Regular Grammars as Effective Representations of Recognizable Sets of Series-Parallel Graphs

Marius Bozga
VERIMAG, CNRS, Université Grenoble Alpes, France

Radu Iosif
VERIMAG, CNRS, Université Grenoble Alpes, France

Florian Zuleger
Technische Universtität Wien, Austria
Technische Universtität München, Germany

Abstract

Series-parallel (SP) graphs are binary edge-labeled graphs with a designated source and target vertex, built using serial and parallel composition. A set of graphs is recognizable if membership depends only on its image under a homomorphism into a finite algebra. For SP-graphs, and more generally, for graphs of bounded tree-width, recognizability coincides with definability in Counting Monadic Second-Order (CMSO) logic. Despite this strong logical characterization, the conciseness and algorithmic effectiveness of syntactic representations of recognizable sets of SP (and bounded-tree-width) graphs remain poorly understood.

Building on previously introduced regular grammars for SP-graphs, we show that recognizable sets admit concise and effective syntactic representations. The main contribution is an improved construction of finite recognizer algebras whose size is singly-exponential in the size of a regular grammar, improving upon the previously known double-exponential bound. As a consequence, the problems of intersection and language inclusion for sets represented by regular grammars are shown to be EXPTIME-complete, thus improving on a previously known 2EXPTIME upper bound.

2012 ACM Subject Classification Theory of computation → Grammars and context-free languages

Keywords and phrases Series-parallel graphs, Regular grammars, Recognizability, Decision problems

Digital Object Identifier 10.4230/LIPIcs.MFCS.2026.79

Related Version *Full Version*: `https://arxiv.org/abs/2604.24151`

Funding Marius Bozga and Radu Iosif wish to acknowledge the support of the French National Research Agency project Non-Aggregative Resource COmpositions (NARCO) under grant number ANR-21-CE48-0011. Florian Zuleger wishes to acknowledge the support of the FWF project AUTOSARD: "Automated Sublinear Amortised Resource Analysis of Data Structures" No. P36623 and the project VASSAL: "Verification and Analysis for Safety and Security of Applications in Life" funded by the European Union under Horizon Europe WIDERA Coordination and Support Action/Grant Agreement No. 101160022.

Acknowledgements All authors thank the anonymous reviewers for their insightful comments.

1 Introduction

A subset of an algebra is recognizable if it is the inverse image of a homomorphism into a finite algebra over the same signature of function symbols. For words and, more generally, for ground terms (i.e., node-labeled ranked trees), this definition of recognizability coincides with the classical one, i.e., existence of finite-state word and tree automata, respectively. A central aspect of automata-based recognizability is the predefined traversal order of the input: words are read either left-to-right or right-to-left and trees are traversed bottom-up or top-down. Graphs, however, are strictly more complex objects that, in general, do not come with a natural traversal order. As a consequence, attempts to generalize automata-based recognizability to graphs resulted in a variety of non-equivalent and largely incomparable formalisms, without a broadly accepted consensus [20, 18]. This motivates the use of the algebraic definition of recognizability instead of the automata-based one [9, 14] and the quest for succint syntactic representations of recognizable sets, such as regular grammars [6] or regular expressions [12].

The series-parallel (SP) class consists of graphs having a designated source and target vertex, that can be composed either in series (i.e., the target of the first is joined with source of the second) or in parallel (i.e., both sources and targets are joined, respectively). SP-graphs arise naturally in circuit design and distributed data stream processing (see, e.g. [2] for a survey) and play a central role in graph theory: a classical result states that a graph has tree-width 2 at most if and only if each of its 2-connected[1] components (blocks) is a disoriented SP-graph (i.e., a graph obtained from an SP-graph by reversing some edges). Hence, each graph of tree-width at most 2 can be represented as a tree of such blocks [21]. This structural characterization has been used in [6] to develop regular grammars for tree-width 2 graphs, by combining regular grammars for trees and SP-graphs. While the present paper focuses, for simplicity, on SP-graphs, our results extend naturally to the more general class of graphs of tree-width at most 2. We consider this generalization for a future extended version.

The recent development of regular grammars [6] and regular expressions [12] that describe the recognizable sets of the tree-width ≤ 2 class of graphs has left the following question unanswered: "*what is the complexity of the translation between a syntactic representation (regular grammar/expression) and an algebraic representation of a recognizable set?*". For words, it is well-known that the minimal algebra that recognizes the language of a nondeterministic automaton with n states, that can be obtained from a regular grammar/expression in polynomial time, has cardinality $2^{\mathcal{O}(n^2)}$, i.e., each element of the algebra is a set of pairs of states. Similarly, for ground terms (ranked trees), the translation of regular grammars/expressions into deterministic tree automata incurs an unavoidable singly-exponential blowup [8].

Contributions. First, we answer the above question for regular grammars [6] that represent recognizable sets of SP-graphs by showing that, for a regular grammar of size n (comprising the number of nonterminals, rules and the size of each rule), there exists an algebra of cardinality $2^{\mathcal{O}(n^9)}$ that recognizes its language. This result is an improvement of the double exponential upper bound found in [6]. This upper bound is matched by an exponential lower bound: there exist a regular grammar having $\mathcal{O}(n)$ nonterminals and $\mathcal{O}(n \log_2 n)$ rules, whose recognizer algebra has no less than 2^{n^2} elements.

Second, we apply these complexity results to find the complexity of several natural decision problems concerning sets of SP-graphs represented as regular grammars. In particular, the problems of joint intersection of regular grammars and inclusion of an arbitrary grammar into a regular one are shown to be EXPTIME-complete. These results show that regular SP-grammars can be effectively manipulated within the same complexity class as standard tree automata [8], which justifies future efforts for implementing our techniques within one of the existing automata libraries [13, 16].

Related work. Automata recognizing a subclass of SP-graphs, called *synchronized SP-graphs* (SSPG), have been introduced in [2]. SSPGs have matching split and join vertex labels, that give automata hints for parsing the input graph. SSPG automata (SSPGA) are closed under boolean operations and their inclusion problem is EXPTIME-complete. The determinization

[1] A graph is k-connected if it cannot be disconnected by removing $k-1$ vertices, $k \geq 2$.

of a nondeterministic SSPGA incurs a $2^{\Theta(n^2)}$ blowup, where n is the number of states of the nondeterministic SSPGA. In contrast, we consider general SP-graphs and the algebraic notion of recognizability, which naturally extends to (disoriented) SP-graphs and graphs of tree-width at most 2.

Nondeterministic automata for unranked trees with unboundedly many ordered siblings have been introduced by Thatcher [19]. These automata are closed under determinization. Several definitions of automata on unranked and unordered trees, equivalent to various fragments of Monadic Second Order (MSO) logic are given in [3]. Algebraic recognizers for unranked (ordered and unordered) trees are also defined in [4], in terms of forest algebras consisting of a horizontal monoid (disjoint union) and a vertical monoid (composition of contexts). To the best of our knowledge, the complexity of converting a nondeterministic automaton for unranked and unordered trees into a (deterministic) forest algebra has not been studied. We conjecture that our encoding of parallel composition as terms over a commutative dioid (Subsection 3.2) could as well apply to bound the cardinality of minimal recognizer algebras of deterministic automata for unranked and unordered trees.

Regular grammars for unranked and unordered trees originate in the seminal work of Courcelle [10]. A more recent definition of regular grammars [6] extends the partitioning of the nonterminals and the syntax of rules, according to this partition, from unranked and unordered trees to SP-graphs and graphs of tree-width 2 at most. Preliminary double-exponential upper bounds on the size of the recognizer for the language of a regular grammar are given, yielding a 2EXPTIME upper bound for the inclusion problem between an arbitrary and a regular grammar. In contrast, in the present paper, we give a (fairly) tight singly-exponential upper bound for the cardinality of the recognizer and show EXPTIME-completeness for inclusion and related problems.

Regular expressions that capture the recognizable sets of graphs of tree-width at most 2 have been introduced in [12]. This definition relies on a semantic *guardedness* condition, which is shown to be decidable, with unknown complexity. The recognizability of the language of a regular expression is established by translation to Counting Monadic Second Order logic (CMSO) in [12], leaving the question about the cardinality of the minimal algebra recognizing the language of a regular expression unanswered.

A natural question is if regular grammars could be extended beyond graphs of tree-width 2. On the one hand, recent results on canonical decompositions of 3- and 4-connected graphs [15] suggest that such an extension might be possible. On the other hand, regular grammars capturing the recognizable sets of graphs of bounded *embeddable* tree-width, an over-approximation of the tree-width that considers only spanning-tree decompositions, have been proposed in [7]. In this work, the bound on the embeddable tree-width of graphs is not fixed. However, the recognizability of the language of a regular grammar is established via translation to CMSO, leaving the cardinality of the recognizer an open problem.

Definitions. The set of natural numbers is denoted by $\mathbb{N}$, and we use $\mathbb{N}^{\geq k} \stackrel{\text{def}}{=} \{n \in \mathbb{N} \mid n \geq k\}$. Given numbers $i, j \in \mathbb{N}$, we write $[i, j] \stackrel{\text{def}}{=} \{i, i+1, \ldots, j\}$, assumed to be empty if $i > j$. The cardinality of a finite set A is denoted by $\|A\|$. The disjoint union $A \uplus B$ is defined as the union of A and B, if $A \cap B = \emptyset$, and undefined, otherwise. The size of a syntactic object x (e.g., a term, expression or grammar) is the length of its representation as a string, denoted $\mathrm{size}(x)$. A function $f : \mathbb{N} \to \mathbb{N}$ is exponential if there exist $c, d, k \in \mathbb{N}^{\geq 1}$ such that $f(n) \leq c \cdot 2^{n^d}$, for all $n \in \mathbb{N}^{\geq k}$.

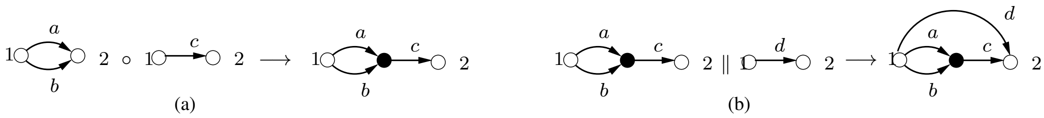

Figure 1 Series (a) and parallel (b) composition.

2 Series-Parallel Graphs

We consider simple binary graphs over a finite alphabet Σ of edge labels. Formally, a graph is a tuple $G = (V, E, \xi)$, where V is a finite set of *vertices*, E is a multiset of edges from $V \times \Sigma \times V$, i.e., multiple edges with the same label are allowed, and $\xi : \{1, 2\} \to V$ is an injective function that designates the *source* $\xi(1)$ and *target* $\xi(2)$ of G, i.e., the source and target of a graph are always distinct. The components of a graph G are denoted as V_G, E_G and ξ_G, respectively. We identify graphs that are isomorphic, i.e., differ only by a renaming of vertices. The *a-bridge* (or simply bridge, when the label a is not important) is the graph consisting of two vertices and a single a-labeled edge, from source to target.

We consider the following operations on graphs (see Figure 1):

- *serial composition* $G_1 \circ G_2$ joins the target of G_1 with the source of G_2. The source and target of $G_1 \circ G_2$ are the source of G_1 and the target of G_2, respectively.
- *parallel composition* $G_1 \parallel G_2$ joins the source and target of G_1 and G_2 into the source and target of the result, respectively. $G^{\parallel k}$ denotes the k-times $\parallel$-composition of G.

The set of *series-parallel* (SP-) graphs is the closure of bridges under the above compositions. An SP-graph G is *atomic* if there are no graphs G_1 and G_2 such that $G = G_1 \circ G_2$. A *P-graph* is an atomic graph that is not a bridge and an *S-graph* is a bridge or a non-atomic graph. A P-graph is the parallel composition of at least two S-graphs, and a non-bridge S-graph is the serial composition of at least two SP-graphs, either bridges or P-graphs [10, Lemma 6.3].

We define recognizable sets of SP-graphs by introducing first their underlying algebra. As usual, an algebra $\mathcal{A} = (\mathsf{A}, \{f^{\mathcal{A}}\}_{f \in \mathcal{F}})$ consists of a domain A and interpretations $f^{\mathcal{A}}$ of a signature $\mathcal{F}$ of functions symbols f, of arity $\#f \in \mathbb{N}$. An algebra is finite if its domain is finite. A term $t[x_1, \ldots, x_n]$ is built as usual from function symbols of matching arities and variables x_i. $\mathsf{Terms}(\mathcal{F}, X)$ denotes the set of terms with variables from X. A *constant* is a term consisting of a single function symbol of zero arity. A *ground* term t has no variables and evaluates to an element $t^{\mathcal{A}} \in \mathsf{A}$, obtained by interpreting the function symbols from t according to $\mathcal{A}$. A *homomorphism* between algebras $\mathcal{A}$ and $\mathcal{B}$ is a mapping $h : \mathsf{A} \to \mathsf{B}$ such that $h(f^{\mathcal{A}}(a_1, \ldots, a_n)) = f^{\mathcal{B}}(h(a_1), \ldots, h(a_n))$, for all $f \in \mathcal{F}$, $\#f = n$ and $a_1, \ldots, a_n \in \mathsf{A}$.

▶ **Definition 1.** *Let $\mathcal{A}$ be an algebra. A set $L \subseteq \mathsf{A}$ is* recognizable *in $\mathcal{A}$ if there exists a finite algebra $\mathcal{B}$ and a homomorphism h between $\mathcal{A}$ and $\mathcal{B}$ such that $L = h^{-1}(F)$, where $F \subseteq \mathsf{B}$ is called the* accepting set *of L.*

We say that L is *recognized* by $\mathcal{B}$, or that $\mathcal{B}$ is a *recognizer* for L, when $\mathcal{A}$ is understood. It is known that, in general, recognizable sets enjoy the following boolean closure properties:

▶ **Proposition 2** (Propositions 2.9 and 2.10 in [17]). *Let $\mathcal{A} = (\mathsf{A}, \{f^{\mathcal{A}}\}_{f \in \mathcal{F}})$ be an algebra and $\mathcal{L}, \mathcal{K} \subseteq \mathsf{A}$ be sets recognized by algebras $\mathcal{B}$ and $\mathcal{C}$, with accepting sets $F \subseteq \mathsf{B}$ and $G \subseteq \mathsf{C}$, respectively. Then, $\mathsf{A} \setminus \mathcal{L}$ is recognized by $\mathcal{B}$ with accepting set $\mathsf{B} \setminus F$ and $\mathcal{L} \cap \mathcal{K}$, $\mathcal{L} \cup \mathcal{K}$ are recognized by $\mathcal{B} \times \mathcal{C}$ with accepting sets $F \times G$ and $(F \times \mathsf{C}) \cup (\mathsf{B} \times G)$, respectively.*

In particular, the signature of the algebra $\mathcal{SP}$ of SP-graphs is the set $\mathcal{F}_{\mathcal{SP}} \stackrel{\text{def}}{=} \{\circ, \|\} \uplus \Sigma$, where $\circ$ and $\|$ are binary function symbols, interpreted as the serial and parallel composition, and each $a \in \Sigma$ is a constant, interpreted as the a-bridge, the domain of the algebra $\mathcal{SP}$ being the set of SP-graphs (up to isomorphism). For instance, the terms $(a \parallel b) \circ c$ and $((a \parallel b) \circ c) \parallel d$ are interpreted in $\mathcal{SP}$ as the rightmost graphs from Figure 1 (a) and (b), respectively. A set of SP-graphs is recognizable if it is recognizable in the $\mathcal{SP}$ algebra (Definition 1).

3 Recognizability of Regular Languages

This section recalls regular grammars for SP-graphs [6]. In particular, we establish that languages defined by regular SP-grammars are recognizable, providing the explicit construction of a recognizer algebra. We note that the existence of such an algebra in itself is not a new result (see Theorem 13 in the extended version [5] of [6]). However, we present a new construction, establishing an exponential bound on its cardinality (Section 4), improving the earlier doubly-exponential construction. We further note that the converse direction (i.e., that each recognizable set of SP-graphs is the language of some regular grammar) has been proved in [6, Theorem 3].

3.1 Regular Grammars

We introduce regular grammars, as representations of the recognizable sets of SP-graphs. A *grammar* $\Gamma = (\mathcal{N}, \mathcal{R}, \mathcal{X})$ consists of a finite set $\mathcal{N} = \{x_1, \ldots, x_n\}$ of *nonterminals*, a set $\mathcal{R} \subseteq \mathcal{N} \times \mathsf{Terms}(\mathcal{F}_{\mathcal{SP}}, \mathcal{N})$ of rules, written $x \to t[x_1, \ldots, x_k]$, where $x, x_1, \ldots, x_k \in \mathcal{N}$ and a set of *axioms* $\mathcal{X} \subseteq \mathcal{N}$. A derivation step is a pair of terms, written $u \Rightarrow_\Gamma v$, such that v is obtained by replacing a single occurrence of a nonterminal x in u by a term t, for some rule $x \to t$ from Γ. A derivation $x \Rightarrow^*_\Gamma t$ is a possibly empty sequence of derivation steps starting with a nonterminal. A derivation is *complete* if t is a ground term. The *language* of Γ is the set of evaluations of ground terms resulting from the complete derivations of Γ that start with an axiom, i.e., $\mathcal{L}(\Gamma) \stackrel{\text{def}}{=} \{t^{\mathcal{A}} \mid x \Rightarrow^*_\Gamma t,\ t \in \mathsf{Terms}(\mathcal{F}_{\mathcal{SP}}, \emptyset),\ x \in \mathcal{X}\}$. A *context-free* set is the language of a grammar. The following definition was introduced in [6]:

▶ **Definition 3.** *A grammar $\Gamma = (\mathcal{P} \uplus \mathcal{S}, \mathcal{R}, \mathcal{X})$ is said to be* regular *if its nonterminals are partitioned into $\mathcal{S}$ and $\mathcal{P}$ and its rules are one of the following forms, for some nonterminals $p, p_1, p_2 \in \mathcal{P}$, $s, s_1, \ldots, s_k \in \mathcal{S}$ and labels $a \in \Sigma$:*
(A) $p \to p \parallel s^{\|\ell}$, *for* $\ell \in \mathbb{N}^{\geq 1}$,
(B) $p \to {s_1}^{\|\ell_1} \parallel \ldots \parallel {s_k}^{\|\ell_k}$, *for* $k, \ell_1, \ldots, \ell_k \in \mathbb{N}^{\geq 1}$, $\sum_{i=1}^{k} \ell_i \geq 2$,
(C) $s \to p \circ s_1$,
(D) $s \to p_1 \circ p_2$,
(E) $p \to a$,
(F) $s \to a$.
We denote by $\mathcal{R}^A$, $\mathcal{R}^B$, $\mathcal{R}^C$, $\mathcal{R}^D$, $\mathcal{R}^E$ and $\mathcal{R}^F$ the subsets of $\mathcal{R}$ consisting rules of the form (A), (B), (C), (D), (E) and (F), respectively. A set L of SP-graphs is regular *if there exists a regular grammar Γ such that $L = \mathcal{L}(\Gamma)$.*

▶ **Example 4.** Consider the regular grammar:

$$\Gamma_{univ} \stackrel{\text{def}}{=} (\underbrace{\{p\}}_{\mathcal{P}} \uplus \underbrace{\{s\}}_{\mathcal{S}}, \mathcal{R}, \{p, s\})$$

having the following rules:

$$\begin{array}{llll} p \rightarrow & p \parallel s & s \rightarrow & p \circ s \\ p \rightarrow & s \parallel s & s \rightarrow & p \circ p \\ p \rightarrow & a & s \rightarrow & a, \text{ for each } a \in \Sigma \end{array}$$

The proof of $\mathcal{L}(\Gamma_{univ}) = \mathsf{SP}$ can be found in [5, Lemma 24]. In contrast, a grammar having rules of the form $p \rightarrow p' \parallel s$ or $s \rightarrow s_1 \circ s_2$, for some $p, p' \in \mathcal{P}$ and $s, s_1, s_2 \in \mathcal{S}$ is not regular.

It is already known that, from a regular grammar Γ, one can build a finite algebra $\mathcal{A}$, of doubly-exponential cardinality in the size of Γ, that recognizes $\mathcal{L}(\Gamma)$ [5, Theorem 13]. The remainder of this section gives a different construction, whose purpose is to show a singly exponential upper bound on the cardinality of the minimal recognizer of $\mathcal{L}(\Gamma)$. The proof that $\|\mathsf{A}\|$ is exponential in size(Γ) will be given in Section 4.

First, we introduce a few technical notions. Given a regular grammar $\Gamma = (\mathcal{P} \uplus \mathcal{S}, \mathcal{R}, \mathcal{X})$ and a nonterminal $p \in \mathcal{P}$, a nonterminal $s \in \mathcal{S}$ is said to be:

- *periodic* for p if Γ contains a rule $p \rightarrow p \parallel s^{\parallel k} \in \mathcal{R}^{\mathrm{A}}$. If there is only one such rule in Γ, we say that k is the *period* of s for p, denoted $\pi_p(s)$.
- *bounded* for p if $s = s_i$ for some rule $p \rightarrow {s_1}^{\parallel \ell_1} \parallel \ldots \parallel {s_k}^{\parallel \ell_k} \in \mathcal{R}^{\mathrm{B}}$ from Γ and some $1 \leq i \leq k$, where $s_1, \ldots, s_k$ are pairwise distinct variables. The *bound* of s for p, denoted $\beta_p(s)$, is the maximum integer ℓ among the occurrences of $s^{\parallel \ell}$ in a rule from $\mathcal{R}^{\mathrm{B}}$ with left-hand side p, plus one.

We denote by $\mathcal{S}^{\pi}_p$ (resp. $\mathcal{S}^{\beta}_p$) the subset of $\mathcal{S}$ consisting of periodic (resp. bounded) nonterminals for p, whenever the grammar in question is understood. Note that $\mathcal{S}^{\pi}_p$ and $\mathcal{S}^{\beta}_p$ are not disjoint and do not cover $\mathcal{S}$, in general.

▶ **Example 5.** In the grammar Γ_{univ} (Example 4), $\mathcal{S}^{\pi}_p = \mathcal{S}^{\beta}_p = \{s\}$, $\pi_p(s) = 1$ and $\beta_p(s) = 2$.

We can however assume w.l.o.g. that $\mathcal{S}^{\pi}_p \cap \mathcal{S}^{\beta}_p = \emptyset$, by putting each regular grammar in the following normal form:

▶ **Definition 6.** *A regular grammar $\Gamma = (\mathcal{P} \uplus \mathcal{S}, \mathcal{R}, \mathcal{X})$ is in* normal form (normalized) *if and only if the following hold:*

1. *for each pair $(p, s) \in \mathcal{P} \times \mathcal{S}$ there is at most one rule $p \rightarrow p \parallel s^{\parallel \ell} \in \mathcal{R}^{A}$,*
2. *$s_1, \ldots, s_k \in \mathcal{S}^{\beta}_p \setminus \mathcal{S}^{\pi}_p$, for each rule $p \rightarrow {s_1}^{\parallel \ell_1} \parallel \ldots \parallel {s_k}^{\parallel \ell_k} \in \mathcal{R}^{B}$, $k \geq 1$.*

We show that each grammar can be normalized at the cost of a polynomial increase in the size of the grammar:

▶ **Proposition 7.** *For each regular grammar $\Gamma = (\mathcal{P} \uplus \mathcal{S}, \mathcal{R}, \mathcal{X})$, one can build, in polynomial time, a normalized regular grammar $\Gamma' = (\mathcal{P} \uplus \mathcal{S}', \mathcal{R}', \mathcal{X})$, such that $\mathcal{L}(\Gamma) = \mathcal{L}(\Gamma')$.*

▶ **Example 8.** Consider the grammar Γ_{univ} (Example 4). The normalized grammar:

$$\Gamma'_{univ} = (\underbrace{\{p\}}_{\mathcal{P}} \uplus \underbrace{\{s, s'\}}_{\mathcal{S}}, \mathcal{R}', \{p, s\})$$

is obtained from Γ_{univ} by replacing the rule $p \rightarrow s \parallel s$ by the rules $p \rightarrow s' \parallel s'$, $s' \rightarrow p \circ p$, $s' \rightarrow p \circ s$ and $s' \rightarrow a$, for all $a \in \Sigma$. This construction, used in the proof of Proposition 7, implies that $\mathcal{L}(\Gamma'_{univ}) = \mathcal{L}(\Gamma_{univ})$.

3.2 A Finite Algebra of Terms

In this section, we assume w.l.o.g. that $\Gamma = (\mathcal{P} \uplus \mathcal{S}, \mathcal{R}, \mathcal{X})$ is a fixed regular grammar in normal form. The definition of a recognizer for the language of a regular grammar given in [5, 6] represents graphs obtained by parallel composition as sets of multisets of nonterminals from $\mathcal{S}$, where the inner level (multisets) counts the number of parallel components up to a certain period and the outer level (sets) acounts for the nondeterminism in the grammar. In the following, we use an equivalent encoding of sets of multisets using terms built from addition, multiplication and nonterminals from $\mathcal{S}$. For instance, the set of multisets $\{\{\!\{s_1, s_2, s_2\}\!\}, \{\!\{s_1, s_1, s_3\}\!\}\}$ is unambiguously encoded by the term $s_1 \cdot s_2^2 + s_1^2 \cdot s_3$.

Formally, the definition of a recognizer for $\mathcal{L}(\Gamma)$ uses finite term algebras $\mathcal{T}_p$, one for each nonterminal $p \in \mathcal{P}$. These algebras have the same signature $\mathcal{F}_\mathcal{T} = \{+, \cdot, 0, 1\}$, where $+$, $\cdot$ are binary operations and 0, 1 are constants. The domain of the algebra $\mathcal{T}_p = (\mathsf{T}_p, +, \cdot, 0, 1)$ is the set of terms built using the function symbols $+$ and $\cdot$ using variables $\mathcal{S}_p^\beta \uplus \mathcal{S}_p^\pi \uplus \mathcal{S}^\theta$, where $\mathcal{S}_p^\beta, \mathcal{S}_p^\pi \subseteq \mathcal{S}$ are the sets of bounded and periodic nonterminals for the given $p \in \mathcal{P}$, respectively, and $\mathcal{S}^\theta$ is a set of *threshold variables*, distinct from $\mathcal{S}$. Note that $\mathcal{S}_p^\beta$ and $\mathcal{S}_p^\pi$ are disjoint, by the assumption that Γ is in normal form (Definition 6). The bound $\beta_p : \mathcal{S}_p^\beta \to \mathbb{N}^{\geq 2}$ and period $\pi_p : \mathcal{S}_p^\pi \to \mathbb{N}^{\geq 1}$ mappings have been defined previously. The threshold variables are artifacts introduced for the purposes of the upcoming complexity proof (Section 4) and do not belong to Γ. We consider a threshold mapping $\theta : \mathcal{S}^\theta \to \mathbb{N}^{\geq 2}$.

Let $p \in \mathcal{P}$ be a fixed nonterminal in the following. The interpretation of $\mathcal{F}_\mathcal{T}$ in $\mathcal{T}_p$ forms an idempotent commutative semiring[2], where $(\mathsf{T}_p, +, 0)$ and $(\mathsf{T}_p, \cdot, 1)$ are commutative monoids, $+$ is idempotent, 0 is cancelling for $\cdot$, and $\cdot$ distributes over $+$. The finiteness of $\mathcal{T}_p$ is ensured by the following additional axioms:

$$(\mathtt{BA})\ s^{\beta_p(s)} = 0,\ s \in \mathcal{S}_p^\beta \qquad (\mathtt{PA})\ s^{\pi_p(s)} = 1,\ s \in \mathcal{S}_p^\pi \qquad (\mathtt{TA})\ s^{\theta(s)} = s^{\theta(s)-1},\ s \in \mathcal{S}^\theta$$

where by s^n we denote the n-times product of s with itself, as usual.

A *monomial* is a term of the form $s_1 \cdot \ldots \cdot s_n$, where $s_1, \ldots, s_n \in \mathcal{S}_p^\beta \uplus \mathcal{S}_p^\pi \uplus \mathcal{S}^\theta$. When no confusion arises, we write $s_1 s_2$ instead of $s_1 \cdot s_2$. A monomial is *reduced* if none of the (BA), (PA) and (TA) axioms can be applied to reduce the number of occurrences of variables in it. For a monomial m, we denote by $\mathit{vars}(m)$ the *support* of m, i.e., the set of variables occurring in m. For a variable $s \in \mathit{vars}(m)$, we denote by $\deg(m, s)$ the degree of s in m and by $\deg(m)$ the degree of m, defined as $\deg(m) \stackrel{\text{def}}{=} \sum_{s \in \mathit{vars}(m)} \deg(m, s)$. A term is *linear* if it is a sum $s_1 + \ldots + s_n$ of variables. A *linear product* is a term of the form $\ell_1 \cdot \ldots \cdot \ell_n$, where ℓ_i are linear terms.

Due to the axioms of distributivity, each term $t \in \mathsf{T}_p$ is equivalent to a sum of distinct reduced monomials, called *normal form*, denoted $\mathrm{nf}_p(t)$. When no confusion arises, by writing $m \in \mathrm{nf}_p(t)$ we mean that the monomial m occurs as a summand in the normal form of t. Note that the normal form of a term is unique modulo associativity and commutativity of the binary operations and the set $\{\mathrm{nf}_p(t) \mid t \in \mathsf{T}_p\}$ is finite. In the following, we shall need an upper bound on the number of monomials from a term in normal form:

▶ **Lemma 9.** *Let $t \in \mathsf{Terms}(\mathcal{F}_\mathcal{T}, \mathcal{S})$ be a term in normal form. Then each monomial $m \in t$ is of size at most $\theta(\Gamma) \cdot \|\mathcal{S}\|$ and there are at most $\theta(\Gamma)^{\|\mathcal{S}\|}$ such monomials.*

[2] Often also called a commutative dioid.

3.3 The Recognizer of a Regular Set

We define a finite algebra $\mathcal{A} = (\mathsf{A}, \circ^{\mathcal{A}}, \|^{\mathcal{A}}, \{a^{\mathcal{A}}\}_{a\in\Sigma})$ and a homomorphism h between $\mathcal{SP}$ and $\mathcal{A}$, such that $\mathcal{L}(\Gamma) = h^{-1}(F)$, where $F \stackrel{\text{def}}{=} h(\mathcal{L}(\Gamma))$, thus establishing that $\mathcal{L}(\Gamma)$ is recognizable. Even though the definitions of $\mathcal{A}$ and h depend on Γ, we omit mentioning it, to avoid clutter.

Intuitively, a P-graph (i.e., a parallel composition of at least two graphs) is represented by a tuple of terms $\langle t_p\rangle_{p\in\mathcal{P}}$, where $t_p \in \mathsf{T}_p$, for each $p \in \mathcal{P}$. Each term $\mathrm{nf}_p(t_p)$ is a sum of monomials, that represent the possible ways in which the P-graph is parsed by Γ starting from p, using only rules from $\mathcal{R}^{\mathrm{A}} \cup \mathcal{R}^{\mathrm{B}}$ (modulo repetitions of rules from $\mathcal{R}^{\mathrm{A}}$). This motivates the following notions of *view*, i.e., a monomial that represents a set of derivations having a common prefix of the form $p \Rightarrow^*_\Gamma s_1 \| \dots \| s_n$, and *profile*, i.e., a sum of monomials that accounts for the nondeterminism in Γ:

▶ **Definition 10.** *Let G be a P-graph. A* view *of G is a monomial $m = s_1 \cdot \ldots \cdot s_n$, for which there exist complete derivations $s_i \Rightarrow^*_\Gamma t_i$, for all $i \in [1, n]$, such that $G = (t_1 \| \dots \| t_n)^{\mathcal{SP}}$. For a nonterminal $p \in \mathcal{P}$, the p-profile of G is the term $h_p(G) \stackrel{\text{def}}{=} \mathrm{nf}_p\left(\sum_{m \text{ view of } G} m\right)$. The* profile *of G is the tuple $h(G) \stackrel{\text{def}}{=} \langle h_p(G)\rangle_{p\in\mathcal{P}}$.*

A view of an S-graph (i.e., either a bridge or a sequential composition of least two graphs) is a pair (s, q), where $s \in \mathcal{S}$ represents the beginning and $q \in \mathcal{P} \uplus \mathcal{S} \uplus \{\bot\}$ the end of a derivation $s \Rightarrow^*_\Gamma p_1 \circ \ldots p_n \circ q$ (resp. $s \Rightarrow^*_\Gamma p_1 \circ \ldots p_n$ if $q = \bot$) using only rules from $\mathcal{R}^{\mathrm{C}} \cup \mathcal{R}^{\mathrm{D}}$, for $p_1, \ldots, p_n \in \mathcal{P}$. This representation of S-graphs is similar to the classical representation of words by pairs of states in a finite word automaton.

For two terms t and u, over the $\mathcal{F}_{\mathcal{SP}}$ signature, we denote by $t \stackrel{\mathsf{a}}{\sim} u$ the fact that t can be rewritten into u by applying zero or more times the associativity law for $\circ$. For a grammar Γ, we write $t \stackrel{\mathsf{a}}{\Rightarrow}{}^*_\Gamma\, u$ if $t \Rightarrow^*_\Gamma w$ and $w \stackrel{\mathsf{a}}{\sim} u$. The profile of an S-graph is a set of views representing the nondeterminism in Γ:

▶ **Definition 11.** *Let G be an S-graph. A* view *of G is a pair $(s, q) \in \mathcal{S} \times (\mathcal{S} \uplus \mathcal{P} \uplus \{\bot\})$, for which there exists a derivation, either $s \stackrel{\mathsf{a}}{\Rightarrow}{}^*_\Gamma\, t \circ q$, if $q \neq \bot$, or $s \stackrel{\mathsf{a}}{\Rightarrow}{}^*_\Gamma\, t$, otherwise, and t is a ground term such that $G = t^{\mathcal{SP}}$. The* profile *of G is the relation $h(G) \stackrel{\text{def}}{=} \{(s, q) \in \mathcal{S} \times (\mathcal{S} \uplus \mathcal{P} \uplus \{\bot\}) \mid (s, q)$ is a view of $G\}$.*

The definition of $\mathcal{A}$ benefits from the notion of *alternative* grammar, that differs from regular grammar only in that we allow rules of the form $p \to s$ (i.e., in violation of condition (B) of Definition 3), where $s \in \mathcal{S}$ occurs in a rule $s \to a$, $a \in \Sigma$ and nowhere else. This allows us to represent the bridges that may occur in the parallel composition of a P-graph using nonterminals from $\mathcal{S}$, which provides a uniform definition of views for P-graphs. Each regular grammar can be transformed into a language-equivalent alternative grammar by adding a rule $s_a \to a$ and replacing each rule $p \to a$ with $p \to s_a$, for each $a \in \Sigma$. In the following, we denote by $\mathcal{R}^\star$ the set of newly introduced rules of this form, in an alternative grammar. If, moreover, $p \in \mathcal{X} \cap \mathcal{P}$ is an axiom of Γ, we promote s_a to be axiom of the alternative grammar. In the following, we consider w.l.o.g. Γ to be an alternative grammar in normal form.

The domain of the recognizer algebra $\mathcal{A}$ is $\mathsf{A} \stackrel{\text{def}}{=} \{h(G) \mid G \in \mathsf{SP}\}$. We denote by A^P and A^S the sets of P-profiles (i.e., profiles of P-graphs) and S-profiles (i.e., profiles of S-graphs), respectively. Note that $\mathsf{A}^P \uplus \mathsf{A}^S = \mathsf{A}$. For technical convenience, in the definition of $\mathcal{A}$, we consider the mappings `par`, that converts an arbitrary profile into a tuple of terms in normal form, and `seq`, that converts an arbitrary profile into a relation between $\mathcal{P} \uplus \mathcal{S}$ and $\mathcal{P} \uplus \mathcal{S} \uplus \{\bot\}$:

$$\mathtt{par}(x) \stackrel{\text{def}}{=} \begin{cases} x & \text{if } x \in \mathsf{A}^P \\ \langle \mathrm{nf}_p \left(\sum \{ s \mid (s,\bot) \in x,\ s \in \mathcal{S}^{\beta}_p \uplus \mathcal{S}^{\pi}_p \} \right) \rangle_{p \in \mathcal{P}} & \text{if } x \in \mathsf{A}^S \end{cases}$$

$$\mathtt{seq}(x) \stackrel{\text{def}}{=} \begin{cases} x & \text{if } x \in \mathsf{A}^S \\ \{(p,\bot) \mid p \rightsquigarrow m \in t_p,\ p \in \mathcal{P}\} \cup \{(s,q) \mid s \to p \circ q \in \mathcal{R}\} & \text{if } x = \langle t_p \rangle_{p \in \mathcal{P}} \in \mathsf{A}^P \end{cases}$$

where $\langle x \rangle_p$ denotes the p-th component of a tuple $x \in \mathsf{A}^P$ and $\mathrm{nf}(\langle t_p \rangle_{p\in\mathcal{P}}) \stackrel{\text{def}}{=} \langle \mathrm{nf}_p(t_p) \rangle_{p \in \mathcal{P}}$ denotes the component-wise normal form. Further, by writing $p \rightsquigarrow m$, we mean the existence of a rule $p \to s_1{}^{\|\ell_1} \| \ldots \| s_n{}^{\|\ell_n} \in \mathcal{R}^{\mathrm{B}}$ such that $m = s_1^{\ell_1} \cdot \ldots \cdot s_n^{\ell_n}$, and by $p \rightsquigarrow^* m$ we mean the existence of a derivation $p \Rightarrow^*_{\Gamma} s_1 \| \ldots \| s_n$, using only rules from $\mathcal{R}^{\mathrm{A}} \cup \mathcal{R}^{\mathrm{B}}$, such that $m = s_1 \cdot \ldots \cdot s_n$. The $\rightsquigarrow$ and $\rightsquigarrow^*$ relations enjoy the following property:

▶ **Lemma 12.** *For each nonterminal $p \in \mathcal{P}$ and monomial $m \in \mathsf{T}_p$, we have $p \rightsquigarrow^* m$ if and only if $p \rightsquigarrow \mathrm{nf}_p(m)$.*

The interpretation of the signature $\mathcal{F}_{\mathcal{SP}}$ in the algebra $\mathcal{A}$ is defined below, for all $a \in \Sigma$ and $x_1, x_2 \in \mathsf{A}$, where $y_1; y_2$ denotes the relational composition, i.e., $y_1; y_2 = \{(a,c) \mid (a,b) \in y_1, (b,c) \in y_2\}$, and $\langle t_p \rangle_{p\in\mathcal{P}} \cdot \langle u_p \rangle_{p \in \mathcal{P}} \stackrel{\text{def}}{=} \langle t_p \cdot u_p \rangle_{p\in\mathcal{P}}$ denotes component-wise multiplication:

$$a^{\mathcal{A}} \stackrel{\text{def}}{=} \{(s,\bot) \mid s \to a \in \mathcal{R}^{\mathrm{F}}\} \cup \{(s,q) \mid s \to p \circ q \in \mathcal{R}^{\mathrm{C}},\ p \to s_a \in \mathcal{R}^{\star}\}$$

$$x_1 \|^{\mathcal{A}} x_2 \stackrel{\text{def}}{=} \mathrm{nf}(\mathtt{par}(x_1) \cdot \mathtt{par}(x_2)) \qquad x_1 \circ^{\mathcal{A}} x_2 \stackrel{\text{def}}{=} \mathtt{seq}(x_1); \mathtt{seq}(x_2)$$

Note that, since bridges are considered to be S-graphs, $a^{\mathcal{A}}$ is an S-profile. Since the definition of $a^{\mathcal{A}}$ assumes the grammar to be alternative, hence there two ways in which a bridge can be parsed, either $s \Rightarrow_{\Gamma} a$, thus yielding the view $(s,\bot)$, or $s \Rightarrow_{\Gamma} p \circ q \Rightarrow_{\Gamma} s_a \circ q \Rightarrow_{\Gamma} a \circ q$, thus yielding the view (s,q).

The parallel composition is interpreted in the recognizer algebra as the multiplication in the term algebra, and the serial composition is interpreted as relational composition. Intuitively, the interpretation of $\mathcal{F}_{\mathcal{SP}}$ in the algebra $\mathcal{A}$ mimicks the parallel and serial composition operations at the level of profiles, as shown by the following example:

▶ **Example 13.** Consider the regular grammar Γ_{univ} from Example 4. The corresponding alternative grammar is obtained by first normalizing the grammar Γ_{univ} (see Example 8) and then replacing the rule $p \to a$ with the rules $p \to s_a$ and $s_a \to a$, where s_a is a fresh nonterminal, for each $a \in \Sigma$. The resulting grammar:

$$\Gamma_{univ} \stackrel{\text{def}}{=} (\underbrace{\{p\}}_{\mathcal{P}} \uplus \underbrace{\{s, s'\} \cup \{s_a \mid a \in \Sigma\}}_{\mathcal{S}}, \mathcal{R}, \{p, s\})$$

has the following rules:

$$\begin{array}{llllllll} p \to & p \| s & s \to & p \circ s & p \to & s_a & s_a \to & a \\ p \to & s' \| s' & s \to & p \circ p & s \to & a & s' \to & a, \text{ for each } a \in \Sigma \\ s' \to & p \circ s & s' \to & p \circ p & & & & \end{array}$$

For some $a \in \Sigma$, we compute, successively:

$$\begin{aligned} a^{\mathcal{A}} &= \{(s_a,\bot),(s,\bot),(s,p),(s,s),(s',\bot),(s',p),(s',s)\} \\ \mathtt{par}(a^{\mathcal{A}}) &= \langle s + s' + s_a \rangle \\ a^{\mathcal{A}} \|^{\mathcal{A}} a^{\mathcal{A}} &= s + s'^2 + s \cdot s' + s \cdot s_a + s_a \cdot s' \\ \mathtt{seq}(a^{\mathcal{A}} \|^{\mathcal{A}} a^{\mathcal{A}}) &= \{(s,p),(s,s),(s',p),(s',s),(p,\bot)\} \\ (a^{\mathcal{A}} \|^{\mathcal{A}} a^{\mathcal{A}}) \circ^{\mathcal{A}} a^{\mathcal{A}} &= \{(s,\bot),(s,p),(s,s),(s',\bot),(s',p),(s',s)\} \end{aligned}$$

It is easy to check that $h((a \parallel a) \circ a) = \{(s, \bot), (s, p), (s, s), (s', \bot), (s', p), (s', s)\}$, because the right-hand side contains all the views of the S-graph $(a \parallel a) \circ a$ for Γ_{univ}. Consequently, we have $h((a \parallel a) \circ a) = (a^{\mathcal{A}} \parallel^{\mathcal{A}} a^{\mathcal{A}}) \circ^{\mathcal{A}} a^{\mathcal{A}}$.

The following lemma proves the correctness of the above definition:

▶ **Lemma 14.** *h is a homomorphism between $\mathcal{SP}$ and $\mathcal{A}$.*

Moreover, $\mathcal{L}(\Gamma)$ does not distinguish graphs with the same image in $\mathcal{A}$:

▶ **Lemma 15.** *SP-graphs having the same image via h are indistinguishable by $\mathcal{L}(\Gamma)$.*

The main result of this section is a consequence of Lemmas 15 and 14 above:

▶ **Theorem 16.** *The language of each regular grammar Γ is recognized by the finite algebra $\mathcal{A}$ with accepting set $F \stackrel{\text{def}}{=} h(\mathcal{L}(\Gamma))$.*

Proof. Since h is a homomorphism between $\mathcal{SP}$ and $\mathcal{A}$, by Lemma 14, it remains to prove that $\mathcal{L}(\Gamma) = h^{-1}(F)$. "$\subseteq$" This direction is trivial, by the definition of F. "$\supseteq$" For each SP-graph $G \in h^{-1}(F)$ there exists $G' \in \mathcal{L}(\Gamma)$ such that $h(G) = h(G')$. By Lemma 15, we obtain $G \in \mathcal{L}(\Gamma)$. ◀

We also need the following constructive definition of the accepting set of the $\mathcal{A}$ recognizer built in the previous:

▶ **Lemma 17.** *The accepting set F consists of the elements $a \in \mathsf{A}$, for which either one of the following holds:*

1. *if $a = \langle t_p \rangle_{p \in \mathcal{P}} \in F \cap \mathsf{A}^P$ then there exists an axiom $p \in \mathcal{P} \cap \mathcal{X}$ and a nonempty monomial $m \in t_p$ such that $p \rightsquigarrow m$.*
2. *if $a \in F \cap \mathsf{A}^S$ then $(s, \bot) \in a$, for an axiom $s \in \mathcal{S} \cap \mathcal{X}$.*

The next proposition gives a lower bound on the cardinality of the minimal recognizer for a regular grammar, which improves on the standard 2^n lower bound of the recognizers for the language of a nondeterministic word or tree automaton [8, Theorem 1.3.1]. The worst-case example has been adapted to our setting from [1, Theorem 3.4], see also [2, Theorem 5.2]:

▶ **Proposition 18.** *There exists a regular grammar Γ having $\mathcal{O}(n)$ nonterminals and $\mathcal{O}(n \log_2 n)$ rules, such that each recognizer for $\mathcal{L}(\Gamma)$ requires at least 2^{n^2} elements.*

4 An Exponential Bound on the Cardinality of the Recognizer

We establish a $2^{\mathcal{O}(n^9)}$ upper bound on the cardinality of the recognizer $\mathcal{A}$ built from the regular grammar Γ, where n is the size of Γ (Corollary 32). This upper bound is (almost) optimal, since a $2^{\Omega(n^2)}$ blowup is unavoidable (Proposition 18).

Because the domain of $\mathcal{A}$ is partitioned into S-profiles (A^S) and P-profiles (A^P), we analyse the two cases separately. Note first that there are at most $\|\mathcal{S}\| \cdot (\|\mathcal{P}\| + \|\mathcal{S}\| + 1)$ pairs of the form $(s, q) \in \mathcal{S} \times (\mathcal{P} \uplus \mathcal{S} \uplus \{\bot\})$ in A^S, hence $\|\mathsf{A}^S\|$ is exponential in the size of Γ (Definition 11). In the rest of this section, we prove that $\|\mathsf{A}^P\|$ is also exponential in the size of Γ, which provides an exponential bound for $\|\mathsf{A}\| = \|\mathsf{A}^S\| + \|\mathsf{A}^P\|$.

For the purpose of proving that $\|\mathsf{A}^P\|$ is exponentially bounded, a key observation is that A^P consists of tuples of normal forms of linear products[3] over variables from $\mathcal{S}$. Indeed, this is the case because each $\langle t_p \rangle_{p \in \mathcal{P}} \in \mathsf{A}^P$ is the profile $h(G) = \langle h_p(G) \rangle_{p \in \mathcal{P}}$ of a P-graph G,

[3] A linear product is a term of the form $(s_{11} + \ldots + s_{1n_1}) \cdot \ldots \cdot (s_{k1} + \ldots + s_{kn_k})$, where $s_{ij} \in \mathcal{S}$.

which is the parallel composition of at least two S-graphs, i.e., $G = G_1 \parallel \ldots \parallel G_n$, $n \geq 2$. This means $h_p(G) = \mathrm{nf}_p(\mathtt{par}(h(G_1))_p \cdot \ldots \cdot \mathtt{par}(h(G_n))_p)$, for each $p \in \mathcal{P}$. Because each G_i is an S-graph, we have $h(G_i) \in \mathsf{A}^S$ hence each term $\mathtt{par}(h(G_i))_p$ is a sum of variables from $\mathcal{S}$, by the definition of $\mathtt{par}$. Then, $\mathtt{par}(h(G_1))_p \cdot \ldots \cdot \mathtt{par}(h(G_n))_p$ is a linear product and $t_p = h_p(G)$ is its normal form, for each $p \in \mathcal{P}$. Consequently, $\|\mathsf{A}^P\|$ is at most the number of linear products over $\mathcal{S}$ that have distinct normal forms, at power $\|P\|$.

4.1 Counting Products of Linear Terms

Let us fix a non-terminal $p \in \mathcal{P}$ and let $\mathcal{S}^\beta$ and $\mathcal{S}^\pi$ denote the sets of bounded and periodic variables associated to p, respectively. Since p is fixed, we omit mentioning it in the following. We also consider a set $\mathcal{S}^\theta$ of threshold variables. For simplicity, in this context, we assume that $\mathcal{P} = \{p\}$ and $\mathcal{S} = \mathcal{S}^\beta \uplus \mathcal{S}^\pi \uplus \mathcal{S}^\theta$.

The term algebra corresponding to p is denoted $\mathcal{T} = (\mathsf{T}, +, \cdot, 0, 1)$. We recall that the domain $\mathsf{T} \stackrel{\text{def}}{=} \mathsf{Terms}(\mathcal{F}_\mathcal{T}, \mathcal{S})$ of the term algebra is the set of terms built from $\mathcal{S}$ using the operations $+$ and $\cdot$ (Subsection 3.2). We assume w.l.o.g. that each variable $s \in \mathcal{S}^\pi$ has period $\pi(s) \geq 2$: if $\pi(s) = 1$ for some variable s, then $\mathrm{nf}(s) = 1$, by the $(\mathtt{PA})$ axiom, hence s either does not occur in the normal form of a linear term, or yields the empty monomial 1.

The goal of this subsection is to provide an exponential bound on the number of terms in normal form $\mathrm{nf}(t)$[4], where t is a linear product of bounded and periodic variables. This is achieved by providing a polynomial cut-off bound on the length (i.e., number of factors) of the linear products having the same normal form. The existence of a cut-off c means that each product longer than c has the same normal form as a product shorter than c. The cut-off allows to bound the number of distinct linear product terms by an exponential, since the number of linear products of length at most c is exponential in c.

The polynomial cut-off is obtained by considering progressively the cases of linear factors consisting of (i) bounded variables only (Lemma 19), (ii) threshold variables only (Lemmas 20 and 22), (iii) bounded and periodic variables containing the 1 monomial (Lemma 23), (iv) bounded and periodic variables containing a distinguished periodic variable (Lemmas 25 and 27) and (v) the general case of arbitrary bounded and periodic variables (Lemma 28). We recall that the threshold variables do not occur in the input grammar Γ, being introduced to support the intermediate steps of the proof.

For each set $V \subseteq \mathcal{S}$ of nonterminals, we define the following measure:

$$\|V\|_1 \stackrel{\text{def}}{=} \sum_{s \in V \cap \mathcal{S}^\beta} (\beta(s) - 1) + \sum_{s \in V \cap \mathcal{S}^\pi} (\pi(s) - 1) + \sum_{s \in V \cap \mathcal{S}^\theta} (\theta(s) - 1) \tag{1}$$

We consider the following sets of linear terms and products, for $V \subseteq \mathcal{S}$ and $U \subseteq \mathsf{T}$:

$$\mathsf{Sums}(V) \stackrel{\text{def}}{=} \left\{ \sum\nolimits_{s \in W} s \mid \emptyset \neq W \subseteq V \right\} \qquad \mathsf{Prods}(U) \stackrel{\text{def}}{=} \{u_1 \ldots u_k \mid u_i \in U,\ i \in [1,k],\ k \geq 0\}$$

The length of a product $t = u_1 \ldots u_n$ of linear terms $u_1, \ldots, u_n$ is denoted $len(t) \stackrel{\text{def}}{=} n$. The below lemma shows that linear products over bounded variables only, that are longer than $\|\mathcal{S}^\beta\|_1$ have normal form 0, which provides a trivial cut-off on their length:

▶ **Lemma 19.** *For each linear product $t \in \mathsf{Prods}(\mathsf{Sums}(\mathcal{S}^\beta))$ of length $len(t) > \|\mathcal{S}^\beta\|_1$, we have $\mathrm{nf}(t) = 0$.*

---

[4] We also write $\mathrm{nf}(t)$ instead of $\mathrm{nf}_p(t)$ because the nonterminal $p \in \mathcal{P}$ is fixed.

Before considering the other cases, we need further notations. Given reduced monomials m_1, m_2 we define $m_1 \sqsubseteq m_2$ iff $vars(m_1) \subseteq vars(m_2)$ and $\deg(m_1, s) \leq deg(m_2, s)$ for all $s \in vars(m_1)$. We also write $m_1 \sqsubset m_2$ iff $m_1 \sqsubseteq m_2$ and $m_1 \neq m_2$. For a term $\mathrm{nf}(t)$ in normal form we denote by $\sup(\mathrm{nf}(t))$ the subterm of $\mathrm{nf}(t)$ containing only the maximal monomials with respect to $\sqsubseteq$. We also define $\max\deg \mathrm{nf}(t) \stackrel{\text{def}}{=} \max_{m \in \mathrm{nf}(t)} \deg(m)$, i.e., the maximal degree of monomials m occurring in $\mathrm{nf}(t)$.

The following two lemmas give the cut-off on the length of linear products over threshold variables only. Lemma 20 gives an auxiliary result concerning variables of threshold 2, called 2-threshold in the following, namely that the maximal (i.e., w.r.t. $\sqsubseteq$) monomials of a linear product using only 2-threshold variables are exactly its monomials of maximal degree. Most of its proof shows that a linear product t of 2-threshold variables enjoys a similar exchange property[5] as matroids, guaranteeing that maximal bases have equal cardinalities [22].

▶ **Lemma 20.** *Assume that $\theta(s) = 2$ for each $s \in \mathcal{S}^\theta$. Then, for each linear product $t \in \mathsf{Prods}(\mathsf{Sums}(\mathcal{S}^\theta))$, we have $\sup(\mathrm{nf}(t)) = \sum\{m \in \mathrm{nf}(t) \mid \deg(m) = \max\deg(\mathrm{nf}(t))\}$.*

▶ **Example 21.** Consider threshold variables $s_1, \ldots, s_4 \in \mathcal{S}^\theta$ such that $\theta(s_i) = 2$, for all $i \in [1, 4]$. Let $t \stackrel{\text{def}}{=} (s_1 + s_2)(s_2 + s_3)(s_3 + s_4)$. We compute:

$$\begin{aligned}
\mathrm{nf}(t) &= s_1s_2s_3 + s_1s_2s_4 + s_1s_3 + s_1s_3s_4 + s_2s_3 + s_2s_3s_4 \\
\sup(\mathrm{nf}(t)) &= s_1s_2s_3 + s_1s_2s_4 + s_1s_3s_4 + s_2s_3s_4
\end{aligned}$$

and observe that $\max\deg(\mathrm{nf}(t)) = 3$ and $\sup(\mathrm{nf}(t)) = \sum\{m \in \mathrm{nf}(t) \mid \deg(m) = 3\}$.

For two products $t_1, t_2 \in \mathsf{Prods}(U)$, we write $t_1 \preceq t_2$ if t_1 is a subproduct of t_2, i.e., t_1 is the product of a subset of the factors of t_2. Note that $\preceq$ agrees with $\sqsubseteq$ over reduced monomials. We write $t_1 \prec t_2$ whenever $t_1 \preceq t_2$ and $t_1 \neq t_2$ and define $[t_1, t_2]_\preceq \stackrel{\text{def}}{=} \{t \mid t_1 \preceq t \preceq t_2\}$. With these notations, Lemma 22 generalizes Lemma 20 from 2 to arbitrary thresholds and provides a cut-off bound on the linear products consisting of threshold variables only:

▶ **Lemma 22.** *For each linear product $t \in \mathsf{Prods}(\mathsf{Sums}(\mathcal{S}^\theta))$, the following hold:*

(i) $\sup(\mathrm{nf}(t)) = \sum\{m \in \mathrm{nf}(t) \mid \deg(m) = \max\deg(\mathrm{nf}(t))\}$.

(ii) $\sup(\mathrm{nf}(t)) = \sup(\mathrm{nf}(t \cdot \ell))$, *for each linear term $\ell \in \mathsf{Sums}(\mathcal{S}^\theta)$ such that* $\max\deg(\mathrm{nf}(t)) = \max\deg(\mathrm{nf}(t \cdot \ell))$.

(iii) *if $len(t) > \|\mathcal{S}^\theta\|_1$ then there exists $t' \prec t$, such that $\sup(\mathrm{nf}(t)) = \sup(\mathrm{nf}(t''))$, for all $t'' \in [t', t]_\preceq$.*

For each $V \subseteq \mathcal{S}$, we denote by $\text{1-}\mathsf{Sums}(V) \stackrel{\text{def}}{=} \{1 + \sum_{s \in W} s \mid \emptyset \neq W \subseteq V\}$ the set of linear products whose factors all contain the empty monomial 1. Lemma 23 gives a cut-off on the length of products of such linear terms, relying on simple observation: a linear term $1 + s$, for $s \in \mathcal{S}^\beta \uplus \mathcal{S}^\pi$, behaves as a threshold variable of threshold equal to $\beta(s)$ resp. $\pi(s)$. Intuitively, this is the case because the normal forms of such linear products are downward closed w.r.t. the $\sqsubseteq$ order on monomials. Hence, the bounded and periodic variables can be syntactically substituted with threshold variables and Lemma 22 can be applied.

▶ **Lemma 23.** *For each linear product $t \in \mathsf{Prods}(\text{1-}\mathsf{Sums}(\mathcal{S}^\beta \uplus \mathcal{S}^\pi))$ of length $len(t) > \|\mathcal{S}^\beta \uplus \mathcal{S}^\pi\|_1$, there exists $t' \prec t$ such that $\mathrm{nf}(t) = \mathrm{nf}(t'')$, for all $t'' \in [t', t]_\preceq$.*

[5] For each $m, n \in \mathrm{nf}(t)$ such that $\deg(m) < \deg(n)$ there exists $z \in vars(n) \setminus vars(m)$ such that $m \cdot z \in \mathrm{nf}(t)$.

▶ **Example 24.** Let $s_1 \in \mathcal{S}^\beta$ and $s_2 \in \mathcal{S}^\pi$ be variables such that $\beta(s_1) = 2$, $\pi(s_2) = 3$ and $t \stackrel{\text{def}}{=} (1 + s_1 + s_2)(1 + s_1)(1 + s_2)(1 + s_1)(1 + s_1)(1 + s_1 + s_2)(1 + s_2)$. Then $\mathrm{nf}(t) = 1 + s_1 + s_2 + s_1 s_2 + s_2^2 + s_1 s_2^2$ and note that $\mathrm{nf}(t)$ contains all reduced monomials constructed from s_1 and s_2. Let $t' \stackrel{\text{def}}{=} (1 + s_1 + s_2)(1 + s_1)(1 + s_2)$, that is, t' contains only the first three factors of t. Then, $\mathrm{nf}(t') = \mathrm{nf}(t)$ is an easy check. Moreover, $\mathrm{nf}(t'') = \mathrm{nf}(t)$ for all $t'' \in [t', t]_\preceq$, because $\mathrm{nf}(t'')$ contains already all the monomials of $\mathrm{nf}(t')$, i.e., of $\mathrm{nf}(t)$.

For each $s_0 \in \mathcal{S}$ and $V \subseteq \mathcal{S}$, we denote by $s_0\text{-}\mathsf{Sums}(V) \stackrel{\text{def}}{=} \{s_0 + \sum_{s \in W} s \mid \emptyset \neq W \subseteq V\}$ the set of linear terms whose factors all contain a distinguished variable s_0. Lemmas 25 and 27 generalize Lemma 23 by considering products of linear terms containing a periodic variable s_0 instead of 1. In particular, Lemma 25 considers the situation where the period of s_0 divides the periods of all other periodic variables occurring in the product. In this case, the normal form of a linear product becomes downward closed w.r.t. $\sqsubseteq$ when s_0 is removed from all monomials. We point out that for the products whose length exceeds the cut-off bound, Lemma 25 only guarantees some strictly shorter product having the same normal form, whereas Lemma 23 guarantees equal normal forms for all linear products whose length exceeds the cut-off bound.

▶ **Lemma 25.** *For each periodic variable $s_0 \in \mathcal{S}^\pi$, whose period $\pi(s_0)$ divides all periods $\{\pi(s) \mid s \in \mathcal{S}^\pi\}$, and each linear product $t \in \mathsf{Prods}(s_0\text{-}\mathsf{Sums}(\mathcal{S}^\beta \uplus \mathcal{S}^\pi))$ of length $len(t) > \|\mathcal{S}^\beta \uplus \mathcal{S}^\pi\|_1$, there exists $t' \prec t$ such that $\mathrm{nf}(t') = \mathrm{nf}(t)$.*

▶ **Example 26.** Consider a bounded variable $s_1 \in \mathcal{S}^\beta$ and periodic variables $s_0, s_2 \in \mathcal{S}^\pi$ such that $\beta(s_1) = 2$, $\pi(s_0) = \pi(s_2) = 3$. Let $t \stackrel{\text{def}}{=} (s_0 + s_1 + s_2)(s_0 + s_1)(s_0 + s_2)(s_0 + s_1)(s_0 + s_1)(s_0 + s_1 + s_2)(s_0 + s_2)$ and observe that $len(t) = 7$. We compute $\mathrm{nf}(t) = s_0 + s_1 + s_2 + s_0^2 s_1 s_2 + s_0^2 s_2^2 + s_0 s_1 s_2^2$. Note that for all monomials $m \in \mathrm{nf}(t)$ we have $\deg(m) \equiv 7 \bmod 3$, i.e., $\deg(m) \equiv len(t) \bmod \pi(s_0)$. Moreover, note that by erasing s_0 from monomials of $\mathrm{nf}(t)$ we obtain all possible reduced monomials of s_1, s_2. Let then $t' \stackrel{\text{def}}{=} (s_0 + s_1 + s_2)(s_0 + s_1)(s_0 + s_2)(s_0 + s_1)$, that is, t' contains the first four factors of t. Then $\mathrm{nf}(t') = \mathrm{nf}(t)$ is an easy check, in particular by taking into account the properties of $\mathrm{nf}(t)$ mentioned above.

Lemma 27 generalizes Lemma 25 by lifting the restriction about the period of s_0 dividing all other periods. This increases the cut-off bound above which factors become redundant by a polynomial in the periods of the variables from $\mathcal{S}^\pi$, where $\mathrm{lcm}(a, b)$ denotes the least common multiple of $a, b \in \mathbb{N}^{\geq 1}$:

▶ **Lemma 27.** *For each periodic variable $s_0 \in \mathcal{S}^\pi$ and linear product $t \in \mathsf{Prods}(s_0\text{-}\mathsf{Sums}(\mathcal{S}^\beta \uplus \mathcal{S}^\pi))$ of length $len(t) > \|\mathcal{S}^\beta\|_1 + \sum_{s \in \mathcal{S}^\pi}(\mathrm{lcm}(\pi(s), \pi(s_0)) - 1)$, there exists $t' \prec t$ such that $\mathrm{nf}(t') = \mathrm{nf}(t)$.*

Lemma 28 below handles the general case of linear products over both bounded and periodic variables (threshold variables are not needed any further). This provides an explicit polynomial cut-off on the length of such products, guaranteeing that each longer product is either 0 or equivalent in normal form to a strict sub-product. The proof uses a reordering of product factors that enables the joint application of Lemmas 19 and 27. Let $\mathcal{S}^\pi \stackrel{\text{def}}{=} \{s_1, \ldots, s_n\}$ be the set of periodic variables and define the bound:

$$b(\mathcal{S}^\beta, \mathcal{S}^\pi) \stackrel{\text{def}}{=} \|\mathcal{S}^\beta\|_1 \cdot (\|\mathcal{S}^\pi\| + 1) + \sum_{1 \leq i \leq j \leq n} \mathrm{lcm}(\pi(s_j), \pi(s_i)) - \|\mathcal{S}^\pi\| \cdot (\|\mathcal{S}^\pi\| + 1)/2 \tag{2}$$

▶ **Lemma 28.** *For each linear product $t \in \mathsf{Prods}(\mathsf{Sums}(\mathcal{S}^\beta \uplus \mathcal{S}^\pi))$ of lenght $len(t) > b(\mathcal{S}^\beta, \mathcal{S}^\pi)$, either $\mathrm{nf}(t) = 0$ or there exists $t' \prec t$ such that $\mathrm{nf}(t') = \mathrm{nf}(t)$.*

We define $K(a,b) \stackrel{\text{def}}{=} \sum_{i=0}^{b}(2^a-1)^i$ and observe that $K(a,b) \leq (2^a-1+1)^b = 2^{a\cdot b}$. Intuitively, $K(a,b)$ represents the number of syntactically distinct products of length at most b, built from non-empty sums of a distinct variables. For a set of terms $U \subseteq \mathsf{T}$, we write $\|U\|_{\mathrm{nf}} \stackrel{\text{def}}{=} \|\mathrm{nf}(t) \mid t \in U,\ \mathrm{nf}(t) \neq 0\|$. Lemma 29 below provides the bounds on the cardinalities on the sets of linear products over both bounded and periodic variables. This lemma simply applies the bounds on lengths of linear products from Lemmas 19, 23 and 28 to obtain bounds on the number of pairwise distinct normal forms of the linear products thereof.

▶ **Lemma 29.**
(i) $\|\mathsf{Prods}(\mathsf{Sums}(\mathcal{S}^\beta))\|_{\mathrm{nf}} \leq K(\|\mathcal{S}^\beta\|, \|\mathcal{S}^\beta\|_1)$,
(ii) $\|\mathsf{Prods}(\text{1-}\mathsf{Sums}(\mathcal{S}^\beta \uplus \mathcal{S}^\pi))\|_{\mathrm{nf}} \leq K(\|\mathcal{S}^\beta \uplus \mathcal{S}^\pi\|, \|\mathcal{S}^\beta \uplus \mathcal{S}^\pi\|_1)$,
(iii) $\|\mathsf{Prods}(\mathsf{Sums}(\mathcal{S}^\beta \uplus \mathcal{S}^\pi))\|_{\mathrm{nf}} \leq K(\|\mathcal{S}^\beta \uplus \mathcal{S}^\pi\|, b(\mathcal{S}^\beta, \mathcal{S}^\pi))$.

4.2 The Main Result

The next lemma gives first an exponential bound on the cardinality of a recognizer built from an alternative grammar in normal form. We distinguish the cases of (i) grammars with no periodic variables, (ii) with periodic variables of period 1, and (iii) the general case.

▶ **Lemma 30.** *The recognizer $\mathcal{A}$ built from an alternative grammar $\Gamma \stackrel{\text{def}}{=} (\mathcal{P} \uplus \mathcal{S}, \mathcal{R}, \mathcal{X})$ in normal form satisfies the following constraints:*

(1) $$\|\mathsf{A}^P\| \leq \begin{cases} 2^{B_{max}\cdot\|\mathcal{S}\|^2\cdot\|\mathcal{P}\|} & \text{if } \mathcal{R}^{\mathrm{A}} = \emptyset \\ 2^{2\cdot B_{max}\cdot\|\mathcal{S}\|^2\cdot\|\mathcal{P}\|} & \text{if } \mathcal{R}^{\mathrm{A}} \neq \emptyset,\ P_{max} = 1 \\ 2^{(B_{max}+P_{max}^2/2)\cdot\|\mathcal{S}\|^2\cdot(\|\mathcal{S}\|+2)\cdot\|\mathcal{P}\|} & \text{if } \mathcal{R}^{\mathrm{A}} \neq \emptyset,\ P_{max} > 1 \end{cases}$$

(2) $\|\mathsf{A}^S\| \leq 2^{\|\mathcal{S}\|\cdot(\|\mathcal{S}\|+\|\mathcal{P}\|+1)}$
where $B_{max} \stackrel{\text{def}}{=} \max_{p\in\mathcal{P}, s\in\mathcal{S}_p^\beta} \beta_p(s)$ *and* $P_{max} \stackrel{\text{def}}{=} \max_{p\in\mathcal{P}, s\in\mathcal{S}_p^\pi} \pi_p(s)$.

We denote by $\theta(\Gamma)$ the maximum size of a term that occurs as the right-hand size of a rule in Γ. The result of this section is that the cardinality of a recognizer built from a regular grammar Γ is bounded by an exponential in the number of nonterminals, rules and $\theta(\Gamma)$:

▶ **Theorem 31.** *The recognizer $\mathcal{A}$ built from the regular grammar $\Gamma = (\mathcal{P} \uplus \mathcal{S}, \mathcal{R}, \mathcal{X})$ has $2^{\mathcal{O}(\|\mathcal{P}\|\cdot\|\mathcal{S}\|^3\cdot\|\mathcal{R}\|^3\cdot\theta(\Gamma)^2)}$ elements.*

Proof. First, the regular grammar Γ is transformed to an equivalent regular grammar $\Gamma' \stackrel{\text{def}}{=} (\mathcal{P} \uplus \mathcal{S}', \mathcal{R}', \mathcal{X})$ in normal form, such that $\|\mathcal{S}'\| = \mathcal{O}(\|\mathcal{S}\| \cdot \|\mathcal{R}\|)$, by Proposition 7. Second, the regular grammar in normal form Γ' is transformed to an alternative grammar $\Gamma'' \stackrel{\text{def}}{=} (\mathcal{P} \uplus \mathcal{S}'', \mathcal{R}'', \mathcal{X})$ in normal form as explained in section 3.3. The set of non-terminals $\mathcal{S}''$ is increased by the elimination of the rules $\mathcal{R}^{\mathrm{E}}$. This increase is bounded by the cardinality of $\mathcal{R}^{\mathrm{E}}$, i.e., $\|\mathcal{S}''\| \leq \|\mathcal{S}'\| + \|\mathcal{R}\|$. Also, the transformation guarantees that the values of maximal periods and/or bases are the same in Γ' and Γ''. The recognizer $\mathcal{A}$ is built from Γ''. By Lemma 30 applied to Γ'' we obtain:

$$\|\mathsf{A}^P\| \leq u^P \stackrel{\text{def}}{=} 2^{(B_{max}+P_{max}^2/2)\cdot\|\mathcal{S}''\|^2\cdot(\|\mathcal{S}''\|+2)\cdot\|\mathcal{P}\|}$$
$$\|\mathsf{A}^S\| \leq u^S \stackrel{\text{def}}{=} 2^{\|\mathcal{S}''\|\cdot(\|\mathcal{S}''\|+\|\mathcal{P}\|)}$$

Then, $\|\mathsf{A}\| = \|\mathsf{A}^P\| + \|\mathsf{A}^S\| \leq u^P + u^S \leq 2 \cdot u^P$, since $u^S \leq u^P$ always holds. Using moreover that $B_{max} \leq \theta(\Gamma)$, $P_{max} \leq \theta(\Gamma)$ and $\|\mathcal{S}''\| = \mathcal{O}(\|\mathcal{S}\| \cdot \|\mathcal{R}\|)$ we obtain $\|\mathsf{A}\| = 2^{\mathcal{O}(\|\mathcal{P}\|\cdot\|\mathcal{S}\|^3\cdot\|\mathcal{R}\|^3\cdot\theta(\Gamma)^2)}$. ◀

We simplify the above bound to a more rough bound in the size of the grammar:

▶ **Corollary 32.** *The recognizer $\mathcal{A}$ built from the regular grammar Γ has $2^{\mathcal{O}(\mathrm{size}(\Gamma)^9)}$ elements.*

Proof. Because $\mathrm{size}(\Gamma)$ is the length of the representation of Γ as a string, we have $\mathrm{size}(\Gamma) = 2n + \sum_{i=1}^{n} \mathrm{size}(t_i)$, where the rules of Γ are $x_i \rightarrow t_i$, for $1 \leq i \leq n$. Moreover, by assuming that each nonterminal occurs as left-hand side in at least one rule, we obtain $\|\mathcal{P}\| + \|\mathcal{S}\| \leq n$. We compute:

$$\|\mathcal{P}\| \cdot \|\mathcal{S}\|^3 \cdot \|\mathcal{R}\|^3 \cdot \theta(\Gamma)^2 \leq \mathrm{size}(\Gamma)^9$$ ◀

As a byproduct, we obtain the following result:

▶ **Corollary 33.** *The accepting set of $\mathcal{L}(\Gamma)$ in $\mathcal{A}$ is computable in time $2^{\mathcal{O}(\|\mathcal{P}\| \cdot \|\mathcal{S}\|^3 \cdot \|\mathcal{R}\|^3 \cdot \theta(\Gamma)^2)}$.*

Proof. Let $F \subseteq \mathsf{A}$ such that $\mathcal{L}(\Gamma) = h^{-1}(F)$. The elements of $F \cap \mathsf{A}^P$ can be enumerated in time $\|\mathsf{A}\| \cdot \|\mathcal{R}\| \cdot \theta(\Gamma)^{\mathcal{O}(\|\mathcal{S}\|)}$), by Lemmas 17 and 9. The elements of $F \cap \mathsf{A}^S$ can be enumerated in time $\|\mathsf{A}\| \cdot \|\mathcal{S}\| \cdot (\|\mathcal{P}\| + \|\mathcal{S}\| + 1)$, because $\|\mathcal{S}\| \cdot (\|\mathcal{P}\| + \|\mathcal{S}\| + 1)$ is the maximum size of an S-profile. By Theorem 31, we obtain an upper bound of the order of $2^{\mathcal{O}(\theta(\Gamma)^2 \cdot \|\mathcal{P}\| \cdot \|\mathcal{S}\|^3 \cdot \|\mathcal{R}\|^3)}$, in both cases. ◀

5 Decision Problems

In this section, we consider standard decision problems for recognisable sets of SP-graphs, represented as regular grammars (Definition 3). We prove that the problems of (i) emptiness of the intersection of n sets given as regular grammars, and (ii) inclusions between a context-free and a regular set, are both EXPTIME-complete.

We denote by $\alpha(\Gamma)$ the maximum number of variables occurring on the right-hand side of a rule from the grammar Γ and recall that $\theta(\Gamma)$ denotes the maximum size of a right-hand side of Γ. A first ingredient is the well-known Filtering Theorem [11, Theorem 3.88]:

▶ **Theorem 34.** *For each grammar $\Gamma = (\mathcal{N}, \mathcal{R}, \mathcal{X})$ and set L of SP-graphs recognized by a finite algebra $\mathcal{A}$, one can build a grammar Γ' such that $\mathcal{L}(\Gamma') = \mathcal{L}(\Gamma) \cap L$. Moreover, Γ' can be built in time $\mathcal{O}(\|\mathcal{R}\| \cdot \|\mathsf{A}\|^{\alpha(\Gamma)} \cdot \tau \cdot \theta(\Gamma))$, where τ is an upper bound on the time needed to compute $f^{\mathcal{A}}(a_1, \ldots, a_n)$, for each $f \in \mathcal{F}_{\mathcal{SP}}$ and $a_1, \ldots, a_n \in \mathsf{A}$.*

Intuitively, the grammar Γ' is obtained from Γ by annotating each nonterminal $x \in \mathcal{N}$ with an element $a \in \mathsf{A}$ and introducing a rule $x^a \rightarrow t[x_1^{a_1}, \ldots, x_n^{a_n}]$ for each rule $x \rightarrow t[x_1, \ldots, x_n] \in \mathcal{R}$ and tuple $a, a_1, \ldots, a_n \in \mathsf{A}$ such that $a = t^{\mathcal{A}}(a_1, \ldots, a_n)$. The axioms of Γ' are $\{x^a \mid x \in \mathcal{X}, a \in h(L)\}$, where h is the homomorphism recognizing L (Definition 1). The upper bound on the time needed to compute Γ' follows immediately from its construction.

To apply the Filtering Theorem to recognizable sets defined by regular grammars, we use the previously computed upper bound on the cardinality of their recognizers (Theorem 31). An estimate on the time τ needed to compute the operations from $\mathcal{F}_{\mathcal{SP}}$ in the recognizer of a regular grammar is given below:

▶ **Lemma 35.** *Let $\Gamma = (\mathcal{P} \uplus \mathcal{S}, \mathcal{R}, \mathcal{X})$ be a regular grammar and $\mathcal{A}$ be a recognizer for $\mathcal{L}(\Gamma)$. Then, $f^{\mathcal{A}}(a_1, \ldots, a_n)$ can be computed in time $\tau = (\|\mathcal{P}\| \cdot \|\mathcal{R}\|)^{\mathcal{O}(1)} \cdot \theta(\Gamma)^{\mathcal{O}(\|\mathcal{S}\|)}$, for each $f \in \mathcal{F}_{\mathcal{SP}}$ and $a_1, \ldots, a_n \in \mathsf{A}$.*

We establish the complexity of two decision problems concerning recognizable sets, described by regular grammars. This tightens an existing 2EXPTIME upper bound, for the problem of inclusion between a context-free and a regular language [6, Theorem 4]. The

EXPTIME upper bound follows from the fact that the language of a regular grammars is recognized by an algebra of cardinality exponentially bounded in the size of the grammar (Theorems 16 and 31). The EXPTIME-hard lower bound for the problems considered in Theorem 36 is established by polynomial many-one reductions for the problems of intersection and inclusion of tree automata [8, Theorems 1.7.5 and 1.7.7].

▶ **Theorem 36.** *The following problems are* EXPTIME*-complete:*
1. *Given regular grammars* $\Gamma_1, \ldots, \Gamma_n$*, does* $\mathcal{L}(\Gamma_1) \cap \ldots \cap \mathcal{L}(\Gamma_n) = \emptyset$ *hold?*
2. *Given grammars* Γ_1 *and* Γ_2 *such that* Γ_2 *is regular, does* $\mathcal{L}(\Gamma_1) \subseteq \mathcal{L}(\Gamma_2)$ *hold?*

Proof. We prove the EXPTIME upper bound for each of the considered problems:

(1) Let $\Gamma_1, \ldots, \Gamma_n$ be an instance of the intersection problem, where $\Gamma_i = (\mathcal{P}_i \uplus \mathcal{S}_i, \mathcal{R}_i, \mathcal{X}_i)$, for each $i \in [1, n]$. We denote $P \stackrel{\text{def}}{=} \sum_{i=1}^{n} \|\mathcal{P}_i\|$, $S \stackrel{\text{def}}{=} \sum_{i=1}^{n} \|\mathcal{S}_i\|$, $R \stackrel{\text{def}}{=} \sum_{i=1}^{n} \|\mathcal{R}_i\|$ and $\Theta \stackrel{\text{def}}{=} \sum_{i=1}^{n} \theta(\Gamma_i)$. The size of the input is then $\sum_{i=1}^{n} \mathrm{size}(\Gamma_i) \leq N \stackrel{\text{def}}{=} P + S + R \cdot \Theta$. Let $\mathcal{A}_i$ be a recognizer for $\mathcal{L}(\Gamma_i)$, for $i \in [1, n]$. By Proposition 2, the set $\mathcal{L}(\Gamma_1) \cap \ldots \cap \mathcal{L}(\Gamma_n)$ is recognized by an algebra $\mathcal{A}$ having the domain $\mathsf{A} = \mathsf{A}_1 \times \ldots \times \mathsf{A}_n$. By Theorem 31, we obtain $\|\mathsf{A}\| = 2^{\mathcal{O}(P \cdot S^3 \cdot R^3 \cdot \Theta^2)}$. By Corollary 33, the accepting set for the language $\mathcal{L}(\Gamma_1) \cap \ldots \cap \mathcal{L}(\Gamma_n)$ can be computed in time that is of the same order of magnitude as $\|\mathsf{A}\|$. Since $P \cdot S^3 \cdot R^3 \cdot \Theta \leq N^7$, the emptiness of the intersection can be decided in time $2^{\mathcal{O}(N^7)}$.

(2) Let $\Gamma_i = (\mathcal{P}_i \uplus \mathcal{S}_i, \mathcal{R}_i, \mathcal{X}_i)$, for $i = 1, 2$, be an instance of the inclusion problem. Let $\mathcal{A}_2$ be a recognizer for $\mathcal{L}(\Gamma_2)$. The size of the input is then $N \stackrel{\text{def}}{=} \sum_{i=1}^{2} \mathrm{size}(\Gamma_i)$. By Proposition 2, the set $\mathsf{SP} \setminus \mathcal{L}(\Gamma_2)$ is also recognized by $\mathcal{A}_2$. Let Γ_1' be the result of applying the Filtering Theorem 34 to Γ_1 and $\mathsf{SP} \setminus \mathcal{L}(\Gamma_2)$, i.e., $\mathcal{L}(\Gamma_1') = \mathcal{L}(\Gamma_1) \cap (\mathsf{SP} \setminus \mathcal{L}(\Gamma_2))$. By Theorem 34 and Lemma 35, Γ_1' can be built in time:

$$
\begin{aligned}
& \|\mathcal{R}_1\| \cdot \|\mathsf{A}_2\|^{\alpha(\Gamma_1)} \cdot \underbrace{(\|\mathcal{P}_2\| \cdot \|\mathcal{R}_2\|)^{\mathcal{O}(1)} \cdot \theta(\Gamma_2)^{\mathcal{O}(\mathcal{S}_2)}}_{\tau} \cdot \theta(\Gamma_1) \\
= {} & \|\mathcal{R}_1\| \cdot \underbrace{2^{\alpha(\Gamma_1) \cdot \mathcal{O}(\|\mathcal{P}_2\| \cdot \|\mathcal{S}_2\|^3 \cdot \|\mathcal{R}_2\|^3 \cdot \theta(\Gamma_2)^2)}}_{\|\mathsf{A}_2\|^{\alpha(\Gamma_1)}} \cdot \\
& \underbrace{(\|\mathcal{P}_2\| \cdot \|\mathcal{R}_2\|)^{\mathcal{O}(1)} \cdot \theta(\Gamma_2)^{\mathcal{O}(\mathcal{S}_2)}}_{\tau} \cdot \theta(\Gamma_1) \\
= {} & 2^{(\|\mathcal{R}_1\| \cdot \alpha(\Gamma_1) \cdot \|\mathcal{P}_2\| \cdot \|\mathcal{S}_2\| \cdot \|\mathcal{R}_2\| \cdot \theta(\Gamma_2))^{\mathcal{O}(1)}} = 2^{N^{\mathcal{O}(1)}}
\end{aligned}
$$

Since the emptiness of $\mathcal{L}(\Gamma_1')$ can be decided in time $\mathcal{O}(\mathrm{size}(\Gamma_1')^2)$, the inclusion problem has an EXPTIME upper bound. The EXPTIME-hard lower bound follows from the EXPTIME-hardness of the same problems for sets of terms represented by finite tree automata [8, Theorems 1.7.5 and 1.7.7]. The encoding of terms as SP-graphs and of bottom-up tree automata as regular grammars are detailed in the proof of [6, Theorem 4]. ◀

6 Conclusions

We prove an exponential bound on the cardinality of the minimal algebra recognizing the language of regular grammar, for series-parallel graphs. Based on this result, we establish the EXPTIME-completeness of the intersection and inclusion problems.

References

1 Rajeev Alur and P. Madhusudan. Adding nesting structure to words. *J. ACM*, 56(3), May 2009. doi:10.1145/1516512.1516518.

2 Rajeev Alur, Caleb Stanford, and Christopher Watson. A robust theory of series parallel graphs. *Proc. ACM Program. Lang.*, 7(POPL), January 2023. doi:10.1145/3571230.

3 Adrien Boiret, Vincent Hugot, Joachim Niehren, and Ralf Treinen. Automata for unordered trees. *Inf. Comput.*, 253:304–335, 2017. doi:10.1016/j.ic.2016.07.012.

4 Mikolaj Bojanczyk and Igor Walukiewicz. Forest algebras. In Jörg Flum, Erich Grädel, and Thomas Wilke, editors, *Logic and Automata: History and Perspectives [in Honor of Wolfgang Thomas]*, volume 2 of *Texts in Logic and Games*, pages 107–132. Amsterdam University Press, 2008.

5 Marius Bozga, Radu Iosif, and Florian Zuleger. Regular grammars for graph sets of tree-width 2. *CoRR*, abs/2408.01226, 2024. doi:10.48550/arXiv.2408.01226.

6 Marius Bozga, Radu Iosif, and Florian Zuleger. Regular grammars for sets of graphs of tree-width 2. In *2025 40th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 704–717, 2025. doi:10.1109/LICS65433.2025.00059.

7 Mark Chimes, Radu Iosif, and Florian Zuleger. Tree-verifiable graph grammars. In Nikolaj S. Bjørner, Marijn Heule, and Andrei Voronkov, editors, *LPAR 2024: Proceedings of 25th Conference on Logic for Programming, Artificial Intelligence and Reasoning, Port Louis, Mauritius, May 26-31, 2024*, volume 100 of *EPiC Series in Computing*, pages 165–180. EasyChair, 2024. doi:10.29007/8113.

8 Hubert Comon, Max Dauchet, Rémi Gilleron, Florent Jacquemard, Denis Lugiez, Christof Löding, Sophie Tison, and Marc Tommasi. *Tree Automata Techniques and Applications*. HAL, 2008. URL: https://inria.hal.science/hal-03367725.

9 Bruno Courcelle. The monadic second-order logic of graphs. i. recognizable sets of finite graphs. *Inf. Comput.*, 85(1):12–75, 1990. doi:10.1016/0890-5401(90)90043-H.

10 Bruno Courcelle. The monadic second-order logic of graphs V: on closing the gap between definability and recognizability. *Theor. Comput. Sci.*, 80(2):153–202, 1991. doi:10.1016/0304-3975(91)90387-H.

11 Bruno Courcelle and Joost Engelfriet. *Graph Structure and Monadic Second-Order Logic: A Language-Theoretic Approach*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2012. doi:10.1017/CBO9780511977619.

12 Amina Doumane. Regular expressions for tree-width 2 graphs. In Mikolaj Bojanczyk, Emanuela Merelli, and David P. Woodruff, editors, *49th International Colloquium on Automata, Languages, and Programming, ICALP 2022, July 4-8, 2022, Paris, France*, volume 229 of *LIPIcs*, pages 121:1–121:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPIcs.ICALP.2022.121.

13 Jacob Elgaard, Nils Klarlund, and Anders Møller. MONA 1.x: new techniques for WS1S and WS2S. In *Proc. 10th International Conference on Computer-Aided Verification, CAV '98*, volume 1427 of *LNCS*, pages 516–520. Springer-Verlag, june/july 1998. doi:10.1007/BFb0028773.

14 Joost Engelfriet. *Context-Free Graph Grammars*, pages 125–213. Springer Berlin Heidelberg, Berlin, Heidelberg, 1997. doi:10.1007/978-3-642-59126-6_3.

15 Jan Kurkofka and Tim Planken. A tutte-type canonical decomposition of 3- and 4-connected graphs. In Kasper Green Larsen and Barna Saha, editors, *Proceedings of the 2026 Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2026, Vancouver, BC, Canada, January 11-14, 2026*, pages 2942–3021. SIAM, 2026. doi:10.1137/1.9781611978971.110.

16 Ondrej Lengál, Jirí Simácek, and Tomás Vojnar. VATA: A library for efficient manipulation of non-deterministic tree automata. In Cormac Flanagan and Barbara König, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 18th International Conference, TACAS 2012, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2012, Tallinn, Estonia, March 24 - April 1, 2012. Proceedings*, volume 7214 of *Lecture Notes in Computer Science*, pages 79–94, Berlin, Heidelberg, 2012. Springer. doi:10.1007/978-3-642-28756-5_7.

17 Jean-Eric Pin. Mathematical foundations of automata theory. `https://www.irif.fr/~jep/PDF/MPRI/MPRI.pdf`, 2025.

18 E. Rémila. *An Introduction to Automata on Graphs*, pages 345–352. Springer Netherlands, Dordrecht, 1999. `doi:10.1007/978-94-015-9153-9_15`.

19 James W. Thatcher. Characterizing derivation trees of context-free grammars through a generalization of finite automata theory. *J. Comput. Syst. Sci.*, 1(4):317–322, 1967. `doi:10.1016/S0022-0000(67)80022-9`.

20 Wolfgang Thomas. Finite-state recognizability and logic: from words to graphs. In Björn Pehrson and Imre Simon, editors, *Technology and Foundations - Information Processing '94, Volume 1, Proceedings of the IFIP 13th World Computer Congress, Hamburg, Germany, 28 August - 2 September, 1994*, volume A-51 of *IFIP Transactions*, pages 499–506. North-Holland, 1994.

21 William T. Tutte. *Graph theory as I have known it*, volume 11 of *Oxford lecture series in mathematics and its applications*. Clarendon Press, 1998.

22 Hassler Whitney. On the abstract properties of linear dependence. *American Journal of Mathematics*, 57(3):509–533, 1935. URL: `http://www.jstor.org/stable/2371182`.