, \tag{4.2.19}$$

for any $m \geq 0$ and $n_i \geq 0, 1 \leq i \leq m$. For $m = 2$ we get

$$w_1 w_2 u^m w_3 w_2 u^n w_3 w_4 w_5 v^n w_6 w_5 v^m w_6 w_7 \in L, \tag{4.2.20}$$

for any $m, n \geq 0$.

If we now compare (4.2.20) with (4.2.14), we easily see that we have a situation similar with Subcase 3.1. Therefore, as in Subcase 3.1, we obtain

$$\begin{array}{ll} u = z_1^r, & z_1 \in \Sigma^+, z_1 \text{ primitive}, r \geq 1, \\ v = t_1^p, & t_1 \in \Sigma^+, t_1 \text{ primitive}, p \geq 1, \\ w_3 = z_1^{s_1} z', & z_1 = z' z'', s_1 \geq 0, \\ w_2 = z'' z_1^{s_2}, & s_2 \geq 0, \\ w_6 = t_1^{q_1} t', & t_1 = t' t'', q_1 \geq 0, \\ w_5 = t'' t_1^{q_2}, & q_2 \geq 0 \end{array}$$

and (4.2.19) becomes

$$w_1 (z''z')^{(s_1+s_2+1)m} (z''z')^{r \sum_{i=1}^m n_i} w_4 (t''t')^{(q_1+q_2+1)m} (t''t')^{p \sum_{i=1}^m n_i} w_7 \in L$$

or, for $z = z''z', t = t''t'$, and $n = \sum_{i=1}^m n_i$,

$$w_1 z^{rn+(s_1+s_2+1)m} w_4 t^{pn+(q_1+q_2+1)m} w_7 \in L \tag{4.2.21}$$

for any $n, m \geq 0$.

At the end, we have

$$|w_1 w_4 w_7| \leq p_L \tag{4.2.21$'$}$$

and, because $|w_2w_3w_4w_5w_6| \leq p_L$ and $|uw_3w_4w_5v| \leq p_L$, we get

$$|z^r t^p| \leq p_L, \tag{4.2.21''}$$
$$|z^{s_1+s_2+1} t^{q_1+q_2+1}| \leq p_L. \tag{4.2.21'''}$$

This is the conclusion of Case 4 and the end of our analysis of two consecutive applications of the steps (3)–(5) in the tree reduction algorithm.

4.2.4 Conclusion

Let us now conclude the reasoning in the previous subsection in a compact form which will be essential for completing the proof.

It is not difficult now to see what happens when we start with an arbitrary loop of L in (4.2.1) together with its derivation tree from Theorem 3.3.3(ii) and apply exhaustively the reduction algorithm. There are the following two possibilities.

(A) When at least one of the Cases 1,2, or 3.2 (or some of their analogues) appear during the entire application of the algorithm, we get (see (4.2.7), (4.2.13), and (4.2.18))

$$\{w_1 z^{\sum_{i=1}^{k} r_i n_i} w_2 \mid n_i \geq 0, 1 \leq i \leq k\} \subseteq L, \tag{4.2.22}$$

for some $w_1, w_2, z \in \Sigma^*$ with $z \neq \lambda, z$ primitive, and some integers $k \geq 1, r_i \geq 1$, for any $1 \leq i \leq k$, such that the following inequalities are satisfied (see (4.2.7′), (4.2.7″), (4.2.7‴), (4.2.13′), (4.2.13″), (4.2.13‴), and (4.2.18′), (4.2.18″), (4.2.18‴)):

$$\begin{aligned} &|w_1w_2| \leq p_L, \\ &|z^{r_i}| \leq p_L, \text{ for any } 1 \leq i \leq k. \end{aligned} \tag{4.2.23}$$

Also, the initial loop is obtained by taking $n_2 = n_3 = \cdots = n_k = 1$ and $n_1 = n$ in (4.2.22), that is,

$$\{w_1 z^{\sum_{i=2}^{k} r_i} (z^{r_1})^n w_2 \mid n \geq 0\}. \tag{4.2.24}$$

The relation (4.2.22) deserves a few explanations. In each of the Cases 1, 2, or 3.2 (or their analogues), no matter what relative position we investigated in the tree, we finally concluded that the pumped words are of the form $z^{r_1}z^{r_2}$, $r_1, r_2 \geq 1$, such that $|z^{r_1}| = |xy|, |z^{r_2}| = |uv|$. This is obvious for Cases 1 and 3.2. In Case 2, we just have to remember that $y = w_3w_4w_5$ and the same result is obtained. Therefore, each nonterminal found by an application of step (4) in Algorithm 4.2.4, say Z, pumps a word of the form $z^r, r \geq 1$, where the length of z^r (given by r, z being fixed) does not depend on the position of the other vertices in discussion in the tree; it depends only on the position of the two vertices labeled Z in the current tree. This is why we have (4.2.22).

(B) When only the Cases 3.1 and 4 (or their analogues) appear during the application of the algorithm, we get (see (4.2.16) and (4.2.21))

$$\{w_1 z^{\sum_{i=1}^{k} r_i n_i} w_2 t^{\sum_{i=1}^{k} p_i n_i} w_3 \mid n_i \geq 0, 1 \leq i \leq k\} \subseteq L, \tag{4.2.25}$$

for some $w_1, w_2, w_3, z, t \in \Sigma^*$ with $z \neq \lambda, t \neq \lambda, z$ and t primitive, and some integers $k \geq 1, r_i \geq 1, p_i \geq 1$, for any $1 \leq i \leq k$, such that the following inequalities are satisfied (see (4.2.16′), (4.2.16″), (4.2.16‴), and (4.2.21′), (4.2.21″), (4.2.21‴)):

$$\begin{aligned} &|w_1 w_2 w_3| \leq p_L, \\ &|z^{r_i} t^{p_i}| \leq p_L, \text{ for any } 1 \leq i \leq k, \end{aligned} \tag{4.2.26}$$

as well as the relation

$$\frac{r_1}{p_1} = \frac{r_2}{p_2} = \cdots = \frac{r_k}{p_k}. \tag{4.2.27}$$

Moreover, the initial loop is again obtained with $n_2 = n_3 = \cdots = n_k = 1$ and $n_1 = n$ in (4.2.25), that is,

$$\{w_1 z^{\sum_{i=2}^{k} r_i} (z^{r_1})^n w_2 (t^{p_1})^n t^{\sum_{i=2}^{k} p_i} w_3 \mid n \geq 0\}. \tag{4.2.28}$$

The relation (4.2.25) deserves a few explanations similar with those we made at (A) for (4.2.22). In each of the Cases 3.1 or 4 (or their analogues), the pairs of pumped words are of the form $(z^{r_1} z^{r_2}, t^{p_1} t^{p_2})$, $r_1, r_2, p_1, p_2 \geq 1$, such that $|z^{r_1}| = |x|, |t^{p_1}| = |y|, |z^{r_2}| = |u|$, and $|t^{p_2}| = |v|$. This is clear in Case 3.1; in Case 4 use the fact that $x = w_2 w_3, y = w_5 w_6$. Hence, each nonterminal found at step (4), say Z, pumps here a pair of words of the form $(z^r, t^p), r, p \geq 1$, where the lengths of z^r and t^p (given by r and p, z and t being fixed) depend only on the position of the two vertices labeled Z in the current tree. Therefore, we have finally (4.2.25).

Consequently, we have started with a loop of L in (4.2.1) and, by the above reasoning, this loop can be written as in (4.2.24) or (4.2.28), at the same time (4.2.22) or (4.2.25), respectively, being fulfilled. We have thus obtained a set bigger than the initial loop (the set in (4.2.22) or (4.2.25)) and still contained in L. Next, we show, using (4.2.23) and, respectively, (4.2.26), (4.2.27), that the new set is a union of loops as required.

4.2.5 Bounded loops

We are now very close to our final goal. We need only the following number-theoretic result for our purpose.

Let us denote, for an integer n, by $\mathcal{M}_n$ the set of all multiples of n.

Lemma 4.2.5. *If $p_1, p_2, \ldots, p_k$ are $k \geq 2$ positive integers and*

$$\gcd(p_1, p_2, \ldots, p_k) = d,$$

then all large enough multiples of d can be expressed as linear combinations of the numbers $p_1, p_2, \ldots, p_k$ with nonnegative integer coefficients. More precisely,

$$\{n \in \mathcal{M}_d \mid n \geq \frac{1}{d}(\max_{1\leq i\leq k} p_i)^2\} \subseteq \{\sum_{i=1}^{k} n_i p_i \mid n_i \geq 0, 1 \leq i \leq k\}.$$

Proof. By induction on $k \geq 2$. Consider first $k = 2$ and p_1, p_2 with $\gcd(p_1, p_2) = d$. Then $\gcd(\frac{p_1}{d}, \frac{p_2}{d}) = 1$ and, clearly, any integer bigger than $\frac{p_1 p_2}{d^2}$ can be obtained as a linear combination of $\frac{p_1}{d}$ and $\frac{p_2}{d}$ with non-negative coefficients. Therefore, by multiplying by d, we get

$$\{n \in \mathcal{M}_d \mid n \geq \frac{p_1 p_2}{d}\} \subseteq \{n_1 p_1 + n_2 p_2 \mid n_1, n_2 \geq 0\}. \tag{4.2.29}$$

Take now $k \geq 3$ and assume the property true for $k - 1$. Put

$$\gcd(p_1, p_2, \ldots, p_{k-1}) = d'.$$

If $d' = d$, then our property follows immediately from the inductive hypothesis, so assume $d' \neq d$. We have $\gcd(d', p_k) = d$ and put $d' = d''d, d'' \geq 2$. By the inductive hypothesis

$$\{n \in \mathcal{M}_{d'} \mid n \geq \frac{1}{d'}(\max_{1\leq i\leq k-1} p_i)^2\} \subseteq \{\sum_{i=1}^{k-1} n_i p_i \mid n_i \geq 0, 1 \leq i \leq k-1\}. \tag{4.2.30}$$

Let us prove the following inclusion

$$\{n \in \mathcal{M}_d \mid n \geq \frac{1}{d'}(\max_{1\leq i\leq k-1} p_i)^2 + \frac{d' p_k}{d}\} \subseteq \{\sum_{i=1}^{k} n_i p_i \mid n_i \geq 0, 1 \leq i \leq k\}. \tag{4.2.31}$$

For consider $n \in \mathcal{M}_d$ such that

$$n \geq \frac{1}{d}(\max_{1\leq i\leq k-1} p_i)^2 + \frac{d' p_k}{d}.$$

Since $n \geq \frac{d' p_k}{d}$ and $n \in \mathcal{M}_d$, it follows from the stronger assertion we proved in case $k = 2$, see (4.2.29), that there are $m_1, m_2 \geq 0$ such that

$$n = m_1 d' + m_2 p_k. \tag{4.2.32}$$

We may assume $m_2 \leq d'' - 1$ since, if this is not the case, we put $m_2 = m_2' d'' + m_2'', 0 \leq m_2'' \leq d'' - 1$, and $p_k = dp_k'$ obtaining

$$n = (m_1 + m_2' p_k') d' + m_2'' p_k.$$

Thus $0 \leq m_2 \leq d'' - 1$, which implies

$$m_1 d' \geq \frac{1}{d'} (\max_{1 \leq i \leq k-1} p_i)^2$$

and so, by (4.2.30),

$$m_1 d' \in \{\sum_{i=1}^{k-1} n_i p_i \mid n_i \geq 0, 1 \leq i \leq k-1\}$$

hence, due to (4.2.32),

$$n \in \{\sum_{i=1}^{k} n_i p_i \mid n_i \geq 0, 1 \leq i \leq k\}$$

and (4.2.31) is proved.

As it is not difficult to show that

$$\frac{1}{d'} (\max_{1 \leq i \leq k-1} p_i)^2 + \frac{d' p_k}{d} \leq \frac{1}{d} (\max_{1 \leq i \leq k} p_i)^2,$$

our statement for k follows from (4.2.31). The induction step is completed and the lemma is proved. ■

Proof of Theorem 4.2.3 (continued). We now apply Lemma 4.2.5 to achieve our final goal. For a positive integer n, denote by $\Sigma^{\leq n}$ the set of words of length at most n over Σ.

Consider first case (A) above and put

$$\gcd(r_1, r_2, \ldots, r_k) = d.$$

By (4.2.22) and Lemma 4.2.5 we get

$$\{w_1 z^{\sum_{i=1}^k r_i n_i} w_2 \mid n_i \geq 0, 1 \leq i \leq k\} = \\ \{w_1 z^{\frac{1}{d}(\max_{1 \leq i \leq k} r_i)^2} (z^d)^n w_2 \mid n \geq 0\} \cup F \subseteq L$$

where, because of (4.2.23),

$$|z^d| \leq p_L,$$
$$|w_1 z^{\frac{1}{d}(\max_{1 \leq i \leq k} r_i)^2} w_2| \leq p_L^2 + p_L, \text{ and}$$
$$F \subseteq \{w_1 (z^d)^n w_2 \mid 0 \leq n < \tfrac{1}{d^2} (\max_{1 \leq i \leq k} r_i)^2\} \subseteq \Sigma^{\leq p_L^2 + p_L}.$$

Consequently, we may replace the initial loop (4.2.24) by several which verify the conditions in the statement.

For case (B), put

$$\begin{aligned}\gcd(r_1, r_2, \ldots, r_k) &= d_1,\\ \gcd(p_1, p_2, \ldots, p_k) &= d_2.\end{aligned}$$

From (4.2.27) we obtain

$$\frac{r_1}{p_1} = \frac{r_2}{p_2} = \cdots = \frac{r_k}{p_k} = \frac{d_1}{d_2}$$

and so, by (4.2.25) and Lemma 4.2.5 we get

$$\begin{aligned}&\{w_1 z^{\sum_{i=1}^{k} r_i n_i} w_2 t^{\sum_{i=1}^{k} p_i n_i} w_3 \mid n_i \geq 0, 1 \leq i \leq k\} =\\ &\{w_1 z^{\frac{1}{d_1}(\max_{1\leq i\leq k} r_i)^2} (z^{d_1})^n w_2 (t^{d_2})^n t^{\frac{1}{d_2}(\max_{1\leq i\leq k} p_i)^2} w_3 \mid n \geq 0\} \cup F \subseteq L\end{aligned}$$

where, because of (4.2.26),

$$\begin{aligned}&|z^{d_1} t^{d_2}| \leq p_L,\\ &|w_1 z^{\frac{1}{d_1}(\max_{1\leq i\leq k} r_i)^2} w_2 t^{\frac{1}{d_2}(\max_{1\leq i\leq k} p_i)^2} w_3| \leq p_L^2 + p_L, \text{ and}\\ &F \subseteq \{w_1 (z^{d_1})^n w_2 (t^{d_2})^n w_3 \mid 0 \leq n < \tfrac{1}{d_1^2}(\max_{1\leq i\leq k} r_i)^2\} \subseteq \Sigma^{\leq p_L^2 + p_L}.\end{aligned}$$

Therefore, as in case (A), we may replace the initial loop (4.2.28) by several which verify the conditions in the statement. Our theorem is proved. ■

Some open problems related with the statement of Theorem 4.2.3 are stated at the end of the chapter.

4.3 The decidability theorem

We give in this section two proofs for the decidability of the slenderness problem for context-free languages. Both rely essentially on the bounds proved by Theorem 4.2.3.

4.3.1 A simple proof

For the first proof we shall need a result from [GiSp]; cf. also [Gi].

A language is called **bounded** if there exists words $w_1, w_2, \ldots, w_n \in \Sigma^*$ such that

$$L \subseteq w_1^* w_2^* \cdots w_n^*.$$

Theorem 4.3.1 (Ginsburg and Spanier [GiSp]). *For two arbitrary context-free languages L_1 and L_2 one of which is bounded, it is decidable whether or not:*
(i) $L_1 \subseteq L_2$;
(ii) $L_1 = L_2$.

We are now ready to prove the announced result.

Theorem 4.3.2. *It is decidable whether or not an arbitrary context-free language is slender.*

First proof. Given a context-free language L, we know, by Theorem 4.2.3, that L is slender if and only if it can be decomposed as a union of paired loops with the conditions (i) and (ii) in the statement of Theorem 4.2.3. Therefore, there are finitely many possibilities for the loops of L. Using Theorem 4.3.1, we may check through all these loops, find which ones are included in L, and then check whether the whole L is obtained. ∎

4.3.2 A direct algorithm

The second proof is more complicated than the first one but it has the advantage that avoids the theory of bounded languages, relying on Theorem 4.2.3 only.

Second proof. Let $L \subseteq \Sigma^*$ be a context-free language. Remember that we denote by $\Sigma^{\leq n}$ the set of words of length at most n over Σ. The following algorithm checks whether or not L is slender. The correctness proof is presented at the same time.

Algorithm 4.3.3.
(1) Find a context-free grammar in Chomsky normal form $G = (N, \Sigma, S, P)$ generating L and set $p_L = 2^{\mathrm{card}(N)-1}$.

(2) Check whether or not

$$L \subseteq \bigcup_{\substack{uwy \in \Sigma^{\leq p_L^2 + p_L} \\ vx \in \Sigma^{\leq p_L}}} uv^*wx^*y$$

and, if not, then answer **no**. Notice that this is decidable and the answer is correct by Theorem 4.2.3.

(3) For each $u, v, w \in \Sigma^*$ such that $|uw| \leq p_L^2 + p_L, |v| \leq p_L$, construct the language

$$L_{u,v,w} = L \cap uv^*w.$$

Claim 1. $L_{u,v,w}$ is effectively regular.

Proof of Claim 1. $L_{u,v,w}$ is effectively context-free and so are the languages $L^{(1)}_{u,v,w} = u^{-1}L_{u,v,w}, L^{(2)}_{u,v,w} = L^{(1)}_{u,v,w}w^{-1}$, and $L^{(3)}_{u,v,w} = g_v(L^{(2)}_{u,v,w})$, where g_v is a gsm which replaces each occurrence of v by the letter $\# \notin \Sigma$. Thus $L^{(3)}_{u,v,w} \subseteq \#^*$ so $L^{(3)}_{u,v,w}$ is regular. Moreover, by Parikh's theorem, it is effectively regular. Performing the above operations in the reverse order, we get that $L_{u,v,w}$ is effectively regular. ■

(4) Construct the language

$$L' = L - \bigcup_{\substack{uw \in \Sigma^{\leq p_L^2 + p_L} \\ v \in \Sigma^{\leq p_L}}} L_{u,v,w}.$$

This construction is effective by Claim 1. Notice that L' does not contain any single loop as in the representation in Theorem 4.2.3. Also, L is slender if and only if L' is slender.

(5) For each $u, v, w, x, y \in \Sigma^*$ with $|uwy| \leq p_L^2 + p_L, |vx| \leq p_L$, construct the language

$$L_1 = L' \cap uv^*wx^*y.$$

Clearly, L' is slender if and only if all such languages L_1 are slender. If L_1 is finite or empty, then it is slender, so suppose that L_1 is infinite.

(6) Construct the languages $L_2 = u^{-1}L_1, L_3 = L_2y^{-1}$, and $L_4 = g(L_3)$, where g is a gsm which replaces each occurrence of v by the word $\#_1^{|v|}$, erases w, and replaces each occurrence of x by the word $\#_2^{|x|}$, for two new letters $\#_1, \#_2 \notin \Sigma$. It is clear that, since v, w, x are given, such a gsm can be constructed. Also, as proved in Claim 2 below, L_1 and L_4 are simultaneously slender.

Claim 2. L_1 is slender if and only if L_4 is slender.

Proof of Claim 2. Suppose first that L_4 is slender and consider an arbitrary $n \geq 0$. We have that

$$\operatorname{card}(\{z \in L_1 \mid |z| = n\}) \leq \operatorname{card}(\{z \in L_4 \mid |z| = n - |uwy|\}),$$

therefore L_1 is slender.

Conversely, suppose that L_1 is slender and take $n \geq 0$. All words of length n in L_4 come from words of length $n + |uwy|$ in L_1. We show that any word of L_1 cannot produce more than $\left\lceil \frac{|vwx|}{|v|} \right\rceil$ different words of L_4. Clearly, this implies the slenderness of L_4.

Suppose that there is $z \in L_1$ which produces at least $\left\lceil \frac{|vwx|}{|v|} \right\rceil + 1$ different words of L_4. It follows that there exist two decompositions of z, $z = uv^n wx^m y$ and $z = uv^r wx^s y$ such that $r - n \geq \left\lceil \frac{|vwx|}{|v|} \right\rceil + 1$. Therefore, the words v^n and x^m have a common factor of length at least $|v| + |x|$. By Fine and Wilf's theorem, v and x are powers of conjugates of the same word $t \in \Sigma^*$, say $x = t^{r_1}, v = (t_2 t_1)^{r_2}$, where $t = t_1 t_2, r_1 \geq 1, r_2 \geq 1$. Thus $w = t_2 t^{r_3}$, for some $r_3 \geq 0$. It follows that $uv^n wx^m y = ut_2 t^{r_2 n + r_1 m + r_3} y$, hence $uv^* wx^* y \subseteq ut_2 t^* y$. But, since $|ut_2 y| \leq p_L^2 + p_L$ and $|t| \leq p_L$, we have, from step 4, $L' \cap ut_2 t^* y = \emptyset$ and so $L_1 = \emptyset$, a contradiction. The claim is proved. ■

(7) By Parikh's theorem (Theorem 2.2.4) the Parikh set of L_4 is effectively semilinear and let it be

$$\Psi(L_4) = \bigcup_{i=1}^{r} P_i,$$

where P_i is a linear set, for any $1 \leq i \leq r$. Put also

$$L_4 = \bigcup_{i=1}^{r} R_i,$$

where R_i corresponds to P_i, $1 \leq i \leq r$. Each R_i is effectively context-free and L_4 is slender if and only if all $R_i, 1 \leq i \leq r$, are slender.

(8) For each $1 \leq i \leq r, R_i$ has the form (which is computable)

$$R_i = \{\#_1^{c_0 + \sum_{j=1}^{q} c_j n_j} \#_2^{d_0 + \sum_{j=1}^{q} d_j n_j} \mid n_j \geq 0, 1 \leq j \leq q\}, \tag{4.3.33}$$

for some $q \geq 0$ and non-negative constant integers $c_j, d_j, 0 \leq j \leq q$. We may suppose that $c_j + d_j > 0$, for any $1 \leq j \leq q$.

Claim 3. R_i is slender if and only if either $q \leq 1$ or else $\dfrac{c_1}{d_1} = \dfrac{c_2}{d_2} = \cdots = \dfrac{c_q}{d_q}$.

Proof of Claim 3. If $q = 0$, then R_i is finite and hence slender. If $q = 1$, then consider a positive integer k. In order to find the number of words of length k in R_i, we have to solve the system

$$\begin{cases} c_0 + c_1 n_1 = n \\ d_0 + d_1 n_1 = m \\ n + m = k \end{cases}$$

in the unknowns n_1, n, and m. Since the determinant of the matrix of the system is $c_1 + d_1 > 0$, the system has a unique solution, so R_i is 1-slender.

Suppose now that $q \geq 2$, $c_j = ld_j$, for $l > 0$, $1 \leq j \leq q$. (Notice that the case when either all c_i's are 0 or all d_i's are 0 is clear.) As above, consider the following system of equations in $n_1, n_2, \ldots, n_q, n, m$:

$$\begin{cases} c_0 + \sum_{j=1}^{q} c_j n_j = n \\ d_0 + \sum_{j=1}^{q} d_j n_j = m \\ n + m = k \end{cases}$$

It implies

$$\begin{cases} n - lm = c_0 - ld_0 \\ n + m = k \end{cases}$$

so n and m are uniquely determined. Consequently, R_i is 1-slender.

In order to prove the converse implication, we argue by contradiction. Suppose that R_i is slender, $q \geq 2$, and, without loss of generality, $\frac{c_1}{d_1} \neq \frac{c_2}{d_2}$. Consider an arbitrary fixed positive integer k and the set

$$S = \{(r(c_2 + d_2), (k - r + 1)(c_1 + d_1)) \mid 1 \leq r \leq k\}.$$

For any $(n_1, n_2) \in S$, we have

$$c_0 + c_1n_1 + c_2n_2 + d_0 + d_1n_1 + d_2n_2 = c_0 + d_0 + (k+1)(c_1 + d_1)(c_2 + d_2),$$

which is constant with respect to n_1 and n_2. Also, for any two different pairs $(n_1, n_2), (m_1, m_2) \in S$, we have that $c_0 + c_1n_1 + c_2n_2 = c_0 + c_1m_1 + c_2m_2$ if and only if $\frac{c_1}{d_1} = \frac{c_2}{d_2}$. Therefore, the words $\#_1^{n_1}\#_2^{n_2}$, $(n_1, n_2) \in S$, are all in R_i, different, and of the same length. Since k has been arbitrarily chosen, it follows that R_i is not slender, a contradiction. The claim is proved. ■

(9) Since the conditions equivalent with the slenderness of R_i in Claim 3 are trivially checkable, it follows that it is decidable whether or not R_i is slender.

If all such languages R_i are slender, then answer **yes**, otherwise answer **no**. The proof of the theorem is concluded. ■

4.4 Effective constructions

We show in this section that a given slender context-free language L can be effectively written as a union of paired loops, even as a disjoint union of paired loops, and that the smallest k such that L is k-slender is computable.

Theorem 4.4.1. *An arbitrarily given slender context-free language can be effectively written as a union of paired loops.*

Proof. Notice first that the theorem follows by the decidability result of [GiSp] (see Theorem 4.3.1) and Theorem 4.2.3, but we give again a proof which avoids the theory of bounded languages and argue directly, based on the second proof of Theorem 4.3.2. We use the same notations. To conclude the statement, it is enough to prove the following two facts:

(i) the language R_i at step 8 in Algorithm 4.3.3 can be effectively written as a union of paired loops, once we know it is slender, and

(ii) having the language L_4 at step 6 given as a union of paired loops, the language L_3 can be effectively written as a union of paired loops.

Let us prove (i). The case when either all c_i's are 0 or all d_i's are 0 is trivial. Suppose this is not the case and put $\frac{c}{d}$ for $\frac{c_1}{d_1} = \frac{c_2}{d_2} = \cdots = \frac{c_q}{d_q}$ in lowest terms. Then $c_j = ce_j, d_j = de_j, 1 \leq j \leq q$, and (4.3.33) becomes

$$R_i = \{\#_1^{c_0}(\#_1^c)^{\sum_{j=1}^q e_j n_j}(\#_2^d)^{\sum_{j=1}^q e_j n_j}\#_2^{d_0} \mid n_j \geq 0, 1 \leq j \leq q\}.$$

Now, by Lemma 4.2.5, any integer which is a multiple of

$$e = \gcd(e_1, e_2, \ldots, e_q)$$

and is bigger than or equal to

$$E = \frac{1}{e}(\max_{1 \leq i \leq q} e_i)^2$$

can be expressed as a linear combination of $e_1, e_2, \ldots, e_q$ with non-negative integer coefficients. Therefore

$$R_i = (R_i \cap \Sigma^{\leq c_0+d_0+E(c+d)-1}) \cup \{\#_1^{c_0+cE}(\#_1^{ce})^n(\#_2^{de})^n\#_2^{d_0+dE} \mid n \geq 0\}, \tag{4.4.34}$$

and (i) is proved.

For (ii), we consider a gsm h which restores the modifications of g, that is, it replaces each word $\#_1^{|v|}$ by v, introduces a w between the last $\#_1$ and the first $\#_2$, and replaces each word $\#_2^{|x|}$ by x. Due to the form of R_i, see (4.4.34), (ii) is proved if h works as intended, that is, $h(L_4) = L_3$.

We show inclusion in both directions. Notice first that both members of the equality to be proved are included in v^*wx^*.

Take first $v^n w x^m \in L_3$. Then $\#_1^{n|v|}\#_2^{m|x|} \in g(L_3) = L_4$ and it is mapped by h into $v^n w x^m$. Thus $v^n w x^m \in h(L_4)$.

Conversely, consider $v^n w x^m \in h(L_4)$. Then, for some $n_1, m_1, v^n w x^m = v^{n_1} w x^{m_1}$ and $\#_1^{n_1|v|}\#_2^{m_1|x|} \in L_4 = g(L_3)$. Hence $v^{n_1} w x^{m_1} \in L_3$. The equality is proved. ∎

We would like to add that, very recently, using our Theorem 4.4.1 in connection with the result of [BeBo], [Do] gave a linear-time algorithm for computing the lexicographically minimal word of a given length in a context-free language.

Having the effective representation of the slender context-free languages in the previous theorem, we can give now the result that strengthens Corollary 3.4.4. It is an immediate consequence of Theorem 4.4.1 and Lemma 3.4.3. Notice the effectiveness of the constructions in Lemma 3.4.3.

Corollary 4.4.2. *The family of slender context-free languages is effectively closed under difference and intersection.*

From Corollary 4.4.2 and the fact that the emptiness problem is decidable for context-free languages, we get the following result. Notice that it follows also from the result of Ginsburg and Spanier in Theorem 4.3.1 but, in this way, we give a proof which avoids the theory of bounded languages.

Theorem 4.4.3. *The inclusion and the equivalence problems are decidable for slender context-free languages.*

Since the family of context-free languages is effectively closed under taking prefixes or factors, we get immediately the following corollary of Theorem 4.4.3.

Corollary 4.4.4. *For an arbitrary slender context-free language L, it is decidable whether or not*
(i) $L = Pref(L)$;
(ii) $L = Fact(L)$.

Notice further that all problems in Theorem 4.4.3 and Corollary 4.4.4 are undecidable for arbitrary context-free languages.

Our next goal is to sharpen the result in Theorem 4.4.1 showing that a given slender context-free language can be effectively written as a disjoint union of paired loops.

We mention that it was proved in [PaSa1] that any union of paired loops is a disjoint union of paired loops, but no effective procedure to construct a disjoint union was given.

Theorem 4.4.5. *An arbitrarily given slender context-free language can be effectively written as a disjoint union of paired loops.*

Proof. Consider a slender context-free language $L \subseteq \Sigma^*$. By Theorem 4.4.1, L can be effectively written as a union of paired loops, say

$$L = \bigcup_{i=1}^{m} \{u_i v_i^n w_i x_i^n y_i \mid n \geq 0\}.$$

We now proceed by induction on $m \geq 0$. If $m \leq 1$, then L is already in the required form. If $m = 2$, we apply Lemma 3.4.3.

Assume $m \geq 3$ and the property true for $m - 1$. Let us prove it for m. By the inductive hypothesis, we can effectively write L as

$$L = \bigcup_{i=1}^{k} \{u_1' v_i'^n w_i' x_i'^n y_i' \mid n \geq 0\} \cup \{u_m v_m^n w_m x_m^n y_m \mid n \geq 0\},$$

where the first part of the union is a disjoint union of paired loops. Next we just apply Lemma 3.4.3 for any pair of loops $\{u_1' v_i'^n w_i' x_i'^n y_i' \mid n \geq 0\}$ and $\{u_m v_m^n w_m x_m^n y_m \mid n \geq 0\}$, for $1 \leq i \leq k$. ∎

Using Theorem 4.4.5, we can show now that, for a given slender context-free language L, the maximal number of words of the same length in L is computable. Clearly, this is the smallest k such that L is k-slender, that is

$$\max_{n \geq 0}(\mathrm{card}(\{w \in L \mid |w| = n\})) = \min(\{k \mid L \text{ is } k\text{-slender}\}).$$

Theorem 4.4.6. *For a given slender context-free language L, the smallest k such that L is k-slender is computable. Equivalently, the maximal number of words of the same length in L is computable.*

Proof. By Theorem 4.4.5, we can write L as a disjoint union of paired loops, say

$$L = \bigcup_{i=1}^{m} \{u_i v_i^n w_i x_i^n y_i \mid n \geq 0\}.$$

In order to determine whether or not two paired loops $\{u_i v_i^n w_i x_i^n y_i \mid n \geq 0\}$ and $\{u_j v_j^m w_j x_j^m y_j \mid m \geq 0\}$ have words of the same length, we have to solve the linear diophantine equation in two variables

$$|u_i w_i y_i| + n|v_i x_i| = |u_j w_j y_j| + m|v_j x_j|, \tag{4.4.35}$$

Moreover, we can find all solutions of (4.4.35) and so all lengths common to the considered paired loops (see, for instance, [Ros]). This extends to any subset of the set of paired loops composing L.

Since the paired loops in the representation of L are pairwise disjoint, any two words of the same length in two different paired loops are different. Therefore, the maximal number of words of the same length in L, hence the minimal k such that L is k-slender, can be computed. ■

Corollary 4.4.7. *For a given $k \geq 0$, it is decidable whether or not an arbitrary context-free language is k-slender.*

4.5 Undecidability

The result in Theorem 4.3.2 is no longer true when we consider complements or intersections of context-free languages; in both cases we have undecidable problems. This fact was proved in [PaSa1] for linear languages. For the context-free case, we give here a much simpler argument which works for both complement and intersection.

Theorem 4.5.1. *It is undecidable whether or not the complement of an arbitrary context-free language is slender.*

Proof. Consider an arbitrary instance of the Post Correspondence Problem,

$$\{(x_1, x_2, \dots, x_n), (y_1, y_2, \dots, y_n)\}, \tag{4.5.36}$$

$n \geq 1, x_i, y_i \in \{a, b\}^+, 1 \leq i \leq n$, and two new letters $c, \# \notin \{a, b\}$. Denote $x_{n+1} = \# = y_{n+1}$ and construct the well-known languages (see, for instance, [Gi])

$$\begin{aligned}
&L_x = \{ba^{i_1}ba^{i_2}\dots ba^{i_k}cx_{i_k}x_{i_{k-1}}\dots x_1 \,|\, k \geq 0, 1 \leq i_j \leq n+1, 1 \leq j \leq k\},\\
&L_y = \{ba^{i_1}ba^{i_2}\dots ba^{i_k}cy_{i_k}y_{i_{k-1}}\dots y_1 \,|\, k \geq 0, 1 \leq i_j \leq n+1, 1 \leq j \leq k\},\\
&L_{x,y} = L_x c L_y^R,\\
&L_{mi} = \{w_1 c w_2 c w_2^R c w_1^R \mid w_1, w_2 \in \{a, b, \#\}^*\},\\
&L_{PCP} = L_{mi} \cap L_{x,y},
\end{aligned}$$

It is known that the complement $\overline{L_{PCP}}$ of L_{PCP} is context-free.

L_{PCP} is the set of all solutions of the following instance of the Post Correspondence Problem

$$\{(x_1, x_2, \dots, x_{n+1}), (y_1, y_2, \dots, y_{n+1})\}. \tag{4.5.37}$$

Since $x_{n+1} = y_{n+1} = \#$, all words in $\#^*$ are solutions of (4.5.37). Moreover, since $\# \notin \{a, b\}$, all solutions of (4.5.37) are catenations of solutions of (4.5.36) and words in $\#^*$. It follows that L_{PCP} is slender if and only if the set of solutions of (4.5.37) is $\#^*$, that is, if and only if (4.5.36) has no solution, which is undecidable. ■

Theorem 4.5.2. *It is undecidable whether or not the intersection of two arbitrary context-free languages is slender.*

Proof. In the notations of the preceding proof, L_{mi} and $L_{x,y}$ are context-free and $L_{mi} \cap L_{x,y}$ is slender if and only if (4.5.36) has no solution. ■

4.6 Further research

We finally present several problems which have not been investigated here but which are related to our considerations and may be of interest for future research in this area.

The first two are connected with the statement of Theorem 4.2.3.

Problem 4.6.1. Are the bounds given by Theorem 4.2.3 optimal?

A short discussion is required here. We worked throughout the paper with a constant p_L depending on L only, whereas it was in fact some $p_{L,G}$, for G a context-free grammar in Chomsky normal form generating L. Since G has been considered to be fixed, this was correct. However, by changing the grammar G, $p_{L,G}$ might decrease; denote its least value by

$$P_L = \min\{p_{L,G} \mid G \text{ a grammar in Chomsky normal form for } L\}.$$

Therefore, Problem 4.6.1 may be understood in two ways: (i) in terms of $p_L = p_{L,G}$, for some fixed G (as assumed in Theorem 4.2.3); (ii) the least bounds irrespective to the grammar. A positive solution to the following problem might connect these two variants.

Problem 4.6.2. Is P_L computable for an arbitrary slender context-free language L?

Let us further remark that the answer to Problem 4.6.2 is negative for arbitrary context-free languages. To see this, take L context-free and compute the minimal alphabet Σ such that $L \subseteq \Sigma^*$. Then $L = \Sigma^+$ if and only if L is infinite and $P_L = 1$. (We assumed $\lambda \notin L$.)

Indeed, with only one nonterminal, the productions are of the form

$$\begin{aligned} &S \longrightarrow SS \text{ or} \\ &S \longrightarrow a, a \in \Sigma. \end{aligned}$$

In this way, the whole Σ^+ can be obtained, that is, $P_{\Sigma^+} = 1$. Conversely, if L is infinite and $P_L = 1$, then the production $S \longrightarrow SS$ as well as some productions of the form $S \longrightarrow a, a \in \Sigma$, must be present. Hence $L = \Sigma_1^+$, for some $\Sigma_1 \subseteq \Sigma$. Since Σ is minimal, $\Sigma_1 = \Sigma$ and so $L = \Sigma^+$.

Since it is undecidable whether or not $L = \Sigma^+$, it is also undecidable whether or not $P_L = 1$. In particular, it follows that P_L is not computable.

Another problem concerns the complexity of the algorithms for deciding the slenderness of context-free languages.

Problem 4.6.3. Compare the complexity of the two algorithms in the proof of Theorem 4.3.2; compare also these with the complexity of other algorithms for the same problem.

Other problems concern the number of loops.

Problem 4.6.4. For an arbitrary slender context-free language L, is the minimal number of loops in the representation of L as a union of paired loops computable?

Problem 4.6.5. The same problem for representations as disjoint unions of paired loops.

Chapter 5

Languages of infinite words

We consider in this chapter languages obtained as prefixes or factors of infinite words and solve several open problems in this area. First, we prove that it is decidable whether or not a context-free language can be written as the set of all prefixes of an infinite word as well as whether or not a regular language can be written as the set of all factors of an infinite word. Second, we prove the existence of infinite words α such that the set of the factors appearing infinitely often in α is (i) linear non-regular, (ii) context-free non-linear.

5.1 Finite words from infinite words

There are several classical ways to associate a set of finite words to an infinite word α. One can take the set of all finite prefixes or finite factors of α, $Pref(\alpha)$ or $Fact(\alpha)$, respectively, [MaPa1], [MaPa2], or the set of all finite words which are not prefixes of α, $Copref(\alpha)$, [AFG], [AuGa], [Be]. Conversely, to a language of finite words, one can associate infinite words considering the notions of limit [Ei] or adherence [BoNi].

This chapter is devoted to the study of languages obtained from infinite words by taking the set of all finite prefixes or finite factors, respectively, as well as by restricting attention to those factors that appear infinitely often.

In both cases, we are interested in the type of the obtained language with respect to the Chomsky hierarchy. In the case when all prefixes or factors are taken we consider the following two decidability problems, which are mentioned as open in [MaPa1] and [MaPa2]:

1. Is it decidable whether or not a context-free language can be written as the set of all finite prefixes of an infinite word?

2. Is it decidable whether or not a regular language can be written as the set of all finite factors of an infinite word?

We give a positive answer to each of these problems. Also, we prove that the first problem is undecidable for any extension of the factor partial order instead of the prefix partial order.

Then, we consider languages obtained by restricting attention to those factors which appear infinitely often. This cannot be regarded as a classical way of going from infinite words to languages but, nevertheless, some natural and interesting problems arise here. One of them, left open in [MaPa1] and [MaPa2], asks whether there exist infinite words α such that the set of factors which appear infinitely many times in α is context-free but not regular. We shall answer this problem also in the affirmative and even in a stronger form.

We mention that [Bea1], [Bea2], and [BeaN] contain results related with our Sections 5.3 and 5.4.

5.2 $\mathcal{F}$-families

We define in this section several notations which will be often used in the last three chapters of the thesis. They have not been defined in Chapter 2 because they are introduced in this thesis, so are not well known.

For a quasi order $\leq$ on Σ^*, the **down-set**[1] of a language $L \subseteq \Sigma^*$ with respect to $\leq$, denoted $\textsc{down}_{\leq}(L)$, is the set

$$\textsc{down}_{\leq}(L) = \{w \in \Sigma^* \mid w \leq u, \text{ for some } u \in L\}.$$

Thus, for instance, for a word w, we have

$$\textsc{down}_{\leq_p}(w) = Pref(w), \qquad \textsc{down}_{\leq_f}(w) = Fact(w).$$

Using this notation, we define the following families of languages, so called **$\mathcal{F}$-families**, which actually describe some of the classical ways of obtaining languages of finite words from infinite words:

$$\begin{aligned}
\mathcal{F}^r_{\leq} &= \{L \subseteq \Sigma^* \mid L = \textsc{down}_{\leq}(Pref(\alpha)), \text{ for some } \alpha \in \Sigma^\omega\},\\
\mathcal{F}^l_{\leq} &= \{L \subseteq \Sigma^* \mid L = \textsc{down}_{\leq}(Suf(\alpha)), \text{ for some } \alpha \in {}^\omega\Sigma\},\\
\mathcal{F}^{bi}_{\leq} &= \{L \subseteq \Sigma^* \mid L = \textsc{down}_{\leq}(Fact(\alpha)), \text{ for some } \alpha \in {}^\omega\Sigma^\omega\},\\
\mathcal{F}_{\leq} &= \mathcal{F}^r_{\leq} \cup \mathcal{F}^l_{\leq} \cup \mathcal{F}^{bi}_{\leq}.
\end{aligned}$$

Thus, $\mathcal{F}^r_{\leq_p}$ is the family of languages obtained as prefixes of right-infinite words, $\mathcal{F}^l_{\leq_p} = \{\emptyset\}$, $\mathcal{F}_{\leq_f}$ contains all languages which are factors of some infinite words, etc.

[1]We use this term and its notation for uniformity; in [EHR], $\textsc{down}_{\leq^{-1}}(L)$ is called "upward closure" of L with respect to $\leq$; in [vL], $\textsc{down}_{\leq_s^{-1}}(L)$ is called the "ideal" generated by L and $\textsc{down}_{\leq_s}(L)$ is called the "co-ideal" generated by L; in [ChKa], $\textsc{down}_{\leq_s}(L)$ and $\textsc{down}_{\leq_s^{-1}}(L)$ are denoted $SW(L)$ and $SW_1(L)$, respectively.

Another notation which we now introduce is not a classical one. For an infinite word $\alpha \in \Sigma^\omega \cup {}^\omega\Sigma \cup {}^\omega\Sigma^\omega$, the set of the factors appearing infinitely often in α is denoted by $Fact_\infty(\alpha)$. It will be used in the last section of the chapter.

Examples

1. $\text{DOWN}_{\leq_p}(a(bc)^*) = \{\lambda\} \cup a(bc)^* \cup ab(cb)^*$.
2. $\text{DOWN}_{\leq_f}(a(bc)^*) = \{\lambda, a\}(bc)^*\{\lambda, b\} \cup (cb)^*\{\lambda, c\}$.
3. $\text{DOWN}_{\leq_s}(a(bc)^*) = a\{b, c\}^*$.
4. $Fact_\infty(a(bc)^\omega) = \{\lambda, b\}(cb)^*\{\lambda, c\}$.

5.3 Prefixes

It is proved in [MaPa1] that it is decidable whether or not a regular language L is the set of prefixes of a right-infinite word. The case when L is context-free is left open. In this section, we give a positive answer to this problem; we give three different proofs for this result.

We start with a lemma which characterizes the context-free languages in the family $\mathcal{F}^r_{\leq_p}$ and will be usefull for our decidability result. (Remember that the notation p_L has been defined in Chapter 3, page 34.)

Lemma 5.3.1. *For a language $L \subseteq \Sigma^*$, the following two assertions are equivalent:*

(i) L is context-free and $L \in \mathcal{F}^r_{\leq_p}$;

(ii) $L = Pref(uv^\omega)$, for some $u, v \in \Sigma^*, |u| \leq p_L, 1 \leq |v| \leq p_L + 1$.

Proof. Assume (i). Because $L \in \mathcal{F}^r_{\leq_p}$, there is a word $z \in L$ such that $|z| = p_L + 1$. From the pumping lemma for context-free languages, z can be written as $z = x_1x_2x_3x_4x_5$ with $x_2x_4 \neq \lambda$, $|x_2x_3x_4| \leq p_L$, and $x_1x_2^nx_3x_4^nx_5 \in L$, for any $n \geq 0$. Put now

$$(u, v) = \begin{cases} (x_1, x_2), & \text{if } x_2 \neq \lambda, \\ (x_1x_3, x_4), & \text{otherwise.} \end{cases}$$

It is not difficult to see that, in either case,

$$L = Pref(uv^\omega).$$

Notice further that, in both cases, $|u| \leq p_L$, $|v| \leq p_L + 1$ and (ii) is fulfilled.

Assume next (ii). Then, obviously, $L \in \mathcal{F}^r_{\leq_p}$ and L is even regular. The lemma is proved. ■

Our next lemma gives a characterization of the languages in the family $\mathcal{F}^r_{\leq_p}$. It is more general than Lemma 5.3.1 in the sense that it holds for arbitrary languages.

Lemma 5.3.2. *For a language $L \subseteq \Sigma^*$, the following two assertions are equivalent:*

(i) $L \in \mathcal{F}^r_{\leq_p}$;

(ii) L is infinite, $L = Pref(L)$, and L is 1-slender.

Proof. If $L \in \mathcal{F}^r_{\leq_p}$, then all conditions at (ii) are obviously fulfilled. Conversely, assume (ii) holds. Then, for any $n \geq 0$, we have that

$$\mathrm{card}\{w \in L \mid |w| = n\} = 1,$$

and put

$$L = \{w_0 = \lambda, w_1, w_2, w_3, \dots\}$$

where $|w_n| = n$, for any $n \geq 0$. It follows that

$$w_0 \leq_p w_1 \leq_p w_2 \leq_p \cdots .$$

Define the right-infinite word $\alpha \in \Sigma^\omega$ by

$$\alpha_i = w_i^{-1} w_{i+1}, \text{ for all } i \geq 0.$$

Then, clearly,

$$L = Pref(\alpha),$$

so $L \in \mathcal{F}^r_{\leq_p}$. The proof is concluded. ∎

We are now ready to prove the announced decidability result. We give three proofs. The first uses the theorem of Ginsburg and Spanier stated in Chapter 4 (Theorem 4.3.1), the second is based on Lemma 5.3.2 and on some decidability results we proved in Chapter 4 for slender languages, and the third is direct, being based on Lemma 5.3.1 only.

Theorem 5.3.3. *It is decidable whether or not an arbitrary context-free language is in the family $\mathcal{F}^r_{\leq_p}$.*

First proof. Consider all pairs of words $u, v \in \Sigma^*$ with $|u| \leq p_L$ and $1 \leq |v| \leq p_L + 1$. For any such pair u, v, check, using Theorem 4.3.1, whether $L = Pref(uv^\omega)$. Notice that $Pref(uv^\omega)$ is a bounded language since, if

$$\begin{aligned} u &= u_1 u_2 \cdots u_{|u|}, \quad u_i \in \Sigma, 1 \leq i \leq |u|, \\ v &= v_1 v_2 \cdots v_{|v|}, \quad v_i \in \Sigma, 1 \leq i \leq |v|, \end{aligned}$$

then

$$Pref(uv^\omega) \subseteq u_1^* u_2^* \cdots u_{|u|}^* (v_1 v_2 \cdots v_{|v|})^* (v_1 v_2 \cdots v_{|v|-1})^* \cdots (v_1 v_2)^* v_1^*.$$

If a pair u, v as above is found, then the answer to our problem is affirmative, otherwise is negative. Either answer is correct by Lemma 5.3.1. ■

Second proof. Relying on Lemma 5.3.2, we give the following algorithm which decides whether or not an arbitrary context-free language $L \subseteq \Sigma^*$ belongs to the family $\mathcal{F}^r_{\leq_p}$.

Algorithm 5.3.4.
(1) Decides whether or not L is infinite; if not, then answer **no**, else go to step 2.

(2) Decide, using Corollary 4.4.7, whether or not L is 1-slender. If not, then answer **no**, else go to step 3.

(3) Decide, using Corollary 4.4.4, whether or not $L = Pref(L)$. If so, then answer **yes**, else answer **no**.

Notice that either answer is correct by Lemma 5.3.2. ■

Remark. We notice that this second proof gives a potentially stronger result as it implies that the problem in discussion is decidable for any family of languages for which (ii) in Lemma 5.3.2 can be checked.

Third proof. For $L \subseteq \Sigma^*$ a context-free language, the following algorithm decides whether or not $L \in \mathcal{F}^r_{\leq_p}$. The correctness proof is presented at the same time.

Algorithm 5.3.5.
(1) For all pairs $u, v \in \Sigma^*$ with $|u| \leq p_L, 1 \leq |v| \leq p_L + 1$, decide whether or not $L \subseteq Pref(uv^\omega)$. [That can be decided because $Pref(uv^\omega)$ is a regular language and it is decidable whether or not a context-free language is included in a regular one.]

If no such pair is found, then answer **no** [correct by Lemma 5.3.1].

(2) $[L \subseteq Pref(uv^\omega)]$ We have

$$Pref(uv^\omega) = F \cup \bigcup_{i=0}^{|v|-1} R_i,$$

where

$$F = \{\lambda, u_1, u_1u_2, u_1u_2u_3, \ldots, u_1u_2\cdots u_{|u|-1}\}$$

is a finite language and

$$R_0 = \{uv^n \mid n \geq 0\},$$
$$R_i = \{uv^n v_1v_2\cdots v_i \mid n \geq 0\}, 1 \leq i \leq |v| - 1,$$

are $|v|$ regular languages.

(3) Check whether $F \subseteq L$. If so, then put

$$L' = u^{-1}(L - F),$$

else answer **no**. [The family of context-free languages is effectively closed under intersection with regular languages ($L - F = L \cap \overline{F}$) as well as under quotient with regular sets.]

(4) For each $i, 0 \leq i \leq |v| - 1$ (supposing that $v_1 v_2 \cdots v_i = \lambda$ for $i = 0$) define the language

$$L_i = (L' \cap \{v^n v_1 v_2 \cdots v_i \mid n \geq 0\})(v_1 v_2 \cdots v_i)^{-1}$$

[as it was pointed out above, this is effectively constructable], the morphism (where we assume $\# \notin \Sigma$)

$$h : \#^* \longrightarrow \Sigma^*, \quad h(\#) = v,$$

and put

$$L_i' = h^{-1}(L_i).$$

[The family of context-free languages is effectively closed under inverse morphisms.]

It follows that $L_i' \subseteq \{\#^n \mid n \geq 0\}$ and $R_i \subseteq L$ if and only if $L_i' = \{\#^n \mid n \geq 0\}$.

(5) Decide whether or not $L_i' = \{\#^n \mid n \geq 0\}$. [$L_i'$ is a context-free language over a one letter alphabet, hence it is effectively regular.]

If the answer is affirmative for all is, then answer **yes**.

(6) If, for no pair u, v a positive answer is obtained at step 5, then answer **no**.

It should be clear that, in all cases, the answer given by the above algorithm is correct. ■

5.4 Factors and regularity

In [MaPa1] it is proved that it is undecidable whether or not an arbitrary context-free language is the set of factors of a right-infinite word. The same problem for regular languages is left open. In the following, we solve this problem in the affirmative.

5.4.1 Strongly minimal automaton

For a regular language $R \subseteq \Sigma^*$, a finite automaton $\mathcal{A} = (Q, \Sigma, \delta, I, F)$ accepting R and having a deterministic transition function $\delta : Q \times \Sigma \longrightarrow Q$ is called **strongly minimal** if and only if no state or transition in $\mathcal{A}$ is useless or redundant. That is, if we eliminate any state or transition in $\mathcal{A}$, the obtained automaton accepts a language strictly contained in R.

Lemma 5.4.1. *Any regular language $R \subseteq \Sigma^*$ closed under taking factors is accepted by a strongly minimal automaton $\mathcal{A} = (Q, \Sigma, \delta, Q, Q)$ in which all states are both initial and final.*

Proof. Let $\mathcal{A}' = (Q', \Sigma, \delta', I, T)$ be the minimal finite deterministic automaton recognizing R. Consider the automaton

$$\mathcal{A}'' = (Q', \Sigma, \delta', Q', Q').$$

Since R is closed under taking factors, $R = L(\mathcal{A}'')$. Because the equivalence problem is decidable for finite automata, we can now iteratively eliminate from $\mathcal{A}''$ all states and transitions which are either useless or redundant. That is, if $s \in Q'$ (resp. $\delta'(s, a) = s'$, for $s, s' \in Q', a \in \Sigma$) and the language accepted by the finite automaton $\mathcal{B}$ obtained from $\mathcal{A}''$ by removing the state s together with all transitions containing it (resp. removing the transition by a from s to s', respectively) is R, then take $\mathcal{B}$ instead of $\mathcal{A}''$ and continue the reduction. Obviously, after a finite number of steps, the automaton $\mathcal{A}$ asked for in the claim of our lemma is obtained. (Note that $\mathcal{A}$ is not necessarily unique, but this will not cause troubles later.) ■

5.4.2 The associated graph

For a finite automaton $\mathcal{A} = (Q, \Sigma, \delta, I, F)$, denote by $G(\mathcal{A})$ the graph associated to $\mathcal{A}$ and define the relation $\longrightarrow \subseteq Q \times Q$ by

$$p \longrightarrow q \text{ if and only if there is a path from } p \text{ to } q \text{ in } G(\mathcal{A}).$$

The relation $\equiv \subseteq Q \times Q$ defined by $p \equiv q$ if and only if $p \longrightarrow q$ and $q \longrightarrow p$ is an equivalence relation which induces an acyclic structure on $Q/_\equiv$ i.e., the graph $G = (Q/_\equiv, E)$ with $Q/_\equiv$ as the set of vertices and with the set of edges

$$E = \{([p], [q]) \mid [p], [q] \in Q/_\equiv \text{ and } p \longrightarrow q\}$$

is acyclic. (We have denoted the equivalence class of $p \in Q$ with respect to $\equiv$ by $[p]$.)

The automaton $\mathcal{A}$ is called **diconnected** if and only if there is a state $q \in Q$ such that $Q = [q]$. (For instance, the restriction of any automaton to an equivalence class with respect to the relation $\equiv$ is a diconnected automaton.)

An equivalence class $[p] \in Q/_{\equiv}$ is called **trivial** if and only if $[p]$ is a singleton ($[p] = \{p\}$) and there is no transition $p \xrightarrow{a} p$, for any $a \in \Sigma$.

A finite automaton $\mathcal{A} = (Q, \Sigma, \delta, I, F)$ is called **ultimately periodic** if and only if there is a state $q \in Q$ such that

$$\begin{array}{rcll} Q - [q] & = & \{s_1, s_2, \ldots, s_k\}, & \text{for some } k \geq 1, \\ [q] & = & \{q_1 = q, q_2, \ldots, q_l\}, & \text{for some } l \geq 1, \end{array}$$

and all transitions in $\mathcal{A}$ are

$$\begin{array}{ll} \delta(s_i, a_i) = s_{i+1}, 1 \leq i \leq k-1, & \text{for some } a_1, a_2, \ldots, a_{k-1} \in \Sigma, \\ \delta(s_k, a_k) = q, & \text{for some } a_k \in \Sigma, \\ \delta(q_j, b_j) = q_{j+1}, 1 \leq j \leq l-1, & \text{for some } b_1, b_2, \ldots, b_{l-1} \in \Sigma, \\ \delta(q_l, b_l) = q, & \text{for some } b_l \in \Sigma. \end{array}$$

Informally speaking, a finite automaton $\mathcal{A}$ is ultimately periodic if and only if $G(\mathcal{A})$ has the form in Figure 5.4.1 (using the notations in the definition).

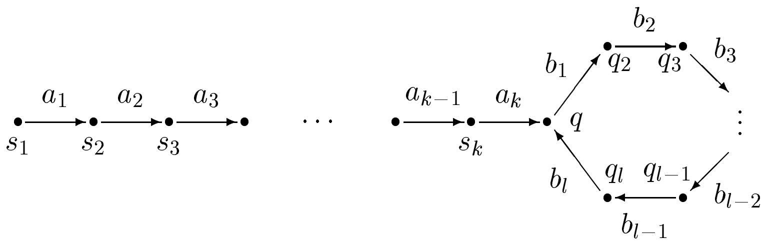


Figure 5.4.1: ultimately periodic finite automaton

(Note that if $\mathcal{A}$ is ultimately periodic, then $\mathcal{A}$ is not diconnected. Moreover, for any $p \in Q - [q]$, $[p]$ is trivial.)

An equivalence class $[p] \in Q/_{\equiv}$ is called a **source** if and only if there is no $[q] \in Q/_{\equiv} - [p]$ such that $q \longrightarrow p$.

Lemma 5.4.2. *For a regular language $R \subseteq \Sigma^*$ such that $R \in \mathcal{F}^r_{\leq_f}$, let $\mathcal{A} = (Q, \Sigma, \delta, Q, Q)$ be a strongly minimal automaton constructed by Lemma 5.4.1 for R. If $\mathcal{A}$ is not diconnected, then there is a unique equivalence class $[p] \in Q/_{\equiv}$ which is a source. Moreover, $[p]$ is trivial and there is exactly one transition leaving p in $\mathcal{A}$.*

Proof. The existence of a source is guaranteed by the fact that the graph $G = (Q/_{\equiv}, E)$ is acyclic.

Let us show that any source is trivial. Take a source $[p] \in Q/_{\equiv}$ and consider the subautomata

$$\mathcal{A}^{[p]} = ([p], \Sigma, \delta|_{[p]}, [p], [p])$$

and

$$\mathcal{A}^{Q-[p]} = (Q - [p], \Sigma, \delta|_{Q-[p]}, Q - [p], Q - [p])$$

of $\mathcal{A}$.

Suppose, contrary to the claim, that $[p]$ is not trivial. Take $p_1, p_2 \in [p]$ such that if $\mathrm{card}([p]) \geq 2$, then $p_1 \neq p_2$, otherwise $p_1 = p_2 = p$. If $p_1 \neq p_2$, then, as $\mathcal{A}^{[p]}$ is diconnected, there must be in $\mathcal{A}^{[p]}$ a path from p_1 to p_2, say $p_1 \xrightarrow{y} p_2, y \in \Sigma^+$, and another one from p_2 to p_1, say $p_2 \xrightarrow{z} p_1, z \in \Sigma^+$. If $p_1 = p_2$, as $[p]$ is not trivial, there must be a transition $p \xrightarrow{a} p$, for some $a \in \Sigma$, and we can take $y = z = a$. So, in what follows, it will not be important whether $p_1 \neq p_2$ or not. What is important is the fact that, in both cases, the word yz is not empty.

As, clearly, $G(\mathcal{A})$ is connected and the sets of states of $\mathcal{A}^{[p]}$ and $\mathcal{A}^{Q-[p]}$, respectively, are non-empty (p is in $\mathcal{A}^{[p]}$ and, if $\mathcal{A}^{Q-[p]}$ is empty, then $\mathcal{A} = \mathcal{A}^{[p]}$ hence diconnected, a contradiction), there must be a path from a state in $\mathcal{A}^{[p]}$ to one in $\mathcal{A}^{Q-[p]}$ and we can find $p_3 \in [p], q_1 \in Q - [p]$, and a transition $p_3 \xrightarrow{a} q_1, a \in \Sigma$, in $\mathcal{A}$. Since $\mathcal{A}$ is strongly minimal, there is a word $w \in \Sigma^*$ which contains a and is accepted by the automaton $\mathcal{A}$ but not accepted by the automaton obtained from $\mathcal{A}$ by removing the transition $p_3 \xrightarrow{a} q_1$. (That is, when $\mathcal{A}$ accepts w, then it must read a from p_3 to q_1.) It follows that we can find the states $p_4 \in [p]$, $q_2 \in Q - [p]$ and the words $w_1, w_2 \in \Sigma^*$ such that $w = w_1 a w_2$ and there are paths $p_4 \xrightarrow{w_1} p_3$ in $\mathcal{A}^{[p]}$ and $q_1 \xrightarrow{w_2} q_2$ in $\mathcal{A}^{Q-[p]}$. Using again the fact that $\mathcal{A}^{[p]}$ is diconnected, we get a path from p_1 to p_4 in $\mathcal{A}^{[p]}$, say $p_1 \xrightarrow{u} p_4, u \in \Sigma^*$, see Figure 5.4.2 below.

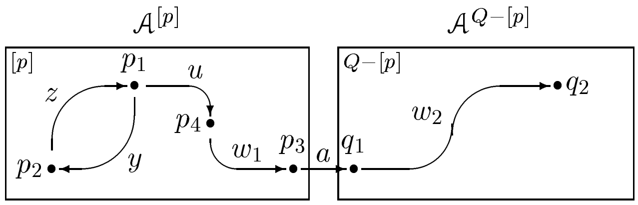
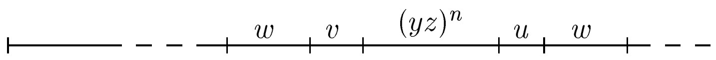


Figure 5.4.2

Then $(yz)^* u w_1 a w_2 \subseteq R$. Assume $R = Fact(\alpha)$ for some infinite word $\alpha \in \Sigma^\omega$ (there is such an α by hypothesis). As the language $(yz)^*$ contains arbitrarily long words, there must be an $n \geq 0$ such that an occurrence of

$(yz)^n uw$ appears in α after an occurrence of w reading from left to right. Supposing that we have the situation depicted in the figure below

α :


for some $v \in \Sigma^*$, α has a factor $wv(yz)^n uw$. But now $wv(yz)^n uw \in R$. Because, when accepting w, $\mathcal{A}$ must read a from p_3 to q_1, using the fact that $[p]$ is a source, we obtain that the word $v(yz)^n uw$ is accepted by $\mathcal{A}$ without reading $p_3 \xrightarrow{a} q_1$. In particular, w is accepted by $\mathcal{A}$ without reading $p_3 \xrightarrow{a} q_1$, a contradiction. It follows that $[p]$ is trivial.

Let us prove now that there is exactly one source in $\mathcal{A}$. For this, suppose that $[p]$ and $[q]$ are two different sources. As shown above, $[p] = \{p\}$ and $[q] = \{q\}$. As $\mathcal{A}$ is strongly minimal, there must be transitions starting from p and q labeled by $a \in \Sigma$ and $b \in \Sigma$, respectively. Moreover, there are some words $ax_1, bx_2 \in L(\mathcal{A})$ such that $\mathcal{A}$ must read the transition which leaves p (q) and is labeled by a (b) in order to accept the word ax_1 (bx_2, respectively). It follows that the word ax_1 cannot be prolonged to the left in R, that is, there is no word $w \in \Sigma^+$ such that $wax_1 \in R$. As $R = Fact(\alpha), \alpha \in \Sigma^\omega$, α must begin with ax_1. Since the same reasoning can be carried out for bx_2, we get that of ax_1 and bx_2 one is a prefix of the other. Suppose that ax_1 is a prefix of bx_2. In this case, ax_1 is accepted by the automaton $\mathcal{A}$ without reading the transition labeled a leaving p, a contradiction.

A similar reasoning shows that for a source $[p]$ the number of transitions leaving p is exactly one. Indeed, it is at least one and if there are two transitions labeled a and b, then $a \neq b$, because $\mathcal{A}$ is deterministic, and then α must start with both a and b, a contradiction. ■

5.4.3 Characterization and decidability

We now give a characterization for the regular languages in the family $\mathcal{F}^r_{\leq_f}$. This will help us to show that it is decidable whether or not a regular language belongs to this family.

Lemma 5.4.3. *For an infinite regular language $R \subseteq \Sigma^*$, $R \in \mathcal{F}^r_{\leq_f}$ if and only if an automaton $\mathcal{A}$ constructed for R in Lemma 5.4.1 is either diconnected or ultimately periodic.*

Proof. Suppose first that $\mathcal{A} = (Q, \Sigma, \delta, Q, Q)$ is diconnected and take an arbitrary $q \in Q$. As R is infinite, there must be at least one cycle $q \longrightarrow q$ in $G(\mathcal{A})$ hence there are infinitely many such cycles, the set of them being

$$C(q) = \{w \in \Sigma^* \mid q \xrightarrow{w} q\} = \{w_1, w_2, w_3, \dots\}.$$

Define

$$\alpha = w_1 w_2 w_3 \cdots .$$

We have $L(\mathcal{A}) = Fact(\alpha)$. The inclusion $Fact(\alpha) \subseteq L(\mathcal{A})$ is trivial. To prove the other one, take $w \in L(\mathcal{A})$. There are $p_1, p_2 \in Q$ such that $p_1 \xrightarrow{w} p_2$. Since $\mathcal{A}$ is diconnected, we have also $q \xrightarrow{u} p_1$, $p_2 \xrightarrow{v} q$, for some $u, v \in \Sigma^*$. It follows that there is an $i \geq 1$ such that $w_i = uwv$, so $w \in Fact(\alpha)$.

If $\mathcal{A}$ is ultimately periodic then $G(\mathcal{A})$ has the form in Figure 5.4.1. Hence (using the notations there)

$$L(\mathcal{A}) = Fact(a_1 a_2 \cdots a_k (b_1 b_2 \cdots b_l)^\omega).$$

Conversely, take a regular language $R \subseteq \Sigma^*$ such that $R \in \mathcal{F}^r_{\leq_f}$ and construct $\mathcal{A}$ as in Lemma 5.4.1. If $\mathcal{A}$ is diconnected, then we are done. Otherwise, by Lemma 5.4.2, there is exactly one source, say $[p_1]$, in $\mathcal{A}$. Moreover, $[p_1]$ is trivial and there is exactly one transition in $\mathcal{A}$ starting from p_1, say $p_1 \xrightarrow{a_1} p_2$, for some $p_2 \in Q - \{p_1\}$ and $a_1 \in \Sigma$. If $\alpha \in \Sigma^\omega$ with $R = Fact(\alpha)$, then from the proof of Lemma 5.4.2, α has the form $\alpha = a_1 \alpha', \alpha' \in \Sigma^\omega$.

Denote the automaton obtained from $\mathcal{A}$ by removing p_1 and the transition leaving p_1 (labeled a_1) by $\mathcal{A}_1$,

$$\mathcal{A}_1 = (Q - \{p_1\}, \Sigma, \delta|_{Q-\{p_1\}}, Q - \{p_1\}, Q - \{p_1\}).$$

If $\mathcal{A}_1$ is not diconnected, then the only source in $\mathcal{A}_1$ is $[p_2] = \{p_2\}$. Suppose that the only transition from p_2 is $p_2 \xrightarrow{a_2} p_3, p_3 \in Q - \{p_1, p_2\}, a_2 \in \Sigma$ and put

$$\mathcal{A}_2 = (Q - \{p_1, p_2\}, \Sigma, \delta|_{Q-\{p_1,p_2\}}, Q - \{p_1, p_2\}, Q - \{p_1 p_2\}).$$

If $\mathcal{A}_2$ is not diconnected, then we continue our procedure. Obviously, after a finite number of steps, say $k \geq 1$, we get a diconnected $\mathcal{A}_k$. Moreover, $L(\mathcal{A}_k)$ is infinite since R is. It remains to show that $G(\mathcal{A}_k)$ is a cycle. $G(\mathcal{A}_k)$ contains at least one cycle; suppose that there are two distinct cycles (meaning that none of them is contained in the other), the second one being $p_k \xrightarrow{u_k} p_k$. As mentioned, α must start with $a_1 a_2 \cdots a_{k-1}$. We have that

$$a_1 a_2 \cdots a_{k-1} w_k, a_1 a_2 \cdots a_{k-1} u_k \in Fact(\alpha).$$

As $a_1 a_2 \cdots a_{k-1}$ appears only at the beginning of α, it follows that w_k is a prefix of u_k or conversely, a contradiction. ■

Because, given a regular language R, a strongly minimal automaton for R is effectively constructable by Lemma 5.4.1 and it is decidable whether or not an arbitrary finite automaton is diconnected as well as ultimately periodic, we obtain as consequences of Lemma 5.4.3 the main results of this section.

Theorem 5.4.4. *It is decidable whether or not an arbitrary regular language is in the family* $\mathcal{F}^r_{\leq_f}$.

The similar result for left-infinite words follows easily from Theorem 5.4.4.

Theorem 5.4.5. *It is decidable whether or not an arbitrary regular language is in the family* $\mathcal{F}^l_{\leq_f}$.

Proof. It is straightforward to see that for a regular language L, L belongs to the family $\mathcal{F}^l_{\leq_f}$ if and only if the language L^R belongs to the family $\mathcal{F}^r_{\leq_f}$. Since the family of regular languages is closed under mirror image, our claim follows by Theorem 5.4.4. ∎

5.5 Bi-infinite words

As in the case of one-sided infinite words, it is very easy to prove that it is *undecidable* whether or not an arbitrary context-free language $L \subseteq \Sigma^*$ is in the family $\mathcal{F}^{bi}_{\leq_f}$ or not (the proof uses the undecidability of the problem of whether or not $L = \Sigma^*$ and is similar to the one of Theorem 6 in [MaPa1]).

In what concerns regular languages, we show in this section that the above problem is decidable.

First, notice that things are different from the case of one-sided infinite words; for instance, we can find a regular language $R \in \mathcal{F}^{bi}_{\leq_f}$ such that its automaton $\mathcal{A}$ constructed using Lemma 5.4.1 has a non-trivial source; see Figure 5.5.1

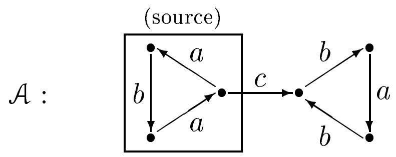


Figure 5.5.1: a non-trivial source in the bi-infinite case

Obviously, $R = Fact((aba)^*c(bab)^*) = Fact(\alpha)$ for $\alpha = {}^\omega(aba)c(bab)^\omega$.

Lemma 5.5.1. *For a regular language* $R \subseteq \Sigma^*$ *such that* $R \in \mathcal{F}^{bi}_{\leq_f}$, *let* $\mathcal{A} = (Q, \Sigma, \delta, Q, Q)$ *be a strongly minimal automaton constructed by Lemma 5.4.1 for* R. *Then there is a unique equivalence class* $[p] \in Q/_\equiv$ *which is a source. Moreover,* $[p]$ *is not trivial.*

Proof. Take $\alpha \in {}^\omega\Sigma^\omega$ such that $R = Fact(\alpha)$.

Suppose that $[p] \in Q/_\equiv$ is a source. Then, we have a transition $p \xrightarrow{a} q, a \in \Sigma, q \in Q - [p]$, and a word $w = w'aw'' \in R$ such that $\mathcal{A}$ must read $p \xrightarrow{a} q$ in order to accept w. Since α is bi-infinite, w can be prolonged arbitrarily long to the left in R, hence the language accepted by the subautomaton

$$\mathcal{A}^{[p]} = ([p], \Sigma, \delta|_{[p]}, [p], [p])$$

must be infinite. It follows that $[p]$ is not trivial.

Suppose now that there are two different sources $[p_1], [p_2] \in Q/_{\equiv}$ and the respective transitions and words as above are:

$$\begin{aligned} p_1 &\xrightarrow{a_1} q_1, a_1 \in \Sigma, q_1 \in Q - ([p_1] \cup [p_2]), w_1 = w_1' a_1 w_1'' \in R, \\ p_2 &\xrightarrow{a_2} q_2, a_2 \in \Sigma, q_2 \in Q - ([p_1] \cup [p_2]), w_2 = w_2' a_2 w_2'' \in R. \end{aligned}$$

(So, for $i = 1, 2$, $\mathcal{A}$ must read $p_i \xrightarrow{a_i} q_i$ in order to accept w_i.)

Since w_1 and w_2 appear as factors of α and $[p_1]$ and $[p_2]$ are sources, the occurrences of $w_1' a_1$ and $a_2 w_2''$ are not overlapped and the same holds for the occurrences of $w_2' a_2$ and $a_1 w_1''$. Consequently, the occurrences of w_1 and w_2 in α are not overlapped and we can find, for instance, $w_1 v w_2 \in Fact(\alpha) = R, v \in \Sigma^*$. But now w_2 can be accepted by $\mathcal{A}$ without reading $p_2 \xrightarrow{a_2} q_2$, a contradiction. The lemma is proved. ∎

We can give now a characterization of the regular languages in the family $\mathcal{F}^{bi}_{\leq_f}$.

Lemma 5.5.2. *For a regular language $R \subseteq \Sigma^*$, consider a strongly minimal automaton $\mathcal{A} = (Q, \Sigma, \delta, Q, Q)$ accepting R. Then $R \in \mathcal{F}^{bi}_{\leq_f}$ if and only if either $\mathcal{A}$ is diconnected or there are $u, v, w \in \Sigma^*$ such that $G(\mathcal{A})$ has the form in Figure 5.5.2.*

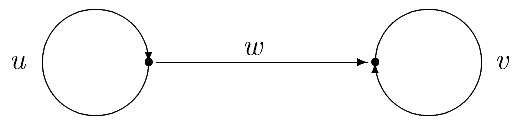


Figure 5.5.2

Proof. If $\mathcal{A}$ is diconnected, then we can prove as in Lemma 5.4.3 that $R \in \mathcal{F}^{bi}_{\leq_f}$.

If $G(\mathcal{A})$ has the form in the statement, then $R = Fact(\alpha)$ for $\alpha = {}^{\omega}uwv^{\omega} \in {}^{\omega}\Sigma^{\omega}$.

Conversely, suppose that $R \in \mathcal{F}^{bi}_{\leq_f}$. If $\mathcal{A}$ is not diconnected, then, by Lemma 5.5.1, we get a $p \in Q$ such that $[p]$ is the only source in $\mathcal{A}$ and $[p]$ is not trivial. If

$$\mathcal{A}^{[p]} = ([p], \Sigma, \delta|_{[p]}, [p], [p]),$$

then, as in the proof of Lemma 5.4.3, one can show that
(i) $G(\mathcal{A}^{[p]})$ is a cycle (u in the figure above),
(ii) $G(\mathcal{A}^{Q-[p]})$ is either of the form in Figure 5.4.1 or a cycle.
In both cases, the form of $\mathcal{A}$ in the statement of our lemma is obtained. ■

The main theorem of this section is a consequence of Lemma 5.5.2 and the fact that it is decidable whether or not an arbitrary finite automaton is diconnected or of the form in Figure 5.5.2.

Theorem 5.5.3. *It is decidable whether or not an arbitrary regular language is in the family* $\mathcal{F}^{bi}_{\leq_f}$.

Our last result in this section is an obviuos consequence of Theorems 5.4.4, 5.4.5, and 5.5.3.

Theorem 5.5.4. *It is decidable whether or not an arbitrary regular language is in the family* $\mathcal{F}_{\leq_f}$.

5.6 Undecidability

As proved in [MaPa1], it is undecidable whether or not an arbitrary context-free language is the set of finite factors of some right-infinite word. Using similar techniques, we prove this undecidability result for any quasi order containing $\leq_f$.

Theorem 5.6.1. *If* $\leq$ *is a quasi order on* Σ^* *which is an extension of* $\leq_f$, *then it is undecidable whether or not an arbitrary context-free language is in any of the families* $\mathcal{F}^r_{\leq}, \mathcal{F}^l_{\leq}$, $\mathcal{F}^{bi}_{\leq}$, *or* $\mathcal{F}_{\leq}$.

Proof. Consider a two-letter alphabet $\{a,b\}$ and

$$\{(x_1,x_2,\ldots,x_n),(y_1,y_2,\ldots,y_n)\},$$

$x_i,y_i \in \{a,b\}^+, 1\leq i\leq n\}$, an instance of the Post Correspondence Problem. Construct the languages (which are very similar with those in the proof of Theorem 4.5.1):

$$\begin{aligned}
&L(x) = \{ba^{i_k}\cdots ba^{i_1}cx_{i_1}\cdots x_{i_k} \mid k\geq 1, 1\leq i_j\leq n, 1\leq j\leq k\},\\
&L(y) = \{ba^{i_k}\cdots ba^{i_1}cy_{i_1}\cdots y_{i_k} \mid k\geq 1, 1\leq i_j\leq n, 1\leq j\leq k\},\\
&L_{x,y} = L(x)c\,L(y)^R,\\
&L_{mi} = \{w_1cw_2c\,w_2^Rc\,w_1^R \mid w_1,w_2\in\{a,b\}^*\},\\
&L_{PCP} = L_{x,y}\cap L_{mi},
\end{aligned}$$

where $c\notin\{a,b\}$ is a new letter. It is well-known (see, for instance, [Gi]) that the complement $\overline{L_{PCP}}$ of L_{PCP} is context-free.

We show that $\overline{L_{PCP}}$ belongs to any of the three families above if and only if the above instance of the Post Correspondence Problem has no solution.

Notice first that, since $\leq_f \subseteq \leq$, if $x \in \text{DOWN}_{\leq_f}(\overline{L_{PCP}})$, then $x \leq_f y$, for some $y \in \overline{L_{PCP}}$, and so $x \leq y$. Thus

$$\overline{L_{PCP}} \subseteq \text{DOWN}_{\leq_f}(\overline{L_{PCP}}) \subseteq \text{DOWN}_{\leq}(\overline{L_{PCP}}).$$

Suppose now that $\overline{L_{PCP}} \in \mathcal{F}_{\leq}$. Then, as $\text{DOWN}_{\leq}$ is idempotent, $\overline{L_{PCP}} = \text{DOWN}_{\leq}(\overline{L_{PCP}})$ and, using the observation above,

$$\overline{L_{PCP}} = \text{DOWN}_{\leq_f}(\overline{L_{PCP}}) = \{a, b, c\}^*$$

since we may assume that any solution of the PCP is a factor of a non-solution. (Indeed, the instances of the PCP which violate this condition are exactly those having all words as solutions, so they are trivial and can be eliminated.) Thus, $L_{PCP} = \emptyset$.

Conversely, if $L_{PCP} = \emptyset$, then $\overline{L_{PCP}} = \{a, b, c\}^*$. Chose now infinite words $\alpha_1 \in \Sigma^\omega, \alpha_2 \in {}^\omega\Sigma, \alpha_3 \in {}^\omega\Sigma^\omega$ such that $Fact(\alpha_i) = \{a, b, c\}^*$, for any $1 \leq i \leq 3$. Now, as $\leq$ is reflexive, it follows that $\overline{L_{PCP}} = \text{DOWN}_{\leq}(Fact(\alpha_3))$, so $\overline{L_{PCP}} \in \mathcal{F}^{bi}_{\leq}$. Using again the fact that $\leq_f \subseteq \leq$, we get

$$\{a, b, c\}^* = Fact(\alpha_1) = \text{DOWN}_{\leq_f}(Pref(\alpha_1)) \subseteq \text{DOWN}_{\leq}(Pref(\alpha_1))$$

so $\overline{L_{PCP}} = \text{DOWN}_{\leq}(Pref(\alpha_1))$ and $\overline{L_{PCP}} \in \mathcal{F}^{r}_{\leq}$. Analogously, $\overline{L_{PCP}} = \text{DOWN}_{\leq}(Suf(\alpha_2))$, thus $\overline{L_{PCP}} \in \mathcal{F}^{l}_{\leq}$. The proof is completed. ■

5.7 Factors appearing infinitely often

We consider in this section languages obtained from infinite words by taking all factors that appear infinitely many times. Answering an open problem in [MaPa1] and [MaPa2], we prove the existence of the infinite words α such that $Fact_\infty(\alpha)$ is context-free non-regular.

The problem was formulated separately for the following two complementary cases: for almost periodic infinite words and for non-almost periodic infinite words. (An infinite word α is *almost periodic* if for any finite factor x of it, there is a factorization of α in factors of equal length, $\alpha = x_1 x_2 x_3 \cdots$, such that each $x_i, i \geq 1$, has x as a factor.) An example of an almost periodic word is the famous Thue-Morse infinite word (cf. [Mo], [Th]).

Obviously, if α is almost periodic, then $Fact_\infty(\alpha) = Fact(\alpha)$. This variant of the problem (open problem 2 in [MaPa1]) is solved negatively in [Is], while the case of non-almost periodic infinite words (and so the unrestricted case) remained open.

We give here a positive solution to this problem. In fact, we prove more. Our answer is sharper than that asked for above in the sense that we shall give two examples of infinite (not almost periodic) words, α_1, α_2, such that $Fact_\infty(\alpha_1)$ is a linear non-regular language and $Fact_\infty(\alpha_2)$ is a context-free non-linear one.

Theorem 5.7.1. *For the infinite word α_1 over the alphabet $\{a, b, c\}$,*

$$\alpha_1 = z_1 z_2 z_3 \cdots$$

where

$$z_i = abc^{f(i)}a^2b^2c^{f(i)+1} \cdots a^ib^ic^{f(i)+i-1}, \text{ for any } i \geq 1,$$

with the mapping f verifying

(i) $f(1) = 1$,

(ii) $f(i) = f(i-1) + i - 1$, for any $i \geq 2$,

the set $Fact_\infty(\alpha_1)$ is linear non-regular.

Proof. Let us prove first that

$$Fact_\infty(\alpha_1) = Fact(\{c^{m-1}a^nb^nc^m \mid n, m \geq 1\}).$$

In order to prove the inclusion " $\subseteq$ ", let us take a factor w of α_1 containing occurrences of the symbols a and b from the both sides of a maximal factor of the form c^k of α_1, that is $w = ub^ic^ka^jv$, for some $u, v \in \{a, b, c\}^*, i, j, k \geq 1$. Because the lengths of maximal factors of the form c^k of α_1 form a strictly increasing sequence of natural numbers, w can not appear infinitely many times in α_1. So, this inclusion is proved.

For the other one, it is sufficient to observe that, for any $n \geq 1$ and $m \geq 1$, the word $c^{m-1}a^nb^nc^m$ appears infinitely many times in α_1 and the equality above is proved.

Obviously, the language

$$L = \{c^{m-1}a^nb^nc^m \mid n, m \geq 1\}$$

is linear. As the family of linear languages is closed under operation $Fact$, it follows that the language $Fact(L) = Fact_\infty(\alpha_1)$ is linear. In order to prove that this language is not regular, let us intersect it with the regular language $c^+a^+b^+c^+$. We obtain

$$Fact_\infty(\alpha_1) \cap c^+a^+b^+c^+ = \{c^ia^nb^nc^j \mid n, i, j \geq 1\}.$$

As the language in the right-hand member of this equality is not regular and the family of regular languages is closed under intersection it follows that $Fact_\infty(\alpha_1)$ is not regular and the proof is completed. ∎

Theorem 5.7.2. *For the infinite word α_2 over the alphabet $\{a, b, c, d, e\}$,*

$$\alpha_2 = z_1 z_2 z_3 \cdots$$

where

$$z_i = abcde^{f(i)}a^2b^2cde^{f(i)+1}abc^2d^2e^{f(i)+2}\cdots a^ib^icde^{f(i)+\frac{i(i-1)}{2}}a^{i-1}b^{i-1}c^2d^2 e^{f(i)+\frac{i(i-1)}{2}+1}\cdots a^2b^2c^{i-1}d^{i-1}e^{f(i)+\frac{i(i+1)}{2}-2}abc^id^ie^{f(i)+\frac{i(i+1)}{2}-1},$$

for any $i \geq 1$, and with the mapping f verifying

(i) $f(1) = 1$,

(ii) $f(i) = f(i-1) + \dfrac{i(i-1)}{2}$, *for any $i \geq 2$,*

the set $Fact_\infty(\alpha_2)$ is context-free non-linear. (In a block z_i, the sum of the powers of a, b and of c, d, takes all values from 2 to $i+1$, in all possible combinations.)

Proof. First, we prove the equality

$$Fact_\infty(\alpha_2) = Fact(\{e^{p-1}a^nb^nc^md^me^p \mid m, n, p \geq 1\}).$$

For the inclusion " $\subseteq$ ", we observe that any factor w of α_2 of the form

$$w = ud^ie^ka^jv, \text{ for some } u, v \in \{a, b, c, d, e\}^*, i, j, k \geq 1,$$

cannot appear infinitely many times in α_2, the argument being the same as in the previous proof.

The other inclusion is true because each word of the form $e^{p-1}a^nb^nc^md^me^p$, for some $m, n, p \geq 1$, appears infinitely many times in α_2.

Denoting

$$L = \{e^{p-1}a^nb^nc^md^me^p \mid m, n, p \geq 1\},$$

it is obvious that L is a context-free language and, as the family of contex-free languages is closed under operation $Fact$, we obtain that the language $Fact_\infty(\alpha_2) = Fact(L)$ is context-free. Intersecting this language with the regular language $e^+a^+b^+c^+d^+e^+$ we obtain

$$Fact_\infty(\alpha_2) \cap e^+a^+b^+c^+d^+e^+ = \{e^ia^nb^nc^md^me^j \mid n, m, i, j \geq 1\}$$

which is a non-linear language. Because the family of linear languages is closed under intersection with regular languages, it follows that $Fact_\infty(\alpha_2)$ is not linear. ■

5.8 Further research

We consider here the problems investigated in Sections 5.3, 5.4, and 5.5 for the case of factors appearing infinitely many times in an infinite word. We introduce first some notations similar with those for $\mathcal{F}$-families in Section 5.2.

$$\begin{aligned}
\mathcal{F}^r_{\leq_f,\infty} &= \{L \subseteq \Sigma^* \mid Fact_\infty(\alpha),\ \text{for some } \alpha \in \Sigma^\omega\},\\
\mathcal{F}^l_{\leq_f,\infty} &= \{L \subseteq \Sigma^* \mid Fact_\infty(\alpha),\ \text{for some } \alpha \in {}^\omega\Sigma\},\\
\mathcal{F}^{bi}_{\leq_f,\infty} &= \{L \subseteq \Sigma^* \mid Fact_\infty(\alpha),\ \text{for some } \alpha \in {}^\omega\Sigma^\omega\},\\
\mathcal{F}_{\leq_f,\infty} &= \mathcal{F}^r_{\leq_f,\infty} \cup \mathcal{F}^l_{\leq_f,\infty} \cup \mathcal{F}^{bi}_{\leq_f,\infty}.
\end{aligned}$$

In the case of context-free languages, things are very similar with the case of usual factors, as shown briefly below.

We use the constructions and notations in the proof of Theorem 5.6.1 and claim that $\overline{L_{PCP}}$ is in any on the four families above if and only if $L_{PCP} = \emptyset$.

If $\overline{L_{PCP}}$ is in some of those families, then, clearly, $\overline{L_{PCP}} = \text{DOWN}_{\leq_f}(\overline{L_{PCP}})$ and, in virtue of the remark in the proof of Theorem 5.6.1, we get $\overline{L_{PCP}} = \{a, b, c\}^*$, thus $L_{PCP} = \emptyset$. For the converse implication, it is enough to notice that, if $Fact(\alpha) = \{a, b, c\}^*$, for some infinite word α, then also $Fact_\infty(\alpha) = \{a, b, c\}^*$.

We have thus proved that it is undecidable whether or not an arbitrary context-free language is in any of the above families.

In the regular case, things are very much different from the case of all factors. For instance, using the main idea of the examples in Theorems 5.7.1 and 5.7.2, we can prove the following statement:

If L is a regular language and $\#$ is a lettter not in alph(L), then

$$\#^*L\#^* \cup \#^* \in \mathcal{F}^r_{\leq_f,\infty} \cap \mathcal{F}^l_{\leq_f,\infty} \cap \mathcal{F}^{bi}_{\leq_f,\infty}.$$

To see this, consider an arbitrary enumeration of the words in L,

$$L = \{w_0, w_1, w_2, \dots\}$$

and construct the infinite words

$$\begin{aligned}
\alpha_r &= w_0\#w_0\#^2w_1\#^3w_0\#^4w_1\#^5w_2\#^6w_0\#^7w_1\#^8w_2\#^9w_3\#^{10}\cdots,\\
\alpha_l &= \alpha_r^R,\\
\alpha_{bi} &= \alpha_l\alpha_r.
\end{aligned}$$

(For a more intuitive description, we abused the reversal and catenation. Precisely, $\alpha_l(n) = \alpha_r(-n)$, for $n \leq 0$, $\alpha_{bi}(n) = \alpha_r(n)$, for $n \geq 0$, $\alpha_{bi}(n) = \alpha_l(n+1)$, for $n < 0$.)

It is clear that any word of the form $\#^n$ or $\#^n w_m \#^p, n, m, p \geq 0$, appears infinitely many times in each of α_r, α_l, and α_{bi} and also that no other word does, due to the strictly increasing sequence of the powers of $\#$. The above statement is now clear.

However, the following problem remains to be investigated.

Problem 5.8.1. Is it decidable whether or not an arbitrary regular language belongs to the family $\mathcal{F}$, for $\mathcal{F} \in \{\mathcal{F}^{r}_{\leq f,\infty}, \mathcal{F}^{l}_{\leq f,\infty}, \mathcal{F}^{bi}_{\leq f,\infty}\}$?

We believe that the answer is affirmative and that the techniques developed in this chapter should work also for this case.

Chapter 6

Confluence of languages

We investigate in this chapter the *confluence* property, that is, the property of a language to contain, for any two words of it, one which is bigger, with respect to a given quasi order on the respective free monoid, than each of the former two. This property emerged from our considerations on the ways of obtaining languages of finite words from infinite words in the previuos chapter. It is investigated mainly for regular and context-free languages with respect to the prefix and the factor partial orders. The main decidability results are established, as well as generalizations of those and related results.

6.1 Confluence

As it is easy to see, a language L obtained by taking all finite factors (or prefixes) of an infinite word has the following property: for any two words $u, v \in L$, there is $w \in L$ (w is not necessarily different from u and v) such that both u and v are factors (prefixes, resp.) of w. From this, there naturally arises the problem of investigating this property for arbitrary languages and, moreover, for arbitrary quasi orders on the free monoid (instead of the factor partial order). We call this property *confluence*[1].

We investigate basic decidability results concerning the confluence property in the next two sections. Due to the basic idea that led us to introducing the confluence property, there is a strong connection between the confluence property and the property of a language being the set of prefixes or factors of an infinite word, as investigated in Chapter 5. This will allow us to use the decidability results we proved in Chapter 5. In this way, we prove that the confluence problem with respect to the factor partial order is decidable for regular languages and the confluence problem with respect to the prefix

[1]The term "confluence" comes from the name of a similar property for binary relations, cf., for instance, [We].

partial order is decidable even for context-free languages; the latter problem for factors is undecidable. We then establish some other undecidability results concerning context-free languages as well as some general undecidability results concerning the confluence property.

The next two sections deal with terminating relations and closure properties, respectively.

Finally, we consider a generalization of the confluence property and extend some of our main results for this. Of great help here are the results we proved in Chapters 3 and 4 about slender languages.

We now give the formal definition of the confluence property and some examples of languages confluent or not with respect to different quasi orders.

Given a quasi order $\leq$ on Σ^*, a language $L \subseteq \Sigma^*$ is **confluent w.r.t.** $\leq$ if and only if for any $x, y \in L$ there is $z \in L$ such that $x \leq z$ and $y \leq z$. Notice that the empty language is confluent w.r.t. any quasi order on Σ^*.

6.1.1 Examples

1. The regular language

$$L_1 = a^* \cup b^*$$

is confluent w.r.t. none of the partial orders $\leq_p, \leq_f$, or $\leq_s$ since, for any nonempty $w_1 \in a^*, w_2 \in b^*$, there is no word in L_1 which contains both w_1 and w_2 as prefixes, factors, or subwords, respectively.

2. The context-free language

$$L_2 = \{a^n b^n \mid n \geq 0\}$$

is confluent w.r.t. $\leq_f$ because, for any $a^n b^n, a^m b^m \in L_2$, we have, for any $p \geq \max\{m, n\}$, $a^n b^n \leq_f a^p b^p, a^m b^m \leq_f a^p b^p$, and $a^p b^p \in L_2$. It follows that L_2 is confluent w.r.t. $\leq_s$ too. But, since there is no word in L_2 which has as prefixes both ab and $a^2 b^2$, L_2 is not confluent w.r.t. $\leq_p$. Notice that also Σ^* is not confluent w.r.t. $\leq_p$, for any alphabet $\Sigma, |\Sigma| \geq 2$.

3. Consider two morphisms $g, h : \Sigma^* \longrightarrow \Delta^*$ and the associated equality set

$$E(g, h) = \{w \in \Sigma^+ \mid g(w) = h(w)\}.$$

Because $E(g, h)$ is closed under concatenation, it follows that it is confluent w.r.t. each of the partial orders $\leq_f$ and $\leq_s$.

In fact, this a particular case of a more general result: if a language L satisfies $L = L^+$, then L is confluent w.r.t. any quasi order $\leq$ which is an extension of $\leq_f$.

6.1.2 Confluence and down-sets

We now show that, when deciding the confluence property, one may work with the down-set of a language instead of the language itself. This result, in spite of the fact that it is not difficult to prove, will be very helpful in proving many decidability results in this chapter and in the next one. This is due to the exceptional regularity of the structure of the down-sets.

Lemma 6.1.1. *For a quasi order $\leq$ on Σ^*, any language $L \subseteq \Sigma^*$ is confluent w.r.t. $\leq$ if and only if the down-set of L, $\textsc{down}_{\leq}(L)$, is confluent w.r.t. $\leq$.*

Proof. Suppose that L is confluent w.r.t $\leq$ and take $x, y \in \textsc{down}_{\leq}(L)$. It follows that there exist two words $u, v \in L$ such that $x \leq u$ and $y \leq v$. As L is confluent w.r.t $\leq$, there is a word $w \in L$ such that $u \leq w$ and $v \leq w$. Since $\leq$ is transitive, we get $x \leq w, y \leq w$. Now, because $\leq$ is reflexive, $L \subseteq \textsc{down}_{\leq}(L)$, so $w \in \textsc{down}_{\leq}(L)$ and, therefore, $\textsc{down}_{\leq}(L)$ is confluent w.r.t. $\leq$.

Conversely, suppose that $\textsc{down}_{\leq}(L)$ is confluent w.r.t $\leq$ and take $x, y \in L$. As $x, y \in \textsc{down}_{\leq}(L)$, we can find $w \in \textsc{down}_{\leq}(L)$ such that $x \leq w$ and $y \leq w$, since $\textsc{down}_{\leq}(L)$ is confluent w.r.t. $\leq$. By the definition of $\textsc{down}_{\leq}(L)$, there must be a word $z \in L$ such that $w \leq z$. But now $x \leq z, y \leq z$, so L is confluent w.r.t. $\leq$, as claimed. ■

6.2 The regular case

We prove in this section that the confluence problem w.r.t. the factor partial order is decidable for regular languages. In this aim, of great help will be our decidability results in Chapter 5 concerning the property of a language of being the set of factors of an infinite word.

We establish first the following lemma which shows a connection between the property of a language of being confluent w.r.t. $\leq_f$ and of belonging to the family $\mathcal{F}_{\leq_f}$.

Lemma 6.2.1. *A language $L \subseteq \Sigma^*$ is in the family $\mathcal{F}_{\leq_f}$ if and only if the following conditions are fulfilled:*
(i) L is infinite,
(ii) $L = \textsc{down}_{\leq_f}(L)$,
(iii) L is confluent w.r.t. $\leq_f$.

Proof. Suppose first that $L \in \mathcal{F}_{\leq_f}$. This is equivalent to the fact that there exists an infinite word $\alpha \in \Sigma^\omega \cup {}^\omega\Sigma \cup {}^\omega\Sigma^\omega$ such that $L = Fact(\alpha)$. The conditions (i) and (ii) are obviously fulfilled.

Let us prove now (iii). Take $x, y \in L$. Then $x, y \in Fact(\alpha)$ and there is $w \in Fact(\alpha) = L$ such that $x \leq_f w, y \leq_f w$. Consequently, L is confluent w.r.t. $\leq_f$.

For the converse part, suppose that $L \subseteq \Sigma^*$ is a language satisfying (i), (ii), and (iii). We construct an infinite word α following the *algorithm* below:

(1) Write L as a totally ordered infinite set (the order being arbitrary)

$$L = \{u_1, u_2, u_3, \dots\}.$$

(2) Define inductively the finite words v_n, for all $n \geq 1$, as follows:

- $v_1 = u_1$,
- For any $n \geq 2$, supposing that $v_1, v_2, \dots, v_{n-1}$ are defined, take v_n as the word of L having the property that $u_{n-1} \leq_f v_n$ and $v_{n-1} \leq_f v_n$ (v_n exists since L is confluent w.r.t. $\leq_f$ by (iii)).

(3) Since $v_n \leq_f v_{n+1}$, it follows that v_{n+1} is obtained from v_n by adding some (possibly empty) words at its ends. Thus, the limit $\alpha = \lim_{n\to\infty} v_n$ is well defined and infinite since L is infinite.

(4) Depending on the direction(s) in which the sequence $(v_n)_{n\geq 1}$ extends unboundedly, α can be a left-, right-, or bi-infinite word. (As we will see in a moment, all the three cases are possible.)

Let us prove that $L = Fact(\alpha)$.

If $w \in L$, then $w = u_n$, for some $n \geq 1$ and so $w \leq_f v_n, v_n \in Fact(\alpha)$. Thus, $w \in Fact(\alpha)$.

Conversely, for $w \in Fact(\alpha)$, since we supposed that the sequence $(v_n)_{n\geq 1}$ grows unboundedly in both direction, there must be an $n \geq 1$ such that v_n contains w as a factor. As $v_n \in L$ and $L = \text{DOWN}_{\leq_f}(L)$, w must be in L and our equality is proved.

As mentioned above, all the three cases are indeed possible. For instance, as it is easy to prove, the language

$$L_1 = a^* \cup ba^* = Fact(ba^\omega)$$

belongs to the family $\mathcal{F}^r_{\leq_f}$ but is not contained in any of the other two families, the language

$$L_2 = a^* \cup a^*b = Fact({}^\omega ab)$$

in only in the family $\mathcal{F}^l_{\leq_f}$, and

$$L_3 = a^*ba^* \cup a^* = Fact({}^\omega aba^\omega)$$

is only in $\mathcal{F}^{bi}_{\leq_f}$.

Let us further remark that the infinite word α in the construction above might not be unique.

Our proof is completed. ■

We can prove now the announced decidability result. The proof itself is simple but it uses quite a lot of work already done. (See also Problem 6.7.1 at the end of the chapter.)

Theorem 6.2.2. *It is decidable whether or not an arbitrary regular language is confluent w.r.t.* $\leq_f$.

Proof. Obviously, the problem of confluence w.r.t $\leq_f$ is decidable for finite languages. (In fact this is decidable for any quasi order $\leq$ for which it can be decided whether or not two arbitrary words are related by $\leq$.) Now, by Theorem 5.5.4, it follows that it is decidable whether or not an arbitrary regular language L belongs to the family $\mathcal{F}_{\leq_f}$. But, by Lemma 6.2.1, an infinite language L with $L = \text{DOWN}_{\leq_f}(L)$ is in the family $\mathcal{F}_{\leq_f}$ if and only if it is confluent w.r.t. $\leq_f$. Since L and $\text{DOWN}_{\leq_f}(L)$ are simultaneously confluent w.r.t. $\leq_f$ (Lemma 6.1.1), using the fact that the finiteness problem is decidable for regular languages, our result follows. ■

Notice that it is possible to prove in a similar way that it is decidable whether or not an arbitrary regular language is confluent w.r.t. $\leq_p$ but we will prove in the next section a more general result, namely that this problem is decidable for context-free languages.

6.3 The context-free case

We now move to the context-free case and prove that the confluence problem w.r.t. prefixes is decidable while the same problem for factors is undecidable. We consider also some general undecidability results.

6.3.1 Decidability for prefixes

We prove in this section that, rather unexpectedly, the confluence problem w.r.t. $\leq_p$ is decidable for context-free languages. The basic idea is similar with the one in the previous section.

We state first a result similar to Lemma 6.2.1 for the prefix order instead of the factor one.

Lemma 6.3.1. *Any language* $L \subseteq \Sigma^*$ *is in the family* $\mathcal{F}^r_{\leq_p}$ *if and only if the following conditions are fulfilled:*

(i) L is infinite,
(ii) $L = \text{DOWN}_{\leq_p}(L)$,
(iii) L is confluent w.r.t. $\leq_p$.

Proof. The proof is similar to the one of Lemma 6.2.1 with the difference that (using the same notations as in the algorithm there) in this case, for all $n \geq 1$, v_n is a prefix of v_{n+1}, and so the infinite word obtained using the algorithm is always a right-infinite one. Consequently, the family $\mathcal{F}_{\leq_f}$ is replaced by $\mathcal{F}^r_{\leq_p}$. ■

We may state now

Theorem 6.3.2. *It is decidable whether or not an arbitrary context-free language is confluent w.r.t. $\leq_p$.*

Proof. Since the property in discussion is decidable for finite languages, we may restrict the problem to infinite languages. Using Lemma 6.1.1, we may further restrict our family of languages to those which are closed under taking prefixes. (Notice that $\text{DOWN}_{\leq_p}(L)$ is effectively context-free for L context-free.) But now, by Lemma 6.3.1, for an infinite context-free language L with $L = \text{DOWN}_{\leq_p}(L)$, L belongs to the family $\mathcal{F}^r_{\leq_p}$ if and only if L is confluent w.r.t. $\leq_p$. Finally, the former problem is decidable by Lemma 5.3.3 and, using the fact that the finiteness problem is decidable for context-free languages, our decidability result follows. ■

Remark. Notice that for any family of languages closed under reversal operation (and the families of regular and context-free languages are so) all results proved for the prefix order hold true also for the suffix order by left-right duality.

6.3.2 Undecidability for factors

When considering factors, the situation changes and we get undecidability. Our theorem will follow as a consequence of the following general undecidability result.

Lemma 6.3.3. *Let $\mathcal{L}$ be a family of languages effectively closed under union, catenation with symbols, and λ-free catenation closure. If the confluence problem w.r.t. $\leq_f$ is decidable in $\mathcal{L}$, then so is the inclusion problem (and hence the equivalence problem).*

Proof. We observe first that the emptiness problem is decidable in $\mathcal{L}$. Indeed, if $L \in \mathcal{L}$, then

$$L' = \#L\#_1 \cup \#L\#_2 \in \mathcal{L},$$

where $\#, \#_1, \#_2$ are new symbols, and L' is confluent w.r.t. $\leq_f$ if and only if $L = \emptyset$. Indeed, if $L = \emptyset$, then L' is confluent w.r.t. $\leq_f$. Conversely, if $L \neq \emptyset$, then choose $w \in L$ and consider the words $\#w\#_1, \#w\#_2 \in L'$. Since there is no $z \in L'$ having both $\#w\#_1$ and $\#w\#_2$ as factors, a contradiction is obtained. Thus, instead of deciding whether or not $L = \emptyset$, it is enough to decide whether or not L' is confluent w.r.t. $\leq_f$ which is possible, by hypothesis, since L' is effectively in $\mathfrak{L}$.

Let us prove that the inclusion problem is decidable for $\mathfrak{L}$. For take two arbitrary languages $L_1, L_2 \in \mathfrak{L}$. Suppose that $L_1 \subseteq \Sigma^*, L_2 \subseteq \Sigma^*$, and construct the language

$$L_3 = \#L_1\# \cup (\#L_2\#\#)^+$$

where $\# \notin \Sigma$ is a new symbol. By hypothesis, L_3 is effectively in $\mathfrak{L}$. We prove first the following claim.

Claim. L_3 is confluent w.r.t. $\leq_f$ if and only if either $L_1 \subseteq L_2$ or else $\mathrm{card}(L_1) = 1$ and $L_2 = \emptyset$.

Proof of Claim. Suppose first that $L_1 \subseteq L_2$. It follows that

$$\begin{aligned} \text{DOWN}_{\leq_f}(L_3) &= \text{DOWN}_{\leq_f}(\#L_1\#) \cup \text{DOWN}_{\leq_f}((\#L_2\#\#)^+) \\ &= \text{DOWN}_{\leq_f}((\#L_2\#\#)^+) \end{aligned}$$

since

$$\text{DOWN}_{\leq_f}(\#L_1\#) \subseteq \text{DOWN}_{\leq_f}(\#L_2\#) \subseteq \text{DOWN}_{\leq_f}((\#L_2\#\#)^+).$$

Because $(\#L_2\#\#)^+$ is confluent w.r.t. $\leq_f$, it follows by Lemma 6.1.1 that $\text{DOWN}_{\leq_f}((\#L_2\#\#)^+)$ is also confluent w.r.t. $\leq_f$. Now, $\text{DOWN}_{\leq_f}(L_3)$ is confluent w.r.t. $\leq_f$ and, using again Lemma 6.1.1, we get that L_3 is confluent w.r.t. $\leq_f$.

Suppose now that $\mathrm{card}(L_1) = 1$ and $L_2 = \emptyset$. If we put $L_1 = \{w\}$, then

$$L_3 = \{\#w\#\}$$

and it is obvious that L_3 is confluent w.r.t. $\leq_f$ since any singleton is. One implication is proved.

For the other one, suppose that L_3 is confluent w.r.t. $\leq_f$ and $L_1 \not\subseteq L_2$. Take $u \in L_1 - L_2$. Suppose now that $L_2 \neq \emptyset$ and take $v \in L_2$. Since L_3 in confluent w.r.t. $\leq_f$, there is a $w \in L_3$ such that $\#u\# \leq_f w$ and $\#v\#\# \leq_f w$. Because of the three symbols $\#$ in the second word, we must have $w \in (\#L_2\#\#)^+$ and put

$$w = \#w_1\#\#\#w_2\#\# \dots \#w_n\#\#, n \geq 1, w_i \in L_2, 1 \leq i \leq n.$$

From $\#u\# \leq_f w$, we get that $u = w_i$, for some $1 \leq i \leq n$. Thus $u \in L_2$, a contradiction. Consequently, $L_2 = \emptyset$ and $L_3 = \#L_1\#$.

Suppose that $\mathrm{card}(L_1) \geq 2$ and take $u, v \in L_1, u \neq v$. Using again the confluence of L_3 w.r.t. $\leq_f$, we find $w \in L_3$ with $\#u\# \leq_f w$ and $\#v\# \leq_f w$. If $w = \#x\#$, then $u = x = v$, a contradiction. Consequently, $\mathrm{card}(L_1) = 1$ ($L_1 \neq \emptyset$ because $L_1 \not\subseteq L_2$) and our claim is proved. ■

We give now an *algorithm* for deciding whether or not $L_1 \subseteq L_2$.

(1) Decide whether or not $L_2 = \emptyset$ [the emptiness problem is decidable for $\mathcal{L}$].

1.1 If *yes*, then go to step 2.

1.2 If *no*, then go to step 3.

(2) Decide whether or not $L_1 = \emptyset$ [again, possible].

2.1 If *yes*, the output **yes** [obviously correct].

2.2 If *no*, then output **no** [again correct].

(3) Decide whether or not L_3 is confluent w.r.t. $\leq_f$ [the confluence problem w.r.t. $\leq_f$ is decidable for $\mathcal{L}$ by hypothesis and L_3 is effectively in $\mathcal{L}$].

3.1 If *yes*, then output **yes** [correct by our claim].

3.2 If *no*, then output **no** [again, correct by our claim].

The proof is concluded. ■

From Lemma 6.3.3, we get immediately the following result.

Theorem 6.3.4. *It is undecidable whether or not an arbitrary context-free language is confluent w.r.t. $\leq_f$.*

6.3.3 More about undecidability

We prove in this section some other undecidability results concerning confluence and down-sets. The first one concerns the equality between a context-free language and its down-set w.r.t. any quasi order which is an extension of the prefix partial order $\leq_p$.

Theorem 6.3.5. *For any quasi order $\leq$ on Σ^* such that $\leq_p \subseteq \leq$, it is undecidable, for an arbitrary context-free language L, whether or not $L = \mathrm{DOWN}_{\leq}(L)$.*

Proof. Consider a two-letter alphabet $\{a, b\}$ and

$$\{(x_1, x_2, \ldots, x_n), (y_1, y_2, \ldots, y_n)\},$$

$x_i, y_i \in \{a,b\}^+, 1 \leq i \leq n\}$, an instance of the Post Correspondence Problem. Construct the languages below which we already used before:

$$\begin{aligned}
&L(x) = \{ba^{i_k} \ldots ba^{i_1} c x_{i_1} \ldots x_{i_k} \mid k \geq 1, 1 \leq i_j \leq n, 1 \leq j \leq k\},\\
&L(y) = \{ba^{i_k} \ldots ba^{i_1} c y_{i_1} \ldots y_{i_k} \mid k \geq 1, 1 \leq i_j \leq n, 1 \leq j \leq k\},\\
&L_{x,y} = L(x) c\, L(y)^R,\\
&L_{mi} = \{w_1 c w_2 c\, w_2^R c\, w_1^R \mid w_1, w_2 \in \{a,b\}^*\},\\
&L_{PCP} = L_{x,y} \cap L_{mi},
\end{aligned}$$

where $c \notin \{a,b\}$ is a new letter. As known, the complement $\overline{L_{PCP}}$ of L_{PCP} is context-free.

We claim that $\overline{L_{PCP}} = \text{DOWN}_{\leq}(\overline{L_{PCP}})$ if and only if the above instance of the Post Correspondence Problem has no solution.

Indeed, if $L_{PCP} = \emptyset$, then $\overline{L_{PCP}} = \{a,b,c\}^*$ and so we have $\overline{L_{PCP}} = \text{DOWN}_{\leq}(\overline{L_{PCP}})$. Conversely, suppose that $\overline{L_{PCP}} = \text{DOWN}_{\leq}(\overline{L_{PCP}})$. From $\leq_p \subseteq \leq$, we get

$$\overline{L_{PCP}} \subseteq \text{DOWN}_{\leq_p}(\overline{L_{PCP}}) \subseteq \text{DOWN}_{\leq}(\overline{L_{PCP}}),$$

so $\overline{L_{PCP}} = \text{DOWN}_{\leq_p}(\overline{L_{PCP}})$. Since we may assume that any solution is a prefix of a non-solution, we have $\text{DOWN}_{\leq_p}(\overline{L_{PCP}}) = \{a,b,c\}^*$, so $L_{PCP} = \emptyset$. (We may indeed make this assumption because the instances of the PCP which violate this condition are trivial, as having all words as solutions, so they can be eliminated.) The proof is concluded. ■

Remark. Notice that the fact that it is undecidable whether or not $L = \text{DOWN}_{\leq_p}(L)$ for an arbitrary context-free language L does not invalidate the proof of Theorem 6.3.2 since there we do not decide whether or not $L = \text{DOWN}_{\leq_p}(L)$, but replace L by $\text{DOWN}_{\leq_p}(L)$, which can be effectively done for L context-free.

We give next a general undecidability result concerning restrictions of the factor partial order $\leq_f$.

Theorem 6.3.6. *Let $\leq$ be a quasi order on Σ^* such that $\leq \subseteq \leq_f$ and $\mathcal{L}$ a family of languages which is effectively closed under union and catenation with symbols such that the emptiness problem is undecidable for languages in $\mathcal{L}$. Then the confluence problem w.r.t. $\leq$ is undecidable for languages in $\mathcal{L}$.*

Proof. Consider a language $L \subseteq \Sigma^*, L \in \mathcal{L}$, and three new symbols $\#, \#_1, \#_2 \notin \Sigma$. Construct the language

$$L_1 = \#L\#_1 \cup \#L\#_2.$$

We have that L_1 is effectively in $\mathcal{L}$ and claim that L_1 is confluent w.r.t. $\leq$ if and only if $L = \emptyset$.

If $L = \emptyset$, then $L_1 = \emptyset$ and so L_1 is confluent w.r.t. $\leq$. Conversely, suppose that L_1 is confluent w.r.t. $\leq$. If L is not empty, then take $w \in L$. Now $\#w\#_1, \#w\#_2 \in L_1$ and, since L_1 is confluent w.r.t. $\leq$, there must be a $v \in L_1$ such that $\#w\#_1 \leq v$ and $\#w\#_2 \leq v$. Without loss of generality, we may suppose that $v \in \#L\#_1$ and let us put $v = \#u\#_1, u \in L$. From $\#w\#_1 \leq \#u\#_1$ we get $u = w$, since $\leq$ is contained in $\leq_f$. But now, $\#w\#_1 \leq v$ means $\#w\#_1 \leq \#w\#_2$, which is impossible. Thus $L = \emptyset$.

Consequently, we have reduced the decidability of the confluence problem w.r.t. $\leq$ to the decidability of the emptiness problem which is, by hypothesis, undecidable for $\mathcal{L}$. The proof is concluded. ■

From the proof of the previous theorem, we obtain directly the following corollary.

Corollary 6.3.7. *For any quasi order $\leq$ on Σ^* such that $\leq \subseteq \leq_f$, it is undecidable whether or not the intersection of two arbitrary context-free languages is confluent w.r.t. $\leq$.*

Another general undecidability result, concerning restrictions of the Parikh subword quasi order $\leq_\Psi$, similar with the one in the above theorem, will be given in Chapter 7 (Theorem 7.4.8).

6.4 Terminating relations

In this section we investigate the confluence w.r.t. terminating partial orders. We need first some definitions.

For a relation $\leq$ on Σ^*, its inverse is denoted $\leq^{-1}$ and defined by $x \leq^{-1} y$ iff $y \leq x$. If $\leq$ is a partial order, then so is $\leq^{-1}$. A quasi order $\leq$ on Σ^* is called **well founded** if there is no infinite descending chain w.r.t. $\leq$, that is, there is no infinite sequence of words $w_1 \leq^{-1} w_2 \leq^{-1} w_3 \leq^{-1} \ldots$ such that for no $n \geq 1$, $w_n \leq w_{n+1}$. A quasi order $\leq$ is **terminating** if its inverse $\leq^{-1}$ is well founded.

We start with a general result which characterizes the languages confluent with respect to a terminating partial order.

Lemma 6.4.1. *If $\leq$ is a terminating partial order on Σ^*, then any language $L \subseteq \Sigma^*$ is confluent w.r.t. $\leq$ if and only if there exists a unique word $x_L \in L$ such that, for all $y \in L$, $y \leq x_L$.*

Proof. Suppose that $\leq$ is a terminating partial order on Σ^* and $L \subseteq \Sigma^*$ is a language confluent w.r.t. $\leq$. We may suppose that L is non-empty since any empty language verifies our property. Thus, there exists a word, say x_1, in L. If there is no $y \in L, y \neq x_1$, such that $x_1 \leq y$, then we choose $x_L = x_1$. Then, for any $y \in L$, since L is confluent, there is $w \in L$ such that $x_L \leq w, y \leq w$. But, we must have $x_L = w$ from the choice of x_L. So $y \leq x_L$ and we are done. If there is a $x_2 \in L - \{x_1\}$ such that $x_1 \leq x_2$ then we make the same reasoning with x_2 instead of x_1. Since $\leq$ is terminating, after a finite number $n \geq 1$ of steps we find $x_n \in L$ such that for no $y \in L, y \neq x_n$, $x_n \leq y$. It follows as above that $x_L = x_n$ is the word looked for. If there is another one, say x'_L, we have $x_L \leq x'_L$ and $x'_L \leq x_L$. As $\leq$ is antisymmetric, $x_L = x'_L$ so x_L is unique.

The converse implication is trivial. ■

We next apply the previous lemma for some particular terminating partial orders.

Lemma 6.4.2. *For any partial order* $\leq \in \{\leq_p^{-1}, \leq_f^{-1}, \leq_s^{-1}\}$ *and any language* $L \subseteq \Sigma^*$, *L is confluent w.r.t.* $\leq$ *if and only if there is a unique word* $x_L \in L$ *such that* $|x_L| = \min_{x \in L} |x|$ *and* $y \leq x_L$ *for all* $y \in L$.

Proof. Follows directly from Lemma 6.4.1 since all the three partial orders in discussion are terminating and $x \leq y$ implies $|x| \leq |y|$ with $|x| = |y|$ if and only if $x = y$. ■

The problem of confluence w.r.t. a terminating partial order seems to be easier than the problem for a non-terminating one. Indeed, we show that for the inverses of all the three relations considered above, the confluence problem is decidable for a very large family of languages.

Theorem 6.4.3. *For any partial order* $\leq \in \{\leq_p^{-1}, \leq_f^{-1}, \leq_s^{-1}\}$ *and any family* $\mathcal{L}$ *of languages such that*

(i) $\mathcal{L}$ *is closed under gsm mappings,*

(ii) for any language in $\mathcal{L}$, *the set of minimal length words is effectivelly computable,*

it is decidable whether or not an arbitrary language in $\mathcal{L}$ *is confluent w.r.t.* $\leq$.

Proof. Suppose that we have a a family of languages $\mathcal{L}$ with the properties (i) and (ii) above and $\leq \in \{\leq_p^{-1}, \leq_f^{-1}, \leq_s^{-1}\}$. For an arbitrary language $L \in \mathcal{L}$, $L \subseteq \Sigma^*$, we use the following *algorithm* to decide whether or not L is confluent w.r.t. $\leq$.

(1) Compute the set of minimal length words of L, say $L_{\min}$ (this is possible by (ii)).

(2) If $\mathrm{card}(L_{\min}) \geq 2$, then answer **no** (the answer is correct by Lemma 6.4.2).

(3) If $\mathrm{card}(L_{\min}) = 0$, then answer **yes**. (It follows that $L_{\min} = \emptyset$, so $L = \emptyset$ and the answer is correct.)

(4) Now $\mathrm{card}(L_{\min}) = 1$ and put $L_{\min} = \{x_L\}$. By Lemma 6.4.2, we have to check only whether or not $y \leq x_L$ for all $y \in L$.

(5) If $x_L = \lambda$, then answer **yes**.

(6) We may suppose now that all words in L are non-empty. Consider a new symbol $\#$ and construct a (deterministic) gsm $g_{\leq}$ such that, for any $y \in L$,

$$g_{\leq}(y) = \begin{cases} \lambda, & y \not\leq x_L, \\ \#, & y \leq x_L, \end{cases}$$

Suppose that $x_L = a_1 a_2 \ldots a_n, a_i \in \Sigma$, $1 \leq i \leq n$. Then our gsms are:

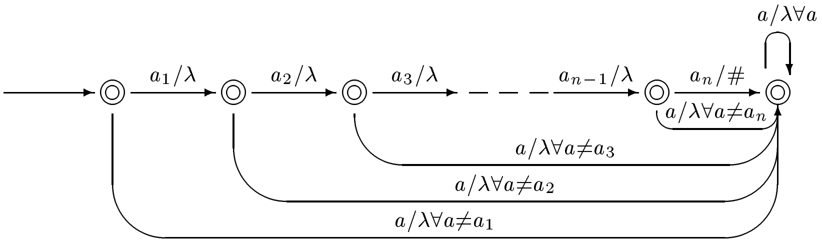


Figure 6.4.1: the gsm $g_{\leq_p^{-1}}$

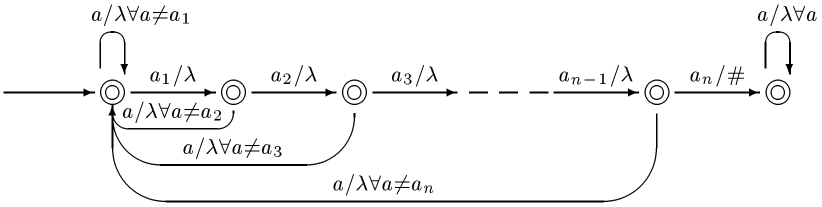


Figure 6.4.2: the gsm $g_{\leq_f^{-1}}$

Notice that, in Figures 6.4.1 and 6.4.2, all states are final for both $g_{\leq_p^{-1}}$ and $g_{\leq_f^{-1}}$ so each of them maps any word into a unique one. Moreover, $g_{\leq_p^{-1}}$ ($g_{\leq_f^{-1}}$) outputs at most one symbol which is $\#$ and this only in the case when the input contains $a_1 a_2 \ldots a_n$ as a prefix (factor, respectively).

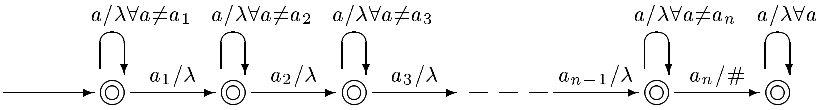


Figure 6.4.3: the gsm $g_{\leq_s^{-1}}$

Also, notice that if $a_1 a_2 \ldots a_n$ appears as a subword of $y \in L$, then we may suppose that a_1 appears as the first occurrence of a_1 in y, a_2

appears as the first occurrence of a_2 after one of a_1, and so on and so forth. The converse is obvious and it follows that $g_{\leq_s^{-1}}$ in Figure 6.4.3 works as required.

(7) Now, for any $L \in \mathcal{L}$, $g_{\leq}(L) \in \{\{\lambda\}, \{\lambda, \#\}, \{\#\}\}$ and L is confluent w.r.t. $\leq$ if and only if $g_{\leq}(L) = \{\#\}$.

(8) Compute $g_{\leq}(L)_{\min}$. [This is possible by (ii), since $g_{\leq}(L) \in \mathcal{L}$.]

(9) If $g_{\leq}(L)_{\min} = \{\#\}$, then answer **yes**, otherwise answer **no**. [Notice that in both cases the answer is correct.]

Our proof is now complete. ∎

Corollary 6.4.4. *For any partial order $\leq \in \{\leq_f^{-1}, \leq_s^{-1}, \leq_p^{-1}\}$ and any full trio $\mathcal{L}$ such that $\mathcal{L}$ containes only recursive languages and the emptiness problem is decidable for all languages in $\mathcal{L}$, it is decidable whether or not an arbitrary language in $\mathcal{L}$ is confluent w.r.t. $\leq$.*

Proof. Both conditions in Theorem 6.4.3 are fulfilled because: (i) any full trio is closed under gsm mappings and (ii) the set of minimal length words $L_{\min}$ can be effectively computed for any language $L \in \mathcal{L}$ by deciding first whether or not $L = \emptyset$ and in the case $L = \emptyset$ set $L_{\min} = \emptyset$ while for $L \neq \emptyset$ compute effectivelly $L_{\min}$ (possible because L is recursive). The result is proved. ∎

As a direct consequence of Corollary 6.4.4 we get

Corollary 6.4.5. *For any partial order $\leq \in \{\leq_f^{-1}, \leq_s^{-1}, \leq_p^{-1}\}$, it is decidable whether or not an arbitrary context-free language is confluent w.r.t. $\leq$.*

6.5 Closure properties for down-operators

We investigate briefly in this subsection the closure under the down-operator $\text{DOWN}_{\leq}$.

A general result concerning the closure under down-operators is the following.

Theorem 6.5.1. *Let $\leq \subseteq \Sigma^* \times \Sigma^*$ be a quasi order for which there is a gsm $g_{\leq}$ such that for any $w \in \Sigma^*, \text{DOWN}_{\leq}(\{w\}) = g_{\leq}(w)$. Then any full trio is closed under the down-operator $\text{DOWN}_{\leq}$.*

Proof. Consider a full trio $\mathcal{L}$ and a language $L \in \mathcal{L}$. We have then

$$\text{DOWN}_{\leq}(L) = \bigcup_{w \in L} \text{DOWN}_{\leq}(\{w\}) = \bigcup_{w \in L} g_{\leq}(w) = g_{\leq}(L).$$

Since any full trio is closed under gsms, it follows that the family $\mathcal{L}$ is closed under the down-operator $\text{DOWN}_{\leq}$. ■

Corollary 6.5.2. *Any full trio is closed under the down-operators* $\text{DOWN}_{\leq_p}$, $\text{DOWN}_{\leq_f}$, *and* $\text{DOWN}_{\leq_s}$.

Proof. We need, according to Theorem 6.5.1, only to show that, for any of the down-operators in the statement, there is a gsm as required. The three constructions are given in Figure 6.5.1.

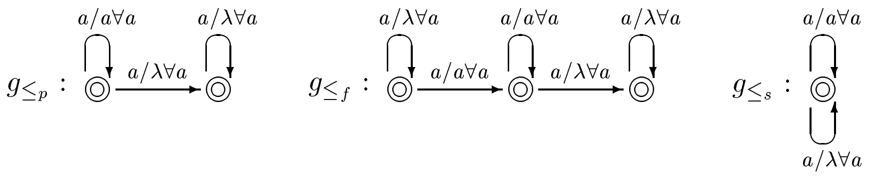


Figure 6.5.1

It is easy to see from the graphs in Figure 6.5.1 that the three gsm's works as required. ■

Another general closure result which covers also the part concerning $\leq_s$ in Corollary 6.5.2 will be proved in the next chapter (Theorem 7.6.3).

6.6 Generalized confluence

We generalize in this section the notion of confluence in a natural way: instead of considering confluent languages, we take finite unions of those.[2]

Formally, for a quasi order $\leq$ on Σ^* and a non-negative integer $k \geq 0$, we say that a language $L \subseteq \Sigma^*$ is ***k*-confluent w.r.t. $\leq$** if

$$L = \bigcup_{i=1}^{k} L_i,$$

for some languages $L_i \subseteq \Sigma^*$, L_i confluent w.r.t. $\leq$, for any $1 \leq i \leq k$. L is called **confluent w.r.t. $\leq$ in generalized sense** if L is k-confluent w.r.t. $\leq$, for some $k \geq 1$.

Notice that this is a generalization of the ordinary confluence since 1-confluent is the same as confluent.

We give next some examples.

[2] In the same way as the semilinear sets generalize the linear ones, cf. Chapter 2, page 21.

Examples

1. The language

$$L_1 = \{a^n b \mid n \geq 0\}$$

is obviously confluent w.r.t. $\leq_f$ but it is not confluent w.r.t. $\leq_p$. In fact, L_1 is not confluent w.r.t. $\leq_p$ even in generalized sense. Indeed, if R is a nonempty subset of L_1 such that R is confluent w.r.t. $\leq_p$, then R has exactly one element, since any two elements of L_1 are incomparable w.r.t. $\leq_p$. Thus, as L_1 is infinite, it cannot be written as a finite union of languages confluent w.r.t. $\leq_p$.

2. For any alphabet Σ such that $\mathrm{card}(\Sigma) \geq 2$, the total language

$$L_2 = \Sigma^*$$

is not confluent w.r.t. $\leq_p$, even in the generalized sense. To prove this, consider two different letters of Σ, say a and b. For any $k \geq 1$, consider the following $k+1$ words in Σ^*: $ab^{k+1}, a^2b^k, \ldots, a^{k+1}b$. Since, for any two words x, y of the previous $k+1$, there is no word $z \in \Sigma^*$ such that $x \leq_p z$ and $y \leq_p z$, it follows that Σ^* is not k-confluent w.r.t. $\leq_p$.

3. For any $k \geq 1$, the language

$$L_{3,k} = \{a^i b a^j \mid i, j \geq 0, i + j = k\}$$

is $(k+1)$-confluent w.r.t. $\leq_s$ but not k-confluent w.r.t. $\leq_s$. Notice that $L_{3,k}$ is confluent w.r.t. $\leq_\Psi$.

Concerning the decidability of the generalized confluence problem w.r.t. a given quasi order $\leq$, there are two main problems. We formulate them for an arbitrary family $\mathcal{L}$ of languages.

1. Given a fixed $k \geq 1$, is it decidable whether or not an arbitrary language $L \in \mathcal{L}$ is k-confluent w.r.t. $\leq$?

2. Is it decidable whether or not an arbitrary language $L \in \mathcal{L}$ is confluent w.r.t. $\leq$ in generalized sense, that is, is there some $k \geq 1$ such that the given language is k-confluent w.r.t. $\leq$?

We investigate below these two problems for context-free languages and prove that both are decidable for the prefix partial order and undecidable for the factor partial order, thus generalizing the corresponding results for ordinary confluence.

For the decidability results in case of prefixes, our results about slender context-free languages in Chapters 3 and 4 will be of essential importance.

6.6.1 k-confluence and down-sets

We now prove that a language and its corresponding down-set are simultaneously confluent w.r.t some quasi order in the generalized sense. Moreover, the respective constant k is the same. This is a generalization of Lemma 6.1.1 and is very important for our forthcoming interests since it allows us, when deciding the k-confluence or the generalized confluence, to work with the down-set of a language instead of the language itself.

Lemma 6.6.1. *For any quasi order $\leq$ on Σ^* and integer $k \geq 1$, a language $L \subseteq \Sigma^*$ is k-confluent w.r.t. $\leq$ if and only if the down-set $\textsc{down}_{\leq}(L)$ is k-confluent w.r.t. $\leq$.*

Proof. Suppose first that L is k-confluent w.r.t. $\leq$. By definition, we have

$$L = \bigcup_{i=1}^{k} L_i,$$

for some languages $L_i \subseteq \Sigma^*, L_i$ confluent w.r.t. $\leq$, for any $1 \leq i \leq k$. We have

$$\textsc{down}_{\leq}(L) = \textsc{down}_{\leq}(\bigcup_{i=1}^{k} L_i) = \bigcup_{i=1}^{k} \textsc{down}_{\leq}(L_i)$$

and, by Lemma 6.1.1, any language $\textsc{down}_{\leq}(L_i), 1 \leq i \leq k$, is confluent w.r.t. $\leq$. Thus $\textsc{down}_{\leq}(L)$ is k-confluent w.r.t. $\leq$.

Conversely, suppose that $\textsc{down}_{\leq}(L)$ is k-confluent w.r.t. $\leq$ and put

$$\textsc{down}_{\leq}(L) = \bigcup_{i=1}^{k} R_i,$$

for some languages $R_i \subseteq \Sigma^*, R_i$ confluent w.r.t. $\leq$, for any $1 \leq i \leq k$. As above, we have

$$\textsc{down}_{\leq}(L) = \bigcup_{i=1}^{k} \textsc{down}_{\leq}(R_i),$$

so we may assume that, for any $1 \leq i \leq k$, $R_i = \textsc{down}_{\leq}(R_i)$. Again, we may indeed make this assumption because, by Lemma 6.1.1, $\textsc{down}_{\leq}(R_i)$ is confluent w.r.t. $\leq$ since R_i is.

Suppose also that, for no $1 \leq j \leq k$,

$$\textsc{down}_{\leq}(L) = \bigcup_{\substack{i=1 \\ i \neq j}}^{k} R_i.$$

This would mean that we can eventually decrease k to $k' < k$. Then, if we prove that L is k'-confluent w.r.t. $\leq$, then it is also k-confluent w.r.t. $\leq$.

Let us define, for any $1 \leq i \leq k$, the language

$$L_i = L \cap R_i.$$

Claim 1. $L = \bigcup_{i=1}^{k} L_i$.

Proof of Claim 1. Since $L_i \subseteq L$, for any $1 \leq i \leq k$, the inclusion "$\supseteq$" follows.

For the converse inclusion, take $w \in L$. Because $\leq$ is reflexive, $L \subseteq \text{DOWN}_{\leq}(L)$, so $w \in \text{DOWN}_{\leq}(L)$ and thus $w \in R_i$, for some $1 \leq i \leq k$. Hence $w \in L \cap R_i = L_i$. ■

Claim 2. For any $1 \leq i \leq k$ and $x \in R_i$, there exists $y \in L \cap R_i$ such that $x \leq y$.

Proof of Claim 2. We argue by contradiction. Suppose that there are $i_0, 1 \leq i_0 \leq k$, and $x \in R_{i_0}$ such that, for any y, $x \leq y$ implies $y \notin L \cap R_{i_0}$.

Take an arbitrary $w \in R_{i_0}$. Since R_{i_0} is confluent w.r.t. $\leq$, there is $y \in R_{i_0}$ such that $w \leq y$ and $x \leq y$. Using our assumption, it follows that $y \notin L$. But $y \in R_{i_0} \subseteq \text{DOWN}_{\leq}(L)$, so $y \leq z$, for some $z \in L$. Then $z \in R_j$, for some $1 \leq j \leq k$. We must have $j \neq i_0$ because $j = i_0$ would imply $z \in L \cap R_{i_0}$ and as $x \leq z$, by the transitivity of $\leq$, this would contradict our assumption. Consequently, $w \leq z$ and, as we supposed that $R_j = \text{DOWN}_{\leq}(R_j)$, we get $w \in R_j$. Because $j \neq i_0$ and w was arbitrarily chosen in R_{i_0}, we have

$$R_{i_0} \subseteq \bigcup_{\substack{i=1 \\ i \neq i_0}}^{k} R_i$$

thus

$$\text{DOWN}_{\leq}(L) = \bigcup_{\substack{i=1 \\ i \neq i_0}}^{k} R_i,$$

a contradiction which proves the claim. ■

Claim 3. For any $1 \leq i \leq k, \text{DOWN}_{\leq}(L_i) = R_i$.

Proof of Claim 3. Since, by definition, $L_i \subseteq R_i$ and, as we supposed, $R_i = \text{DOWN}_{\leq}(R_i)$, the inclusion $\text{DOWN}_{\leq}(L_i) \subseteq R_i$ follows.

For the converse inclusion, take $x \in R_i$. By Claim 2, there is $y \in L \cap R_i = L_i$ such that $x \leq y$. Consequently, $x \in \text{DOWN}_{\leq}(L_i)$ and the equality is proved. ■

Finally, the fact that R_i is confluent w.r.t. $\leq$ implies, by Claim 3, that $\text{DOWN}_{\leq}(L_i)$ is confluent w.r.t. $\leq$. By Lemma 6.1.1, it follows that also L_i is confluent w.r.t. $\leq$. Using Claim 1, we have that L is k-confluent w.r.t. $\leq$ as claimed. ■

6.6.2 k-confluence and slenderness

We establish next some connections between the property of a language of being confluent w.r.t. the prefix partial order in generalized sense and of being slender.

Our first result is a connection between k-confluence and the family $\mathcal{F}_{\leq_p}$.

Lemma 6.6.2. *For any $k \geq 1$ and $L \subseteq \Sigma^*$, the language $\text{DOWN}_{\leq_p}(L)$ is k-confluent w.r.t. $\leq_p$ if and only if there exist the languages $L_i \subseteq \Sigma^*, 1 \leq i \leq k$, such that*

$$\text{DOWN}_{\leq_p}(L) = \bigcup_{i=1}^{k} L_i$$

and, for any $1 \leq i \leq k$,

- *either $L_i = Pref(w)$, for some $w \in \Sigma^*$,*
- *or $L_i = Pref(\alpha)$, for some infinite word $\alpha \in \Sigma^\omega$.*

Proof. Consider $k \geq 1$ and $L \subseteq \Sigma^*$. Assume first that $\text{DOWN}_{\leq_p}(L)$ is k-confluent w.r.t. $\leq_p$. By definition, there are $R_i \subseteq \Sigma^*, 1 \leq i \leq k$, such that

$$\text{DOWN}_{\leq_p}(L) = \bigcup_{i=1}^{k} R_i$$

and R_i is confluent w.r.t. $\leq_p$, for any $1 \leq i \leq k$. We have then

$$\text{DOWN}_{\leq_p}(L) = \text{DOWN}_{\leq_p}(\text{DOWN}_{\leq_p}(L)) = \text{DOWN}_{\leq_p}(\bigcup_{i=1}^{k} R_i) = \bigcup_{i=1}^{k} \text{DOWN}_{\leq_p}(R_i).$$

If R_i is finite, for some $1 \leq i \leq k$, then, clearly, $\text{DOWN}_{\leq_p}(R_i) = Pref(w)$, for some $w \in \Sigma^*$.

If R_i is infinite, then R_i confluent w.r.t. $\leq_p$ implies, by Lemma 6.1.1, $\text{DOWN}_{\leq_p}(R_i)$ confluent w.r.t. $\leq_p$ and so, by Lemma 6.3.1, $\text{DOWN}_{\leq_p}(R_i) \in \mathcal{F}^r_{\leq_p}$. Consequently, we may take

$$L_i = \text{DOWN}_{\leq_p}(R_i),$$

for any $1 \leq i \leq k$, and the required representation of $\text{DOWN}_{\leq_p}(L)$ is obtained.

The converse implication is obvious and the lemma is proved. ■

The next corollary follows directly from the representation given above in Lemma 6.6.2.

Corollary 6.6.3. *For any $k \geq 1$ and $L \subseteq \Sigma^*$, if L is k-confluent w.r.t. $\leq_p$, then $\text{DOWN}_{\leq_p}(L)$ is k-slender.*

Proof. If L is k-confluent w.r.t. $\leq_p$, then, by Lemma 6.1.1, $\text{DOWN}_{\leq_p}(L)$ is also k-confluent w.r.t. $\leq_p$ and in virtue of the representation in the previous lemma, it is enough to observe that both languages $Pref(w)$, for $w \in \Sigma^*$, and $Pref(\alpha)$, for $\alpha \in \Sigma^\omega$, are 1-slender. ■

Next, using the theory we developed for slender context-free languages in Chapters 3 and 4, we improve the result in Lemma 6.6.2 as follows.

Lemma 6.6.4. *For any $k \geq 1$ and $L \subseteq \Sigma^*$, a context-free language L is k-confluent w.r.t. $\leq_p$ if and only if*

(i) $\text{DOWN}_{\leq_p}(L)$ is k-slender;

(ii) there exist the languages $L_i \subseteq \Sigma^*, 1 \leq i \leq k$, such that

$$\text{DOWN}_{\leq_p}(L) = \bigcup_{i=1}^{k} L_i$$

and, for any $1 \leq i \leq k$,

– *either $L_i = Pref(w)$, for some $w \in \Sigma^*$,*

– *or $L = Pref(uv^\omega)$, for some $u, v \in \Sigma^*, v \neq \lambda$.*

Proof. Consider $k \geq 1$ and $L \subseteq \Sigma^*$ context-free and confluent w.r.t. $\leq_p$. Part (i) follows from Corollary 6.6.3 so let us prove part (ii).

Since L is k confluent w.r.t. $\leq_p$, by Lemma 6.1.1, so is $\text{DOWN}_{\leq_p}(L)$ and we have the representation in Lemma 6.6.2,

$$\text{DOWN}_{\leq_p}(L) = \bigcup_{i=1}^{k} L_i,$$

where each L_i is the set of prefixes of a finite or right-infinite word. All we have to do is to prove that when some L_i is the set of prefixes of some infinite word, then this infinite word is ultimately periodic.

Since $\mathrm{DOWN}_{\leq_p}(L)$ is context-free and slender, it follows by Theorem 3.3.1 that it is a finite union of paired loops, say

$$\mathrm{DOWN}_{\leq_p}(L) = \bigcup_{i=1}^{m} \{u_i v_i^n w_i x_i^n y_i \mid n \geq 0\},$$

for some $m \geq 0$ and $u_i, v_i, w_i, x_i, y_i \in \Sigma^*, 1 \leq i \leq m$.

Consider now $i, 1 \leq i \leq k$, for which L_i is infinite and put $L_i = Pref(\alpha)$, for some $\alpha \in \Sigma^\omega$. Then, there is $j, 1 \leq j \leq m$, such that the set

$$Pref(\alpha) \cap \{u_j v_j^n w_j x_j^n y_j \mid n \geq 0\}$$

is infinite.

Consider $n_1 \geq 0$ such that

$$u_j v_j^{n_1} w_j x_j^{n_1} y_j \in Pref(\alpha)$$

and

(i) $n_1 \geq 2$,
(ii) $|x_j^{n_1}| \geq |x_j| + |v_j|$.

Consider also $n_2 \geq 0$ such that

$$u_j v_j^{n_2} w_j x_j^{n_2} y_j \in Pref(\alpha)$$

and

$$|v_j^{n_2}| \geq |v_j^{n_1} w_j x_j^{n_1} y_j|.$$

It follows that

$$u_j v_j^{n_1} w_j x_j^{n_1} y_j \leq_p u_j v_j^{n_2} w_j x_j^{n_2} y_j$$

and, applying Fine and Wilf's theorem, we get, using also Theorem 2.1.1, that

$$\begin{array}{ll}
v_j = z^{p_1}, & z \in \Sigma^+, z \text{ primitive}, p_1 \geq 1, \\
x_j = (z_2 z_1)^{p_2}, & z = z_1 z_2, p_2 \geq 1, \\
w_j = z^{q_1} z_1, & q_1 \geq 0, \\
y_j = \left\{ \begin{array}{l} z_2 z^{q_2} z_3, \\ z_5 \end{array} \right. & \begin{array}{ll} z = z_3 z_4, q_2 \geq 0, & \text{if } |y| \geq |z_2|, \\ z_2 = z_5 z_6, z_6 \neq \lambda, & \text{otherwise.} \end{array}
\end{array}$$

We obtain that, for any $n \geq 0$,

$$\begin{aligned}
u_j v_j^n w_j x_j^n y_j &= u_j z^{(p_1+p_2)n+q_1} z_1 y_j \\
&= \left\{ \begin{array}{ll} u_j z_3 (z_4 z_3)^{q_1+q_2+1} ((z_4 z_3)^{p_1+p_2})^n, & \text{if } |y| \geq |z_2|, \\ u_j z_1 z_5 (z_6 z_1 z_5)^{q_1} ((z_6 z_1 z_5)^{p_1+p_2})^n, & \text{otherwise.} \end{array} \right.
\end{aligned}$$

In either case, we get that $\alpha = uv^\omega$, for some $u, v \in \Sigma^*, v \neq \lambda$, as claimed. ■

Corollary 6.6.5. *If $k \geq 1$ and $L \subseteq \Sigma^*$ is a context-free language which is k-confluent w.r.t. $\leq_p$, then L is regular.*

Remark. It is clear that condition (i) in Lemma 6.6.4 does not imply that L is k-confluent w.r.t. $\leq_p$. We remark that, even with the hypothesis that L is confluent w.r.t. $\leq_p$ in generalized sense, it still does not follow that L is k-confluent w.r.t. $\leq_p$. For instance, the language

$$L = Pref(ab^\omega) \cup Pref(a^2b^\omega) \cup \{b\}$$

is 2-slender but it is not 2-confluent w.r.t. $\leq_p$. Notice that L is 3-confluent w.r.t. $\leq_p$.

6.6.3 Prefixes

We prove in this subsection that both problems, the k-confluence problem w.r.t. $\leq_p$ and the generalized confluence problem w.r.t. $\leq_p$ are decidable for context-free languages.

We start with the result for the generalized confluence. The proof is based on the characterization given by Lemma 6.6.4 and on the effective representations we proved in Chapter 4 for slender context-free languages.

Theorem 6.6.6. *It is decidable whether or not an arbitrary context-free language is confluent w.r.t. $\leq_p$ in generalized sense.*

Proof. Let $L \subseteq \Sigma^*$ be a context-free language. The following algorithm checks whether or not L is confluent w.r.t. $\leq_p$ in generalized sense.

Since, by Lemma 6.6.1, L and $\text{DOWN}_{\leq_p}(L)$ are confluent w.r.t. $\leq_p$ in generalized sense at the same time and $\text{DOWN}_{\leq_p}(L)$ can be effectively constructed from L, we may assume in the sequel that $L = \text{DOWN}_{\leq_p}(L)$.

Algorithm 6.6.7.
(1) Decide whether or not L is slender [using Theorem 4.3.2] and if it is not, then answer **no**. [correct by Lemma 6.6.4]

(2) [L is slender] Find a representation of L as a union of paired loops [using Theorem 4.4.1], say

$$L = \bigcup_{i=1}^{m} \{u_i v_i^n w_i x_i^n y_i \mid n \geq 0\}.$$

(3) Check whether this representation verifies the conditions in the statement of Lemma 6.6.4 (ii) for some $k \geq 0$. This is done as follows.

(4) Check whether any infinite paired loop of L (in the representation at step 2) is of the form $\{uv^n \mid n \geq 0\}$ [this is clearly possible; see the proof of Lemma 6.6.4] and, if not, then answer **no** [correct by Lemma 6.6.4].

(5) [All loops of L are of this form, so L is effectively regular] For any infinite loop of L, say $\{uv^n \mid n \geq 0\}$, check whether $Pref(uv^\omega) \subseteq L$. If so, then replace L by $L - Pref(uv^\omega)$, else answer **no** [correct by Lemma 6.6.4].

(6) [L is finite] Check whether L is a finite union of sets of prefixes of some finite words. [this is clearly possible] If so, then answer **yes**, otherwise answer **no** [both answers are correct again by Lemma 6.6.4].

In a similar manner, we prove the corresponding result for k-confluence. It generalizes Theorem 6.3.2. We preferred to present the result for ordinary confluence separately in order to emphasize the similarity with the case of confluence w.r.t. factors for regular languages (Theorem 6.2.2).

Theorem 6.6.8. *For any fixed $k \geq 0$, it is decidable whether or not an arbitrary context-free language is k-confluent w.r.t. $\leq_p$.*

Proof. The proof is similar with the one given for the confluence in the generalized sense, except that here we have to check whether or not the representation of L as a union of paired loops verifies the conditions in the statement of Lemma 6.6.4 (ii) for the given k.

For this, we only have to notice that, if the intersection of two sets of the form $Pref(uv^n)$ is infinite, then the two sets must coincide. Therefore, the number of infinite L_i's in the characterization given by Lemma 6.6.4 is precisely determined, thus we have to count carefully only the number of those L_i's which are finite (at step 6 in the above algorithm). This is clearly possible. ∎

It is easy to see that at the above result we proved actually more, namely that we can compute the minimal number of L_i's there. This is stated explicitly in the next corollary.

Corollary 6.6.9. *For a context-free language L which is confluent w.r.t. $\leq_p$ in generalized sense, the minimal k such that L is k-confluent w.r.t. $\leq_p$ is computable. Also*

$$\min\{k \mid \text{DOWN}_{\leq_p}(L) \textit{ is } k\textit{-slender}\} \leq \min\{k \mid L \textit{ is } k\textit{-confluent w.r.t. } \leq_p\}.$$

6.6.4 Factors

We deal in this section with the factor partial order $\leq_f$ and prove that the k-confluence problem w.r.t. $\leq_f$ and the generalized confluence problem w.r.t. $\leq_f$ are undecidable for context-free languages.

We prove first a general undecidability result concerning the decidability of the k-confluence problem w.r.t. $\leq_f$. It generalizes Lemma 6.3.3. Also the proof is a generalization but, for the sake of completeness, we present it in all details.

Lemma 6.6.10. *Let $k \geq 1$ and $\mathfrak{L}$ be a family of languages effectively closed under union, catenation with letters, and λ-free catenation closure such that the inclusion problem is undecidable in $\mathfrak{L}$. Then also the k-confluence problem is undecidable in $\mathfrak{L}$.*

Proof. We argue by contradiction. Suppose that the k-confluence problem w.r.t. $\leq_f$ is decidable in $\mathfrak{L}$.

Claim 1. The emptiness problem is decidable in $\mathfrak{L}$.

Proof of Claim 1. Take an arbitrary language $L \in \mathfrak{L}, L \subseteq \Sigma^*$, and construct the language

$$L_1 = \bigcup_{i=1}^{k+1} \# L \#_i,$$

where $\#, \#_1, \#_2, \ldots, \#_{k+1} \notin \Sigma$ are new letters. It follows by hypothesis that L_1 is effectively in $\mathfrak{L}$ and we claim that L is empty if and only if L_1 is k-confluent w.r.t. $\leq_f$.

If $L = \emptyset$, then $L_1 = \emptyset$, so L_1 is k-confluent w.r.t. $\leq_f$.

Conversely, suppose that L_1 is k-confluent w.r.t. $\leq_f$ but $L \neq \emptyset$ and take $w \in L$. We have by definition

$$L_1 = \bigcup_{i=1}^{k} R_i,$$

for some $R_i \subseteq \Sigma^*, R_i$ confluent w.r.t. $\leq_f$. Consider the words $\#w\#_i, 1 \leq i \leq k+1$. They belong all to L_1. But no R_i can contain two of them. Indeed, if $\#w\#_{i_1}, \#w\#_{i_2} \in R_i$, for some $1 \leq i_1, i_2 \leq k+1, i_1 \neq i_2$, then, as R_i is confluent w.r.t. $\leq_f$, there is $z \in R_i$ such that $\#w\#_{i_1} \leq_f z$ and $\#w\#_{i_2} \leq_f z$. Then, from $\#w\#_{i_1} \leq_f z$, we get $z = \#w\#_{i_1}$ so $\#w\#_{i_2} \not\leq_f z$, a contradiction. Therefore L is empty.

Consequently, to decide whether or not L is empty, we decide whether or not L_1 as constructed above is k-confluent w.r.t. $\leq_f$, which is possible by our assumption, since L_1 is effectively in $\mathfrak{L}$. The claim is proved. ■

We show that also the inclusion problem is decidable in $\mathfrak{L}$, thus obtaining a contradiction. For, take $L_1, L_2 \in \mathfrak{L}, L_1 \subseteq \Sigma^*, L_2 \subseteq \Sigma^*$, and construct the language

$$L = \#L_1\#_0 \cup \bigcup_{i=1}^{k} (\#L_2\#_0\#_i)^+,$$

where $\#, \#_0, \#_1, \dots, \#_k \notin \Sigma$ are new letters. Then, by hypothesis, L is effectively in $\mathfrak{L}$.

Claim 2. L is k-confluent w.r.t. $\leq_f$ if and only if either $L_1 \subseteq L_2$ or else $1 \leq \operatorname{card}(L_1) \leq k$ and $L_2 = \emptyset$.

Proof of Claim 2. Suppose first that $L_1 \subseteq L_2$. Then

$$\begin{aligned}\text{DOWN}_{\leq_f}(L) &= \text{DOWN}_{\leq_f}(\#L_1\#_0) \cup \bigcup_{i=1}^{k} \text{DOWN}_{\leq_f}((\#L_2\#_0\#_i)^+) \\ &= \bigcup_{i=1}^{k} \text{DOWN}_{\leq_f}((\#L_2\#_0\#_i)^+)\end{aligned}$$

since

$$\text{DOWN}_{\leq_f}((\#L_1\#_0)) \subseteq \text{DOWN}_{\leq_f}(\#L_2\#_0\#_i) \subseteq \text{DOWN}_{\leq_f}((\#L_2\#_0\#_i)^+),$$

for any $1 \leq i \leq k$. As, clearly, $(\#L_2\#_0\#_i)^+$ is confluent w.r.t. $\leq_f$, we have, by Lemma 6.6.1, that $\text{DOWN}_{\leq_f}((\#L_2\#_0\#_i)^+)$ is confluent w.r.t. $\leq_f$. Thus $\text{DOWN}_{\leq_f}(L)$ if k-confluent w.r.t. $\leq_f$ and so, again by Lemma 6.6.1, L is k-confluent w.r.t. $\leq_f$.

Suppose now that $1 \leq \operatorname{card}(L_1) \leq k$ and $L_2 = \emptyset$. If, for some $1 \leq p \leq k$, $L_1 = \{w_1, w_2, \dots w_p\}$, then

$$L = \bigcup_{i=1}^{p} \{\#w_i\#_0\}.$$

As $\{\#w_i\#_0\}$ is confluent w.r.t. $\leq_f$, it follows that L is p-confluent w.r.t. $\leq_f$ hence also k-confluent w.r.t. $\leq_f$.

For the converse part, suppose that L is k-confluent w.r.t. $\leq_f$. Suppose also that $L_1 \not\subseteq L_2$ and take $w_1 \in L_1 - L_2$. If $L_2 \neq \emptyset$, then take $w_2 \in L_2$. It follows that

$$\{\#w_1\#_0, \#w_2\#_0\#_1, \#w_2\#_0\#_2, \dots, \#w_2\#_0\#_k\} \subseteq L$$

and a contradiction with the k-confluence w.r.t. $\leq_f$ of L is obtained as in the proof of Claim 1. Thus $L_2 = \emptyset$ and so

$$L = \#L_1\#_0.$$

If $\mathrm{card}(L_1) \geq k+1$, then take $z_1, z_2, \ldots, z_{k+1} \in L_1, z_i \neq z_j$, for any $i \neq j$. We have that

$$\{\#z_1\#_0, \#z_2\#_0, \ldots \#z_{k+1}\#_0\} \subseteq L$$

and again a contradiction with the k-confluence w.r.t. $\leq_f$ of L is obtained. Thus $\mathrm{card}(L_1) \leq k$. As $L_1 \not\subseteq L_2 = \emptyset$, we get also $\mathrm{card}(L_1) \geq 1$. The claim is proved. ■

Claim 3. The inclusion problem is decidable in $\mathcal{L}$.

Proof of Claim 3. Take two arbitrary language $L_1, L_2 \in \mathcal{L}$ and construct the language L as above. We give the following algorithm to decide whether or not $L_1 \subseteq L_2$ and prove the claim.

Algorithm 6.6.11.

(1) Decide whether or not $L_2 = \emptyset$ [the emptiness problem is decidable in $\mathcal{L}$ by Claim 1]. If yes, then go to step 2, else go to step 3.

(2) Decide whether or not $L_1 = \emptyset$ [possible by Claim 1]. If yes, then answer **yes** [obviously correct], else answer **no** [again correct].

(3) Decide whether or not L is k-confluent w.r.t. $\leq_f$ [the k-confluence problem is decidable in $\mathcal{L}$ by our assumption and L is effectively in $\mathcal{L}$]. If yes, then answer **yes** [correct by Claim 2], else answer **no** [correct also by Claim 2]. ■

Consequently, we have proved that the decidability of the k-confluence problem w.r.t. $\leq_f$ for languages in $\mathcal{L}$ would entail the decidability of the inclusion problem for languages in $\mathcal{L}$. Since the latter is undecidable in $\mathcal{L}$, it follows that also the former is undecidable in $\mathcal{L}$ and the result is proved. ■

As a direct corollary of Lemma 6.6.10, we obtain the undecidability of the k-confluence problem w.r.t. $\leq_f$ for context-free languages.

Theorem 6.6.12. *For any $k \geq 1$, it is undecidable whether or not an arbitrary context-free language is k-confluent w.r.t. $\leq_f$.*

We consider now the generalized confluence problem w.r.t. $\leq_f$ and give the following general undecidability result.

Lemma 6.6.13. *Let $\mathcal{L}$ be a family of languages effectively closed under union, catenation, and λ-free catenation closure such that all regular languages are in $\mathcal{L}$. If it is undecidable for an arbitrary language $L \in \mathcal{L}, L \subseteq \Sigma^*$, whether or not $L = \Sigma^*$, then the generalized confluence problem w.r.t. $\leq_f$ is undecidable for languages in $\mathcal{L}$.*

Proof. We argue by contradiction. Suppose that the generalized confluence problem w.r.t. $\leq_f$ is decidable in $\mathcal{L}$. Then we prove that it is decidable for an arbitrary language $L \in \mathcal{L}, L \subseteq \Sigma^*$, whether or not $L = \Sigma^*$. For, take an arbitrary such language L and construct the language

$$L_1 = \#\Sigma^*\#_1\#_2^*\#_1 \cup (\#L\#_1\#_2^*\#_1)^+,$$

where $\#, \#_1, \#_2 \notin \Sigma$ are new letters. Then, by hypothesis, $L_1 \in \mathcal{L}$.

Claim. L_1 is confluent w.r.t. $\leq_f$ in generalized sense if and only if $L = \Sigma^*$.

Proof of Claim. Suppose first that $L = \Sigma^*$. Then

$$L_1 = \#\Sigma^*\#_1\#_2^*\#_1 \cup (\#\Sigma^*\#_1\#_2^*\#_1)^+ = (\#\Sigma^*\#_1\#_2^*\#_1)^+.$$

Since $(\#\Sigma^*\#_1\#_2^*\#_1)^+$ is confluent w.r.t. $\leq_f$, L_1 is confluent w.r.t. $\leq_f$. It follows also that L_1 is confluent w.r.t. $\leq_f$ in generalized sense.

Conversely, suppose that L_1 is k-confluent w.r.t. $\leq_f$, for some $k \geq 1$, but $L \subset \Sigma^*$ and take $w \in \Sigma^* - L$. Put also

$$L_1 = \bigcup_{i=1}^{k} R_i,$$

for some $R_i \subseteq \Sigma^*, R_i$ confluent w.r.t. $\leq_f$. We have then

$$\bigcup_{i=0}^{k}\{\#w\#_1\#_2^i\#_1\} \subseteq L_1$$

and we can find an $i, 1 \leq i \leq k$, and $0 \leq i_1, i_2 \leq k, i_1 \neq i_2$, such that $\#w\#_1\#_2^{i_1}\#_1$, $\#w\#_1\#_2^{i_2}\#_1 \in R_i$. As R_i is confluent w.r.t. $\leq_f$, there is $z \in R_i$ such that $\#w\#_1\#_2^{i_1}\#_1 \leq_f z$ and $\#w\#_1\#_2^{i_2}\#_1 \leq_f z$. Because $w \notin L$, it follows that $z \notin (\#L\#_1\#_2^*\#_1)^+$, and so $z = \#x\#_1\#_2^j\#_1$, for some $x \in \Sigma^*, j \geq 0$. From $\#w\#_1\#_2^{i_1}\#_1 \leq_f \#x\#_1\#_2^j\#_1$, we obtain $w = x$ and $i_1 = j$. But, since $i_1 \neq i_2$, $\#w\#_1\#_2^{i_2}\#_1 \not\leq_f z$, a contradiction. It follows that $L = \Sigma^*$. As k above has been chosen arbitrarily, the claim is proved. ■

Hence, to decide whether or not $L = \Sigma^*$, we can decide whether or not L_1 is confluent w.r.t. $\leq_f$ in generalized sense, which is possible by our assumption, since L_1 is effectively in $\mathcal{L}$. But this contradicts the hypothesis. Therefore the generalized confluence problem w.r.t. $\leq_f$ is undecidable in $\mathcal{L}$ and the result is proved. ■

We get immediately the following result.

Theorem 6.6.14. *It is undecidable whether or not an arbitrary context-free language is confluent w.r.t. $\leq_f$ in generalized sense.*

6.7 Further research

We mention finally two open problems related with our considerations in this chapter. They focus on regular languages.

First, the confluence property came out of our study on the ways of obtaining languages of finite words from infinite words in Chapter 5. Therefore, of essential use in proving the decidability of the confluence problem with respect to the factor partial order for regular languages (Theorem 6.2.2) was the decidability result in Chapter 5 concerning the property of a language being the set of factors of an infinite word (Theorem 5.5.4). It might be then of interest to solve

Problem 6.7.1. Prove directly Theorem 6.2.2, without using Theorem 5.5.4.

The other problems concern generalizations of the result in Theorem 6.2.2.

Problem 6.7.2. For any fixed $k \geq 0$, is it decidable whether or not an arbitrary regular language is k-confluent w.r.t. $\leq_f$?

Problem 6.7.3. Is it decidable whether or not an arbitrary regular language is confluent w.r.t. $\leq_f$ in the generalized sense?

We believe that the answer to both of the latter two problems is affirmative and that they can be solved by considerations similar with those in Sections 5.4, 5.5, and 6.2.

Chapter 7

Well quasi orders

This chapter investigates down-sets associated to well quasi orders. Of particular language-theoretic interest is the quasi order $u \leq_s v$ (resp. $u \leq_\Psi v$) of u being a subword (resp. a Parikh subword) of v, as well as their inverses. The main decidability problems in the previous chapter are solved for these two quasi orders. Also, it is shown that a quasi order being a well quasi order is closely connected with arbitrary languages being confluent. Further, we establish a number of results about the regularity and effective regularity of the down-sets. Finally, we generalize the factor partial order and study the finite basis property of the relations obtained; the characterization theorem which is obtained is a generalization of Higman's theorem in the restricted form for words.

7.1 Problems

The results of Higman, [Hi], and Haines, [Hai], are often used in the theory of formal languages. In the particular form for words, Higman's theorem says that if we have a set S of words such that no two words in S are comparable with respect to $\leq_s$, then necessarily S is finite. Therefore, any language L whatsoever contains only a finite number of words minimal with respect to the partial order $\leq_s$. Based on similar ideas, Haines also showed that the languages

$$\begin{aligned}&\text{DOWN}_{\leq_s}(L) = \{w \mid w \leq_s u \text{ for some } u \in L\} \text{ and}\\ &\text{DOWN}_{\leq_s^{-1}}(L) = \{w \mid u \leq_s w \text{ for some } u \in L\}\end{aligned}$$

are regular for any language L. Of course, the languages mentioned are effectively regular only in exceptional cases.

Several aspects are to be investigated here. First we consider the problem of extending Haines' result. We prove that it still holds for any monotone well quasi order instead of the subwords partial order. Second, consider the effective regularity of these down-sets. Solving an open problem of Haines, van Leeuwen [vL] showed when it is possible to find effectively regular expressions in the case of superwords and gave a method to find the set of subwords for context-free languages. It remained open whether there exist families with a decidable emptiness problem but for which the regularity of the sets of subwords is not effective. We give a positive answer to this problem. Third, we give a generalization of van Leeuwen's result concerning superwords.

Continuing the investigations in the previous chapter, we consider here the confluence problem, mainly with respect to subwords and Parikh subwords, and the problem for a language of being the set of (Parikh) subwords of an infinite word. In what concerns the confluence, we get, unexpectedly, that for both relations the problem is decidable for context-free languages. For the other problem we obtain decidability for regular languages but undecidability in the context-free case. We show also that there is a close connection between the (generalized) confluence and the concept of well quasi order.

Finally, we generalize the factor partial order in a natural way. We obtain nondenumerably many different relations lying in between the factor and the subword relation. We then study the finiteness of the antichains of these relations and give a precise characterization of it. This result generalizes the particular case for words of Higman's theorem.

7.2 Well quasi orders

The concept of a well quasi order has been frequently discovered; see, for instance, [ErRa], [Hi], [Kr1], [Hai]. A complete account in this matter is given by [Kr2]. The definitions we use here are from [ChKa].

Let $\leq$ be a quasi order on Σ^*. A set $L \subseteq \Sigma^*$ is an **antichain** of $\leq$ if all elements in L are pairwise incomparable w.r.t. $\leq$, that is, for any $u, v \in L$, neither $u \leq v$ nor $v \leq u$. The quasi order $\leq$ is called **well founded** if there is no infinite descending sequence $w_1 \leq^{-1} w_2 \leq^{-1} w_3 \leq^{-1} \cdots$ such that, for no $i \geq 1, w_i \leq w_{i+1}$, where $\leq^{-1}$ denotes the inverse of $\leq$. If $\leq$ is well founded and any antichain of it is finite, then it is a **well quasi order**.

We mention some results on well quasi orders which we shall need later on. Some of them can be also found in [dLVa2] and [Lo].

The first one is a particular form of the well-known theorem by Higman [Hi] which easily implies the fact the the subword partial order is a well partial order. (A simple proof of this result is given by Conway [Co], cf. also [Lo].)

Theorem 7.2.1 (Higman [Hi]). *If L is a language such that any two words in L are incomparable w.r.t. the subword partial order $\leq_s$, then L is finite.*

Corollary 7.2.2. *The Parikh subword quasi order $\leq_\Psi$ is a well quasi order.*

Haines [Hai], seems to have been the first to notice that both sets, of subwords and of superwords, are regular for any language.

Theorem 7.2.3 (Haines [Hai]). *For any language L, both sets* $\text{DOWN}_{\leq_s}(L)$ *and* $\text{DOWN}_{\leq_s^{-1}}(L)$ *are regular.*

It may happen that regular expressions for $\text{DOWN}_{\leq}(L)$ and $\text{DOWN}_{\leq^{-1}}(L)$ cannot be found effectively. As noticed by van Leeuwen in [vL], if this is possible for a family of languages, then the emptiness problem is decidable for the same family of languages. The converse of this statement has been shown to be true in the case of $\text{DOWN}_{\leq_s^{-1}}(L)$ for families effectively closed under intersection with regular sets in [vL].

Theorem 7.2.4 (van Leeuwen [vL]). *For any family $\mathfrak{L}$ effectively closed under intersection with regular sets, one can effectively determine the down-set* $\text{DOWN}_{\leq_s^{-1}}(L)$ *for each $L \in \mathfrak{L}$ if and only if $\mathfrak{L}$ has a decidable emptiness problem.*

Solving an open problem of Haines [Hai], van Leeuwen [vL] proved also the following result which we shall need later on.

Theorem 7.2.5 (van Leeuwen [vL]). *For a context-free language L, one can effectively determine* $\text{DOWN}_{\leq_s}(L)$.

7.3 Subwords of infinite words

We deal in this section with languages consisting of subwords of infinite words. We prove that it is decidable whether or not a given regular language is the set of subwords of a right-, left-, or bi-infinite word. The same problem for context-free languages is undecidable.

We make the following convention which will be used in Sections 7.3 and 7.4: whenever we write $L \subseteq \Sigma^*$, Σ is supposed to be the minimal alphabet with this property.

Also, for a finite word $w \in \Sigma^*$, we denote by $Sub(w)$ the set of subwords of w, that is, $Sub(w) = \text{DOWN}_{\leq_s}(\{w\})$.

7.3.1 Infinite words

We consider first right-infinite words. The next lemma is a characterization for a language to be the set of subwords of a right-infinite word.

Lemma 7.3.1. *For $L \subseteq \Sigma^*$, denote by Σ_I the set of letters $a \in \Sigma$ such that*

$$L \cap \Sigma^* a$$

is infinite. Then $L \in \mathcal{F}^r_{\leq_s}$ if and only if
- *either $L = \Sigma^*$ and $\Sigma_I = \Sigma$,*
- *or else $L = Sub(w)\Sigma_I^*$ and $\emptyset \neq \Sigma_I \subset \Sigma$, for some $w \in \Sigma^M$, where*

$$M = \max_{a \in \Sigma - \Sigma_I} \max\{|w| \mid w \in L \cap \Sigma^* a\}.$$

Proof. Consider $L = \text{DOWN}_{\leq_s}(Pref(\alpha))$, for some $\alpha = \alpha_1\alpha_2\alpha_3 \ldots \in \Sigma^\omega$, $\alpha_i \in \Sigma$, for all $i \geq 1$.

Suppose first $\Sigma_I = \Sigma$ and prove that $L = \Sigma^*$. Take $w = a_1 a_2 \ldots a_n \in \Sigma^*$. If $n = 0$, then $w = \lambda \in L$. Assume $n \geq 1$. Since $a_1 \in \Sigma = \Sigma_I$, there is $i_1 \geq 1$ such that $\alpha_{i_1} = a_1$. Now, $a_2 \in \Sigma_I$, so $\alpha_{i_2} = a_2$, for some $i_2 > i_1$. Continuing inductively, we find $i_1 < i_2 < \cdots < i_n$ such that $\alpha_{i_k} = a_k$, for any $1 \leq k \leq n$. Therefore $w = \alpha_{i_1}\alpha_{i_2} \ldots \alpha_{i_n} \in \text{DOWN}_{\leq_s}(Pref(\alpha)) = L$.

Suppose now $\emptyset \neq \Sigma_I \subset \Sigma$. We prove by inclusion in both directions that

$$L = Sub(\alpha_1\alpha_2 \ldots \alpha_M)\Sigma_I^*.$$

Take $w = \alpha_{i_1}\alpha_{i_2} \ldots \alpha_{i_n} \in L, n \geq 1$. If $i_n \leq M$, then $w \in Sub(\alpha_1\alpha_2 \ldots \alpha_M)$ and if $M < i_1$, then, using an inductive argument, $w \in \Sigma_I^*$. The remaining possibility is $i_k \leq M < i_{k+1}$, for some $1 \leq k \leq n-1$. Then $\alpha_{i_1}\alpha_{i_2} \ldots \alpha_{i_k} \in Sub(\alpha_1\alpha_2 \ldots \alpha_M)$ and $\alpha_{i_{k+1}}\alpha_{i_{k+2}} \ldots \alpha_{i_n} \in \Sigma_I^*$, hence $w \in Sub(\alpha_1\alpha_2 \ldots \alpha_M)\Sigma_I^*$. One inclusion is proved. For the converse inclusion, it is enough to see that $\Sigma_I^* \subseteq \text{DOWN}_{\leq_s}(Pref(\alpha_{M+1}\alpha_{M+2} \ldots))$.

The converse implication is clear since, in the second case, $\Sigma_I \neq \emptyset$. ■

From Lemma 7.3.1 we get the following corollary.

Corollary 7.3.2. *It is decidable whether or not an arbitrary regular language is in the family $\mathcal{F}^r_{\leq_s}$.*

Proof. Consider a regular language $R \subseteq \Sigma^*$. If R is finite, then $R \notin \mathcal{F}^r_{\leq_s}$, so suppose R infinite. Using the notations in Lemma 7.3.1, we can effectively determine Σ_I. Now, if $\Sigma = \Sigma_I$, then $R \in \mathcal{F}^r_{\leq_s}$ if and only if $R = \Sigma^*$ which is decidable. If $\Sigma_I \subset \Sigma$, then $R \in \mathcal{F}^r_{\leq_s}$ if and only if $R = Sub(w)\Sigma_I^*$, for some $w \in \Sigma^M$. Since M is effectively computable and there are only finitely many words in Σ^M, this is also decidable. ■

By left-right duality, we get the next result.

Corollary 7.3.3. *It is decidable whether or not an arbitrary regular language is in the family $\mathcal{F}^l_{\leq_s}$.*

We now turn our attention to the bi-infinite case and give a characterization for a language to be the set of subwords of a bi-infinite word.

Lemma 7.3.4. *For $L \subseteq \Sigma^*$, denote by Δ_E, Δ_F the sets of pairs (a, b) of letters in Σ such that*

$$L \cap a\Sigma^* \cap \Sigma^* b$$

is empty or finite non-empty, respectively. Then $L \in \mathcal{F}^{bi}_{\leq_s}$ if and only if

– *either $L = \Sigma^*$ and $\Delta_E = \Delta_F = \emptyset$,*

– *or $L = \Sigma_1^*\Sigma_2^*$ and $\Delta_E \neq \emptyset, \Delta_F = \emptyset$, for some non-empty $\Sigma_1, \Sigma_2 \subset \Sigma$,*

– *or else $L = \Sigma_1^* Sub(w)\Sigma_2^*$ and $\Delta_F \neq \emptyset$, for some non-empty $\Sigma_1, \Sigma_2 \subset \Sigma$, $w \in \Sigma^M$, where*

$$M = \max_{(a,b)\in\Delta_F} \{|w| \mid w \in L \cap a\Sigma^* \cap \Sigma^* b\}.$$

Proof. If one of the three conditions is fulfilled, then clearly $L \in \mathcal{F}^{bi}_{\leq_s}$.

Consider then $L = \text{DOWN}_{\leq_s}(Fact(\alpha))$, $\alpha = \ldots\alpha_{-2}\alpha_{-1}\alpha_0\alpha_1\alpha_2\ldots \in {}^{\omega}\Sigma^{\omega}$.

Suppose first $\Delta_E = \Delta_F = \emptyset$ and prove $L = \Sigma^*$. For any $a \in \Sigma$, denote

$$m_+(a) = \sup\{n \mid n \geq 0, \alpha_n = a\},$$
$$m_-(a) = \sup\{-n \mid n \leq 0, \alpha_n = a\}.$$

We have that, for any $a \in \Sigma$, at least one of $m_+(a)$ and $m_-(a)$ is infinite. Moreover, if one of the two suprema is finite for some $a \in \Sigma$, then, for any $b \in \Sigma$, the other supremum corresponding to b is infinite. Therefore, clearly, $L = \Sigma^*$.

Suppose now that $\Delta_E \neq \emptyset, \Delta_F = \emptyset$ and construct the sets

$$\Sigma_1 = \{a \in \Sigma \mid m_-(a) = \infty\},$$
$$\Sigma_2 = \{a \in \Sigma \mid m_+(a) = \infty\}.$$

It should be clear that $\Sigma_1 \cup \Sigma_2 = \Sigma$ and, if $ab \in L$, then either $a \in \Sigma_1$ or $b \in \Sigma_2$.

We claim that $L = \Sigma_1^*\Sigma_2^*$. The inclusion $\Sigma_1^*\Sigma_2^* \subseteq L$ is clear. For the converse, take $a_1a_2\ldots a_n \in L$. Then, for any $1 \leq i \leq n$, either $m_-(a_i) = \infty$ or $m_+(a_{i+1}) = \infty$. If $m_-(a_i) = \infty$ for all $1 \leq i \leq n$, then $a_1a_2\ldots a_n \in \Sigma_1^*$. Similarly, $m_+(a_i) = \infty$ for all $1 \leq i \leq n$ implies $a_1a_2\ldots a_n \in \Sigma_2^*$. The remaining possibility is that there exists $1 \leq i_0 \leq n-1$ such that $m_-(a_i) = \infty$ for $1 \leq i \leq i_0$ and $m_+(a_i) = \infty$ for $i_0 + 1 \leq i \leq n$. Then $a_1a_2\ldots a_{i_0} \in \Sigma_1^*$, $a_{i_0+1}a_{i_0+2}\ldots a_n \in \Sigma_2^*$, hence $a_1a_2\ldots a_n \in \Sigma_1^*\Sigma_2^*$. The equality is proved.

Finally, suppose $\Delta_F \neq \emptyset$ and M is reached for $w = bu = vc \in L$, $b, c \in \Sigma, u, v \in \Sigma^*$, $|w| = M$. Then there is $n_0 \in \mathbb{Z}$ such that $\alpha_{n_0} = b, \alpha_{n_0+M-1} = c$. Put

$$\Sigma_1 = \{a \in \Sigma \mid \text{ there is } n \leq n_0 - 1 \text{ such that } \alpha_n = a\},$$
$$\Sigma_2 = \{a \in \Sigma \mid \text{ there is } n \geq n_0 + M \text{ such that } \alpha_n = a\}.$$

It is easy to see that $m_-(a) = \infty$ for $a \in \Sigma_1$ and $m_+(a) = \infty$ for $a \in \Sigma_2$. It follows that $L = \Sigma_1^* Sub(w) \Sigma_2^*$. The proof is concluded. ■

From Lemma 7.3.4 we get

Corollary 7.3.5. *It is decidable whether or not an arbitrary regular language is in the family* $\mathcal{F}^{bi}_{\leq_s}$.

Proof. All conditions in Lemma 7.3.4 are decidable for regular languages. ■

From Corollaries 7.3.2, 7.3.3, and 7.3.5, we obtain the main result of this section.

Theorem 7.3.6. *It is decidable whether or not an arbitrary regular language is in the family* $\mathcal{F}_{\leq_s}$.

The same problem is undecidable for context-free languages.

Theorem 7.3.7. *It is undecidable whether or not an arbitrary context-free language is in the family* $\mathcal{F}_{\leq_s}$.

Proof. Consider an arbitrary Turing machine M and its set of invalid computations L_M. It is known (see, for instance, [HoUl]) that L_M is context-free. We claim that L_M belongs to the family $\mathcal{F}_{\leq_s}$ if and only if its complement is empty.

One implication is obvious. For the other one, suppose that $L_M \in \mathcal{F}_{\leq_s}$. It follows that any subword of a word in L_M is also in L_M. Thus, any subword of an invalid computation is also an invalid computation. Since any valid computation is a subword of an invalid one, it follows that there is actually no valid computation, hence $\overline{L_M} = \emptyset$.

Consequently, $L_M \in \mathcal{F}_{\leq_s}$ if and only if the set of valid computations of M is empty which is equivalent to the fact that the language accepted by M is empty which is undecidable. The result follows. ■

Observe that, rather than containing new aspects about the family $\mathcal{F}_{\leq_s}$, Theorem 7.3.7 is another manifestation of the undecidability of the equality between a given context-free language and Σ^*. Theorem 7.3.7, as opposed to Theorem 7.3.6, shows the power of the devices defining context-free languages, not the power of the languages themselves.

7.3.2 Confluence for subwords

We prove now that the confluence problem with respect to the subword partial order is decidable for regular langauges. It turns out, unexpectedly, that this can be extended to the context-free case.

Theorem 7.3.8. *It is decidable whether or not an arbitrary regular language is confluent w.r.t.* $\leq_s$.

Proof. Consider a regular language L and construct the language

$$K = \{u\#v^R \mid u, v \in L \text{ such that there is } w \in L \text{ with } u \leq_s w, v \leq_s w\}.$$

It is easy to see that L is confluent w.r.t. $\leq_s$ if and only if $K = L\#L^R$.

Claim 1. K is context-free.

Proof of Claim 1. A push-down automaton $\mathcal{A}$ for K works as follows.
– any input without the marker $\#$ or with at least two markers is rejected.
– for an input of the form $u\#v$, $\mathcal{A}$ does:
– when reading u, it pushes nondeterministically in the stack a word $w \in L$ suah that $u \leq_s w$;
– after the marker $\#$, $\mathcal{A}$ checks whether or not $v \leq_s w^R$ (which is equivalent to $v^R \leq_s w$).

A formal construction for $\mathcal{A}$ follows. Suppose that $\mathcal{B} = (\Sigma, Q, \delta_{\mathcal{B}}, q_0, F)$ is a complete finite automaton recognizing L. Then we construct

$$\mathcal{A} = (\Sigma, Q \times Q \cup Q, \Gamma = \Sigma \cup \{Z\}, \delta_{\mathcal{A}}, (q_0, q_0), Z, \emptyset)$$

with the transition mapping $\delta_{\mathcal{A}}$ given by

$$\begin{aligned}
&\delta_{\mathcal{A}}((q_1, q_2), a, Y) = \{(\delta_{\mathcal{B}}(q_1, a), \delta_{\mathcal{B}}(q_2, a)), aY\}, q_1, q_2 \in Q, a \in \Sigma, Y \in \Gamma,\\
&\delta_{\mathcal{A}}((q_1, q_2), \lambda, Y) = \{((q_1, \delta_{\mathcal{B}}(q_2, a)), aY) \mid a \in \Sigma\}, q_1, q_2 \in Q, Y \in \Gamma,\\
&\delta_{\mathcal{A}}((q_1^F, q_2^F), \#, Y) = \{(q_0, Y)\}, q_1^F, q_2^F \in F, Y \in \Gamma,\\
&\delta_{\mathcal{A}}(q_1, a, a) = \{(\delta_{\mathcal{B}}(q_1, a), \lambda)\}, q_1 \in Q, a \in \Sigma,\\
&\delta_{\mathcal{A}}(q_1, \lambda, a) = \{(q_1, \lambda)\}, q_1 \in Q, a \in \Sigma,\\
&\delta_{\mathcal{A}}(q_1^F, \lambda, Z) = \{(q_1^F, \lambda)\}, q_1^F \in F.
\end{aligned}$$

Clearly, for any $w \in \Sigma^*$, we have

$$((q_0, q_0), w, Z) \overset{*}{\vdash} (q_1, \lambda, \lambda) \text{ iff } w \in K.$$

The claim is proved. ∎

Claim 2. $K = L\#L^R$ if and only if $\text{DOWN}_{\leq_s}(K) = \text{DOWN}_{\leq_s}(L\#L^R)$.

Proof of Claim 2. In one direction the implication is obvious. Suppose that $\text{DOWN}_{\leq_s}(K) = \text{DOWN}_{\leq_s}(L\#L^R)$. We have anyway $K \subseteq L\#L^R$. Take $u\#v^R \in L\#L^R$. We have that $L\#L^R \subseteq \text{DOWN}_{\leq_s}(L\#L^R) = \text{DOWN}_{\leq_s}(K)$, so $u\#v^R \in \text{DOWN}_{\leq_s}(K)$. Thus, for some $w \in K, u\#v^R \leq_s w$. Let us put $w = x\#y^R$. It follows that $u \leq_s x$ and $v \leq_s y$. But now, since $x\#y^R \in K$, there must be a word $z \in L$ with $x \leq_s z, y \leq_s z$, by the definition of K. We get $u \leq_s z, v \leq_s z$ and so $u\#v^R \in K$, which proves the claim. ■

Claim 3. It is decidable whether or not $K = L\#L^R$.

Proof of Claim 3. Since the languages $L\#L^R$ and K are context-free, we obtain by Theorem 7.2.5 that the down-sets $\text{DOWN}_{\leq_s}(L\#L^R)$ and $\text{DOWN}_{\leq_s}(K)$ are effectivelly regular. As the equivalence problem is decidable for regular languages, using Claim 2, we are done. ■

Consequently, it is decidable whether or not L is confluent w.r.t. $\leq_s$ and the proof is concluded. ■

The result in Theorem 7.3.8 can be extended to the context-free case.

Corollary 7.3.9. *It is decidable whether or not an arbitrary context-free language is confluent w.r.t. $\leq_s$.*

Proof. Take an arbitrary context-free language L. By Lemma 6.1.1, L is confluent w.r.t. $\leq_s$ if and only if $\text{DOWN}_{\leq_s}(L)$ is confluent w.r.t. $\leq_s$. By Theorem 7.2.5, the down-set $\text{DOWN}_{\leq_s}(L)$ is effectively regular and thus its confluence w.r.t. $\leq_s$ can be decided by Theorem 7.3.8. ■

Remark. Since, by Theorem 6.3.2 and Corollary 7.3.9, the confluence problem w.r.t. any of the partial orders $\leq_p$ and $\leq_s$ is decidable for context-free languages, it follows that the result in Lemma 6.3.3 cannot be extended for any partial order $\leq$ such that either $\leq \subseteq \leq_f$ or $\leq_f \subseteq \leq$. In this sense, Lemma 6.3.3 characterizes the factor partial order $\leq_f$.

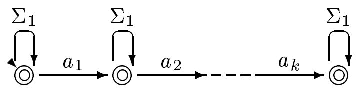
7.4 Parikh subwords

We consider in this section the Parikh subword quasi order. We prove that the confluence problem w.r.t. $\leq_\Psi$ is decidable for regular languages. From this, we deduce that it is also decidable whether or not an arbitrary regular language is in the family $\mathcal{F}_{\leq_\Psi}$. Using the effective regularity of down-sets w.r.t. $\leq_\Psi$ for context-free languages, we prove that the confluence problem w.r.t. $\leq_\Psi$ is decidable even in the context-free case.

7.4.1 Chain automata

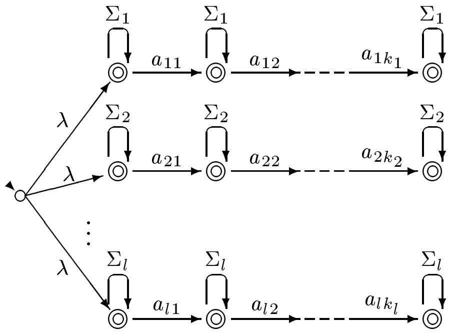
We now show that $\text{DOWN}_{\leq_\Psi}(R)$ is effectively regular for any language R and a finite automaton of a special form can be constructed for it. We need some definitions.

A **chain automaton** over the alphabet Σ is a finite nondeterministic (incomplete) automaton defined by the graph



where $k \geq 0, a_i \in \Sigma, 1 \leq i \leq k, \Sigma_1 \subseteq \Sigma$, and $\Sigma_1 \cap \{a_1, a_2, \ldots, a_k\} = \emptyset$. Observe that all states are final and Σ_1 may be empty.

A **multichain automaton** is defined similarly:



The notation is the same as before: each Σ_i is a subset of Σ and each a_{ij} is a letter of Σ. Also $\Sigma_i \cap \{a_{i1}, a_{i2}, \ldots, a_{ik_i}\} = \emptyset$, for any $1 \leq i \leq l$, and all states, except the initial state, are final.

A multichain automaton is **chain-reduced** if none of its chains is superfluous in the sense that every word accepted by the chain is accepted by some of the other chains as well.

The following lemma is obvious, since inclusion is decidable for regular languages.

Lemma 7.4.1. *For an arbitrarily given multichain automaton, an equivalent chain-reduced multichain automaton can be effectively constructed.*

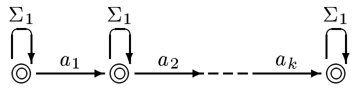
Lemma 7.4.2. *For a given regular language R, a chain-reduced multichain automaton accepting the language* $\text{DOWN}_{\leq_\Psi}(R)$ *can be effectively constructed.*

Proof. The proof is by induction on the regular expression defining R. It suffices to construct a multichain automaton for $\text{DOWN}_{\leq_\Psi}(R)$; the chain-reduction can be carried out using the preceding lemma.

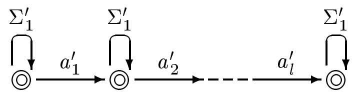
The basis of the induction is clear: the statement holds if R is a letter or $\emptyset$. (Observe that $\lambda = \emptyset^*$.) We make the inductive hypothesis: the statement holds true if R is defined by a regular expression α or by a regular expression β. We show that it holds true if R is defined by any of $(\alpha + \beta), (\alpha\beta)$, or α^*.

As regards union, we only have to combine the two multichain automata in an obvious fashion. (Recall that at this stage we do not have to worry about the result being chain-reduced.)

Consider $(\alpha\beta)$. The multichain automaton for it is constructed as follows. Let



be a chain in the automaton for α and



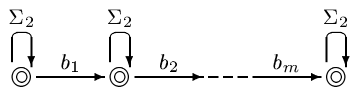
a chain in the automaton for β. Put

$$\Sigma_2 = \Sigma_1 \cup \Sigma_1'$$

and

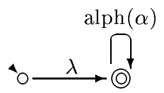
$$\{b_1, b_2, \dots, b_m\} = (\{a_1, a_2, \dots, a_k\} \cup \{a_1', a_2', \dots, a_l'\}) - \Sigma_2.$$

Then



as well as every permutation of it, is a chain in the new automaton.

Finally, consider α^*. Let $\mathrm{alph}(\alpha) \subseteq \Sigma$ be the set of letters appearing in α. The multichain automaton for α^* is simply



■

From Lemma 7.4.2 and Theorem 7.2.5, we deduce the following corollary.

Corollary 7.4.3. *For any context-free language L, the down-set* $\mathrm{DOWN}_{\leq_\Psi}(L)$ *is effectively regular.*

7.4.2 A direct algorithm

We give here a direct algorithm for constructing a finite automaton for the set of Parikh subwords of a given regular language. We notice that the same problem for subwords is very simple. As pointed out by [vL], it is sufficient to add, for each transition $p \xrightarrow{a} q$ in the automaton for the given regular language R, a transition $p \xrightarrow{\lambda} q$ and the obtained finite automaton (with λ-moves) accepts the down-set $\text{DOWN}_{\leq_s}(R)$.

The next algorithm performs the corresponding task for Parikh subwords. The difference with respect to Lemma 7.4.2 is that the algorithm works directly and not by induction.

Algorithm 7.4.4.
Input: a deterministic finite automaton $\mathcal{A}$ accepting the regular language R.
Output: a finite automaton with λ-moves $\mathcal{A}_\Psi$ accepting the regular language $\text{DOWN}_{\leq_\Psi}(R)$. (Notice that a deterministic finite automaton for $\text{DOWN}_{\leq_\Psi}(R)$ can be constructed from $\mathcal{A}_\Psi$ using the usual procedure; see, for instance, [HoUl].)

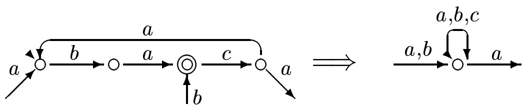
(1) For any cycle C of $G(\mathcal{A})$ do:

(1.1) put all states of C together into one, say q;

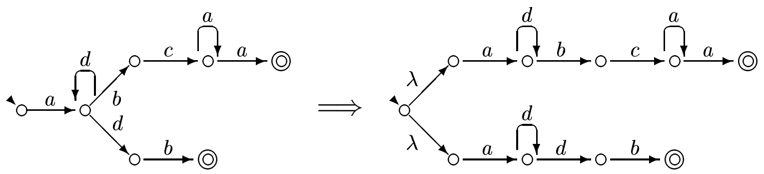
(1.2) put a loop on q labeled by all letter in C;

(1.3) q is initial if the initial state is in C and final if a final state is in C;

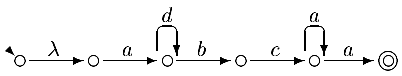
(1.4) if p is a state in C, then any incoming (outcoming) transition in (from) p from (to) a state not in C is preserved as it is, just that p is replaced by q; see the example below:



(2) By making copies of the states, construct a new equivalent automaton in which all simple paths (that is, paths which do not autointersect) from the initial state to a final one are disjoint with respect to both states and transitions; a transition labeled λ is added at the beginning of each simple path; see the example below:

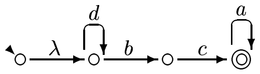


(3) for any simple path P from the initial state to a final one, as below,

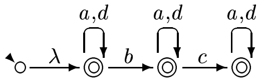
P: 

do:

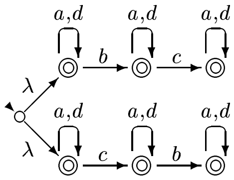
(3.1) for any $p \xrightarrow{a} q$ in P such that there is a loop labeled a on a state in P, put p and q together; the new state inherits the final-state character: the obtained simple path from P in our example is P_1 below;

P_1:


(3.2) for any loop on a state of P_1, add this loop to any state of P_1, then make all non-initial states final; we obtain P_2 below;

P_2:


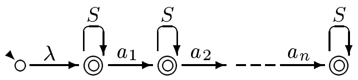
(3.3) make a new distinct simple path for any distinct permutation of P_2; see below;



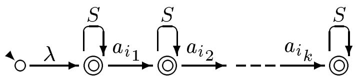
(4) Remove all redundant simple paths. The obtained automaton is the required $\mathcal{A}_\Psi$.

Theorem 7.4.5. *The above algorithm is correct, that is, if the automaton computed by the algorithm is $\mathcal{B}$, then $L(\mathcal{B}) = \text{DOWN}_{\leq_\Psi}(R)$.*

Proof. Take first $x \in \text{DOWN}_{\leq_\Psi}(R)$. Then $x \leq_\Psi w$, for some $w \in R$. The word w is accepted by $\mathcal{B}$ using a subautomaton as below:



where $a_i \in \Sigma, 1 \leq i \leq n$, and $S \subseteq \Sigma$. Denote by y the word obtained from x by removing all letters not in $\text{alph}(x) \cap \{a_1, a_2, \ldots, a_n\}$; put $y = a_{i_1} a_{i_2} \cdots a_{i_k}$, $a_{i_j} \in \Sigma$. Then $\mathcal{B}$ contains the subautomaton



and, obviuosly, x is accepted by it, so $x \in L(\mathcal{B})$.

Conversely, take $x \in L(\mathcal{B})$ and assume x accepted by the subautomaton of $\mathcal{B}$ in the first picture of the proof. Consider also y constructed as above.

From the algorithm, there is a simple path P in $\mathcal{B}$ from the initial state to a final one such that:

(i) for any letter $a_i, 1 \leq i \leq n$, a_i appears among the letters of P at least as many times as it appears among $a_1, a_2, \ldots, a_n$;

(ii) any letter in S labels a cycle on a state in P.

Hence, there is w accepted by $\mathcal{B}$ such that $y \leq_\Psi w$ and $|w|_a$, for any $a \in S$, is arbitrarily high. Hence $x \leq_\Psi w$ and the proof is concluded. ■

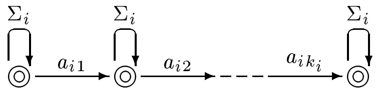
A problem connected with the above construction is stated at the end of the chapter; see Problem 7.8.4.

7.4.3 Confluence

We now prove that the confluence problem with respect to Parikh subwords is decidable for regular languages and that this result extends to the context-free case.

Theorem 7.4.6. *It is decidable whether or not an arbitrary regular language is confluent w.r.t.* $\leq_\Psi$.

Proof. Let R be a regular language. Using Lemma 7.4.2, we construct a chain-reduced multichain automaton accepting the language $\text{DOWN}_{\leq_\Psi}(R)$. Suppose that this automaton consists of n chains, $c_1, c_2, \ldots, c_n$, and, for any $1 \leq i \leq n, c_i$ is defined by the graph



where $\Sigma_i = \{b_{i1}, b_{i2}, \ldots, b_{il_i}\}$.

Notice that each c_i is uniquely identified by the word $w_i = a_{i1}a_{i2}\ldots a_{ik_i}$ and the set Σ_i.

Claim. $\text{DOWN}_{\leq_\Psi}(R)$ is confluent w.r.t. $\leq_\Psi$ if and only if, for any $1 \leq i, j \leq n$, $\Sigma_i = \Sigma_j$ and the words w_i and w_j are a permutation of each other.

Proof of Claim. If the conditions on c_i's in the statement of our claim are fulfilled, then clearly $\text{DOWN}_{\leq_\Psi}(R)$ is confluent w.r.t. $\leq_\Psi$.

Conversely, suppose that $\text{DOWN}_{\leq_\Psi}(R)$ is confluent w.r.t. $\leq_\Psi$ and consider two arbitrary $i, j, 1 \leq i, j \leq n$. Denote

$$M = \max_{1 \leq i \leq n} |w_i| + 1$$

and consider the words

$$u_i = w_i b_{i1}^M b_{i2}^M \ldots b_{il_i}^M,$$
$$u_j = w_j b_{j1}^M b_{j2}^M \ldots b_{jl_j}^M.$$

Since $\text{DOWN}_{\leq_\Psi}(R)$ is confluent w.r.t. $\leq_\Psi$, there is $w \in \text{DOWN}_{\leq_\Psi}(R)$ such that $u_i \leq_\Psi w, u_j \leq_\Psi w$. Suppose that w is recognized by the chain c_m, for some $1 \leq m \leq n$. From the choice of M it follows that any word $b_{ir}^M, 1 \leq r \leq l_i$, must be generated, at least partially, in a loop. Thus $\Sigma_i \subseteq \Sigma_m$ and we claim that $\Sigma_i = \Sigma_m$. If this is not the case, then take $b \in \Sigma_m - \Sigma_i$. There is a permutation of w, say w', such that $u_i \leq_s w'$. Since the word $w'b^M$ belongs to $\text{DOWN}_{\leq_\Psi}(R)$, it is recognized by a chain different from c_i, say c_r. It follows that any word recognized by c_i is recognized by c_r as well, a contradiction. Therefore $\Sigma_i = \Sigma_m$.

Now $w_i \leq_\Psi w_m$ and it can be proved by contradiction as above that w_i and w_m must be a permutation of each other.

The same reasoning can be done with k instead of i. Thus, we have $\Sigma_i = \Sigma_j$ and w_i is a permutation of w_j. The claim is proved.

Since the conditions on c_i's in our claim above can be effectively checked, it follows that it can be decided whether or not $\text{DOWN}_{\leq_\Psi}(R)$ is confluent w.r.t. $\leq_\Psi$. As, by Theorem 6.1.1, R and $\text{DOWN}_{\leq_\Psi}(R)$ are simultaneously confluent w.r.t. $\leq_\Psi$, the proof is concluded. ■

We can prove now the following result.

Theorem 7.4.7. *It is decidable whether or not an arbitrary context-free language is confluent w.r.t. $\leq_\Psi$.*

Proof. Let L be a context-free language. By Corollary 7.4.3, the down-set $\text{DOWN}_{\leq_\Psi}(L)$ is effectively regular and by Theorem 6.1.1, L is confluent w.r.t. $\leq_\Psi$ if and only if $\text{DOWN}_{\leq_\Psi}(L)$ is confluent w.r.t. $\leq_\Psi$, which is decidable by Theorem 7.4.6. ■

Remark. The result in Theorem 7.4.7 is just a manifestation of the fact that, for any context-free language L, a letter equivalent regular language R can be effectively constructed. Indeed, in this case $\text{DOWN}_{\leq_\Psi}(L) = \text{DOWN}_{\leq_\Psi}(R)$ and hence any property decidable for the sets of Parikh subwords of regular languages is decidable also for the sets of Parikh subwords of context-free languages.

We present next a general undecidability result concerning the confluence problem with respect to restrictions of the Parikh subword quasi order.

Theorem 7.4.8. *Let $\leq$ be a quasi order on Σ^* such that $\leq \subseteq \leq_\Psi$ and $\mathfrak{L}$ a family of languages which is effectively closed under union with singletons and catenation with simbols such that the emptiness problem is undecidable for languages in $\mathfrak{L}$. Then the confluence problem w.r.t. $\leq$ is undecidable for languages in $\mathfrak{L}$.*

Proof. Consider a language $L \subseteq \Sigma^*, L \in \mathcal{L}$, and two new symbols $\#_1, \#_2 \notin \Sigma$. Construct the language

$$L_1 = L\#_1 \cup \{\#_2\}.$$

We have that L_1 is effectively in $\mathcal{L}$ and we claim that L_1 is confluent w.r.t. $\leq$ if and only if $L = \emptyset$. One implication is clear. For the converse one, suppose that $L \neq \emptyset$ and take $w \in L$. Since L_1 is confluent w.r.t.$\leq$, it follows that there must be an word $v \in L_1$ such that $w\#_1 \leq v$ and $\#_2 \leq v$. Now, either $v \in L\#_1$ and $\#_2 \leq v$ does not hold, or else $v = \#_2$ and $w\#_1 \leq v$ cannot hold. In any case a contradiction is obtained. Because $\mathcal{L}$ has an undecidable emptiness problem, our theorem is proved. ■

7.4.4 Infinite words

Notice that it can be proved as in Theorem 7.4.6 that it is decidable whether or not an arbitrary regular language belongs to the family $\mathcal{F}_{\leq_\Psi}$ but we shall do this using the following lemma which establishes a connection between the property of a language being confluent w.r.t. $\leq_\Psi$ and belonging to the family $\mathcal{F}_{\leq_\Psi}$.

Lemma 7.4.9. *A language $L \subseteq \Sigma^*$ is in the family $\mathcal{F}_{\leq_\Psi}$ if and only if the following conditions are fulfilled:*
(i) L is infinite;
(ii) $L = \text{DOWN}_{\leq_\Psi}(L)$;
(iii) L is confluent w.r.t. $\leq_\Psi$.

Proof. If $L \in \mathcal{F}_{\leq_\Psi}$, then it is easy to see that L is infinite and $L = \text{DOWN}_{\leq_\Psi}(L)$. To show that L is confluent w.r.t. $\leq_\Psi$, put

$$L = \text{DOWN}_{\leq_\Psi}(Fact(\alpha)),$$

for some $\alpha \in {}^\omega\Sigma \cup \Sigma^\omega \cup {}^\omega\Sigma^\omega$ and take $u, v \in L$. Then $u = \alpha_{i_1}\alpha_{i_2}\ldots\alpha_{i_r}$, $v = \alpha_{j_1}\alpha_{j_2}\ldots\alpha_{j_s}$, $\alpha_{i_k}, \alpha_{j_l} \in \Sigma$, and, if $w = \alpha_{k_1}\alpha_{k_2}\ldots\alpha_{k_t}$, $\{k_1, \ldots, k_t\} = \{i_1, \ldots, i_r\} \cup \{j_1, \ldots, j_s\}$, then $w \in L, u \leq_\Psi w$, and $v \leq_\Psi w$.

Let us prove the converse implication. Suppose $L \subseteq \Sigma^* = \{a_1, a_2, \ldots a_n\}^*$ with the three properties (i)–(iii) in the statement.

Consider the projections $\pi_i : \Sigma^* \longrightarrow \{a_i\}^*, 1 \leq i \leq n$, given by

$$\pi_i(a_j) = \begin{cases} a_i, & \text{if } i = j, \\ \lambda, & \text{if } i \neq j. \end{cases}$$

We put

$$\Sigma_1 = \{a_i \in \Sigma \mid \pi_i(L) \text{ is infinite }\} = \{a_{j_1}, a_{j_2}, \ldots, a_{j_k}\},$$
$$\Sigma_2 = \Sigma - \Sigma_1 = \{a_{i_1}, a_{i_2}, \ldots, a_{i_l}\},$$

for some $k \geq 1, l \geq 0$, and $1 \leq j_r \leq n, 1 \leq i_s \leq n$, for any $1 \leq r \leq k, 1 \leq s \leq l$. (Notice that, because L is infinite, Σ_1 is non-empty.)

Consider also

$$\max_{w \in \pi_{i_s}(L)} |w| = m_s < \infty, \text{ for any } 1 \leq s \leq l.$$

We claim that

$$L = \text{DOWN}_{\leq_\Psi}(Pref(a_{i_1}^{m_1} a_{i_2}^{m_2} \ldots a_{i_l}^{m_l}(a_{j_1} a_{j_2} \ldots a_{j_k})^\omega)).$$

The part "$\subseteq$" is clear since no word in L can contain more than m_s occurrences of the symbol a_{i_s}, for any $1 \leq s \leq l$.

In order to prove the converse inclusion, "$\supseteq$", consider a word w in the right hand member of the equality to be proved. Now, for any $1 \leq s \leq l$, there is a word $w_{i_s} \in L$ with $|w_{i_s}|_{a_{i_s}} \geq |w|_{a_{i_s}}$, since there are in L words containing m_s occurrences of a_{i_s} and w cannot contain more. Also, for any $1 \leq r \leq k$, there is a word $w_{j_r} \in L$ with $|w_{j_r}|_{a_{j_r}} \geq |w|_{a_{j_r}}$, because there are in L words containing arbitrarily many occurrences of a_{j_r}. Since L is confluent w.r.t. $\leq_\Psi$, we get inductively a word $w' \in L$ such that

$$\begin{aligned} w_{i_s} &\leq_\Psi w', \quad 1 \leq s \leq l, \\ w_{j_r} &\leq_\Psi w', \quad 1 \leq r \leq k, \end{aligned}$$

and so $w \leq_\Psi w'$. Thus $w \in \text{DOWN}_{\leq_\Psi}(L) = L$ and the equality is proved. ∎

Remark. A similar result does not hold for the subword partial order $\leq_s$. Indeed, the language $L = a^*b^*c^*$ is infinite, $L = \text{DOWN}_{\leq_s}(L)$, and it is confluent w.r.t. $\leq_s$. But, clearly, L is not the set of subwords of a right-, left-, or bi-infinite word.

From the previous results in this section, we obtain

Theorem 7.4.10. *It is decidable whether or not an arbitrary regular language belongs to the family $\mathcal{F}_{\leq_\Psi}$.*

Proof. Let R be a regular language. If R is finite, then it does not belong to the family $\mathcal{F}_{\leq_\Psi}$. Suppose then that R is infinite. By Lemma 7.4.9, if $R \neq \text{DOWN}_{\leq_\Psi}(R)$, then $R \notin \mathcal{F}_{\leq_\Psi}$. Since, using Lemma 7.4.2, it can be decided whether or not $R = \text{DOWN}_{\leq_\Psi}(R)$, we can restrict the problem to languages R with $R = \text{DOWN}_{\leq_\Psi}(R)$. Now, by Lemma 7.4.9, $R \in \mathcal{F}_{\leq_\Psi}$ if and only if R is confluent w.r.t. $\leq_\Psi$, which is decidable by Theorem 7.4.6. ∎

From the proof of Lemma 7.4.9 we get another consequence.

Corollary 7.4.11. $\mathcal{F}^l_{\leq_\Psi} = \mathcal{F}^r_{\leq_\Psi} = \mathcal{F}^{bi}_{\leq_\Psi} = \mathcal{F}_{\leq_\Psi}$.

Proof. We show that, for any language $L \subseteq \Sigma^*, L$ belongs to any of the four families in the statement if and only if the conditions (i)-(iii) in Lemma 7.4.9 are fulfilled. As seen in the proof there, if L belongs to any of the families, then (i)-(iii) are fulfilled. Conversely, using the notations in the proof of Lemma 7.4.9, we have

$$\begin{aligned} L &= \mathrm{DOWN}_{\leq_\Psi}(Pref(a_{i_1}^{m_1}a_{i_2}^{m_2}\ldots a_{i_l}^{m_l}(a_{j_1}a_{j_2}\ldots a_{j_k})^\omega)) \\ &= \mathrm{DOWN}_{\leq_\Psi}(Suf({}^\omega(a_{j_1}a_{j_2}\ldots a_{j_k})a_{i_1}^{m_1}a_{i_2}^{m_2}\ldots a_{i_l}^{m_l})) \\ &= \mathrm{DOWN}_{\leq_\Psi}(Fact({}^\omega(a_{j_1}a_{j_2}\ldots a_{j_k})a_{i_1}^{m_1}a_{i_2}^{m_2}\ldots a_{i_l}^{m_l}(a_{j_1}a_{j_2}\ldots a_{j_k})^\omega)). \end{aligned}$$

The equalities in the statement are now clear. ∎

The last result is particular for Parikh subwords. It does not hold for any quasi order $\leq \subseteq \leq_s$, a fact not surprising because of the permutation of letters allowed in the Parikh subword quasi order. Indeed, we have the following examples: $a^*b^* \in \mathcal{F}_\leq^{bi} - (\mathcal{F}_\leq^l \cup \mathcal{F}_\leq^r)$, $a^*b \in \mathcal{F}_\leq^l - (\mathcal{F}_\leq^r \cup \mathcal{F}_\leq^{bi})$, and $ba^* \in \mathcal{F}_\leq^r - (\mathcal{F}_\leq^l \cup \mathcal{F}_\leq^{bi})$.

Notice that the next result can be proved as Theorem 7.3.7. Remarks similar to our comparison between Theorems 7.3.6 and 7.3.7 can be made in this case as well.

Theorem 7.4.12. *It is undecidable whether or not an arbitrary context-free language is in the family* $\mathcal{F}_{\leq_\Psi}$.

7.5 The relation with confluence

In this section, we show that the generalized confluence property is strongly connected with the concept of well quasi order.

7.5.1 Finite antichains

In one direction we have the following result.

Theorem 7.5.1. *If $\leq$ is a quasi order on Σ^* such that any language $L \subseteq \Sigma^*$ is confluent w.r.t. $\leq$ in generalized sense, then any antichain of $\leq$ is finite.*

Proof. Suppose that $L \subseteq \Sigma^*$ is an antichain of $\leq$. By hypothesis, L is confluent w.r.t. $\leq$ in generalized sense. Therefore, we have

$$L = \bigcup_{i=1}^{k} L_i,$$

for some $k \geq 1, L_i \subseteq \Sigma^*, L_i$ confluent w.r.t. $\leq$, for any $1 \leq i \leq k$.

Consider an $i, 1 \leq i \leq k$, and two elements $u, v \in L_i$. Since L_i is confluent w.r.t. $\leq$, there is $w \in L_i$ such that $u \leq w$ and $v \leq w$. But $u, v, w \in L$ which is an antichain of $\leq$. It follows that $u = w$ and $v = w$, thus $u = v$. As u and v were arbitrarily chosen in L_i, we get that

$$\mathrm{card}(L_i) = 1, \text{ for any } 1 \leq i \leq k.$$

Consequently,

$$\mathrm{card}(L) = k,$$

hence L is finite. ∎

Corollary 7.5.2. *If $\leq$ is a well founded quasi order on Σ^* such that any language $L \subseteq \Sigma^*$ is confluent w.r.t. $\leq$ in generalized sense, then $\leq$ is a well quasi order.*

There is also an obvious connection between the ordinary confluence property and total orders.

Theorem 7.5.3. *For a partial order $\leq$ on Σ^*, $\leq$ is total if and only if any language $L \subseteq \Sigma^*$ is confluent w.r.t. $\leq$.*

7.5.2 Extensions of subwords

In order to find a connection in the other direction, that is, to find some conditions according to which the property of being well for a quasi order $\leq$ implies the generalized confluence w.r.t. $\leq$ for any language, we need a result concerning the form of the down-sets w.r.t. $\leq_s$. It is known by Theorem 7.2.3 that, for any language L, the down-set $\mathrm{DOWN}_{\leq_s}(L)$ is regular.

Lemma 7.5.4. *For any language $L \subseteq \Sigma^*, \mathrm{DOWN}_{\leq_s}(L)$ is a finite union of sets of the form*

$$w_0\Sigma_1^* w_1 \Sigma_2^* w_2 \ldots w_{n-1}\Sigma_n^* w_n,$$

for some $n \geq 0, w_i \in \Sigma^*, \Sigma_i \subseteq \Sigma$, for any $1 \leq i \leq n$.

Proof. Since $\mathrm{DOWN}_{\leq_s}(L)$ is regular for any L, we may restrict ourselves to regular languages L; just take $\mathrm{DOWN}_{\leq_s}(L)$ instead of L and use the fact that the operator $\mathrm{DOWN}_{\leq_s}$ is idempotent.

The proof now continues by induction on the regular expression defining L. The basis of the induction, for $L = \emptyset$ or L is a letter, is clear. Suppose then that the statement holds true for L defined be any of the regular expressions α and β and let us prove it for L defined by $(\alpha + \beta)$, $(\alpha\beta)$ or α^*.

The cases of union and catenation are clear. (The catenation of two sets of the form in the statement has the same form.) For α^*, the required representation is just $(\mathrm{alph}(\alpha))^*$. ∎

We can state now the connection result already announced.

Theorem 7.5.5. *If $\leq$ is a well quasi order on Σ^* which is an extension of $\leq_s$, then any language $L \subseteq \Sigma^*$ is confluent w.r.t. $\leq$ in generalized sense.*

Proof. Take $L \subseteq \Sigma^*$. By Lemma 7.5.4, the down-set $\mathrm{DOWN}_{\leq_s}(L)$ is a finite union of sets of the form $R = w_0\Sigma_1^*w_1\Sigma_2^*w_2 \ldots w_{n-1}\Sigma_n^*w_n$, using the notations there. If $x, y \in R$, then $x = w_0u_1w_1 \ldots u_nw_n$, $y = w_0v_1w_1 \ldots v_nw_n$, for some $u_i, v_i \in \Sigma_i^*, 1 \leq i \leq n$. Consider the word $z = w_0u_1v_1w_1 \ldots u_nv_nw_n \in R$. Obviuosly $x \leq_s z, y \leq_s z$, thus R is confluent w.r.t. $\leq_s$. Therefore $\mathrm{DOWN}_{\leq_s}(L)$ is confluent w.r.t. $\leq_s$ in generalized sense. By Lemma 6.6.1, we obtain that also L is confluent w.r.t. $\leq_s$ in generalized sense and put

$$L = \bigcup_{i=1}^{k} L_i,$$

for some $k \geq 1, L_i \subseteq \Sigma^*, L_i$ confluent w.r.t. $\leq_s$, for any $1 \leq i \leq k$. Since $\leq$ is an extension of $\leq_s$, it follows that L_i is confluent w.r.t. $\leq$, for any $1 \leq i \leq k$. Therefore, L is confluent w.r.t. $\leq$ in generalized sense, as claimed. ∎

Using Theorem 7.2.1 and Corollary 7.2.2, we get the following corollary of Theorem 7.5.5.

Corollary 7.5.6. *If $\leq \in \{\leq_s, \leq_\Psi\}$, then any language is confluent w.r.t. $\leq$ in generalized sense.*

7.6 Regularity of down-sets

As mentioned before, for any language L, both down-sets of L, with respect to the subword partial order and its inverse, are regular. We consider several aspects of the regularity of down-sets in this section. First, we give a generalization of Haines' theorem. Second, we prove the existence of the families of languages with a decidable emptiness problem but for which the regularity of the sets of subwords is not effective. Third, we give a generalization of van Leeuwen's result concerning superwords.

7.6.1 A generalization of Haines' theorem

We give in this subsection a generalization of Haines'theorem by proving that it still holds for any monotone well quasi order instead of the subword partial order.

We need first some definitions. A quasi order $\leq$ on Σ^* is **monotone** if for any $x, y, u, v \in \Sigma^*$, $x \leq y$ implies $uxv \leq uyv$; L is **$\leq$-closed** if $x \in L$ and $x \leq y$ imply $y \in L$.

We will use the following generalization of Myhill-Nerode theorem from [EHR].

Theorem 7.6.1 (Ehrenfeucht, Haussler, Rozenberg, [EHR]). *A language $L \subseteq \Sigma^*$ is regular if and only if L is $\leq$-closed under some monotone well quasi order $\leq$ on Σ^*.*

We can prove now the theorem concerning the regularity of down-sets.

Theorem 7.6.2. *For any monotone well quasi order $\leq \subseteq \Sigma^* \times \Sigma^*$ and any language $L \subseteq \Sigma^*$, both sets $\mathrm{DOWN}_{\leq}(L)$ and $\mathrm{DOWN}_{\leq^{-1}}(L)$ are regular.*

Proof. Since $\mathrm{DOWN}_{\leq^{-1}}(L)$ is $\leq$-closed, the regularity of the down-set $\mathrm{DOWN}_{\leq^{-1}}(L)$ follows from Theorem 7.6.1.

In order to show that $\mathrm{DOWN}_{\leq}(L)$ is also regular, it is enough to prove that its complement $\overline{\mathrm{DOWN}_{\leq}(L)}$ is regular. We claim that

$$\overline{\mathrm{DOWN}_{\leq}(L)} = \mathrm{DOWN}_{\leq^{-1}}(\overline{\mathrm{DOWN}_{\leq}(L)}).$$

If this equality were proved, then, in virtue of the first part of our theorem, $\overline{\mathrm{DOWN}_{\leq}(L)}$ would be proved to be regular.

The inclusion of the left-hand member into the right-hand one is by the definition of $\mathrm{DOWN}_{\leq^{-1}}$ and the reflexivity of $\leq^{-1}$. For the converse inclusion, suppose that there is a word

$$w \in \mathrm{DOWN}_{\leq^{-1}}(\overline{\mathrm{DOWN}_{\leq}(L)}) - \overline{\mathrm{DOWN}_{\leq}(L)} = \mathrm{DOWN}_{\leq^{-1}}(\overline{\mathrm{DOWN}_{\leq}(L)}) \cap \mathrm{DOWN}_{\leq}(L).$$

It follows that there must be two words $u \in \overline{\mathrm{DOWN}_{\leq}(L)}$ and $v \in L$ such that $w \leq^{-1} u$ and $w \leq v$. Then $u \leq w \leq v$ and so, $\leq$ being transitive, $u \leq v$. But now $u \in \mathrm{DOWN}_{\leq}(L) \cap \overline{\mathrm{DOWN}_{\leq}(L)} = \emptyset$, which is absurd. Our proof is complete. ■

Notice that the result of Haines concerning the regularity of the sets of subwords and superwords follows from Theorem 7.6.2 and the restricted form of Higman's theorem in Theorem 7.2.1.

Using Theorem 7.6.2, we obtain a closure result which covers also the part concerning $\leq_s$ in Corollary 6.5.2.

Theorem 7.6.3. *Any full trio is closed under the down-operators $\mathrm{DOWN}_{\leq}$ and $\mathrm{DOWN}_{\leq^{-1}}$, for any monotone quasi order $\leq$.*

Proof. Any full trio contains all regular languages. Consequently, the result follows from Theorem 7.6.2. ■

7.6.2 Emptiness and effective regularity

Haines raised also the following problem: when is it possible to find effectively the regular sets $\mathrm{DOWN}_{\leq_s}(L)$ and $\mathrm{DOWN}_{\leq_s^{-1}}(L)$? It is easy to see that if it is possible to find effectively the two sets for all languages in a family $\mathcal{L}$, then $\mathcal{L}$ has a decidable emptiness problem. Van Leeuwen [vL] proved that for the superword partial order $\leq_s^{-1}$ the converse statement holds (Theorem 7.2.4).

For the subword partial order $\leq_s$, he gave an algorithm to find for any context-free language L the set $\mathrm{DOWN}_{\leq_s}(L)$; actually, he proved a more general result, namely that it is possible to find effectively the sets $\mathrm{DOWN}_{\leq_s}(L)$ and $\mathrm{DOWN}_{\leq_s^{-1}}(L)$ for all languages in a family $\mathcal{L}$ if and only if this is possible for the algebraic closure of $\mathcal{L}$.

It remained open whether or not there are families of languages for which the emptiness problem is decidable but the regularity of the sets of subwords is not effective. In what follows, we give a positive answer to this problem and construct a wide range of families having these properties.

Theorem 7.6.4. *Le $\mathcal{L}$ be a family of languages such that the emptiness problem is undecidable for languages in $\mathcal{L}$. Construct the family $\mathcal{L}'$ of languages by*

$$\mathcal{L}' = \{L\Sigma_L^* \cup \Delta_L^* \mid L \in \mathcal{L}\}$$

where

$$\Sigma_L = \begin{cases} \{a_1, a_2, \ldots, a_{p-1}\}, & \textit{if } \mathrm{alph}(L) = \{a_1, a_2 \ldots, a_p\}, p \geq 2, \\ \{a_2\}, & \textit{if } \mathrm{alph}(L) \subseteq \{a_1\}, \end{cases}$$
$$\Delta_L = \begin{cases} \{a_1, a_2, \ldots, a_{p-1}, a_{p+1}\}), & \textit{if } \mathrm{alph}(L) = \{a_1, a_2, \ldots, a_p\}, p \geq 2, \\ \{a_3\}, & \textit{if } \mathrm{alph}(L) \subseteq \{a_1\}, \end{cases}$$

and two letters with different subscripts are considered to be different. Then the emptiness problem is decidable for languages in $\mathcal{L}'$ but there is no algorithm to compute the sets of subwords of the languages in $\mathcal{L}'$.

Proof. The emptiness problem is trivially decidable for languages in $\mathcal{L}'$. For the second part of the statement, we prove first the next claim.

Claim. For any $L \in \mathcal{L}$, the language $L\Sigma_L^* \cup \Delta_L^*$ is confluent w.r.t. $\leq_s$ if and only if L is empty.

Proof of Claim. If L is empty, then $L\Sigma_L^* \cup \Delta_L^* = \Delta_L^*$ which is confluent w.r.t. $\leq_s$.

Conversely, suppose that $L\Sigma_L^* \cup \Delta_L^*$ is confluent w.r.t. $\leq_s$ but L is not empty. If $\mathrm{alph}(L) = \{a_1, a_2, \ldots, a_p\}, p \geq 2$, take $w \in L$ such that $a_p \in \mathrm{alph}(w)$. Consider the words $w, a_{p+1} \in L\Sigma_L^* \cup \Delta_L^*$. Since $L\Sigma_L^* \cup \Delta_L^*$ is confluent w.r.t. $\leq_s$, there must be an $x \in L\Sigma_L^* \cup \Delta_L^*$ such that $w \leq_s x$ and $a_{p+1} \leq_s x$.

Because $a_{p+1} \notin \mathrm{alph}(L\Sigma_L^*)$, we must have $x \in \Delta_L^*$. But now $a_p \notin \mathrm{alph}(x)$ thus $w \not\leq_s x$, a contradiction.

In the case when $\mathrm{alph}(L) \subseteq \{a_1\}$, as L is not empty, we have two cases: either $L = \{\lambda\}$ or there is a word, say w, in L such that $a_1 \in \mathrm{alph}(w)$. In the former case, we get that $L\Sigma_L^* \cup \Delta_L^* = a_2^* \cup a_3^*$ is confluent w.r.t. $\leq_s$, which is false. For the latter one, consider $w, a_3 \in La_2^* \cup a_3^*$. As above, there is no word $x \in La_2^* \cup a_3^*$ with $w \leq_s x, a_3 \leq_s x$, again a contradiction. The claim is proved. ∎

Suppose that there is an algorithm to compute the sets of subwords for languages in $\mathcal{L}'$. Now, for any $L \in \mathcal{L}$, by the Claim above, L is empty if and only if $L\Sigma_L^* \cup \Delta_L^*$ is confluent w.r.t. $\leq_s$, which is, in turn, equivalent, by Lemma 6.1.1, to the fact that the set $\textsc{down}_{\leq_s}(L\Sigma_L^* \cup \Delta_L^*)$ is confluent w.r.t. $\leq_s$, which is decidable by Theorem 6.4.3 since $\textsc{down}_{\leq_s}(L\Sigma_L^* \cup \Delta_L^*)$ is effectively regular by our assumption. Thus, the emptiness problem is decidable for languages in $\mathcal{L}$, a contradiction. The theorem is proved. ∎

7.6.3 Effective regularity for inverse relations

In this subsection, we give a generalization of van Leeuwen's result concerning the regularity of the set of superwords for well quasi orders with certain properties.

We remind that for any monotone well quasi order $\leq$ on Σ^*, the down-set $\textsc{down}_{\leq^{-1}}(L)$ is regular, for any language $L \subseteq \Sigma^*$, see Theorem 7.6.2.

Let us denote the length quasi order on Σ^* by $\leq_l$; for $u, v \in \Sigma^*$,

$$u \leq_l v \text{ iff } |u| \leq |v|.$$

Theorem 7.6.5. *Let $\leq$ be a monotone well quasi order on Σ^* such that $\leq \subseteq \leq_l$ and, for any $u, v \in \Sigma^*$, if $u \leq v$ and $|u| = |v|$, then also $v \leq u$. Suppose that, for any finite set F, the down-set $\textsc{down}_{\leq^{-1}}(F)$ is effectively regular. If $\mathcal{L}$ is a family of languages effectively closed under intersection with regular sets, then the regularity of the down-sets $\textsc{down}_{\leq^{-1}}(L)$ is effective for languages L in $\mathcal{L}$ if and only if the emptiness problem is decidable in $\mathcal{L}$.*

Proof. Suppose first that, for any $L \in \mathcal{L}$, the down-set $\textsc{down}_{\leq^{-1}}(L)$ is effectively regular. By the definition of $\textsc{down}_{\leq^{-1}}(L)$, it follows that if L is empty, then $\textsc{down}_{\leq^{-1}}(L)$ is empty. Since $\leq^{-1}$ is reflexive, $L \subseteq \textsc{down}_{\leq^{-1}}(L)$, so, if $\textsc{down}_{\leq^{-1}}(L)$ is empty, then L is empty as well. Consequently, L and $\textsc{down}_{\leq^{-1}}(L)$ are empty at the same time. The decidability of the emptiness problem in $\mathcal{L}$ follows now from its decidability for regular languages.

Conversely, suppose that the emptiness problem is decidable in $\mathcal{L}$ and let us show that, for any $L \in \mathcal{L}$, the down-set $\text{DOWN}_{\leq^{-1}}(L)$ is effectively regular.

Consider an arbitrary total order on $\Sigma = \{a_1, a_2, \dots, a_{\text{card}(\Sigma)}\}$, $a_i < a_j$, for any $1 \leq i < j \leq \text{card}(\Sigma)$, and the total lexicographical order on Σ^*; for $u, v \in \Sigma^*$,

$$\begin{aligned} u \leq_{lex} v \text{ iff either } & |u| < |v|, \text{ or } u = v, \text{ or else} \\ & u = a_{i_1} a_{i_2} \dots a_{i_{|u|}}, v = a_{i_1} \dots a_{i_{k-1}} a_{j_k} \dots a_{j_{|u|}}, \\ & \text{for some } 1 \leq k \leq |u|, 1 \leq i_l, j_l \leq \text{card}(\Sigma), i_k < j_k. \end{aligned}$$

For any language L, define

$$\min_{\leq} L = \{w \in L \mid \text{ for any } x \in L, \text{if } x \leq w, \text{then } w \leq x \text{ and } w \leq_{lex} x\}.$$

Claim 1. For any language L, $\min_{\leq} L$ is finite.

Proof of Claim 1. It is enough to show that the set $A = \{|w| \mid w \in \min_{\leq} L\}$ is finite. In order to prove this, consider, for any $n \in A$, $w_n \in \min_{\leq} L, |w_n| = n$, and take the set $B = \{w_n \mid n \in A\}$. If B is finite, then A is finite too. Suppose that, for some $n, m \in A, n \neq m$, we have $w_n \leq w_m$. Then, as $\leq \subseteq \leq_l$, we have $n \leq m$ and so $n < m$. From $w_m \in \min_{\leq} L$ we get $w_m \leq w_n$. It follows as above that $m < n$, a contradiction. Consequently, B is an antichain of $\leq$. As $\leq$ is a well quasi order, B is finite, thus A is finite. ■

Claim 2. For any language L,

$$\text{DOWN}_{\leq^{-1}}(\min_{\leq} L) = \text{DOWN}_{\leq^{-1}}(L).$$

Moreover, if $L \neq \emptyset$, then, for any $R \subset \min_{\leq} L$, $\text{DOWN}_{\leq^{-1}}(R) \subset \text{DOWN}_{\leq^{-1}}(L)$.

Proof of Claim 2. Since $\min_{\leq} L \subseteq L$, it follows that $\text{DOWN}_{\leq^{-1}}(\min_{\leq} L) \subseteq \text{DOWN}_{\leq^{-1}}(L)$. To prove the converse inclusion, take $w_0 \in L$. Since $\leq$ is well founded, we can find $w_1 \in L$ such that $w_1 \leq w_0$ and, for any $w \in L$, $w \leq w_1$ implies $w_1 \leq w$. If, for some $w \in L$, $w \leq w_1$ and $w_1 \leq w$, then, by hypothesis, $|w| = |w_1|$. Hence, we can find $w_2 \in L$ such that $w_2 \leq w_1, w_1 \leq w_2$, and, for any $w \in L$ with $w \leq w_1$ and $w_1 \leq w$, we have $w_2 \leq_{lex} w$. Therefore $w_2 \in \min_{\leq} L$ and, since $w_0 \leq^{-1} w_2$, it follows that $w_0 \in \text{DOWN}_{\leq^{-1}}(\min_{\leq} L)$. The equality is proved.

Take now $R \subset \min_{\leq} L$ and $w \in \min_{\leq} L - R$. (Since L is nonempty, $\min_{\leq} L$ is nonempty too.) We claim that $w \notin \text{DOWN}_{\leq^{-1}}(R)$. Indeed, if $w \in \text{DOWN}_{\leq^{-1}}(R)$, then $x \leq w$, for some $x \in R$. As $w \in \min_{\leq} L$, it follows that $w \leq x$ and $w \leq_{lex} x$. As also $x \in \min_{\leq} L$, we get $x \leq_{lex} w$, thus $x = w$ and so $w \in R$, a contradiction. The claim is proved. ■

We now give an algorithm to compute a regular expression for $\text{DOWN}_{\leq^{-1}}(L)$.

Algorithm 7.6.6.
Input: a language $L \in \mathcal{L}, L \subseteq \Sigma^*$.
Output: a regular expression representing the regular language $\text{DOWN}_{\leq^{-1}}(L)$.

(1) Consider Σ^* totally ordered lexicographically and put

$$\Sigma^* = \{w_1, w_2, w_3, \dots\},$$

such that $w_i \leq_{lex} w_{i+1}$, for any $i \geq 1$;

(2) Initialize $\text{MIN}_\text{L} \longleftarrow \emptyset$

[the variable MIN_L will reach, at the end of the algorithm, the value $\min_\leq L$];

(3) $i \longleftarrow 1$

[start with w_1];

(4) If $L \cap (\Sigma^* - \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L})) \neq \emptyset$, then go to step 5, else go to step 6;

[When $L \cap (\Sigma^* - \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L})) = \emptyset$, that is, $L \subseteq \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L})$, the algorithm has to stop because the value of $\min_\leq L$ has been found. Indeed, as it will be seen at step 5, always $\text{MIN}_\text{L} \subseteq \min_\leq L$ and so, by Claim 2, $L \subseteq \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L})$ if and only if $\text{MIN}_\text{L} = \min_\leq L$. Otherwise, the algorithm goes to step 5 attempting to include w_i in MIN_L. By hypothesis, $\text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L})$ is effectively regular since, by Claim 1, $\min_\leq L$ is finite. Then $\Sigma^* - \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L})$ is effectively regular too. Thus $L \cap (\Sigma^* - \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L}))$ is effectively in $\mathcal{L}$ and its emptiness can be decided.]

(5) If $(L \cap (\Sigma^* - \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L}))) \cap \{w_i\} \neq \emptyset$, then $\text{MIN}_\text{L} \longleftarrow \text{MIN}_\text{L} \cup \{w_i\}, i \longleftarrow i+1$, and go to step 4, else $i \longleftarrow i+1$, and go to step 5.

[Notice first that, as seen above, $L \cap (\Sigma^* - \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L}))$ is effectively in $\mathcal{L}$, so $(L \cap (\Sigma^* - \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L}))) \cap \{w_i\}$ is effectively in $\mathcal{L}$ too, as $\{w_i\}$ is regular. Therefore, the emptiness of the language $(L \cap (\Sigma^* - \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L}))) \cap \{w_i\}$ can be decided. If it is not empty, then $w_i \in L$ and $w_i \notin \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L})$. Now, if $w_j \leq w_i$, for some $w_j \in L$, then $|w_j| \leq |w_i|$. If $|w_j| < |w_i|$, then $j \leq i-1$. Hence, w_j has been considered at step 5 before, so either $w_j \in \text{MIN}_\text{L}$ or $w_k \leq w_j$, for some $1 \leq k \leq j-2$ such that $w_k \in \text{MIN}_\text{L}$. In both cases we find $w_l \in \text{MIN}_\text{L}$ such that $w_l \leq w_i$, thus $w_i \in \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L})$, a contradiction. Therefore $|w_j| = |w_i|$ and so, by hypothesis, $w_i \leq w_j$. Consider $j_0 = \min\{j \mid w_j \in L, w_j \leq w_i, |w_j| = |w_i|\}$. Then $j_0 \leq i$. If $j_0 < i$, then $w_{j_0} \in \text{MIN}_\text{L}$ and so $w_i \in \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L})$, a contradiction. Thus $i = j_0$ and $w_i \leq_{lex} w_j$, for any $w_j \in L, w_j \leq w_i, |w_j| = |w_i|$. Consequently $w_i \in \min_\leq L$ and it is correctly added to MIN_L. On the other hand, if $(L \cap (\Sigma^* - \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L}))) \cap \{w_i\} = \emptyset$, then, as $L \cap (\Sigma^* - \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L}) \neq \emptyset$ from step 4, either $w_i \notin L$ or $w_j \leq w_i$, for some $j < i$ such that $w_j \in \text{MIN}_\text{L} \subseteq \min_\leq L$. In both cases, w_i is correctly skipped and w_{i+1} is tested at step 5. If w_i is added, then, before testing w_{i+1}, the algorithm checks at step 4 whether there is any word to be added to MIN_L.]

(6) Output a regular expression for $\text{DOWN}_{\leq^{-1}}(\min_\leq L)$.

[As, by Claim 1, $\min_\leq L$ is finite, the algorithm will certainly reach this step and end. By hypothesis, it is possible to find effectively a regular expression for $\text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L})$. Step 6 is reached when $L \cap (\Sigma^* - \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L})) = \emptyset$, so $L \subseteq \text{DOWN}_{\leq^{-1}}(\text{MIN}_\text{L})$. It follows by Claim 2 that MIN_L has reached the value $\text{MIN}_\text{L} = \min_\leq L$ and the regular expression for $\text{DOWN}_{\leq^{-1}}(\min_\leq L)$ at output is also a regular expression for $\text{DOWN}_{\leq^{-1}}(L)$, as required.] ■

As an application of our general result in Theorem 7.6.5, we give the next corollary. Notice that a remark as the one in page 130 can be made here as well.

Corollary 7.6.7. *For any context-free language L, the down-set $\text{DOWN}_{\leq_\Psi^{-1}}(L)$ is effectively regular.*

Proof. The quasi order $\leq_\Psi$ is a monotone well quasi order on Σ^* and, obviously, $\leq_\Psi \subseteq \leq_l$. Also, if $u \leq_\Psi v$ and $|u| = |v|$, then u and v have the same Parikh vector, so also $v \leq u$.

For a finite set $F = \{w_1, w_2, \ldots, w_n\} \subset \Sigma^*$, if $\lambda \in F$, then $\text{DOWN}_{\leq_\Psi^{-1}}(F) = \Sigma^*$. Otherwise, if $w_i = w_{i,1}w_{i,2}\ldots w_{i,n_i}, w_{i,j} \in \Sigma, 1 \leq j \leq n_i, 1 \leq i \leq n$, then

$$\text{DOWN}_{\leq_\Psi^{-1}}(F) = \bigcup_{i=1}^{n} \bigcup_{\pi \in \mathcal{S}_{n_i}} \Sigma^* w_{i,\pi(1)} \Sigma^* w_{i,\pi(2)} \Sigma^* \ldots \Sigma^* w_{i,\pi(n_i)} \Sigma^*,$$

so it is effectively regular.

Since the family of context-free languages fulfills the conditions in Theorem 7.6.5, the result is proved. ■

7.7 Generalized factors of words

We introduce and study relations on words which generalize the factor relation, being restrictions of the subword relation. We give an equivalent condition for the finite basis property for these relations, which generalizes the form for words of Higman's theorem (Theorem 7.2.1), as well as some language-theoretic considerations on infinite antichains.

7.7.1 Generalized factor relations

Roughly speaking, a word is a factor of another one it the former appears inside the latter exactly as it is whereas a subword may appear scattered as several factors. Therefore, the idea of in-between cases naturally arises, that is, of imposing different restrictions on the number of factors as which a word is allowed to be scattered. Following this idea, we define in this subsection a class of (uncountably many) relations which are extensions of the factor partial order $\leq_f$ and restrictions of the subword partial order $\leq_s$. We then study in the next two subsections the finite basis property for these relations and give a characterization of those having this property. In the last subsection, we consider some language-theoretic properties of the infinite antichains of some of the relations introduced.

The first step in defining our generalized factor relations is to consider, for any positive integer k, the relation $\leq_{f,k}$ on Σ^*, defined by

$$u \leq_{f,k} v \quad \text{iff} \quad \begin{array}{l} v = v_0u_1v_1u_2v_2\ldots u_kv_k, \\ v_0, u_i, v_i \in \Sigma^*, 1 \leq i \leq k, \\ u = u_1u_2\ldots u_k. \end{array}$$

It is easy to see that $\leq_{f,1}=\leq_f$ and, for any $k \geq 2$, $\leq_{f,k}$ is reflexive, antisymmetric, but not transitive.

We have obtained an infinite strict hierarchy of relations between the factor and the subword relation.

Theorem 7.7.1. $\leq_f=\leq_{f,1}\subset\leq_{f,2}\subset\leq_{f,3}\subset \cdots \subset\leq_s$, *all inclusions being strict.*

Proof. Obviously, for any $k \geq 1$, $\leq_{f,k}\subseteq\leq_{f,k+1}$. We have also $(ab)^ka \leq_{f,k+1} (ab^2)^ka$ but $(ab)^ka \not\leq_{f,k} (ab^2)^ka$ since $(ab)^ka$ appears in $(ab^2)^ka$ scattered in at least $k+1$ factors, as none of them can contain two a's. The strictness of the inclusion follows. ∎

We now define the finite basis property for relations exactly as it was defined for quasi orders, see [ErRa], [Hi]. Consider an arbitrary relation $\leq$ on Σ^*. The notion of antichain is defined as for quasi orders; a set $L \subseteq \Sigma^*$ is an **antichain** of $\leq$ if all elements of L are pairwise incomparable with respect to $\leq$, that is, for any $u, v \in L$, neither $u \leq v$ nor $v \leq u$. The relation $\leq$ has the **finite basis property**[1] if (i) any antichain of it is finite and (ii) there is no infinite descending sequence $w_1 \leq^{-1} w_2 \leq^{-1} w_3 \cdots$ such that, for no $i \geq 1$, $w_i \leq w_{i+1}$, where $\leq^{-1}$ denotes the inverse of $\leq$.

[1]Since most of the generalized factor relations are not transitive, we prefer to use this name and to avoid the term "well" from "well founded" and "well quasi order".

Remark. Since we work only with relations which are included in $\leq_s$, the condition (ii) will always be fulfilled, so we shall consider explicitely the condition (i) only.

Theorem 7.7.2. *For any $k \geq 1$, the relation $\leq_{f,k}$ does not have the finite basis property.*

Proof. Take $k \geq 1$ and consider the language

$$L_k = \{(ab^n)^k a \mid n \geq 1\}.$$

We observe that L_k is infinite and an antichain of $\leq_{f,k}$. Indeed, if, for some $k' \geq 1$, $(ab^n)^k a \leq_{f,k'} (ab^m)^k a$, for some $n < m$, then $(ab^n)^k a$ must be scattered in such a way that no factor contains two a's. Thus $k' \geq k+1$ so $(ab^n)^k \not\leq_{f,k} (ab^m)^k a$. ■

We generalize further the above idea as follows. Consider the following set of natural functions

$$\mathfrak{G} = \{g : \mathbb{N} \longrightarrow \mathbb{N} \mid g(0) = 0 \text{ and } 1 \leq g(n) \leq n, \text{ for any } n \geq 1\}.$$

For any function $g \in \mathfrak{G}$, we define the relation $\leq_{f,g}$ on Σ^*, for $u, v \in \Sigma^*$, by

$$u \leq_{f,g} v \quad \text{iff} \quad \begin{array}{l} v = v_0 u_1 v_1 u_2 v_2 \ldots u_{g(|u|)} v_{g(|u|)}, \\ v_0, u_i, v_i \in \Sigma^*, 1 \leq i \leq g(|u|), \\ u = u_1 u_2 \ldots u_{g(|u|)}. \end{array}$$

Notice that

$$\leq_f = \leq_{f,\mathbf{1}},$$

where $\mathbf{1} : \mathbb{N} \longrightarrow \mathbb{N}, \mathbf{1}(0) = 0,\ \mathbf{1}(n) = 1$, for any $n \geq 1$, and

$$\leq_s = \leq_{f,\mathbf{1}_{\mathbb{N}}},$$

where $\mathbf{1}_{\mathbb{N}} : \mathbb{N} \longrightarrow \mathbb{N}, \mathbf{1}_{\mathbb{N}}(n) = n$, for any $n \geq 0$.

As seen in the next result, we have now uncountably many relations between $\leq_f$ and $\leq_s$.

Theorem 7.7.3. *(i). For any $g \in \mathfrak{G}$, $\leq_f \subseteq \leq_{f,g}$ and $\leq_f = \leq_{f,g}$ if and only if $g \equiv \mathbf{1}$.*

(ii). For any $g_1, g_2 \in \mathfrak{G}$ such that $g_1(n) \leq g_2(n)$, for any $n \geq 1$, we have $\leq_{f,g_1} \subseteq \leq_{f,g_2}$. If $g_1 \not\equiv g_2$, then the inclusion is strict.

(iii). For any $g \in \mathfrak{G}, \leq_{f,g} \subseteq \leq_s$ and $\leq_{f,g} = \leq_s$ if and only if $g \equiv \mathbf{1}_{\mathbb{N}}$.

Proof. (i). We prove only that $\leq_f = \leq_{f,g}$ implies $g \equiv \mathbf{1}$. Indeed, if, for some $n_0 \geq 1, g(n_0) \geq 2$, then $a^{n_0} \leq_{f,g} aba^{n_0-1}$ but $a^{n_0} \not\leq_f aba^{n_0-1}$.

(ii). To prove the strictness of the inclusion, suppose that, for some $n_0 \geq 1$, $g_1(n_0) < g_2(n_0)$. Then $a^{n_0} \leq_{f,g_2} (ab)^{g_2(n_0)-1} a$ but $a^{n_0} \not\leq_{f,g_1} (ab)^{g_2(n_0)-1} a$.

(iii). The first part is obvious and the second is a special case of (ii). ■

7.7.2 Finite basis property

We present in this section a class of relations $\leq_{f,g}$ which allow infinite antichains. It will turn out in the next section that these are actually all relations in the family $(\leq_{f,g})_{g\in\mathfrak{G}}$ with this property.

Theorem 7.7.4. *For any $g \in \mathfrak{G}$ such that*

$$\overline{\lim}_{n\to\infty}\frac{n}{g(n)} = \infty,$$

the relation $\leq_{f,g}$ allows infinite antichains and, therefore, it does not have the finite basis property.

Proof. Take $g \in \mathfrak{G}$ such that

$$\overline{\lim}_{n\to\infty}\frac{n}{g(n)} = \infty.$$

We can find a subsequence $(k_n)_{n\geq 1}$ such that

$$\lim_{n\to\infty}\frac{k_n}{g(k_n)} = \infty.$$

We construct the following natural functions. First, define $k : \mathbb{N} \longrightarrow \mathbb{N}$ by

$$k(n) = \min\{k_m \in \mathbb{N} \mid k_m \geq 1 \text{ and } g(k_m) \leq \frac{k_m}{2(n+1)}\}, \text{ for any } n \geq 0.$$

For any fixed $n \geq 1$, since $\lim\limits_{m\to\infty}\frac{k_m}{g(k_m)} = \infty$, there is an $m \in \mathbb{N}$ such that

$$\frac{k_m}{g(k_m)} \geq 2(n+1)$$

and so

$$g(k_m) \leq \frac{k_m}{2(n+1)}.$$

Therefore, there is also a minimal such m and the function k is well defined.

The second function is $h : \mathbb{N} \longrightarrow \mathbb{N}$ defined by

$$h(n) = \max\{m \in \mathbb{N} \mid m(n+1)+1 \leq k(n)\}, \text{ for any } n \geq 0.$$

For any fixed $n \geq 0$, $k(n)$ is a finite number and the set of those $m \in \mathbb{N}$ with $m(n+1)+1 \leq k(n)$ is finite and nonempty. Thus, the above maximum always exists and h is well defined. Notice also that both h and k are increasing functions.

The last function we use is $r : \mathbb{N} \longrightarrow \mathbb{N}$, defined by

$$r(n) = k(n) - h(n)(n+1) - 1, \text{ for any } n \geq 0.$$

Notice that, from the definition of h, we have that $k(n) \geq h(n)(n+1)+1$, for any $n \geq 0$, so r is well defined.

Consider now the set

$$L_g = \{(ab^n)^{h(n)}ab^{r(n)} \mid n \geq 1\}.$$

Obviously L_g is infinite, and we claim that L_g is an antichain of $\leq_{f,g}$. To prove this, suppose that, for some $n, m \geq 1, n < m$, and $k \geq 1$, we have

$$(ab^n)^{h(n)}ab^{r(n)} \leq_{f,k} (ab^m)^{h(m)}ab^{r(m)}.$$

It follows that $(ab^n)^{h(n)}ab^{r(n)}$ must be scattered in at least $h(n)+1$ factors, thus

$$k \geq h(n) + 1. \tag{7.7.1}$$

Using the definitions of the functions k, h, and r, we can write

$$g(|(ab^n)^{h(n)}ab^{r(n)}|) = g(h(n)(n+1) + r(n) + 1) = g(k(n)) \leq \frac{k(n)}{2(n+1)} \tag{7.7.2}$$

and

$$k(n) \leq (h(n)+1)(n+1) \leq 2h(n)(n+1), \tag{7.7.3}$$

since $h(n) \geq 1$, for any $n \geq 1$. From (7.7.2) and (7.7.3) we get

$$g(|(ab^n)^{h(n)}ab^{r(n)}|) \leq h(n)$$

thus, using (7.7.1), we have

$$(ab^n)^{h(n)}ab^{r(n)} \not\leq_{f,g} (ab^m)^{h(m)}ab^{r(m)}.$$

Therefore, indeed L_g is an antichain of $\leq_{f,g}$. The proof is concluded. ■

We give below two examples of the construction in Theorem 7.7.4.

Example 7.7.5. Consider the function $g_1 : \mathbb{N} \longrightarrow \mathbb{N}$ given by

$$g_1(n) = \begin{cases} [\sqrt{n}] & \text{if } n \in \{4(k+1)^2 \mid k \geq 1\}, \\ n & \text{otherwise,} \end{cases}$$

where $[x]$ denotes the integer part of x. Then $g_1 \in \mathfrak{G}$ and it verifies the condition in Theorem 7.7.4 as

$$\overline{\lim}_{n\to\infty}\frac{n}{g_1(n)} = \infty,$$

a subsequence realizing this being $k_n = 4(n+1)^2, n \geq 1$. We find, as in the proof of Theorem 7.7.4, an infinite antichain of $\leq_{f,g_1}$. Compute first the three functions there:

$$\begin{aligned}
k(n) &= \min\{k_m \mid k_m \geq 1 \text{ and } g_1(k_m) \leq \frac{k_m}{2(n+1)}\} = \\
&= \min\{4(m+1)^2 \mid 2(m+1) \leq \frac{4(m+1)^2}{2(n+1)}\} = 4(n+1)^2;\\
h(n) &= \max\{m \in \mathbb{N} \mid m(n+1)+1 \leq k(n)\} = \\
&= \max\{m \in \mathbb{N} \mid m(n+1)+1 \leq 4(n+1)^2\} = 4(n+1)-1;\\
r(n) &= 4(n+1)^2 - (4(n+1)-1)(n+1) - 1 = n.
\end{aligned}$$

The obtained infinite antichain of $\leq_{f,g_1}$ is

$$L_{g_1} = \{(ab^n)^{4(n+1)-1}ab^n \mid n \geq 1\}.$$

Another example, where the antichain obtained is much more complicated, is given below.

Example 7.7.6. Take the function $g_2 : \mathbb{N} \longrightarrow \mathbb{N}$ given by

$$g_2(n) = \begin{cases} \left[\frac{n}{\log_2\log_2 n}\right] & \text{if } n \in \{2^{2^k} \mid k \geq 1\},\\ n & \text{otherwise.}\end{cases}$$

Then $g_2 \in \mathfrak{G}$ and it verifies the condition on the superior limit in the theorem, namely

$$\overline{\lim}_{n\to\infty}\frac{n}{g_2(n)} = \infty.$$

A subsequence that realizes this limit is $k_n = 2^{2^n}, n \geq 1$. The three functions are:

$$\begin{aligned}
k(n) &= \min\{2^{2^m} \mid \left[\frac{2^{2^m}}{m}\right] \leq \frac{2^{2^m}}{2(n+1)}\} = 2^{2^{2(n+1)}},\\
h(n) &= \max\{m \in \mathbb{N} \mid m(n+1)+1 \leq 2^{2^{2(n+1)}}\} = \left[\frac{2^{2^{2(n+2)}}-1}{n+1}\right],\\
r(n) &= 2^{2^{2(n+1)}} - \left[\frac{2^{2^{2(n+2)}}-1}{n+1}\right](n+1) - 1.
\end{aligned}$$

In this case, we obtain the following infinite antichain of $\leq_{f,g_2}$:

$$L_{g_2} = \{(ab^n)^{\left[\frac{2^{2^{2(n+2)}}-1}{n+1}\right]} ab^{2^{2^{2(n+1)}} - \left[\frac{2^{2^{2(n+2)}}-1}{n+1}\right](n+1)-1} \mid n \geq 1\}.$$

7.7.3 A generalization of Higman's theorem

We now show that also the converse of Theorem 7.7.4 holds true, thus obtaining a complete characterization of the relations in the family $(\leq_{f,g})_{g\in\mathfrak{G}}$ having the finite basis property. This generalizes the form for words of Higman's theorem, cf. Theorem 7.2.1. The proof has a basic idea similar with the one for the subword relation $\leq_s$ by Conway [Co]. (According to [Lo], the basic idea is due to Nash-Williams [NW].)

Theorem 7.7.7. *For any $g \in \mathfrak{G}$ such that*

$$\overline{\lim}_{n\to\infty} \frac{n}{g(n)} < \infty,$$

the relation $\leq_{f,g}$ has the finite basis property.

Proof. Take $g \in \mathfrak{G}$ such that

$$\overline{\lim}_{n\to\infty} \frac{n}{g(n)} < \infty.$$

Then, there is a positive integer $c \geq 1$ such that

$$\frac{n}{g(n)} \leq c, \text{ for any } n \geq 1.$$

So $g(n) \geq \dfrac{n}{c}$. Consider the function $h : \mathbb{N} \longrightarrow \mathbb{N}$ defined by

$$h(n) = \begin{cases} 0 & \text{if } n = 0, \\ \max\left(1, \left[\dfrac{n}{c}\right]\right) & \text{if } n \geq 1. \end{cases}$$

Obviously, $h \in \mathfrak{G}$ and, since

$$h(n) = \max\left(1, \left[\frac{n}{c}\right]\right) \leq \max\left(1, \frac{n}{c}\right) \leq g(n),$$

for any $n \geq 1$, it follows that if $\leq_{f,h}$ has the finite basis property, then so has $\leq_{f,g}$. Therefore, it is enough to prove that $\leq_{f,h}$ has the finite basis property. We argue by contradiction. Suppose that $\leq_{f,h}$ does not have the finite basis property and consider an infinite antichain of $\leq_{f,h}$, say

$$\{w_1, w_2, w_3, \ldots\},$$

where $w_i \in \Sigma^*$, for any $i \geq 1$. It follows that, in particular, we have

$$\text{if } i < j, \text{ then } w_i \not\leq_{f,h} w_j. \tag{7.7.4}$$

Consider now the sequence $w_1, w_2, w_3, \ldots$ built as follows: w_1 is a shortest word begining a sequence satisfying (7.7.4), w_2 is a shortest word such that w_1, w_2 begin a sequence satisfying (7.7.4), and so on and so forth. Thus, there is no sequence $v_1, v_2, v_3, \ldots$ satisfying (7.7.4) such that, for some $i \geq 1$, $|v_i| < |w_i|$.

Now, as Σ is finite, there must be a word w such that $|w| = c$ and infinitely many words of the sequence $(w_i)_{i\geq 1}$ have w as a prefix. Suppose that all those beginning with w are $(w_{i_k})_{k\geq 1}$ and put $w_{i_k} = wu_k$, for any $k \geq 1$.

Consider the sequence

$$w_1, w_2, \ldots, w_{i_1-1}, u_1, u_2, \ldots . \tag{7.7.5}$$

We claim that this sequence satisfies (7.7.4). We notice first that $w_i \not\leq_{f,h} w_j$, for all $1 \leq i < j \leq i_1 - 1$, and $w_i \not\leq_{f,h} u_j$, for any $1 \leq i \leq i_1 - 1, j \geq 1$.

If, for some $p, q, p < q$ and $u_p \leq_{f,h} u_q$, then, by definition,

$$u_q = u_{q,0}u_{p,1}u_{q,1}u_{p,2}u_{q,2} \ldots u_{p,h(|u_p|)}u_{q,h(|u_p|)},$$

where $u_{q,0}, u_{p,i}, u_{q,i} \in \Sigma^*, 1 \leq i \leq h(|u_p|)$, and

$$u_p = u_{p,1}u_{p,2} \ldots u_{p,h(|u_p|)}.$$

But now, since

$$\begin{aligned} w_q &= wu_{q,0}u_{p,1}u_{q,1}u_{p,2}u_{q,2} \ldots u_{p,h(|u_p|)}u_{q,h(|u_p|)}, \\ w_p &= wu_{p,1}u_{p,2} \ldots u_{p,h(|u_p|)}, \end{aligned}$$

and

$$h(|w_p|) = h(|u_p| + c) = \left\lceil \frac{|u_p| + c}{c} \right\rceil = \left\lceil \frac{|u_p|}{c} \right\rceil + 1 = h(|u_p|) + 1,$$

we obtain that $w_p \leq_{f,h} w_q$, contradicting the fact that $(w_i)_{i\geq 1}$ satisfies (7.7.4). Therefore, the sequence (7.7.5) satisfies the condition (7.7.4).

Since $|u_1| < |w_{i_1}|$, this contradicts our assumption on $(w_i)_{i\geq 1}$ as being the "earliest" sequence satisfying (7.7.4). Consequently, $\leq_{f,h}$ has the finite basis property and, as noticed above, so has $\leq_{f,g}$. The proof is concluded. ■

Remark. The subword relation $\leq_s$ is obtained for $g = \mathbf{1}_{\mathbb{N}}$ (see above). Since

$$\lim_{n\to\infty} \frac{n}{\mathbf{1}_{\mathbb{N}}(n)} = \lim_{n\to\infty} \frac{n}{n} = 1 < \infty,$$

Higman's theorem is obtained as a consequence of our Theorem 7.7.7 for $g = \mathbf{1}_{\mathbb{N}}$.

From Theorems 7.7.4 and 7.7.7, we obtain a complete characterization of those relations in the family $(\leq_{f,g})_{g \in \mathfrak{G}}$ which have the finite basis property.

Theorem 7.7.8. *For any $g \in \mathfrak{G}$, the relation $\leq_{f,g}$ has the finite basis property if and only if*

$$\overline{\lim}_{n \to \infty} \frac{n}{g(n)} < \infty.$$

As a consequence of Theorem 7.7.8, we obtain immediately the following decidability result.

Corollary 7.7.9. *For any arbitrary $g \in \mathfrak{G}$ such that the limit*

$$\overline{\lim}_{n \to \infty} \frac{n}{g(n)},$$

is computable, it is decidable whether or not the relation $\leq_{f,g}$ has the finite basis property.

7.7.4 Language-theoretic gaps for antichains

In this section, we consider infinite antichains for some of the relations mentioned in the previous sections and investigate their regularity and context-freeness.

The first result concerns the relations which are restrictions of $\leq_f$.

Theorem 7.7.10. *There exists infinite regular antichains for any relation $\leq$ which is a restriction of the factor relation $\leq_f$.*

Proof. It is enough to prove the statement for $\leq_f$ since any antichain of $\leq_f$ is an antichain of $\leq$ too. We can take then

$$L = \{ab^n a \mid n \geq 0\}.$$

Indeed, L is an antichain of $\leq_f$ and it is an infinite regular language. ■

A natural question arises: are there relations which do not have the finite basis property but for which there is no infinite regular antichain ? The answer is positive, as proved in the next result. Moreover, the result holds also in the context-free case, as we show in a moment.

Theorem 7.7.11. *The relation $\leq_{f,2}$ has no infinite regular antichains. There are infinite context-free antichains for $\leq_{f,2}$.*

Proof. For the second part of the statement, it is enough to consider the language

$$L = \{ab^nab^na \mid n \geq 1\}.$$

Obviuosly, L is an antichain of $\leq_{f,2}$ and it is an infinite context-free (even linear) language.

For the first part, we argue by contradiction. Suppose that there is an infinite regular antichain R of $\leq_{f,2}$. By the pumping lemma for regular languages, there is a constant $p \geq 1$ such that any word $w \in R$ with $|w| \geq p$ can be written as $w = uvx$, with $v \neq \lambda$ and $uv^ix \in R$, for any $i \geq 0$. Consider such a $w \in R$ (there is such a w because R is infinite) and also its decomposition as above. We have $uvx, uvvx \in R$ and $uvx \leq_{f,2} uvvx$. Since $uvx \neq uvvx$, a contradiction is obtained. The theorem is proved. ■

Theorem 7.7.12. *The relation $\leq_{f,3}$ has no infinite context-free antichains. There are infinite antichains for $\leq_{f,2}$ which are intersections of two context-free languages.*

Proof. The second part is proved by the infinite language

$$L = \{ab^nab^nab^na \mid n \geq 0\}$$

which is an antichain of $\leq_{f,3}$ and is the intersection of the context-free (linear) languages $\{ab^nab^nab^ma \mid n, m \geq 0\}$ and $\{ab^mab^nab^na \mid n, m \geq 0\}$.

The proof of the first assertion is similar to the one in the previous theorem. Arguing by contradiction, we suppose that P is an infinite context-free language which is an antichain of $\leq_{f,3}$. By the pumping lemma for context-free languages, there is a constant $n \geq 1$ such that, for any $z \in P$ with $|z| \geq n$, z can be decomposed as $z = uvwxy$, where $vx \neq \lambda$ and $uv^iwx^iy \in P$, for any $i \geq 0$. Take such a z and its decomposition as above. It follows that $uvwxy$ and uv^2wx^2y are both in P and different. Since $uvwxy \leq_{f,3} uv^2wx^2y$, a contradiction is obtained and the result is proved. ■

7.8 Further research

We mention here some possible research directions connected with the problems studied in this chapter.

First, we are concerned with the connection between the decidability of the confluence problem and the effective regularity of down-sets. As proved in Lemma 6.1.1, for any quasi order $\leq$, a language L and its down-set $\text{DOWN}_{\leq}(L)$

are at the same time confluent w.r.t. $\leq$. On the other hand, for any monotone well quasi order $\leq$, the down-set $\text{DOWN}_{\leq}(L)$ of any language L is regular, as proved in Theorem 7.6.2. It follows that, for any such quasi order $\leq$ such that the confluence problem w.r.t. $\leq$ is decidable for regular languages, the effectiveness of this regularity for a certain family of languages $\mathcal{L}$ implies the decidability of the confluence problem w.r.t. $\leq$ in $\mathcal{L}$. Therefore, the confluence property is suitable for finding families for which the regularity of down-sets w.r.t. $\leq$ is not effective (as in Theorem 7.6.4).

There are two general problems which should be investigated (or their restrictions to some particular quasi orders):

Problem 7.8.1. When does the decidability of the confluence problem entail the effectiveness of the regularity of down-sets ?

Problem 7.8.2. Find general conditions which are equivalent to the effectiveness of the regularity of down-sets (as, for instance, the one by van Leeuwen for $\leq_s^{-1}$ in Theorem 7.2.4).

Second, we consider the state complexity problem for the operation $\text{DOWN}_{\leq}$, $\leq \in \{\leq_s, \leq_\Psi\}$, in the sense of [YZS].

Problem 7.8.3. What is the number of states that is sufficient and necessary in the worst case for a DFA to accept the regular language $\text{DOWN}_{\leq_s}(L)$, for L a regular language accepted by a n-state DFA?

Problem 7.8.4. The same problem for the down-operator $\text{DOWN}_{\leq_\Psi}$.

Bibliography

[ADPS] M. Andraşiu, J. Dassow, Gh. Păun, and A. Salomaa, Language-theoretic problems arising from Richelieu cryptosystems, *Theoret. Comput. Sci.* **116** (1993) 339 – 357.

[AFG] J.-M. Autebert, P. Flajolet, and J. Gabarro, Prefixes of infinite words and ambiguous context-free languages, *Inform. Process. Lett.* **25** (1987) 211 – 216.

[AuGa] J.-M. Autebert and J. Gabarro, Iterated GSM's and Co-CFL, *Acta Inform.* **26** (1989) 749 – 769.

[Bea1] D. Beauquier, Thin homogeneous sets of factors, *Proc. of the 6th Conference of Foundations of Software Technology and Theoret. Comput. Sci.* (New Delhi, 1986) Lecture Notes in Comput. Sci., 241, Springer, Berlin-New York, 1986, 239 – 251.

[Bea2] D. Beauquier, Minimal automaton of a rational cover, *Proc. of the 14th ICALP* (Karlsruhe 1987), Lecture Notes in Comput. Sci., 267, Springer, Berlin-New York, 1987, 174 – 189.

[BeaN] D. Beauquier and M. Nivat, About rational sets of factors of a bi-infinite word, *Proc. of the 12th ICALP* (Nafplion 1985) Lecture Notes in Comput. Sci., 194, Springer, Berlin-New York, 1985, 33 – 42.

[Be] J. Berstel, Every iterated morphism yields a Co-CFL, *Inform. Process. Lett.* **22** (1986) 7 – 9.

[BeBo] J. Berstel, L. Boasson, The set of minimal words of a context-free language is context-free, *J. Comput. Syst. Sci.* **55** (1997) 477 – 488.

[BePe] J. Berstel and D. Perrin, *Theory of Codes* (Academic Press, New York, 1985).

[BoNi] L. Boasson and M. Nivat, Adherence of languages, *J. Comput. System Sci.* **20** (1980) 285 – 309.

[CaYu] C. Calude and S. Yu, Language-theoretic complexity of disjunctive sequences, *Discrete Appl. Math.* **80** (1997) 203 – 209.

[ChKa] C. Choffrut and J. Karhumäki, Combinatorics of Words, in G. Rozenberg, A. Salomaa, eds., *Handbook of Formal Languages, Vol. 1, Word, Language, Grammar* (Springer-Verlag, Berlin, Heidelberg, 1997) 329 – 438.

[CrRy] M. Crochemore and W. Rytter, *Text Algorithms* (Oxford Univ. Press., New York, Oxford, 1994).

[Co] J. H. Conway, *Regular Algebra and Finite Machines* (Chapman and Hall, 1971).

[DPS] J. Dassow, G. Păun, and A. Salomaa, On thinness and slenderness of L languages, *EATCS Bull.* **49** (1993) 152 – 158.

[dLVa1] A. de Luca and S. Varricchio, Well quasi-orders and regular languages, *Acta Inform.* **31** (1994) 539 – 557.

[dLVa2] A. de Luca and S. Varricchio, Unavoidable Regularity and Finiteness Conditions, in G. Rozenberg, A. Salomaa, eds., *Handbook of Formal Languages, Vol. 1, Word, Language, Grammar* (Springer-Verlag, Berlin, Heidelberg, 1997) 747 – 810.

[Do] P. Dömösi, Unusual algorithms for lexicographical enumeration, manuscript.

[EHR] A. Ehrenfeucht, D. Haussler, and G. Rozenberg, On regularity of context-free languages, *Theoret. Comput. Sci.* **27** (1983) 311 – 332.

[Ei] S. Eilenberg, *Automata, Languages and Machines, Vol. 1* (Academic Press, New York, 1974).

[ErRa] P. Erdös and R. Rado, Sets having divisor property, Solution to problem 4358, *Amer. Math. Monthly* **59** (1952) 255 – 257.

[FiWi] N. J. Fine and H. S. Wilf, Uniqueness theorem for periodic functions, *Proc. Amer. Math. Soc.* **16** (1965) 109 – 114.

[Gi] S. Ginsburg, *The Mathematical Theory of Context-free Languages* (McGraw-Hill, New York, 1966).

[GiSp] S. Ginsburg and E.H. Spanier, Bounded ALGOL-like Languages, *Trans. Amer. Math. Soc.* **113** (1964) 333 – 368.

[Hai] L. H. Haines, On free monoids partially ordered by embedding, *J. Combin. Theory* **6** (1969) 94 – 98.

[HaIl1] T. Harju and L. Ilie, Languages obtained from infinite words, *RAIRO Inform. Thèor. Appl.* **31** (1997) 445 – 455.

[HaIl2] T. Harju and L. Ilie, On quasi orders of words and the confluence property, *Theoret. Comput. Sci.* **200** (1998) 205 – 224.

[HaKa] T. Harju and J. Karhumäki, Morphisms, in G. Rozenberg, A. Salomaa, eds., *Handbook of Formal Languages, Vol. 1, Word, Language, Grammar* (Springer-Verlag, Berlin, Heidelberg, 1997) 439 – 510.

[Har] M. A. Harrison, *Introduction to Formal Language Theory* (Addison-Wesley, Reading, MA., 1978).

[Hi] G. Higman, Ordering by divisibility in abstract algebras, *Proc. London Math. Soc.* **2**(3) (1952) 326 – 336.

[Ho1] J. Honkala, On Parikh slender languages and power series, *J. Comput. System Sci.* **52** (1996) 185 – 190.

[Ho2] J. Honkala, On images of algebraic series, *J. Univ. Comp. Sci.* **2**(4) (1996) 217 – 223.

[Ho3] J. Honkala, A decision method for Parikh slenderness of context-free languages, *Discrete Appl. Math.* **73** (1997) 1 – 4.

[Ho4] J. Honkala, Decision problems concerning thiness and slenderness of formal languages, *Acta Inform.* **35** (1998) 625 – 636.

[HoUl] J. E. Hopcroft and J. D. Ullman, *Introduction to Automata Theory, Languages, and Computation* (Addison-Wesley, Reading, MA., 1979).

[Il1] L. Ilie, On a conjecture about slender context-free languages, *Theoret. Comput. Sci.* **132** (1994) 427 – 434.

[Il2] L. Ilie, On subwords of infinite words, *Discrete Appl. Math.* **63** (1995) 277 – 279.

[Il3] L. Ilie, On lengths of words in context-free languages, *Theoret. Comput. Sci.*, to appear.

[Il4] L. Ilie, The decidability of the generalized confluence problem for context-free languages, in Gh. Păun, A. Salomaa, eds., *New Trends in Formal Languages. Control, Cooperation, and Combinatorics*, Lecture Notes in Comput. Sci, 1218, Springer, Berlin-New York, 1997, 454 – 464.

[Il5] L. Ilie, Remarks on well quasi orders of words, *Proc. of the 3rd DLT* (Thessaloniki 1997), Aristotle Univ. of Thessaloniki (S. Bozapalidis, ed.), 1998, 399 – 411.

[Il6] L. Ilie, Generalized factors of words, *Fund. Inform.* **33** (1998) 239 – 247.

[IlSa] L. Ilie and A. Salomaa, On well quasi orders of free monoids, *Theoret. Comput. Sci.* **204** (1998) 131 – 152.

[Is] G. Istrate, Some remarks on almost periodic sequences and languages, in vol. *Mathematical Linguistics and Related Topics* (Gh. Păun, ed.), The Publ. House of the Romanian Academy, Bucharest, 1994, 191 – 194.

[Ko] T. Koshiba, On a hierarchy of slender languages based on control sets, *Fund. Inform.* **31** (1997) 41 – 47.

[Kr1] J. B. Kruskal, The theory of well-partially-ordered sets, Doctoral Thesis, Princeton, 1954.

[Kr2] J. B. Kruskal, The theory of well-quasi-ordering; a frequently discovered concept, *J. Combin. Theory Ser. A* **13** (1972) 297 – 305.

[Lo] M. Lothaire, *Combinatorics on Words* (Addison-Wesley, Reading, MA., 1983).

[Mak] G. S. Makanin, The problem of solvability of equations in a free semigroup, *Mat. Sbornik* **103** (1977) 147 – 236. (English transl. in *Math. USSR Sbornik* **32** (1977) 129 – 198.)

[MaPa1] S. Marcus and Gh. Păun, Infinite (almost periodic) words, formal languages and dynamical systems, *EATCS Bull.* **54** (1994) 224 – 231.

[MaPa2] S. Marcus and Gh. Păun, Infinite words and their associated formal languages, in: C. Calude, M. J. J. Lennon, H. Maurer (eds.), *Salodays in Auckland* (Auckland Univ. Press, 1994), 95 – 99.

[Mat] Y. Matiyasevich, Enumerable sets are diophantine, *Soviet Math. Doklady* **11** (1970) 354 – 357. (English transl. *Dokl. Akad. Nauk. SSSR* **191** (1971) 279 – 282.)

[Mo] M. Morse, Recurrent geodesics on a surface of negative curvature, *Trans. Amer. Math. Soc.* **22** (1921) 84 – 100.

[NW] C. Nash-Williams, On well quasi-ordering finite trees, *Proc. Cambridge Philos. Soc.* **59** (1963) 833 – 835.

[NiSa] T. Nishida and A. Salomaa, Slender 0L languages, *Theoret. Comput. Sci.* **158** (1996) 161 – 176.

[Pa] R. J. Parikh, On context-free languages, *J. Assoc. Comput. Mach.* **13** (1966) 570 – 581.

[PaSa1] Gh. Păun and A. Salomaa, Thin and slender languages, *Discrete Appl. Math.* **61** (1995) 257 – 270.

[PaSa2] Gh. Păun and A. Salomaa, Decision problems concerning the thinness of D0L languages, *EATCS Bull.* **46** (1992) 171 – 181.

[PaSa3] Gh. Păun and A. Salomaa, Closure properties of slender languages, *Theoret. Comput. Sci.* **120** (1993) 293 – 301.

[Po] E. Post, A variant of a recursively unsolvable problem, *Bull. Amer. Math. Soc.* **53** (1946) 264 – 268.

[Ra1] D. Raz, On slender context-free languages, *Proc. of the 12th STACS*, (Munich 1995) Lecture Notes in Comput. Sci., 900, Springer, Berlin-New York, 1995, 445 – 454.

[Ra2] D. Raz, Length considerations in context-free languages, *Theoret. Comput. Sci.* **183** (1997) 21 – 32.

[Rog] H. Rogers Jr., *Theory of Recursive Functions and Effective Computability* (Mc-Graw-Hill, New York, 1967).

[Ros] K. Rosen, *Elementary Number Theory and its Applications* (Addison-Wesley, Reading, MA., 1988).

[RoSa] G. Rozenberg and A. Salomaa, *Cornerstones of Undecidability* (Prentice Hall, New York, London, 1994).

[Sa1] A. Salomaa, *Theory of Automata* (Pergamon Press, Oxford, 1969).

[Sa2] A. Salomaa, *Formal Languages* (Academic Press, New York, 1973).

[Sh] J. Shallit, Numeration systems, linear recurrences, and regular sets, *Inform. and Comput.* **113** (1994) 331 – 347.

[SYZS] A. Szilard, S. Yu, K. Zhang and J. Shallit, Characterizing regular languages with polynomial densities, *Proc. of the 17th MFCS* (Prague 1992) Lecture Notes in Comput. Sci., 629, Springer, Berlin-New York, 1992, 494 – 503.

[Th] A. Thue, Über unendliche Zeichenreihen, *Norske Vid. Selsk. Skr., I. Mat. Nat. Kl.*, Kristiania, 7 (1906), 1 – 22.

[vL] J. van Leeuwen, Effective constructions in well-partially-ordered free monoids, *Discrete Math.* **21** (1978) 237 – 252.

[We] W. Wechler, *Universal Algebra for Computer Scientists*, EATCS Monographs on Theoretical Computer Science (Springer-Verlag, Berlin, Heidelberg, 1991).

[YZS] S. Yu, Q. Zhuang, and K. Salomaa, The state complexities of some basic operations on regular languages, *Theoret. Comput. Sci.* **125** (1994) 315 – 328.

Turku Centre for Computer Science
Lemminkäisenkatu 14
FIN-20520 Turku
Finland

http://www.tucs.abo.fi

University of Turku
- **Department of Mathematical Sciences**

Åbo Akademi University
- **Department of Computer Science**
- **Institute for Advanced Management Systems Research**

Turku School of Economics and Business Administration
- **Institute of Information Systems Science**