Logical Methods in Computer Science
Vol. 11(1:9)2015, pp. 1–26
www.lmcs-online.org

Submitted Jan. 9, 2014
Published Mar. 25, 2015

COMPLEXITY OF PROBLEMS OF COMMUTATIVE GRAMMARS *

ERYK KOPCZYŃSKI

Institute of Informatics, University of Warsaw
e-mail address: erykk@mimuw.edu.pl

ABSTRACT. We consider commutative regular and context-free grammars, or, in other words, Parikh images of regular and context-free languages. By using linear algebra and a branching analog of the classic Euler theorem, we show that, under an assumption that the terminal alphabet is fixed, the membership problem for regular grammars (given v in binary and a regular commutative grammar G, does G generate v?) is P, and that the equivalence problem for context free grammars (do G_1 and G_2 generate the same language?) is in Π_2^{P}.

1. INTRODUCTION

Let Σ be a finite alphabet. By Σ^* we denote the set of words over Σ, or finite sequences of elements of Σ. For a word $w \in \Sigma^*$, by $\Psi(w)$ (the Parikh image of w) we denote the function from Σ to non-negative integers $\mathbb{N}$, such that each $x \in \Sigma$ appears $\Psi(w)(x)$ times in w. For a language $L \subseteq \Sigma^*$, $\Psi(L) = \{\Psi(w) : w \in L\} \subseteq \mathbb{N}^\Sigma$.

Context free and regular languages are one of the most important classes of languages in computer science [HU79]. By a famous result of Parikh [Par66], a subset of $\mathbb{N}^\Sigma$ is a Parikh image of a context free language if and only if it is a *semilinear set*, or a union of finitely many *linear sets*.

In this paper, we explore the complexity of various problems related to Parikh images of context free languages, such as the following:

- Membership: Given a context-free grammar G and $v \in \mathbb{N}^\Sigma$ (given in binary). Is v a member of the $\Psi(G)$, the Parikh image of the language generated by G?
- Universality: Given two context-free grammars G, is $\Psi(G)$ equal to $\mathbb{N}^\Sigma$?
- Inclusion: Given two context-free grammars G_1 and G_2, does $\Psi(G_1) \subseteq \Psi(G_2)$?
- Equality: Given two context-free grammars G_1 and G_2, does $\Psi(G_1) = \Psi(G_2)$?
- Disjointness: Given two context-free grammars G_1 and G_2, is $\Psi(G_1) \cap \Psi(G_2)$ nonempty?

Since in this paper we are never interested in the order of terminals or non-terminals, we treat everything in a commutative way. This allows us to identify the commutative languages (subsets of Σ^*) with their Parikh images (subsets of $\mathbb{N}^\Sigma$).

2012 ACM CCS: [**Theory of computation**]: Models of computation—Abstract machines; Formal languages and automata theory—Grammars and context-free languages.

Key words and phrases: Euler theorem, Parikh image, commutative grammar, fixed alphabet, equivalence, membership.

* This paper is the full version of the author's part of [KT10], with some differences.

DOI:10.2168/LMCS-11(1:9)2015

In the non-commutative case, the size of alphabet usually does not matter very much: larger alphabets can be encoded as words over smaller alphabets, for example the alphabet $\{a, b, c\}$ can be encoded as $\{b, ba, baa\}$ or $\{a, ba, bb\}$. This changes in the commutative case: each new letter in the alphabet literally brings a new dimension to the Parikh image. If we do not fix the size of the alphabet, it can be easily shown that even the membership problem for regular languages is NP complete.

There are many practical uses of regular and context-free languages which do not care about the order of the letters in the word. For example, when considering regular languages of trees, we might be not interested in the ordering of children of a given node. [BM99] and [NS99] consider XML schemas allowing marking some nodes as unordered.

Some complexity results regarding semilinear sets and commutative grammars have been obtained by D. Huynh [Huy80, Huy85], who has shown that equivalence is Π_2^{P}-hard both for semilinear sets and commutative grammars (where Π_2^{P} is the dual of the second level of the polynomial-time hierarchy, [Sto76]).

Some research has also been done in the field of communication-free Petri nets, or Basic Parallel Processes (BPP). We say that a Petri net ([Pet81], [Rei85]) is communication-free if each transition has only one input. This restriction means that such a Petri net is essentially equivalent to a commutative context-free grammar. [Yen97] shows that the reachability equivalence problem for BPP-nets can be solved in $DTIME\left(2^{2^{ds^3}}\right)$, where d is a constant and s is the size of the problem instance. For general Petri nets, reachability (membership in terms of grammars) is decidable [Kos82, Ler10], although the known algorithms require non-primitive recursive space; and reachability equivalence is undecidable [Hac75]. Also, some harder types of equivalence problems are undecidable for BPP nets [Hüt94]. See [EN94] for a survey of decidability results regarding Petri nets.

As mentioned above, we will assume in this paper that the alphabet is fixed. We have the following two main results:

- The membership problem for commutative regular languages over a fixed alphabet is in P. This was open for a long time (even for a binary alphabet), until it was solved independently by the author of this paper and Anthony Widjaja Lin. This result, and its applications, was presented as a merged paper at the LICS conference [KT10].
- The equivalence problem for commutative context-free languages over a fixed alphabet is in Π_2^{P}. As far as we know, there have been no successful previous attempts in this direction, except for the much simpler case where the alphabet has only one symbol [Huy84].

In context free grammars, usually the order of transitions used in a derivation is important: we are never allowed to use a non-terminal which has not yet been produced. For example, a transition $X \to aX$ allows us to produce an arbitrary amount of the terminal symbol a, but only if we have access to the non-terminal symbol X. However, derivations also can be defined commutatively. The well known Euler's theorem gives a necessary and sufficient condition for whether there is a path or cycle in a graph which uses each edge e exactly n_e times: the condition says that each vertex has to be entered and left exactly the same number of times (*Euler condition*) and reachable from the starting vertex (*connectedness*). Thus, we can forget the order of edges on such a path or cycle, count them, and just check that the conditions are satisfied; a roughly similar approach is used in the definition of a cycle in the construction of homology groups in algebraic topology [Hat02]. The same approach works in our more general case — we can define a *commutative run* and a *commutative cycle* by counting the number of times each transition has been used, and just like in Euler's theorem, a very simple condition can be used to check whether such a function from transitions to integers is indeed a Parikh image of a valid derivation. Similar technique has been used previously in [Esp97]; it also has been used successfully to solve an open problem in database theory [Kop11].

Although we are not allowed to use a non-terminal which has not yet been produced, our framework straghtforwardly allows the following interesting generalization: we allow our terminal symbols to be produced in negative quantities. In this case, these negative productions do not have to be balanced by positive productions. Thus, our commutative languages are interpreted as subsets of $\mathbb{Z}^\Sigma$ rather than $\mathbb{N}^\Sigma$. Although such languages do not commonly appear in the theory of languages, they turn out to be interesting and useful. For example, such negative production allow to reduce disjointness of A and B to membership very easily – just check for membership of 0 in $A - B = \{a - b : a \in A, b \in B\}$ (see Proposition 5.12 below) . This generalization was also discovered independently by Anthony Widjaja Lin.

1.1. **Related papers.** This paper is the full version of [KT10], with the following major differences:

- [KT10] is a result of merging of two submissions. This paper includes only results obtained by the author.
- Equivalence of commutative context-free grammars (Theorem 3.2) has been generalized to the case where we allow terminal symbols to be produced in negative quantities.
- Universality of commutative context-free languages has been only proven to be in Π_2^{P} in [KT10]. Here, we prove that it is in fact Π_2^{P}-complete.
- Full proofs are included.

1.2. **Structure of the paper.** The paper is structured as follows.

Section 2 introduces basic definitions and facts from linear algebra and language theory, and then presents bounds on the size of runs and cycles of commutative grammars. The techniques used to obtain these bounds are almost the same for regular and non-regular grammars, but the results are much stronger in the regular case. This section culminates in Theorem 2.13, which gives a compact representation of a commutative regular language (equivalently, a Parikh image of a regular language). This compact representation will be used in Section 5 to solve the membership problem of commutative regular languages in P.

The whole Section 3 presents a “window theorem” (Theorem 3.2) for (non-regular) commutative grammars. This theorem roughly says that, in order to decide whether two grammars are equivalent, we only have to look at a window of exponential size, and it will be used in Section 5 to show that the equivalence of commutative grammars is in Π_2^{P}. In fact, most of Section 3 is the proof of Lemma 3.1 about semilinear sets, which does not refer commutative grammars directly; Theorem 3.2 is a simple corollary. Lemma 3.1, as well as some other lemmas in this section, thus could potentially have applications to semilinear sets in general, whether they come from commutative grammars or not.

Section 4 presents a non-regular grammar over a three letter alphabet which has no compact representation similar to one given by Theorem 2.13. We believe this example is of interest, because it shows why we cannot prove Theorem 3.2 using a simpler approach.

Section 5 gives the answers to questions about the complexity of the problems mentioned in the introduction; some results are obtained directly, and for some we need our compact representation and window theorems from Sections 2 and 3.

There is a conclusion in Section 6.

A reader interested only in the first main result (that the membership problem of commutative regular languages in P) only has to read Section 2 (assuming that the grammar is regular) and the respective part of Section 5. Due to the lack of branching, results in subsections 2.3 and 2.4 can be obtained in a much more straightforward way for regular grammars. A reader interested in the second main result (that the equivalence problem for commutative context-free languages over a fixed alphabet is in Π_2^{P}) has to read Section 2 (without concentrating on regular grammars), Section 3, and the respective part of Section 5; Section 4 should also be of interest for such readers.

2. Preliminaries

2.1. **Vectors and matrices, and semilinear sets.** In this subsection we present the basic notation, notions and facts from linear algebra, which will be used throughout the paper.

As usual, we denote the set of integers by $\mathbb{Z}$, the set of non-negative integers by $\mathbb{N}$, the set of rational numbers with $\mathbb{Q}$, and the set of real numbers with $\mathbb{R}$. We also denote the set of non-negative real numbers with $\mathbb{P}$ (we do not use the more standard notation $\mathbb{R}^+$ to avoid double indexing).

We also use the notation $[a, b]$ for the interval of all real numbers between a and b, and $[a..b]$ for the interval of all integers between a and b.

Let X be a finite set. We interpret $\mathbb{N}^X$ as multisets over X: $z \in \mathbb{N}^X$ represents a multiset where each $x \in X$ appears exactly z_x times. For a $x \in X$, by $[x] \in \mathbb{N}^X$ we represent the multiset which represents x: $[x](y) = 1$ iff $y = x$, 0 otherwise. The set $\mathbb{N}^X$ is a subset of $\mathbb{Z}^X$ (intuitively, we allow the elements of X to appear in negative quantities), $\mathbb{Q}^X$, $\mathbb{P}^X$, and $\mathbb{R}^X$.

For $v \in \mathbb{R}^X$, let $|v| = \sum_{x \in X} |v_x|$, and $||v|| = \max_{x \in X} |v_x|$.

We also interpret elements of $\mathbb{R}^X$ as (column) vectors. By $\mathbb{R}^{X \times Y}$ we denote the set of matrices over $\mathbb{R}$ with columns indexed by X, and rows indexed by Y; $A^{X \times Y} \subseteq \mathbb{R}^{X \times Y}$ denotes matrices where all entries are in $A \subseteq \mathbb{R}$. For $M \in \mathbb{R}^{X \times Y}$, we use the notation M_y^x for the coefficient in row $y \in Y$ and column $x \in X$. For $M \in \mathbb{R}^{X \times Y}$, $N \in \mathbb{R}^{Y \times Z}$, and $v \in \mathbb{R}^X$ we define the multiplication in the usual way:

$$Mv \in \mathbb{R}^Y,\ (Mv)_y = \sum_{x \in X} M_y^x v_x$$

$$NM \in \mathbb{R}^{X \times Z},\ (NM)_z^x = \sum_{y \in Y} N_y^x M_z^y$$

For a matrix $M \in \mathbb{R}^{X \times Y}$ and a set $S \subseteq \mathbb{R}^X$, by $MS \subseteq \mathbb{R}^Y$ we denote $\{Ms : s \in S\}$. Similarly, for sets $S_1, S_2 \subseteq \mathbb{R}^X$, we define $S_1 + S_2 = \{s_1 + s_2 : s_1 \in S_1, s_2 \in S_2\}$ and $S_1 - S_2 = \{s_1 - s_2 : s_1 \in S_1, s_2 \in S_2\}$ (the algebraic sum and difference).

For a finite set $S \subseteq \mathbb{R}^X$ and $A \subseteq \mathbb{R}$, we define

$$S^{\oplus A} = \left\{ \sum_{s \in S} s a_s : \forall s\ a_s \in A \right\}.$$

Typically, A will be one of $\mathbb{Z}$ ($S^{\oplus \mathbb{Z}}$ is the additive group generated by S), $\mathbb{N}$ ($S^{\oplus \mathbb{N}}$ is the additive monoid), $\mathbb{R}$ ($S^{\oplus \mathbb{R}}$ the linear space spanned by S), $\mathbb{P}$ ($S^{\oplus \mathbb{P}}$ is the cone spanned by S), or $[0..H]$ (we limit the number of uses of each element of S).

A set $V \subseteq \mathbb{Z}^X$ is **linear** iff it can be written as $V = v_0 + P^{\oplus \mathbb{N}}$, where $v_0 \in \mathbb{Z}^X$ and the set $P \subseteq \mathbb{Z}^X$ is finite. The vector v_0 is called the **base** of V, and the elements of P are called **periods**. Moreover, we say that a linear set V is a **simple linear set** iff the elements of P are linearly independent.

A set $V \subseteq \mathbb{Z}^X$ is **semilinear** iff it is a union of finitely many linear sets.

A semilinear set $V \subseteq \mathbb{Z}^X$ is called a **simple bundle** iff it can be written as $V = W + P^{\oplus \mathbb{N}}$, where $W \subseteq \mathbb{Z}^X$ is finite and the elements of P are linearly independent. We say that a bundle is **bounded** by (B, Y) iff $||v|| \leq B$ for each $v \in W$, and $||p|| \leq Y$ for each $p \in P$.

The following lemmas from the linear algebra will be important for us. We denote the determinant of $M \in \mathbb{R}^{X \times X}$ by $\det M$.

Lemma 2.1. Let $M \in [-C, C]^{X \times X}$ be a matrix. Then the determinant of M is bounded by $H_{|X|}(C) = |X|!C^{|X|}$. Moreover, if $M \in \mathbb{Z}^{X \times X}$, the determinant of M is an integer.

Proof. This follows straightforwardly from the Leibniz formula for determinant. In fact, Hadamard [Had93] has shown a better bound: $H_{|X|}(C) = C^{|X|}|X|^{|X|/2}$. □

Let us recall the well known Cramer's rule [Cra50]:

Lemma 2.2. Let $A \in \mathbb{R}^X_X$ be a non-degenerate matrix, and $b \in \mathbb{R}^X$. Then the system of equations $Av = b$ has a unique solution given by $v_x = \det(A_x)/\det(A)$, where A_x is obtained by replacing the x-th column of A with the vector b. If A is not a non-degenerate matrix, the system of equations either has no solution, or infinitely many solutions.

The following corollary is straightforward:

Lemma 2.3. Let X be a fixed set of indices, and $A \in \mathbb{N}$. There is a number B, polynomial in A, such that whenever the system of equations $Mx = v$, where $M \in [-A..A]^{X \times X}$, $v \in [-A..A]^X$ has a unique solution $x \in \mathbb{R}^X$, we have $x = x'/C$, where $x' \in [-B..B]^X$, and $C \in [-B..B]$.

Proof. We apply the Cramer's rule (Lemma 2.2) and the Hadamard bound (Lemma 2.1). □

Lemma 2.4. Let $M \in [-C..C]^{X \times X}$ be a non-degenerate matrix. Then

$$(\det M)\mathbb{Z}^X \subseteq M\mathbb{Z}^X.$$

Proof. Again, this is immediate from Cramer's rule (Lemma 2.2). The intuition is as follows. For $u, v \in \mathbb{Z}^\Sigma$, we say that $u \equiv v$ iff $u - v \in M\mathbb{Z}^X$. The quotient group $\mathbb{Z}^\Sigma/_\equiv$ has $\det M$ elements (intuitively, for $|X| = 2$, the number of elements is equal to the area of the parallelogram given by columns of M; this intuition also works in other dimensions). Thus, $(\det M)v \equiv 0$. □

Lemma 2.5. Let $P \subseteq [-C..C]^X$ be a linearly dependent set of vectors. Then for some $\alpha \in \mathbb{Z}^P$ we have $\sum_{v\in P} \alpha_v v = 0$, where $||\alpha|| \leq H_{|X|}(C)$, and $\alpha_v > 0$ for some $v \in P$.

Proof. Without loss of generality we can assume that P is a minimal linearly dependent set. Let $(\beta_v) \in \mathbb{R}^P$ be the set of coefficients for which $\sum_{v\in V} \beta_v v = 0$, and such that $\beta_v \neq 0$ for some $v \in P$. Since P is minimal, we know that (β_v) is unique up to a constant: that is, if (β_v) and (β'_v) have this property, then $\beta'_v = q\beta_v$ for some $q \in \mathbb{R}$.

Let $u \in P$ be the element such that $|\beta_u| \geq |\beta_v|$ for each $v \in P$. Let $P_0 = P - \{u\}$. We know that the $P_0^{\oplus\mathbb{R}} = P^{\oplus\mathbb{R}}$.

Since P was minimal, we know that P_0 is linearly independent set. Let $P_1 \supseteq P_0$ be such that $P_1 \subseteq [-C..C]^X$, and P_1 is a base of $\mathbb{R}^X$. This can be done by extending P_0 by unit vectors which are not yet in the subspace spanned by P_0.

Let $M \in [-C..C]^{X\times X}$ be a matrix whose columns are the elements of P_1. Since P_1 is a base, each vector v can be written as a linear combination of elements of P_1 with real coefficients in a unique way. From Lemma 2.4 we know that for $v = (\det M)u$ the coefficients are integers; moreover, since $(\det M)u \in P^{\oplus\mathbb{R}} = P_0^{\oplus\mathbb{R}}$, the coefficients are 0 for elements of $P_1 \backslash P_0$. Thus, we get $\sum_{v\in P} \alpha_v v = 0$, where $\alpha_u = \det M$ and α_v are integers. Since (β_v) was unique up to a constant, and u was the vector with the largest coefficient, we know that the same holds for (α_v). From Lemma 2.1 we know that $\alpha_v \leq \det M \leq H_{|X|}(C)$ for all $v \in P$. □

Lemma 2.6. Let $P \subseteq [-C..C]^X$ be a finite set of vectors, and $w \in P^{\oplus\mathbb{N}}$. Let $H = H_{|X|}(C)$. Then there is a linearly independent set $P_0 \subseteq P$ such that $w \in P^{\oplus[0..H]} + P_0^{\oplus\mathbb{N}}$. In other words, we can assume that all the periods except ones from P_0 are multiplied by factors bounded by H.

Proof. Assume that $w = \sum_{v\in P} n_v v$. Let $H = H_{|X|}(C)$.

Let $P^* = \{v \in P : n_v \geq H\}$. Induction over the cardinality of P^*.

If P^* is linearly independent, we are done (we can simply set $P_0 = P^*$).

Otherwise, from Lemma 2.5 we know that there is a set of coefficients $(\alpha_v) \in \mathbb{Z}^{P_0}$ such that $\sum_{v\in P_0} \alpha_v v = 0$, and $\alpha_v \leq H$ for each $v \in P_0$, and at least one α_v is positive. For each $v \in P_0$, we subtract α_v from n_v; this does not effect the equation $w = \sum_{v\in P} n_v v$. We repeat this until we get $n_v - \alpha_v < 0$ for some $v \in P_0$ (this will have to eventually happen, since at least one α_v is positive). Since $n_v - \alpha_v < 0$ and $\alpha_v \leq H$, we get that $n_v < H$. Thus, the new set P^* is a subset of the old one, and we can apply the induction hypothesis. □

2.2. **Commutative grammars, runs, and cycles.** A commutative grammar (see for example [EN94]) is a tuple $G = (\Sigma, Q, q_I, \delta)$, where Σ is a finite alphabet of **terminal symbols**, Q is a set of **non-terminal symbols**, $q_I \in Q$ is the **initial non-terminal symbol**, and $\delta \subseteq Q \times \mathbb{Z}^\Sigma \times \mathbb{N}^Q$ is the **transition relation**. We will write the transition $(q, a, t) \in \delta$ as $q \xrightarrow{a} t$, or using a multiplicative notation: $q \xrightarrow{xy^{-1}} q_1 q_2^2$ denotes the transition $(q, [x] - [y], [q_1] + 2[q_2])$. The intuition here is that our commutative grammar allows us to use a non-terminal symbol q to produce the multiset of terminal and non-terminal symbols, given by a and t; we allow our terminal symbols to be produced in negative quantities. For a transition $d = (q, a, t) \in \delta$, we denote q, a and t by source(d), $\Psi(d)$ and target(d), respectively.

We say that a grammar is **positive** iff $\Psi(d) \geq 0$ for each $d \in \delta$. Note that our definition is more general than the usual one (e.g., [EN94]), since we allow non-positive commutative grammars.

Runs, paths, and cycles are usually defined as sequences or trees of transitions. However, we will define them as multisets of transitions. The next subsection will show how our definitions are related to the usual ones.

Let $R \in \mathbb{N}^\delta$ be a multiset of transitions. Let $\mathrm{source}(R) = \sum_{d\in\delta} R_d\ [\mathrm{source}(d)]$, $\Psi(R) = \sum_{d\in\delta} R_d\ \Psi(d)$, $\mathrm{target}(R) = \sum_{d\in\delta} R_d\ \mathrm{target}(d)$. Intuitively, the number $\mathrm{source}(R)$ counts times each non-terminal symbol has been used, while $\mathrm{target}(R)$ counts the number of times each non-terminal symbol has been produced, and $\Psi(R)$ counts the number of times each terminal symbol has been produced. For $p, q \in Q$ we say that $p \rightarrow_R q$ iff for some $d \in \delta$ such that $R_d > 0$, $p = \mathrm{source}(d)$ and $\mathrm{target}(d)(q) > 0$; i.e., a transition which produces q from p appears in R with a non-zero quantity. By $\rightarrow^*_R$ we denote the transitive reflexive closure of $\rightarrow_R$. For a multiset $s \in \mathbb{N}^Q$ and $q \in Q$, we say that $s \rightarrow^*_R q$ iff $p \rightarrow^*_R q$ for some $p \in s$. By $\mathrm{supp}(R) \subseteq Q$ we denote the set of non-terminals $q \in Q$ such that $(\mathrm{source}(R))_q > 0$.

We say that a multiset of transitions $R \in \mathbb{N}^\delta$ is a **(commutative) subrun** from $s \in \mathbb{N}^Q$ to $t \in \mathbb{N}^Q$ if the following two conditions are satisfied:

- *Euler condition:* $\mathrm{source}(R) - s = \mathrm{target}(R) - t$;
- *Connectivity:* whenever $q \in \mathrm{source}(R)$, we have $s \rightarrow^*_R q$.

Intuitively, in a subrun, for each state q, we start with s_q copies of state q, during the run we use up $\mathrm{source}_q(R)$ copies and produce $\mathrm{target}_q(R)$ more copies, and t_q copies remain at the end. The Euler condition says that these numbers agree. In the case where $s = t$ it means that each state is produced exactly as many times as it is produced, just like in the classic Euler condition which says that each vertex has the same in-degree and out-degree. In the next subsection we will show that, as long as our conditions on a commutative subrun are satisfied, there is an ordering such that we never use up something which has not yet been produced.

The following kinds of subruns will be of most interest for us:

- A **run from** p is a subrun from $[p]$ to 0.
- A **path from** p_1 **to** p_2 is a subrun from $[p_1]$ to $[p_2]$.
- A **cycle from** p is a path from p to p.

By $\Psi(G)$ we denote the (commutative) language of G, or the set of $\Psi(R)$ for all commutative runs R from q_I.

We say that G is **in normal form** iff for each transition (q, a, t) in δ we have $|t| \leq 2$ and $|a| \leq 1$. Moreover, we say that G is **regular** iff for each transition (q, a, t) in δ we have $|t| \leq 1$ and $|a| \leq 1$. If G is not in normal form, it is straightforward to construct a grammar G' in normal form such that $\Psi(G) = \Psi(G')$. This is done by replacing each transition which does not satisfy the restriction by several simpler transitions, adding additional non-terminals. For example, $q \overset{a_1 a_2}{\rightarrow} q_1 q_2 q_3$ is replaced with $q \overset{a_1}{\rightarrow} q_1 q'$ and $q' \overset{a_2}{\rightarrow} q_2 q_3$, where q' is a new non-terminal.

Usually, we will assume that our grammars are in normal form; in this case, the size of the grammar can be described by stating the number of non-terminals N and the size of the alphabet A, since for a grammar in normal form the number of transitions is bounded by $O(N^3 A)$. In the sequel, many results will work both for regular grammars and the general case, but be much stronger in the regular case – as a typical example, for regular grammars we can achieve a polynomial bound, but for non-regular grammars in normal form we can only achieve an exponential one. Both the polynomial bound in the regular case and the exponential bound in the general case will be of interest for us. The notation Γ_G^N introduced in Lemma 2.11 below, which is linear in N for a regular G and exponential for a non-regular G, should make it clear that a result is given both for regular and general grammars, but it is stronger for regular ones.

We say that a cycle C is **simple** iff it is not a sum of two smaller non-zero cycles. We say that a run R is a **skeleton run** iff it cannot be written as $R = R_0 + C$, where C is a cycle, and $\text{supp}(R) = \text{supp}(R_0)$.

2.3. **Commutative subruns can be ordered.** In this subsection we show that our commutative subruns can be ordered correctly, that is, in each non-empty commutative subrun from s we can choose the first transition δ such that the remaining part is a subrun from $s - \text{source}(\delta) + \text{target}(\delta)$; in other words, we can order the transitions in a subrun in such a way that, if we start from s, and each transition consumes $\text{source}(\delta)$ and produces $\text{target}(\delta)$, we can arrange the transitions in a way that we never consume something which has not been produced yet. This result is practically equivalent to Theorem 3.1 in [Esp97], where it has been stated in the setting of communication-free Petri nets. Also note that, for regular grammars and $|s| = 1$, this can be seen as a restatement of the well known Euler's theorem.

Theorem 2.7. If R is a non-empty subrun from s to t, then there is a $\delta \in R$ such that $\text{source}(\delta) \in s$, and $R' := R - \delta$ is a subrun from $s' := s - \text{source}(\delta) + \text{target}(\delta)$ to t.

Proof. There are two cases:

- the relation $\to_R$ has a cycle starting in s, i.e., there is a $p \in s$ such that $p \to_R q$ and $q \to_R^* p$. In this case, there is a $\delta \in R$ such that $\text{source}(\delta) = p$ and $\text{target}(\delta) = q$. We have to show that R' is a subrun from s' to t. Let $r \in \text{source}(R')$. Since R was a subrun from s, we know that $s \to_R^* r$, and we have to show that $s' \to_{R'}^* r$. Take a minimal path π witnessing $s \to_R^* r$. Since π is minimal, only its first transition starts from an element of s, and in particular, can be equal to δ.
  - If π did not start in p, it is still a path for $\to_{R'}$ (δ was not in π).
  - If π started with δ, we just remove δ from π, thus obtaining a path witnessing $s' \to_R^* r$.
  - If π started in p but not with δ, we know that $q \to_R^* p \to_R^* r$, and both subpaths do not contain δ (provided that we take a minimal path $q \to_R^* p$) and thus they are still paths for $\to_{R'}$.
- otherwise, take any δ such that $\text{source}(\delta) = p \in s$. Such a δ must exist from the connectedness condition. Again, we have to show that R' is a subrun from s' to t. Let $r \in \text{source}(R')$. Since R was a subrun from s, we know that $s \to_R^* r$, and we have to show that $s' \to_{R'}^* r$. Take a minimal path witnessing $s \to_R^* r$. Since π is minimal, only its first transition starts from an element of s, and in particular, can be equal to δ. There are four subcases:
  - If π did not start in p, it is still a path for $\to_{R'}$ (δ was not in π).
  - If π started with δ, we just remove δ from π, thus obtaining a path witnessing $s' \to_R^* r$.
  - If π started in p but not with δ, and $s_p \geq 2$, then $s'(p) \geq 1$, so $p \in s'$ and the path is still valid for R'.
  - If π started in p with $\delta' \neq \delta$, and $s_p = 1$, we know that $\text{source}_p(R) - s_p = \text{target}_p(R) - t_p$. Since $\text{source}(\delta) = \text{source}(\delta') = p$ and $s_p = 1$, $\text{target}_p(R) > 0$, and thus there must be a transition δ'' such that $p \in \text{target}(\delta'')$. We know that $s \to_R^* \text{source}(\delta'')$, and the path cannot include δ – otherwise, we get a sequence of paths $s \to_R^* p \to_R \text{target}(\delta) \to_R^*$

source(δ'') $\rightarrow_R p$, and thus we get a cycle, which was dealt with in the previous case. Thus, we have $s \rightarrow^*_{R'}$ source(δ'') $\rightarrow_{R'} p \rightarrow^*_{R'} r$. □

2.4. **Derivation trees.** In this subsection we compare our commutative runs and cycles with the usual derivation trees. We obtain a subrun from a derivation tree simply by counting how many times each transition has been used; using the result of the previous section, we also show that this process can be reversed, i.e., a derivation tree exists for each commutative subrun. We then use the derivation trees to show upper bounds on the size of simple cycles and skeleton runs.

Again, this is much easier for regular grammars — in this case, derivation trees are simply paths, i.e., sequences of transitions $\partial_0, \partial_1, \partial_2, \ldots, \partial_d$ such that [source(∂_{i+1})] equals target(∂_i).

Let G be a commutative grammar in normal form. A **derivation tree** from $p \in Q$ is a tuple $T = (V, v_0, P, \partial)$, such that:

- V is an arbitrary set of vertices,
- $v_0 \in V$ is a special vertex, called the root of V,
- P is a function from $V - \{v_0\}$ to V (parent), such that for each $v \in V$ there is a $n \in \mathbb{N}$ (called the **depth** of v) such that $P^n(v) = v_0$,
- ∂ is a function from V to δ. We will use source(v) and target(v) for source($\partial(v)$) and target($\partial(v)$), respectively.
- source($\partial(v_0)$) $= p$,
- for each $v \in V$, we have $F(v) \geq 0$, where

$$F(v) = \mathrm{target}(v) - \sum_{w : P(w) = v} [\mathrm{source}(w)].$$

We also denote $F(T) = \sum_{v \in V} F(v)$, source($T$) $= \sum_{v \in V}$[source(v)], and target(T) $= \sum_{v \in V}$ target(v).

Intuitively, each vertex v of the derivation tree represents that we are using a non-terminal and produce new terminals and non-terminals, according to the transition $\partial(v)$. Children of v can use the non-terminals produced. $F(v)$ represents the "free" non-terminals which have been produced, but have not been used by children; $F(v) \geq 0$ represents the fact that children cannot use non-terminals which have not been produced.

Let $\mathrm{U}(T) : \delta \rightarrow \mathbb{N}$ be the function counting the number of times each transition has been used in the derivation tree T: $(\mathrm{U}(T))_d = |\{v \in V : \partial(v) = d\}|$.

We say that a derivation tree T from p is **full** iff $F(T) = 0$, and a **path to** p_2 iff $F(T) = [p_2]$. We say that a derivation tree T is **cyclic** from p iff it is a path derivation tree from p to p.

Lemma 2.8. If T is a derivation tree from p, then $\mathrm{U}(T)$ is a commutative subrun from $[p]$ to $F(T)$.

Proof.

$$\begin{aligned} F(T) &= \sum_{v\in V}\left(\mathrm{target}(v) - \sum_{w:P(w)=v}[\mathrm{source}(w)]\right) \\ &= \sum_{v\in V}\mathrm{target}(v) - \sum_{v\in V-\{v_0\}}[\mathrm{source}(v)] \\ &= \mathrm{target}(T) - \mathrm{source}(T) + [\mathrm{source}(v_0)] \\ &= \mathrm{target}(U(T)) - \mathrm{source}(U(T)) + [p]. \end{aligned}$$

We also have that $P(v) \to_{\mathrm{U}(T)} v$ for each $v \in V$, thus $U(T)$ is indeed connected from $P(v_0) = p$. □

Theorem 2.9. If $p \in Q$, and $R \in \mathbb{N}^\delta$ is a commutative subrun from $[p]$ to $t \in \mathbb{N}^Q$, then $R = \mathrm{U}(T)$ for some derivation tree T from p such that $F(T) = t$.

Proof. We will construct the tree T inductively. We start with $T_0 = (V, v_0, P, \partial)$, where $\partial(v_0)$ is δ from Theorem 2.7 for R, and $V = v_0$. We will keep the following invariant: $U(T) \le R$, and $R - U(T)$ is a subrun from $F(T)$ to t. From Theorem 2.7 the invariant is satisfied for T_0. Suppose that the invariant is also satisfied for T, and $U(T) < R$. . From Theorem 2.7 again there exists a $\delta' \in R - U(T)$ such that $\mathrm{source}(\delta') \in F(T)$. Since $\mathrm{source}(\delta') \in F(T)$, there exists a v such that $\mathrm{source}(\delta') \in F(v)$. We obtain a new tree T' by adding a new vertex w such that $P(w) = v$ and $\partial(w) = \delta'$. From Theorem 2.7, the invariant is also satisfied for T'.

Finally, we get a tree T such that $U(T) = R$. From Lemma 2.8 we know that $F(T) = t$. □

Corollary 2.10. The following two conditions are equivalent:
(1) $R \in \mathbb{N}^\delta$ is a commutative run from p,
(2) $R = \mathrm{U}(T)$ for some full derivation tree T from p.

Also, the following two conditions are equivalent:
(1) $C \in \mathbb{N}^\delta$ is a path from p_1 to p_2,
(2) $C = \mathrm{U}(T)$ for some path derivation tree T from p_1 to p_2.

In particular, the following two conditions are equivalent:
(1) $C \in \mathbb{N}^\delta$ is a cycle from p,
(2) $C = \mathrm{U}(T)$ for some cyclic derivation tree T from p.

Proof. Straightforward from Lemma 2.8 and Theorem 2.9. □

Lemma 2.11. Let G be a grammar in normal form. Let D be a derivation tree such that D has no vertices at depth M. Then $|D| < \Gamma_G^M$, where $\Gamma_G^M = M + 1$ if G is regular, and 2^{M+1} otherwise.

Proof. If each vertex has at most C children, then there are at most C^d vertices at depth d. For regular grammars $C = 1$, and for grammars in normal form, $C = 2$. All the hypotheses follow from simple calculations. □

Theorem 2.12. Let G be a grammar in normal form with N non-terminals. If C is a simple cycle, then $|C| < \Gamma_G^N$. If R is a skeleton run, then $|R| < \Gamma_G^{N^2}$.

Proof. First, let C be a simple cycle. From Corollary 2.10 we know that $C = \mathrm{U}(T)$ for some cyclic derivation tree $T = (V, v_0, P, d)$ from p.

We will show that T has no vertex at depth N. Indeed, suppose otherwise that T has a vertex v at depth $n > N$. Consider a branch of the derivation tree: $v, p(v), p^2(v), \ldots, p^n(v) = v_0$. Since $n \geq N$, we have $\mathrm{source}(p_i(v)) = \mathrm{source}(p_j(v))$ for some $i < j$. Let V_1 be all the descendants of $p_i(v)$ (inclusive), V_2 be all the other descendants of $p_j(v)$, and V_3 be all the other vertices. Then $T_1 = (V_2, p_i(v), P, d)$ is a cyclic derivation tree, and so is $T_2 = (V_1 \cup V_3, p_i(v), P^*, d)$, where $P^*(p_i(v)) = P(p_j(v))$ and $P^*(w) = P(w)$ for all other $w \in V_1 \cup V_3$. Hence, C is not a simple cycle ($C = \mathrm{U}(T) = \mathrm{U}(T_1) + \mathrm{U}(T_2)$).

Now, let R be a skeleton run. From Corollary 2.10 we know that $R = \mathrm{U}(T)$ for some full derivation tree $T = (V, v_0, P, d)$ from q_0.

We will show that T has no vertex at depth N^2. Indeed, suppose otherwise that T has a vertex v at depth $n \geq N^2$. Consider a branch of the derivation tree: $v, p(v), p^2(v), \ldots, p^n(v) = v_0$. Since $n \geq N^2$, there are indices $i_0, i_1, \ldots, i_N$ such that $\mathrm{source}(p_{i_k}(v))$ is the same non-terminal q for each k. By repeating the construction above N times, we can decompose $R = R_0 + \sum_{k=1}^{N} C_k$, where R_0 is a run and C_k is the cycle between $p_{i_{k-1}}(v)$ and $p_{i_k}(v)$. Let $U_K = \mathrm{supp}(R_0 + \sum_{k=1}^{K} C_k$. Since $U_0 \subseteq U_1 \subseteq U_2 \ldots \subseteq U_N \subseteq Q$, U_0 is not empty, and $|Q| = N$, there must be k such that $U_k = U_{k-1}$. Since C_k does not add any new non-terminals to the support, we have that $\mathrm{supp}(R - C_k) = \mathrm{supp}(R)$. This contradicts the assumption that R was a skeleton run ($R - C_k$ is a run).

All the hypotheses follow from the Lemma 2.11. □

2.5. **Compact Representation of a Commutative Regular Language.** In this subsection, we will show how to obtain a compact representation of a regular commutative language over a fixed alphabet: such a language is a union of polynomially many polynomially bounded simple bundles. This compact representation will be used in Section 5 below to prove that the membership problem is in P for regular commutative grammars over a fixed terminal language. We also obtain a less compact representation in the non-regular case.

Theorem 2.13. Let G be a grammar in normal form with N non-terminals over an alphabet of size A, and let R be a run. Let $B_G = \Gamma_G^{N^2} + (2\Gamma_G^N)^{1+A} H_A\left(\Gamma_G^N\right)$. Then $\Psi(R) = \Psi(R_1) + \sum \Psi(C_k) n_k$, where:

- R_1 is a run such that $|R_1| \leq B_G$,
- each C_k is a simple cycle from some $q \in \mathrm{supp}(R_1)$,
- $\Psi(C_k)$ are linearly independent,
- $n_k \in \mathbb{N}$.

Proof. Let R be a run. As long as R is not a skeleton run, we can decompose R as a sum of a smaller run, and a simple cycle. Thus, we obtain that $\Psi(R) = \Psi(R_0) + \sum_{k \in K} \Psi(C_k) n_k$.

Suppose that $\Psi(C_k) = \Psi(C_l)$ for some $k \neq l$. Then we remove l from K, and add n_l to n_k. The equation $\Psi(R) = \Psi(R_0) + \sum_{k \in K} \Psi(C_k) n_k$ still holds.

Let $K^* = \left\{k \in K : n_k \geq H_A\left(\Gamma_G^N\right)\right\}$. From Lemma 2.6 we can assume that $\Psi(C_k)$ are linearly independent for $k \in K^*$.

Since for each k we have $|C_k| < \Gamma_G^N$, and $\Psi(C_k)$ is different for each k, we have at most $(2\Gamma_G^N)^A$ cycles there. Let $R_0 = R_1 + \sum_{k \in K - K^*} C_k n_k$. We have $|R_0| \leq |R_1| + (2\Gamma_G^N)^{A+1} H_A\left(\Gamma_G^N\right)$, and $\Psi(R) = \Psi(R_0) + \sum_{k \in K^*} \Psi(C_k) n_k$. □

Corollary 2.14. If G is a regular grammar with N non-terminals over an alphabet of size A, then $\Psi(G)$ is a union of at most N^{A^2} simple bundles bounded by (B_G, N).

Also note that B_G is polynomial for regular grammars over a fixed alphabet.

Proof. From Theorem 2.13 we know that for each run R we have $\Psi(R) = \Psi(R_1) + \sum \Psi(C_k)n_k$, where $\Psi(R_1)$ is bounded by B_G and C_k is a simple cycle. We bundle the runs which use the same cycles together. Since simple cycles are bounded by N, there are at most N^A of them, and there are at most $(N^A)^A$ sets of linearly independent simple cycles. □

It is possible to get a better bound on the number of simple bundles in dimension 2, even for non-regular grammars.

Corollary 2.15. If G is a grammar in normal form with N non-terminals over a two-letter alphabet, then $\Psi(G)$ is a union of $O(N^2)$ simple bundles bounded by (B_G, Γ_G^N).

Proof. Let $\Sigma = \{a_1, a_2\}$. First assume that the grammar is positive.

For each non-terminal q, among all the cycles from q, let $C_i(q)$ be the one with the greatest proportional amount of a_i, for $i = 1, 2$. All the other cycles from q fall in the angle between $C_1(q)$ and $C_2(q)$.

Now, for each run R, let $C_i(R)$ be the one with the greatest proportional amount of a_i, among all cycles from $q \in \mathrm{supp}(R)$.

Proceed as in the proof of Corollary 2.14, except now we can assume that the cycles C_1, C_2 are always $C_1(R)$ and $C_2(R)$ (instead of arbitrary cycles), as all other cycles can be written as positive linear combinations of these two. Since $C_1(R)$ and $C_2(R)$ are chosen from $C_i(q)$, there at at most N^2 simple bundles.

For non-positive grammars, for each non-terminal q, one of the two cases holds:

- all cycles from q (and their positive combinations) fall in the angle between the two extreme cycles $C_1(q)$ and $C_2(q)$,
- positive linear combinations of cycles from q cover the full plane, and can be written as positive combinations of $C_1(q)$, $C_2(q)$ and $C_3(q)$ which are cycles from q.

Thus, we can always use one of the at most three cycles $C_i(q)$ for some $q \in \mathrm{supp}(R)$. There are still $O(N^2)$ combinations of them. □

3. Window Theorem for Commutative Grammars

This section is devoted to the following result, which we call the *window theorem*: in order to determine whether two commutative languages $\Psi(G_1)$ and $\Psi(G_2)$ over a fixed alphabet Σ of size A defined by grammars G_1 and G_2 with N non-terminals are disjoint or equal, it is enough to only examine a small window of size which is single exponential in N. This result will be instrumental in the proof that inclusion, universality, and disjointness problems are in Π_2^{P} for commutative grammars over an alphabet of fixed size (Theorem 5.5 below).

The situation is much easier for $A = 2$ than in the general case. From Corollary 2.15 we know that each of $\Psi(G_1)$ and $\Psi(G_2)$ is a union of $O(N^2)$ simple bundles bounded by (B_G, Γ_G^N). In this case, we can show the following result: for each $v \in \mathbb{Z}^\Sigma$ we can find a v_0 of single exponential magnitude such that v_0 is in exactly the same of our $O(N^2)$ bundles as v. This is achieved in Lemma 3.5 in the following way:

- Let $W + \mathcal{Z}^{\oplus\mathbb{N}}$ be one of the bundles. From Lemma 2.4 we know that each member of $(\det M)\mathbb{Z}^\Sigma$, where M is the matrix whose columns are $\mathcal{Z}$, is a member of $\mathcal{Z}^{\oplus\mathbb{Z}}$. Hence, if $w \in (\det M)\mathbb{Z}^\Sigma$, then $v \in W + \mathcal{Z}^{\oplus\mathbb{Z}}$ iff $v + w \in W + \mathcal{Z}^{\oplus\mathbb{Z}}$. Since there is just a polynomial number of bundles, we can use the least common multiple of the determinants to ensure that the equivalence above is satisfied for each bundle.
- The bundle is $W + \mathcal{Z}^{\oplus\mathbb{N}}$, not $v \in W + \mathcal{Z}^{\oplus\mathbb{Z}}$, thus we need to ensure that the signum of the coefficients remains unchanged. This is done by partitioning $\mathbb{R}^\Sigma$ into *regions*. Two points v, v_0 are in the same region if $v \in w + \mathcal{Z}^{\oplus\mathbb{P}}$ iff $v_0 \in w + \mathcal{Z}^{\oplus\mathbb{P}}$ for each $w, \mathcal{Z}$ satisfying the necessary bounds. We show that we can additionally ensure that the single exponentially bounded v_0 is in the same region as v, which proves the lemma.

However, this approach no longer works for $A > 2$, as $\Psi(G_i)$ no longer needs to be a union of polynomial number of simple bundles; section 4 below is devoted to showing an example of such a grammar.

We solve this problem by using the regions again. Although $\Psi(G_i)$ need not be a union of a polynomial number of simple bundles, this is true when we consider regions separately: for each region r, $\Psi(G_i) \cap r$ equals $U \cap r$, where U is a union of polynomial number of simple bundles. This is proven in Lemma 3.4 below.

The rest of this section provides a detailed statement and proof of our window theorem. In fact, we will prove the following Lemma about semilinear sets, without directly using the assumption that our semiliner sets come from commutative grammars; the window theorem about commutative grammars (Theorem 3.2 below) will follow easily from it.

Lemma 3.1. Let $C \in \mathbb{N}$. Let S_1 and S_2 be two semilinear sets over the same fixed alphabet Σ of size A, given as $S_k = \bigcup_{i \in I_k} W_i + \mathcal{Z}_i^{\oplus\mathbb{N}}$, where $W_i \subseteq [-C..C]^\Sigma$, $\mathcal{Z}_i \subseteq [-Y..Y]^\Sigma$, for every $i \in I_1 \cup I_2$. Then there exists a number $B_{3.1} = O((C+Y)^{|I_1 \cup I_2|})$ such that $S_1 \subseteq S_2$ iff $S_1 \cap [-B_{3.1}..B_{3.1}]^\Sigma \subseteq S_2 \cap [-B_{3.1}..B_{3.1}]^\Sigma$, and S_1 is disjoint with S_2 iff $S_1 \cap [-B_{3.1}..B_{3.1}]^\Sigma$ is disjoint with $S_2 \cap [-B_{3.1}..B_{3.1}]^\Sigma$.

Theorem 3.2. Let G_1 and G_2 be two commutative grammars in normal form with at most N non-terminals each, over the same fixed alphabet Σ of size A. Then there exists a number $B_{3.2}$ which is single exponential in N, such that $\Psi(G_1) \subseteq \Psi(G_2)$ iff $\Psi(G_1) \cap [-B_{3.2}..B_{3.2}]^\Sigma \subseteq \Psi(G_2) \cap [-B_{3.2}..B_{3.2}]^\Sigma$, and $\Psi(G_1)$ is disjoint with $\Psi(G_2)$ iff $\Psi(G_1) \cap [-B_{3.2}..B_{3.2}]^\Sigma$ is disjoint with $\Psi(G_2) \cap [-B_{3.2}..B_{3.2}]^\Sigma$.

Proof of Theorem 3.2. Let $G_u = (\Sigma, Q_u, q_u, \delta_u)$. Without loss of generality we can assume that the sets Q_u are disjoint for $u \in \{1, 2\}$; this way, we will be able to identify the grammar of each run and cycle by mentioning one of the non-terminals used.

For $q \in Q_u$, let $\mathcal{Z}_q = \{\Psi(C) : C \text{ is a simple cycle from } q\}$, and for $S \subseteq Q_u$, let $\mathcal{Z}_S = \bigcup_{q \in S} \mathcal{Z}_q$.

From Theorem 2.13 we know that $v \in \Psi(G_u)$ iff there exists a run R_1 from q_u such that $|R_1|$ is bounded by B_G and a linearly independent set of simple cycle outputs $P \subseteq \mathcal{Z}_{\mathrm{supp} R_1}$ such that $v \in \Psi(R_1) + P^{\oplus\mathbb{N}}$.

In particular, we know that P is of cardinality at most A. Let I_u be the set of all subsets of Q_u of cardinality at most A. Thus, we know that $v \in \Psi(G_u)$ iff there exists $J \in I_u$, a run R_1 of G_u such that $J \subseteq \mathrm{supp}(R_1)$ and $|R_1|$ is bounded by B_G, and a set $P \subseteq \mathcal{Z}_J$ such that $v \in \Psi(R_1) + P^{\oplus\mathbb{N}}$. Let $W_J = \{\Psi(R) : R \text{ is a run in } G_u \text{ bounded by } B_G \text{ such that } J \subseteq \mathrm{supp}(R)\}$. Thus, we get that $\Psi(G_u) = \bigcup_{J \in I_u} (W_J + \mathcal{Z}_J^{\oplus\mathbb{N}})$. We know that W_J is bounded by B_G which is single exponential, and $\mathcal{Z}_J$ is bounded by $Y = \Gamma_G^N$, which is also single exponential.

Since the cardinality of I_u is polynomial in N, we get our claim by applying Lemma 3.1. □

Proof of Lemma 3.1. We will need to introduce the notion of a *region*.

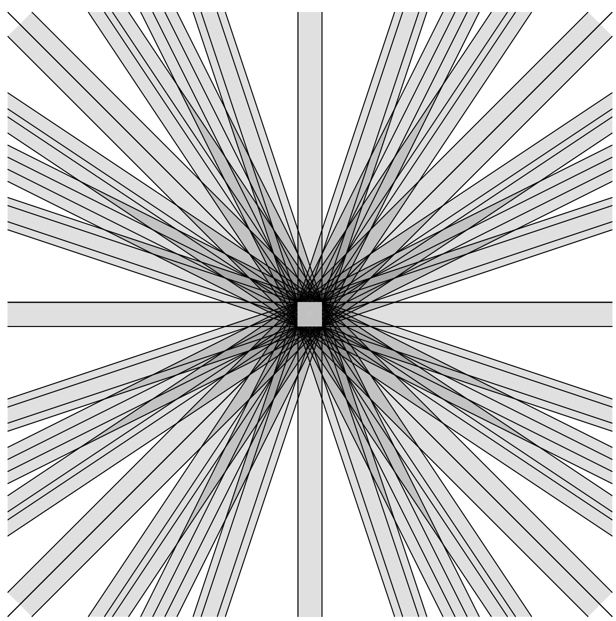
Let $\mathcal{S}_Y$ be the family of all linearly independent subsets of $[-Y..Y]^\Sigma$ containing $A-1$ elements. For each element S of $\mathcal{S}_Y$ and $v \in \mathbb{R}^\Sigma$, let $\phi_S(v)$ be the determinant of the matrix $M_{S,v}$ whose $A-1$ columns are the elements of S (ordered in an arbitrary way), and the last column is v. From the properties of the determinant, we know that $\phi_S(v) = \sum_{a\in\Sigma} \alpha_a v_a$, where α_a is the (possibly negated) determinant of the $(A-1)\times(A-1)$ submatrix of $M_{S,v}$ which misses the column v and the row a. In particular, $\alpha_a \in [-C_\phi..C_\phi]$, where $C_\phi = H_{A-1}(Y)$. Intuitively, for each S, $\phi_S^{-1}(0)$ is the $A-1$-dimensional subspace containing all elements of S. By calculating $\phi_S(v)$ we can tell whether v is above or below this subspace. Let $\Phi_Y = \{\phi_S : S \in \mathcal{S}_Y\}$. The cardinality of Φ_Y is bounded by $(2Y+1)^{A^2}$, and its elements have coefficients bounded by C_ϕ. In particular, if $\phi \in \Phi_Y$, then $|\phi(v)| \leq C_\phi |v|$, and $||\phi(v)|| \leq AC_\phi ||v||$.

Now, let $L_B = [-2B, 2B]^\Sigma$, and $\mathcal{R}_{B,Y}$ be the set of all functions from $\Phi_Y \times L_B$ to $\{-1, 0, 1\}$.

For $r \in \mathcal{R}_{B,Y}$, we define strict regions $\mathrm{Reg}(r)$, weak regions $\mathrm{reg}(r)$, and ray regions $\tau(r)$ as follows:

$$\begin{aligned}
\mathrm{Reg}(r) &= \left\{x \in \mathbb{R}^\Sigma : \forall \phi \in \Phi_Y\ \forall l \in L_B\ \mathrm{sgn}(\phi(x-l)) = r_{\phi,l}\right\} \\
\mathrm{reg}(r) &= \left\{x \in \mathbb{R}^\Sigma : \forall \phi \in \Phi_Y\ \forall l \in L_B\ \mathrm{sgn}(\phi(x-l))\ r_{\phi,l} \geq 0\right\} \\
\tau(r) &= \left\{x \in \mathbb{R}^\Sigma : \forall \phi \in \Phi_Y\ \forall l \in L_B\ \mathrm{sgn}(\phi(x))\ r_{\phi,l} \geq 0\right\}
\end{aligned}$$

The following picture shows what is going on for $A = 2$ and $Y = 3$.



The gray square in the center is L_B. From each point of the square, and for each tuple of $A-1$ vectors in $[-Y..Y]^\Sigma$, we shoot a hyperplane which is parallel to each of these vectors. For $A = 2$, this means that we shoot a line in each direction given by some vector in

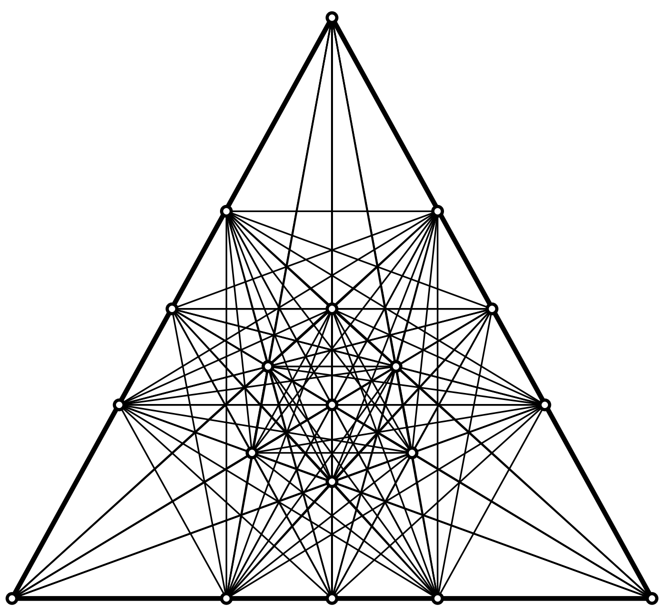
$[-Y..Y]^\Sigma$. There are 32 directions, and for each of them we get a bundle of lines, by taking different points of L_B to shoot from; such bundles of lines are marked gray on the picture above. Each strict region $\mathrm{Reg}(r)$ is given by its relation to each of these lines (above, below, or on one of these lines). There are four types of regions:

- empty regions (the relations are inconsistent),
- a single point (r says that the point is on two (non-parallel) lines at once),
- a semiline (r says that the point is on a line and above some other line),
- a wedge (r says that the point is below line l_1 and above line l_2, where l_1 and l_2 are not parallel).

There are 32 wedge-shaped regions, an infinite number of semiline-shaped regions (bundled into 32 packs of parallel semilines), and an infinite number of points.

Strict regions $\mathrm{Reg}(r)$ are disjoint and partition the plane, while the weak regions $\mathrm{reg}(r)$ are their closures. Ray regions $\tau(r)$ look similar to $\mathrm{Reg}(r)$, but we shoot lines only from 0, instead of each point from L_B. There are 32 wedge-shaped and 32 semiline-shaped regions, each of them start at 0 (which is included into each $\tau(r)$). If $\mathrm{Reg}(r)$ is a single point or the empty set, then $\tau(r) = \{0\}$.

The situation is more complicated in three dimensions, and it is also harder to draw. We will show how $\tau(r)$ looks for $A = 3$ and $Y = 2$. From the definition of $\tau(r)$ we know that for each $\alpha > 0$ we have $x \in \tau(r)$ iff $\alpha x \in \tau(r)$. Thus, it is sufficient to draw only a situation on a planar section of $\mathbb{R}^\Sigma$. Let $\Delta = \{v \in \mathbb{R}^\Sigma : v \geq 0, |v| = 1\}$. The set Δ is a triangle; this triangle is shown on the picture below.



The 19 white dots on the picture denote elements of $[0..Y]^\Sigma$; for each such v, its projection $\frac{v}{|v|}$ is shown as a white dot on the triangle (in fact, we should also have white dots for elements of $[-Y..Y]^\Sigma$ with negative coordinates, but this would obfuscate the picture too much — their projections on our planar section do not fit in the triangle). $\mathcal{S}$ is the set of pairs of linearly independent elements of $[-Y..Y]^\Sigma$, or pairs of distinct white dots. For each $S \in \mathcal{S}$, we have the linear function ϕ_S, which equals 0 on both elements of $\mathcal{S}$. These are represented by the black lines: for each two white dots, we have a black line going through these white dots.

Now, let $r \in \mathcal{R}_{B,Y}$. The set $\tau(r)$ is the set of points which are on the side of the semispace ϕ declared by $r_{\phi,l}$, for each ϕ and l (or on this semispace, in case if $r_{\phi,l} = 0$). In our case, semispaces are the black lines. Therefore, $\tau(r) \cap \Delta$ could be either an empty set (if this is inconsistent – this happens for example when $r_{\phi,l_1} \neq r_{\phi,l_2}$), or a region of the triangle bounded by the black lines, or one of the white dots, or one of the points where the black lines cross, or a segment of a black line between two points where it crosses the other lines. The set $\tau(r)$ can be described by multiplying $\tau(r) \cap \Delta$ by each scalar $\alpha \geq 0$. Thus, each black line becomes a black plane, each white dot or black line crossing becomes a semiline, each black line segment becomes an infinite triangle, and each bounded region becomes an infinite cone.

Regions $\mathrm{reg}(r)$ and $\mathrm{Reg}(r)$ are harder to visualize — we have to replace each black plane by a pack of black planes by moving it by each vector in L_B, and again split $\mathbb{R}^\Sigma$ by these new black planes.

The following lemma describes the shape of the regions.

Lemma 3.3. Let $r \in \mathcal{R}_{B,Y}$. Then:

- $\tau(r)$ can be written as $T_r^{\oplus\mathbb{P}}$, where $T_r \subseteq [-C_{3.3}..C_{3.3}]^\Sigma$,
- $\mathrm{reg}(r) = W_r + \tau(r)$, where $W_r \subseteq [-B_{3.3}, B_{3.3}]^\Sigma$.

In particular, if $\mathrm{reg}(r)$ is bounded, then $\tau(r) = \{0\}$. $C_{3.3}$ is bounded polynomially by Y, and $B_{3.3}$ is bounded polynomially by B and Y.

For example, let us consider the two-dimensional case from the picture on page 14. Take r such that $\mathrm{reg}(r)$ is an infinite wedge with a vertex in some point w. Then $\tau(r)$ is a congruent wedge with a vertex in point 0, bounded by semilines parallel to some two vectors $v_1, v_2 \in [-Y..Y]^\Sigma$. T_r will be simply $\{v_1, v_2\}$, and W_r will be $\{w\}$. Our lemma says that w is bounded polynomially in B and Y; since the picture is big enough to already show all the 32 wedge vertices, we can take $B_{3.3}$ to be (half of) the edge of the square shown on the picture. In the three-dimensional case the situation will be more difficult: elements ot T_r will be bounded by a polynomial in Y (instead of Y itself), W_r will no longer have to be a singleton, and the cardinality of set T_r might be relatively large – for example, on the picture on page 15 there are ray regions $\tau(r) \cap \Delta$ bounded by five black lines, so we will need a T_r with five elements to describe the respective $\tau(r)$.

Having introduced the regions, we can use the following lemma:

Lemma 3.4. Let $B, Y \in \mathbb{N}$. Then there is a $C_{3.4}$, bounded polynomially in B and Y, such that the following holds:

Let $S = W + \mathcal{Z}^{\oplus\mathbb{N}}$, where $\mathcal{Z} \subseteq [-Y..Y]^\Sigma$ and $W \subseteq [-B..B]^\Sigma$, and $r \in \mathcal{R}_{B,Y}$. Then there is a linearly independent subset $\mathcal{Z}_0 \subseteq \mathcal{Z}$ and $W_1 \subseteq [-C_{3.4}..C_{3.4}]^\Sigma$ such that $S \cap \mathrm{reg}(r) = (W_1 + \mathcal{Z}_0^{\oplus\mathbb{N}}) \cap \mathrm{reg}(r)$.

This lemma says that, if we restrict ourselves to a single region, we can improve our presentation $S_u = \bigcup_{i \in I_u} (W_i + \mathcal{Z}_i^{\oplus\mathbb{N}})$ by making the sets $\mathcal{Z}_i$ linearly independent, while still keeping the bound on W_i exponential. Thus, in each region, S_u is a union of $|I_u|$ simple bundles bounded by $(C_{3.4}, Y)$.

Now, we use the following lemma:

Lemma 3.5. Let Σ be a fixed alphabet, $C_{3.4}, Y \in \mathbb{N}$, and I be a set of indices. Then there exists a number $B_{3.5} = O((C_{3.4}+Y)^{|I|})$ such that the following holds:

Let $V_i = W_i + \mathcal{Z}_i^{\oplus\mathbb{N}}$ be a simple bundle bounded by $(C_{3.4}, Y)$, for each $i \in I$, and $r \in \mathcal{R}_{C_{3.4},Y}$. Then for each $v \in \mathbb{Z}^\Sigma \cap \mathrm{Reg}(r)$ there exists a $v_0 \in [-B_{3.5}, B_{3.5}]^\Sigma \cap \mathrm{Reg}(r)$ such that for each $i \in I$ we have $v \in V_i$ iff $v_0 \in V_i$.

By applying the lemma above to $I = I_1 \cup I_2$ and taking $B_{3.1} = B_{3.5}$, we obtain our hypothesis. □

Proof of Lemma 3.3. Without loss of generality we can assume that $\mathrm{Reg}(r) \geq 0$. Indeed, for each $a \in A$, The set Φ_Y includes the linear function ϕ_a given by $\phi_a(x) = x_a$. If $r_{\phi,0} \geq 0$, then $x_a \geq 0$ for all $x \in \mathrm{Reg}(r)$; similarly if $r_{\phi,0} \leq 0$. In the second case, we can take the mirror image, which is also a region.

Take $\phi \in \Phi_Y$. We can write the definitions of $\mathrm{reg}(r)$ and $\tau(r)$ as follows: $\mathrm{reg}(r) = \bigcap_\phi H_\phi^{\mathrm{reg}}$, $\tau(r) = \bigcap_\phi H_\phi^\tau$, where

$$H_\phi^{\mathrm{reg}} = \bigcap_{l \in L_B} \left\{x \in \mathbb{R}^\Sigma :\ \mathrm{sgn}(\phi(x-l))\ r_{\phi,l} \geq 0\right\},$$

$$H_\phi^{\tau} = \bigcap_{l \in L_B} \left\{x \in \mathbb{R}^\Sigma :\ \mathrm{sgn}(\phi(x))\ r_{\phi,l} \geq 0\right\}.$$

The sets H_ϕ^* are half-spaces or closed hyperplanes; H_ϕ^τ (or its boundary) goes through 0, and H_ϕ^{reg} is parallel.

Let $\sigma(v) = \sum_i v_i$, and $\Delta = \{v \geq 0 : \sigma(v) = 1\}$. The set $\tau(r)$ is described by $\tau(r) \cap \Delta$: for $x \in \Delta$ and $\alpha > 0$, we have $\alpha x \in \tau(r)$ iff $x \in \tau(r) \cap \Delta$. The set Δ is a bounded $A-1$-dimensional polytope (simplex), and $\tau(r) \cap \Delta$ is its intersection with a finite number of closed halfspaces and hyperplanes (H_ϕ^τ). Thus, $\tau(r) \cap \Delta$ is also a bounded polytope. Each vertex of this polytope is the point where $A-1$ hyperplanes of form $\phi(v) = 0$ (where $\phi \in \Phi$) cross Δ. The coefficients of ϕ are bounded polynomially by Y, so each vertex can be written as $t_i' = t_i/\sigma(t_i)$, where $\sigma(t_i)$ is bounded polynomially by some $C_{3.3}$ (by Lemma 2.3). Let $T_r = \{t_1, t_2, \ldots, t_N\}$. Note that the number of vertices, N, is bounded polynomially.

It is easy to check that the set $\mathrm{reg}(r)$ is *upward closed*, i.e., if $x \in \mathrm{reg}(r)$, $i \in [1..N]$, and $\alpha \geq 0$, then $x + \alpha t_i \in \mathrm{reg}(r)$.

Let W_r be the set of points in $\mathrm{reg}(r)$ such that for all $\alpha > 0$ and i, $W_r - \alpha t_i \notin \mathrm{reg}(r)$. We have that $\mathrm{reg}(r) = W_r + T_r^{\oplus\mathbb{P}}$. Indeed, $\supseteq$ follows from upward closedness, and for $\subseteq$, take $x_0 \in \mathrm{reg}(r)$. For i from 1 to N, let $x_i = x_{i-1} - \alpha_i t_i$, where $\alpha_i \in \mathbb{P}$ is the greatest number such that $x_{i-1} - \alpha_i t_i$ is still in $\mathrm{reg}(r)$ – such an α must exist, since $\mathrm{reg}(r)$ is closed, $\mathrm{reg}(r) \geq 0$, and $x_{i-1} - \alpha t_i$ has a negative coordinate for large enough values of α. We obtain a point x_N, which must be in W_r. Indeed, suppose that $x' = x_N - \alpha t_i \in \mathrm{reg}(r)$. In that case, $x_{i-1} - (\alpha_i + \alpha)t_i = x' + \sum_{j=i+1}^N \alpha_j t_j \in \mathrm{reg}(r)$ because $x' \in \mathrm{reg}(r)$ and $\mathrm{reg}(r)$ is upward closed. This contradicts the assumption that the value of α_i was maximum.

Take $x \in W_r$. Let $H_\phi^b = H_\phi^{\mathrm{reg}}$ in case if H_ϕ^{reg} is a hyperplane, or the boundary hyperplane of H_ϕ^{reg} if H_ϕ^{reg} is a half-space. Let $\Psi_x \subseteq \Phi_Y = \left\{\phi : x \in H_\phi^b\right\}$. The set $H_x := \bigcap_{\phi \in \Psi_x} H_\phi^b$ is an intersection of hyperplanes, and thus it is a subspace.

The set $H_x \cap W_r$ has to be bounded. Indeed, suppose that $H_x \cap W_r$ is not bounded. Take a sequence (v_i) of elements of $H_x \cap W_r$ such that $\lim_{i\to\infty} \sigma(v_i) = \infty$. Let $w_i = v_i/\sigma(v_i)$; we have $w_i \in \Delta$. Since Δ is compact, w_i has a cluster point, w. For $\phi \in \Phi_Y$ and $l \in L_B$, we have $\phi(w) = \phi(w_i - l)/\sigma(w_i) + \phi(l)/\sigma(w_i)$. The first component is zero or has sign $r(\phi, l)$ (since $v_i \in \mathrm{reg}(r)$) and the second component tends to 0, thus $\phi(w)$ is either 0, or it has sign $r(\phi, l)$. Thus, $w \in \tau(r)$, and also $w \in H_x - H_x$ (the subspace parallel to H_x going through 0). The set $H_x \cap W_r$ is a polytope which is unbounded in the direction of w, but this is impossible, since x and $x + \alpha w$ cannot be both in W_r (from the definition of W_r). Hence, a contradiction.

Moreover, the set $H_x \cap W_r$ has to be bounded polynomially. Indeed, the vertices of $H_x \cap W_r$ are points where some A hyperplanes of form H_ϕ^{reg} cross. These hyperplanes have coefficients bounded polynomially, and from Lemma 2.3 we get that the coordinates of vertices of $H_x \cap W_r$ are bounded polynomially.

Thus, $W_r = \bigcup_{x \in W_r} (H_x \cap W_r)$ is bounded polynomially. □

Proof of Lemma 3.4. If $\mathrm{Reg}(r)$ is bounded, then it is bounded polynomially in B and Y (Lemma 3.3), and therefore we are done. Thus, we can assume that $\mathrm{Reg}(r)$ is unbounded.

Suppose that there is a $t \in \tau(r)$ such that $t \notin \mathcal{Z}^{\oplus\mathbb{P}}$. We will show that in this case $\mathrm{Reg}(r)$ and $S = W + \mathcal{Z}^{\oplus\mathbb{P}}$ are disjoint, and therefore we are done.

Indeed, note that $\mathcal{Z}^{\oplus\mathbb{P}}$ is a d-dimensional cone (for some d), whose faces are hyperplanes going through 0 and parallel to sets of $(d-1)$ vectors in $[-Y, Y]^\Sigma$. The ray region $\tau(r)$ cannot be subdivided into two smaller regions by such a hyperplane, therefore there is one of these hyperplanes, say ϕ, such that $\phi(t) < 0$, $\phi(\tau(r)) \leq 0$, and $\phi(\mathcal{Z}) \geq 0$. Take $x \in \mathrm{Reg}(r)$ such that $x = w + \sum \alpha_i \mathcal{Z}_i$, where $w \in [-B..B]^\Sigma$. Since $\phi(t) < 0$, for each $l \in L_B$ we have $r_{\phi,l} < 0$, and thus, since $x \in \mathrm{Reg}(r)$, we have $\phi(x - l) \leq 0$. Since $L_B = [-2B, 2B]^\Sigma$, there is a $l_0 \in L_B$ such that $\phi(l_0) < \phi(w)$. Thus, $\phi(x) \leq \phi(l_0) < \phi(w) \leq \phi(w) + \sum \alpha_i \phi(\mathcal{Z}_i) = \phi(x)$, which is a contradiction.

Thus, we can assume that $\tau(r) \subseteq \mathcal{Z}^{\oplus\mathbb{P}}$. Each point in $\mathcal{Z}^{\oplus\mathbb{P}}$ can be written as $\sum \alpha_i \mathcal{Z}_i$, where α_i is positive only for a linearly independent set of vectors in $\mathcal{Z}$ (otherwise, if α_i are positive in a linearly dependent subset of $\mathcal{Z}$, we can modify the values so that one of them is 0 – just like in Lemma 2.6). Therefore, $\tau(r) \subseteq \bigcup_{F \in \mathcal{F}} F^{\oplus\mathbb{P}}$, where $\mathcal{F}$ is the family of all linearly independent subsets of $\mathcal{Z}$. Again, each $F^{\oplus\mathbb{P}}$ is a d-dimensional cone, whose faces are hyperplanes going through 0 and parallel to sets of $(d-1)$ vectors in $[-Y, Y]^A$. The ray region $\tau(r)$ cannot be subdivided into two smaller regions by such a hyperplane, thus there is a $F \in \mathcal{F}$ such that $\tau(r) \subseteq F^{\oplus\mathbb{P}}$. Let $\mathcal{Z}_0$ be this F, and let D be determinant of the matrix whose columns are $\mathcal{Z}_0$ (we add unit vectors if there are less than A vectors in $\mathcal{Z}_0$). From Lemma 2.1 we know that $D \leq H_A(Y)$, which is single exponential.

Now, let $W_1 = S \cap [-C_{3.4}..C_{3.4}]^\Sigma$; the sufficient value of $C_{3.4}$ will be apparent from the sequel of the proof. By induction over $|v|$, we will prove that each $v \in S$ can be written as $v = w_1 + \mathcal{Z}_0^{\oplus\mathbb{N}}$, where $w_1 \in W_1$.

For $v \in S$ such that $|v| \leq C_{3.4}$, the hypothesis obviously holds.

For $v \in S$ such that $|v| \geq C_{3.4}$ we apply the following Lemma:

Lemma 3.6. Let Σ be a fixed alphabet, and $B, Y, D \in \mathbb{N}$. Then there exists a constant $B_{3.6}$ polynomial in B, Y and D, such that for each $r \in \mathcal{R}_{B,Y}$ and each $v \in \mathrm{reg}(r)$, if $||v|| \geq B_{3.6}$, then $v = v_0 + Dt$, where $v_0 \in \mathrm{reg}(r)$ and $t \in \tau(r) \cap [-C_{3.3}..C_{3.3}]^\Sigma$ (where $C_{3.3}$ is from Lemma 3.3). Moreover, $v_0 \in \mathrm{Reg}(r)$ iff $v \in \mathrm{Reg}(r)$.

Take $v \in S$. By applying Lemma 3.6 iteratively, we know that $v = v_0 + D(t_1 + \ldots + t_K)$, where $t_i \in \tau(r) \cap [-C_{3.3}..C_{3.3}]^\Sigma$, and $||v_0|| \leq B_{3.6}$.

On the other hand, we know that $v = w_0 + w_{12}$, where $w_0 \in W$ and $w_{12} = \sum_{v\in\mathcal{Z}} \beta_v v$, where $\beta_v \in \mathbb{N}$. From Lemma 2.6 we can assume that β_v exceed $H = H_A(Y)$ only in a linearly independent $P \subseteq \mathcal{Z}$. Take a large enough $J_0 \geq H$, polynomial in B and Y (the required value will be apparent below). We can write $v = w_0 + w_1 + w_2$, where $w_1 \in \mathcal{Z}^{\oplus[0..J_0]}$ and $w_2 = \sum_{v\in P} \beta_v v$, where each $\beta_v \geq J_0$, and P is linearly independent.

Suppose that, for some i, $t_i \in P^{\oplus\mathbb{P}}$. From Lemma 2.4 we obtain that $Dt_i = \sum_{v\in P} \gamma_v v$, where $\gamma_v \in \mathbb{Z}$. Moreover, from Lemma 2.3, $\gamma_v \leq J_0$, if J_0 is large enough. Therefore, $w_2 = w_2' + Dt_i$, where $w_2' = \sum_{v\in P}(\beta_v - \gamma_v v)$. Take $v' = v - Dt_i$; we have $v' = w_0 + w_1 + w_2' \in W + \mathcal{Z}^{\oplus\mathbb{N}} = S$. From the induction hypothesis, $v' \in W_1 + \mathcal{Z}_0^{\oplus\mathbb{N}}$, and $Dt_i \in \mathcal{Z}_0^{\oplus\mathbb{N}}$ from Lemma 2.4. Since $v = v' + Dt_i$, we obtain that $v \in W_1 + \mathcal{Z}_0^{\oplus\mathbb{N}}$.

Otherwise, $\{t_i : i = 1, \ldots, K\}$ is disjoint from $P^{\oplus\mathbb{P}}$. Therefore, $\{t_i\}^{\oplus\mathbb{P}}$ is disjoint from $P^{\oplus\mathbb{P}}$ (except 0). By the same arguments as in the beginning of this proof, there is a hyperplane $\phi \in \Phi_Y$ such that $\phi(t_i) < 0$ for each i, and $\phi(P) \geq 0$.

What is the value of $\phi(v)$? On one hand, we have $\phi(v) = \phi(v_0) + D\sum_i \phi(t_i) < AC_\phi B_{3.6} - DK$. On the other hand, we have $\phi(v) = \phi(w_0+w_1+w_2) = \phi(w_0)+\phi(w_1)+\phi(w_2) = \phi(w_0) + \sum_{v\in\mathcal{Z}} \alpha_v \phi(v) + \sum_{v\in P} \beta_v \phi(v) \geq -BAC_\phi - J_0|\mathcal{Z}|YAC_\phi$. From applying the triangle inequality to $v = v_0 + D\sum_i t_i$, we get that $|v| \leq |v_0| + DKAC_{3.3}$, and hence, $K \geq (C_{3.4} - B_{3.6})/DAC_{3.3}$. Thus, $AC_\phi HB_{3.6} - D(C_{3.4} - B_{3.6})/DAC_{3.3} \geq -BAC_\phi - J_0|\mathcal{Z}|YAC_\phi$. This is a contradiction for a large enough (but still polynomial) $C_{3.4}$. □

Proof of Lemma 3.5. Let D_i be the determinant of $\mathcal{Z}_i$ (if $\mathcal{Z}_i$ includes less than A vectors, add unit vectors as usual). Let D^* be the least common multiple of all D_i. We know that D_i is bounded polynomially in Y, so D^* is bounded polynomially by $Y^{|I|}$.

By applying Lemma 3.6 (with $D = D^*$) we get constants $C_{3.3}$ and $B_{3.6}$. We know that each $v \in \mathrm{Reg}(r)$ such that $|v| \geq B_{3.6}$ can be written as $v_0 + D^* t$, where $t \in \tau(r)$. We repeat this construction (adding all the t's together), until we get $|v_0| < B_{3.6}$.

Now, we have to show that $v \in V_i$ iff $v_0 \in V_i$.

Suppose that $v \in V_i$. Thus, we have $v = w_0 + \sum_k \alpha_k P_k$, where $\alpha_k \geq 0$, and (P_k) are the members of $\mathcal{Z}_k$. We can write similarly $v_0 = w_0 + \sum_k \beta_k P_k$. Note that v and v_0 are in the same $(C_{3.4}, Y)$-region r. In particular, $v - w_0$ and $v_0 - w_0$ are on the same side of each hyperplane going through a member of $L_{C_{3.4}}$ and parallel to $A-1$ members of $[-Y..Y]^\Sigma$; therefore, since $\alpha_k \geq 0$, we also have $\beta_k \geq 0$. Moreover, since $v - v_0 = D^* t$, by Lemma 2.4, $v - v_0 \in \mathcal{Z}_i^{\oplus\mathbb{Z}}$. Therefore, $\alpha_k - \beta_k$ has to be an integer, and therefore $\beta_k \in \mathbb{N}$. Thus, $v_0 \in V_i$.

The proof in the other direction goes in the same way. □

Proof of Lemma 3.6. Take $v \in \mathrm{reg}(r)$. From Lemma 3.3 we know that $v = w + \sum \alpha_i t_i$, where $\{t_i : i \in \{1, \ldots, N\}\} = T_i \subseteq [-C_{3.3}..C_{3.3}]^\Sigma$, and $w \in [-B_{3.3}..B_{3.3}]^\Sigma$. By taking a large enough $B_{3.6}$, we can ensure that if $|x| \geq B_{3.6}$, then some α_i is greater than D. We have that $v_0 = v - Dt_i$ is a proper convex combination of v and $v_1 = v - \alpha_i t_i$, which are both in $\mathrm{reg}(r)$, thus v_0 is also in $\mathrm{reg}(r)$. And if $v \in \mathrm{Reg}(r)$, then so is v_0 ($\phi(v_0 - l)$ from the definition of $\mathrm{reg}(r)$ is a proper convex combination of $\phi(v - l)$ and $\phi(v_1 - l)$, $\phi(v - l)$ has the correct sign, and $\phi(v_1 - l)$ either has the correct sign or is 0). □

4. A hard grammar

As mentioned in the introduction to Section 3, for $A > 2$ $\Psi(G)$ is not a union of polynomial number of simple bundles, which makes it impossible to use the simple reasoning mentioned.

In this section, we will show an example of a grammar over an alphabet with three symbols where this is the case. We believe this example is interesting in its own right.

The proof has two parts. In the first part, we create a sequence of grammars (G_n) over four terminal symbols (x, y, and two temporary ones), such that, if we ignore the two temporary terminal symbols, the convex hull of $\Psi(G_n)$ is a bounded polygon with 2^n vertices. It is then straightforward to create a grammar over three terminal symbols (x, y, z) which has the required property.

Theorem 4.1. There is a positive grammar G_n of linear size with terminal symbols x, y, S_{n+1} and A_n, and non-terminal symbols S_0 and X_n (where S_0 is initial), such that the vertices of the convex hull of $\Psi(G_n)$ are exactly the points $y^i x^{i(i+1)/2} S_{n+1}^{2i+1} A_n^{2N-2i-1}$ for $i = 0, \ldots, N-1$, and X_n generates only x^N, where $N = 2^n$.

Proof. The proof is by induction. For $n = 0$ we simply put $S_0 \to S_1 A_0$, $X_0 \to x$.

Take the grammar G_n; we will construct the grammar G_{n+1}. The symbol X_{n+1} is obtained by the following rule: $X_{n+1} \to X_n^2$.

We replace the terminal A_n with a non-terminal with two transition rules:

- (1) $A_n \to A_{n+1}^2$
- (2) $A_n \to S_{n+2}^2 X_n y$

The new vertices of the convex hull will be obtained by choosing one of the two rules for A_n, and applying it consistently. Thus, from each vertex $y^i x^{i(i+1)/2} S_{n+1}^{2i+1} A_n^{2N-2i-1}$ of the convex hull of $\Psi(G_n)$, we get the following vertices:

- (1) $y^i x^{i(i+1)/2} S_{n+1}^{2i+1} A_{n+1}^{4N-4i-2}$
- (2) $y^{2N-1-i} x^{i(i+1)/2+N(2N-2i-1)} S_{n+1}^{2i+1} S_{n+2}^{4N-4i-2}$

Now, we replace the terminal S_{n+1} with a non-terminal with a single transition rule $S_{n+1} \to A_{n+1} S_{n+2}$. Hence, we get the following vertices:

- (1) $y^i x^{i(i+1)/2} S_{n+2}^{2i+1} A_{n+1}^{4N-2i-1}$
- (2) $y^{2N-1-i} x^{i(i+1)/2+N(2N-2i-1)} A_{n+1}^{2i+1} S_{n+2}^{4N-2i-1}$

By taking $j = 2N - 1 - i$, we can rewrite the second row of vertices as follows:

- (1) $y^i x^{i(i+1)/2} S_{n+2}^{2i+1} A_{n+1}^{4N-2i-1}$
- (2) $y^j x^{j(j+1)/2} S_{n+2}^{2j+1} A_{n+1}^{4N-2j-1}$

Note that i runs from 0 to $N-1$, and j runs from N to $2N-1$. We can thus combine our two rows of vertices into one, indexed by i running from 0 to $2N-1$, and thus obtain the induction thesis. □

Now, let G'_n be obtained from G_n by removing all the terminals A_{n+1} and S_{n+2}. The convex hull of $\Psi(G'_n)$ is the polygon P with vertices $x^i y^{i(i+1)/2}$, for $i \in [0..N-1]$.

Now, we create a new grammar G_n° over x, y, z whose Parikh image will be geometrically the infinite cone with base P. This is done simply by adding a new initial symbol S° to G'_n, with two transitions $S^\circ \xrightarrow{z} S_0 S^\circ$ and $S^\circ \xrightarrow{0} 0$. It can be easily seen that periods are single exponential, and since $\Psi(G_n^\circ)$ is an infinite cone with 2^n edges, we cannot write $\Psi(G_n^\circ)$ as a union of a polynomial number of simple bundles – indeed, each simple bundle can cover only at most 3 edges.

5. Complexity results

In this section, we provide the tight complexity bounds for the pproblems we are considering. Some of the results have been previously known (e.g., [Huy84, Esp97]), we include all the proofs for completeness.

Theorem 5.1. The following membership problem in commutative regular grammars is in P for a fixed Σ:

Given: a regular grammar G, $v \in \mathbb{Z}^\Sigma$

Decide whether $v \in \Psi(G)$

To prove this Theorem, we will need the following two lemmas:

Lemma 5.2. For $P \subseteq Q$, let

$$\mathcal{R}_n(P,q) = \{\Psi(R) : R \text{ is a run from } q,\ |R| \leq n, \text{ and } P \subseteq \mathrm{supp}(R)\}.$$

For a regular grammar G over Σ of size A with N nonterminals, the sets $\mathcal{R}_n(P,q)$ for all P of cardinality at most k and all $n \leq B$ can be calculated in time $O((2B)^A|G|N^{k+1})$ and space $O((2B)^A N^{k+1})$.

Proof. Note that a run of length n from q consists of a transition from q to $[q']$ and a run from q' of length $n-1$. Thus, we get the following recursive formula:

$$\begin{aligned}
\mathcal{R}_1(P,q) &= \{\Psi(d) : d \in \delta, \mathrm{source}(d) = q, \mathrm{target}(d) = 0, \{q\} \subseteq P\} \\
\mathcal{R}_n(P,q) &= \mathcal{R}_{n-1}(P,q) \cup \mathcal{R}_n(P - \{q\}, q) \cup \\
&\quad \{\Psi(d) + \mathcal{R}_{n-1}(P,r) : d \in \delta, \mathrm{target}(d) = [r], \mathrm{source}(d) = q\}
\end{aligned}$$

We know that $\mathcal{R}_n(P,q) \subseteq [-n..n]^\Sigma$. This allows us to calculate all the sets $\mathcal{R}_n(p,q)$ using dynamic programming. □

Lemma 5.3. Let

$$\mathcal{P}_n(q_1,q_2) = \{\Psi(R) : R \text{ is a path from } q_1 \text{ to } q_2 \text{ and } |R| \leq n\}.$$

For a regular grammar G over Σ of size A with N nonterminals, the sets $\mathcal{P}_n(q_1,q_2)$ for all $n \leq B$ can be calculated in time $O((2B)^A|G|N^2)$ and space $O((2B)^A N^{k+2})$.

Proof. The idea is essentially the same as in Lemma 5.2. We use the following recursive formula:

$$\begin{aligned}
\mathcal{P}_0(q_1,q_2) &= \emptyset \text{ if } q_1 \neq q_2 \\
\mathcal{P}_0(q,q) &= \{0\} \\
\mathcal{P}_n(q_1,q_2) &= \mathcal{P}_{n-1}(P,q) \cup \\
&\quad \{\Psi(d) + \mathcal{P}_{n-1}(r,q_2) : d \in \delta, \mathrm{target}(d) = [r], \mathrm{source}(d) = q_1\}
\end{aligned}$$

□

Proof of Theorem 5.1. The algorithm is as follows.

(1) Let B_G be the bound on R_1 from Theorem 2.13.
(2) Using Lemma 5.2 we find the sets $\mathcal{R}_{B_G}(P, q_I)$ for all P such that $|P| \leq A$.
(3) Using Lemma 5.3 we find the sets $\mathcal{P}_N(q, q)$ for all P such that $|P| \leq A$.
(4) For each P such that $|P| \leq A$:
    For each linearly independent subset Z of $\cup_{q \in P} \mathcal{P}_N(q, q)$:
        For each element w of $\mathcal{R}_{B_G}(P, q_I)$:
            if $v - w \in Z^{\oplus \mathbb{N}}$:
                return YES.
(5) Otherwise return NO.

Since Z is linearly independent, we can check whether $v - w \in Z^{\oplus \mathbb{N}}$ using Gaussian elimination.

It is straightforward from Theorem 2.13 that this algorithm will return YES if $v \in \Psi(G)$. It is also straightforward that $v \in \Psi(G)$ if the algorithm answers YES.

This algorithm runs in time $O\left((2B_G)^A |G| N^{A+1} + N^{A+1}\right)$. It uses space $O((2B)^A N^{k+2})$. □

Theorem 5.4. The membership problem in commutative grammars is in NP.

This result was previously known [Esp97].

Proof. From Theorem 2.13 we know that if $v \in \Psi(G)$, then $v = \Psi(R_1) + \sum \Psi(C_k) n_k$, where R_1 and C_k are bounded exponentially, and C_k are linearly independent. We can simply guess R_1 and C_k. □

Theorem 5.5. The inclusion, universality, and disjointness problems in commutative grammars over an alphabet of fixed size are in Π_2^{P}.

Proof. We will consider inclusion; the other problems can be solved in the same way.

Let G_1 and G_2 be two commutative grammars in normal form with at most N non-terminals each, over the same fixed alphabet Σ.

By Theorem 3.2, there exists a number $B_{3.2}$ which is single exponential in N, such that $\Psi(G_1) \subseteq \Psi(G_2)$ iff $\Psi(G_1) \cap [-B_{3.2}..B_{3.2}]^\Sigma \subseteq \Psi(G_2) \cap [-B_{3.2}..B_{3.2}]^\Sigma$. Thus, we need to check the membership in $\Psi(G_2)$ for all the elements of the set $[-B_{3.2}..B_{3.2}]^\Sigma \cap \Psi(G_1)$. This can be done in Π_2^{P}. □

Theorem 5.6. The inclusion and universality problems in regular grammars over an alphabet of fixed size are in coNP.

Proof. The proof is just like for Theorem 5.5. Again, we need to check the membership in $\Psi(G_2)$ for all the elements v of $[-B_{3.2}..B_{3.2}]^\Sigma \cap \Psi(G_1)$. However, this time checking membership of v can be done in P, so the whole algorithm works in coNP. □

Theorem 5.7. The problem of inclusion of commutative grammars over $\Sigma = \{a\}$ is Π_2^{P}hard.

This result was previously known [Huy84].

Proof. We will reduce the following problem (**3-CNF-QSAT**$_2$). Let the clauses C_j $(0 \leq j < m)$ be disjunctions of at most three literals of form x_i, $\neg x_i$, $\neg y_i =$ or $\neg y_i$. Does the formula

$$\forall x_0 \forall x_1 \ldots \forall x_{k-1} \exists y_0 \exists y_1 \ldots \exists y_{l-1} \bigwedge_{j<m} C_j$$

hold? Quantifiers run over two possible values of each variable (either x_j or $\neg x_j$ is true).

The symbol A_i ($i < k$) generates a^{2^i}, for $i < k$. This can be realized with the following rules: $A_0 \to a$, $A_i \to A_{i-1}A_{i-1}$ for $i > 0$.

The symbol $A_i^?$ ($i < k$) generates a^{2^i} or nothing. This can be realized with the following rules: $A_i^? \to 0$, $A_i^? \to A_i$.

The symbol C_j ($j < m$) generates $(a^{2^k})^{4^j}$. This can be realized with the following rules: $C_j \to A_{k-1}A_{k-1}$, $C_j \to C_{j-1}C_{j-1}C_{j-1}C_{j-1}$ for $j > 0$.

The symbol $C_j^?$ ($j < m$) generates $(a^{2^k})^{4^j}$ or nothing. This can be realized with the following rules: $C_j^? \to 0$, $C_j^? \to C_j$.

The symbol X_i ($i < k$) has the following two rules: $X_i \to A_i \prod_{j<m}(C_j^? : x_i \in C_j) | \prod_{j<m}(C_j^? : \neg x_i \in C_j)$.

The symbol Y_i ($i < l$) has the following rules: $Y_i \to \prod_{j<m}(C_j^? : y_i \in C_j) | \prod_{j<m}(C_j^? : \neg y_i \in C_j)$.

The symbol S_1 has the following rules: $S_1 \to \prod_{i<k} A_i^? \prod_{j<m} C_j$.

The symbol S_2 has the following rules: $S_2 \to \prod_{i<k} X_i \prod_{i<l} Y_i$.

We ask whether the language generated by S_1 is a subset of the language generated by S_2.

Note that each A_i is generated at most once, and each C_j is generated at most three times. The definitions of these symbols ensure that we can treat them as independent: for each combinations of A_i and C_j in S_1, we need to find the same combination in S_2.

The language generated by S_1 has 2^k elements (for each A_i, we can either add it or not). These correspond to possible valuations of variables of x_i. To match the given element of S_1 in S_2, we need to make the same choices in X_i. We also need to generate $C_0C_1C_2\ldots C_{j-1}$. This means that we have to make choices in Y_i which generate all the clauses which were not covered by our choices in X_i. Therefore, $S_1 \subseteq S_2$ iff the formula is true. □

Theorem 5.8. The problem of universality (is $\Psi(G) = \mathbb{Z}^\Sigma$?) of commutative grammars over $\Sigma = \{a\}$ is Π_2^{P}hard.

Proof. We use the same reduction as in Theorem 5.7. It is enough to have a symbol S_3 which generates the complement of S_1. Indeed, add a new symbol S_4, with rules $S_4 \to S_2|S_3$. The universality of S_4 is equivalent to S_1 (the complement of S_3) being included in S_2, which is equivalent to our **3-CNF-QSAT**$_2$ formula being true.

This can be done as follows:

The symbol C_j^H generates 0, 2, or 3 copies of C_j. This can be done with the following rules: $C_j^H \to 0|C_jC_j|C_jC_jC_j$.

The symbol Z^+ generates any number of a's which is at least 2^k4^m. This can be realized as follows: $Z^+ \to C_{m-1}C_{m-1}C_{m-1}C_{m-1}|aZ^+$.

The symbol Z^- generates any negative number of a's. This can be realized as follows: $Z^- \to a^{-1}Z^-|a^{-1}$.

Now, we can define S_3 as $S_3 \to Z^+|Z^-|\prod_{i<k} A_i^? \prod_{j<m} C_j^H$. □

Proposition 5.9. The problem of membership in commutative grammars over $\Sigma = \{a\}$ is NP hard.

Proof. We reduce the **3-CNF-SAT** problem. The proof is the same as in Theorem 5.7, except that we take $k = 0$. □

Proposition 5.10. Let G be a commutative regular grammar over $\Sigma = \{a\}$. Then the problem of deciding universality ($\prod(G) = \mathbb{N}^\Sigma$) is coNP-hard.

Proof. We reduce the 3CNF-SAT problem. Let $\phi = \bigwedge_{1 \leq i \leq k} C_i$ be a 3CNF-formula with n variables $x_1 \dots x_n$ (which can be 0 or 1) and k clauses. Let $p_1, p_2, \dots, p_n$ be n distinct prime numbers. Let $i \in [1..k]$. Suppose that clause C_i is of form $\bigvee_{k \in [1..3]} x_{a_k} = v_{a_k}$. Our grammar will have states S^i_j, where $0 \leq j < M_i = p_{a_1} p_{a_2} p_{a_3}$; we have cyclic transitions $S^i_j \xrightarrow{a} S^i_{(j+1) mod M_i}$, and $S^i_j \xrightarrow{0} 0$ for each j not satisfying $\bigvee_{k \in [1..3]} j \bmod p_{a_k} = v_{a_k}$. We also have transitions $s_0 \xrightarrow{0} S^i_0$ for each i.

From simple number theoretic arguments we get that $x \notin \prod(G)$ iff the formula ϕ is satisfied for $x_i = x \bmod p_i$. □

Proposition 5.11. For a commutative regular grammar G over Σ (whose size is not fixed), and $K \in \mathbb{N}^\Sigma$, the problem of deciding whether $K \in \prod(G)$ is NP-hard.

Proof. We show a reduction from the Hamiltonian circuit problem. Let (V, E) be a graph. We take $\Sigma = Q = V$, and for each edge (v_1, v_2) we add a transition $v_1 \xrightarrow{v_2} v_2$. We pick an initial state s_0 and add a final transition $s_0 \xrightarrow{0} 0$. The graph (V, E) has a Hamiltonian circuit iff $K = (1, 1, \dots) \in \Psi(G)$. □

Proposition 5.12. The disjointness problem in commutative regular grammars (over an alphabet of unfixed size) is coNP-complete.

Proof. From Proposition 5.11 we know that membership is NP-hard. We can easily construct a grammar G_2 such that $\Psi(G_2) = \{K\}$ (where K is from the proof of Proposition 5.11), and ask for non-disjointness of G and G_2.

From Theorem 5.4 we also know that membership is in NP (even for non-regular grammars). We reduce non-disjointness of G_1 and G_2 to membership in the following way. It is sufficient to check whether $0 \in \Psi(G_1) - \Psi(G_2)$, where $\Psi(G_1) - \Psi(G_2) = \{v_1 - v_2 : v_1 \in \Psi(G_1), v_2 \in \Psi(G_2)\}$. We create a grammar $G_1 - G_2$ such that $\Psi(G_1 - G_2) = \Psi(G_1) - \Psi(G_2)$: first, create the grammar $-G_2$ such that $\Psi(-G_2) = \{-v : v \in \Psi(G_2)$ by replacing each production by its negative, and then create the grammar $G_1 - G_2$ by replacing each final transition in G_1 (i.e., δ such that $\mathrm{target}(\delta) = 0$) by transitions going to each initial state of G_2. □

The following table summarizes the complexities of various problems regarding commutative grammars. Alphabet size F means fixed, and U means unfixed. We include the very recent result regarding unfixed alphabets [HH14] that inclusion for regular grammars over an unfixed alphabet is coNEXP hard; it is likely that the proof can be also adapted for universality. On the other hand, it is known from [Huy85] that inclusion and universality for context-free grammars over an unfixed alphabet is in coNEXP. For completeness, we also include N (the non-commutative case – note that the membership problem is actually a different problem in the non-commutative case, since we cannot encode long words succinctly with the length of the word in binary); these results are known from other sources ([MS72], also see [HU79] for a reference). The letter c means complete. Note that inclusion and equivalence problems easily reduce to each other.

regular languages					
alphabet size	1	2	F	U	N
membership	P	P	P	NPc	P
universality	coNPc	coNPc	coNPc	?	PSPACEc
inclusion	coNPc	coNPc	coNPc	coNEXPc	PSPACEc
disjointness	P	P	P	coNPc	P
context-free languages					
alphabet size	1	2	F	U	N
membership	NPc	NPc	NPc	NPc	P
universality	Π_2^{P}c	Π_2^{P}c	Π_2^{P}c	?	undecidable
inclusion	Π_2^{P}c	Π_2^{P}c	Π_2^{P}c	coNEXPc	undecidable
disjointness	coNPc	coNPc	coNPc	?	undecidable

6. Conclusion

The table above contains question marks for languages of unfixed size. It is known that these problems are Π_2^{P}-hard for context-free grammars, and they are in NEXP. Our method heavily uses the fact that the size of the alphabet is fixed, so we probably cannot easily generalize it. As mentioned above, coNEXP hardness of inclusion for regular grammars has been shown recently [HH14].

A natural question extending this research is counting. We can answer the question whether $v \in \Psi(G)$, but what about the number of paths (or words) which lead to the given v? This number is exponential in $|v|$, and thus it could be very large, so we cannot always hope for the exact answer—but we could count up to some threshold T, modulo M, or count approximately. We can answer the question whether $\Psi(G_1) \subseteq \Psi(G_2)$, but is the number of paths smaller in the first case than in the second case?

Thanks to everyone on AUTOBÓZ 2009 for the great atmosphere of research, especially to Sławek Lasota for introducing me to these problems. Also I would like to thank Anthony Widjaja Lin for our collaboration on the merged paper [KT10], and the anonymous referees for their insightful comments. This work is supported by the Polish National Science Centre Grant DEC - 2012/07/D/ST6/02435.

References

[BM99] Catriel Beeri and Tova Milo. Schemas for integration and translation of structured and semi-structured data. In *In Proceedings of the International Conference on Database Theory*, pages 296–313, 1999.

[Cra50] Gabriel Cramer. Introduction à l'analyse des lignes courbes algébriques. *Geneva: Europeana*, pages 656–659, 1750.

[EN94] Javier Esparza and Mogens Nielsen. Decidability issues for petri nets - a survey. *Bulletin of the EATCS*, 52:244–262, 1994.

[Esp97] J. Esparza. Petri nets, commutative context-free grammars, and basic parallel processes. *Fundamenta Informaticae*, 31:13–26, 1997.

[Hac75] Michel Hack. *Decidability Questions for Petri Nets*. Outstanding Dissertations in the Computer Sciences. Garland Publishing, New York, 1975.

[Had93] Jacques Hadamard. Résolution d'une question relative aux déterminants. *Bulletin des Sciences Mathématiques*, 17:240–246, 1893.

[Hat02] Allen Hatcher. *Algebraic topology*. Cambridge University Press, Cambridge, New York, 2002. Autre(s) tirage(s) : 2003,2004,2005,2006.

[HH14] Christoph Haase and Simon Halfon. Integer vector addition systems with states. In Joël Ouaknine, Igor Potapov, and James Worrell, editors, *Reachability Problems - 8th International Workshop, RP 2014, Oxford, UK, September 22-24, 2014. Proceedings*, volume 8762 of *Lecture Notes in Computer Science*, pages 112–124. Springer, 2014.

[HU79] John E. Hopcroft and Jefferey D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Adison-Wesley Publishing Company, Reading, Massachusets, USA, 1979.

[Hüt94] Hans Hüttel. Undecidable equivalences for basic parallel processes. In Masami Hagiya and John C. Mitchell, editors, *TACS*, volume 789 of *Lecture Notes in Computer Science*, pages 454–464. Springer, 1994.

[Huy80] Thiet-Dung Huynh. The complexity of semilinear sets. In J. W. de Bakker and Jan van Leeuwen, editors, *ICALP*, volume 85 of *Lecture Notes in Computer Science*, pages 324–337. Springer, 1980.

[Huy84] Dung T. Huynh. Deciding the inequivalence of context-free grammars with 1-letter terminal alphabet is sigma-p-2-complete. *Theor. Comput. Sci.*, 33:305–326, 1984.

[Huy85] Dung T. Huynh. The complexity of equivalence problems for commutative grammars. *Information and Control*, 66(1/2):103–121, 1985.

[Kop11] Eryk Kopczynski. Trees in trees: Is the incomplete information about a tree consistent? In Marc Bezem, editor, *CSL*, volume 12 of *LIPIcs*, pages 367–380. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2011.

[Kos82] S. Rao Kosaraju. Decidability of reachability in vector addition systems (preliminary version). In Harry R. Lewis, Barbara B. Simons, Walter A. Burkhard, and Lawrence H. Landweber, editors, *STOC*, pages 267–281. ACM, 1982.

[KT10] Eryk Kopczynski and Anthony Widjaja To. Parikh images of grammars: Complexity and applications. In *LICS*, pages 80–89. IEEE Computer Society, 2010.

[Ler10] Jérôme Leroux. The general vector addition system reachability problem by presburger inductive invariants. *Logical Methods in Computer Science*, 6(3), 2010.

[MS72] Albert R. Meyer and Larry J. Stockmeyer. The equivalence problem for regular expressions with squaring requires exponential space. In *SWAT (FOCS)*, pages 125–129. IEEE Computer Society, 1972.

[NS99] Frank Neven and Thomas Schwentick. Xml schemas without order, 1999.

[Par66] Rohit Parikh. On context-free languages. *J. ACM*, 13(4):570–581, 1966.

[Pet81] James Lyle Peterson. *Petri Net Theory and the Modeling of Systems*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1981.

[Rei85] Wolfgang Reisig. *Petri Nets: An Introduction*, volume 4 of *Monographs in Theoretical Computer Science. An EATCS Series*. Springer, 1985.

[Sto76] Larry J. Stockmeyer. The polynomial-time hierarchy. *Theor. Comput. Sci.*, 3(1):1–22, 1976.

[Yen97] Hsu-Chun Yen. On reachability equivalence for bpp-nets. *Theor. Comput. Sci.*, 179(1-2):301–317, 1997.