

SLP compression for solutions of equations with constraints in free and hyperbolic groups

Volker Diekert*, Olga Kharlampovich[†]
and Atefeh Mohajeri Moghaddam[‡]

August 19th, 2013

Abstract

The paper is a part of an ongoing program which aims to show that the existential theory in free groups (hyperbolic groups or even toral relatively hyperbolic) is NP-complete. For that we study compression of solutions with straight-line programs (SLPs) as suggested originally by Plandowski and Rytter in the context of a single word equation. We review some basic results on SLPs and give full proofs in order to keep this fundamental part of the program self-contained. Next we study systems of equations with constraints in free groups and more generally in free products of abelian groups. We show how to compress minimal solutions with extended Parikh-constraints. This type of constraints allows to express semi linear conditions as e.g. alphabetic information. The result relies on some combinatorial analysis and has not been shown elsewhere. We show similar compression results for Boolean formula of equations over a torsion-free δ -hyperbolic group. The situation is much more delicate than in free groups. As byproduct we improve the estimation of the “capacity” constant used by Rips and Sela in their paper “Canonical representatives and equations in hyperbolic groups” from a double-exponential bound in δ to some single-exponential bound. The final section shows compression results for toral relatively hyperbolic group using the work of Dahmani: We show that given a system of equations over a fixed toral relatively hyperbolic group, for every solution of length N there is an SLP for another solution such that the size of the SLP is bounded by some polynomial $p(s + \log N)$ where s is the size of the system.

Introduction

This work is motivated by the conjecture that the problem of satisfiability of a system of equations in a free group or free semigroup is NP-complete. There is a polynomial-time reduction from satisfiability in free groups to satisfiability in free semigroups; and it is also known that this problem is NP-hard for free groups

*Universität Stuttgart, FMI, Universitätsstraße 38, 70569 Stuttgart, Germany
diekert@fmi.uni-stuttgart.de

[†]Dept. Math and Stats, Hunter College and Graduate Center, CUNY, 695 Park Ave, New York, NY, USA, 10065 okharlampovich@gmail.com

[‡]Dept. Math and Stats, McGill University, Montreal, Canada, H3A 0B9
mohajeri@math.mcgill.ca

(even in the special case of quadratic equations, [13]). So one should prove that it is in NP. The roadmap how to prove this was suggested by Plandowski and Rytter in [22]. The idea is to prove that the length of a minimal solution is bounded by a single-exponential function $2^{p(s)}$, where $p(s)$ is a polynomial in the size s of the system of equations. Once this bound is established an NP-algorithm can guess a compressed version of the solution. An additional deterministic polynomial time algorithm can verify that the guess is indeed a solution. The result of [22, Thm. 3] is as follows. Assume that the length of a minimal solution of a word equation of length s is bounded by some function $f(s)$. Then for a word equation of length s and $f(s)$ written in binary, the satisfiability of the equation can be decided in non-deterministic polynomial time. This result was shown via Lempel-Ziv encodings of minimal solutions [22, Thm. 2], but it has been apparent that the result holds also for encodings via straight-line programs (SLPs) and for systems of equations. Moreover it extends to Boolean formulae of equations in free groups and free semigroups, as shown in [10]. Actually, a more general result was established concerning systems of equations “with rational constraints”. Rational constraints are given by regular languages (specified by NFAs, i.e., by non-deterministic finite automata) which, algebraically (by the transformation monoids of the NFAs), can be reinterpreted by conditions on images in finite monoids. This approach dates back to the work of Schulz [25] and is also explained in details e.g. in [19, Ch. 12] or [7]. Hence, for systems of equations with rational constraints the sizes of finite monoids become crucial. If the sizes of these monoids are at most polynomial size with respect to the input size of equations then [22, Thm. 2] and [22, Thm. 3] are true in this more general setting by [10, Chapter 8]. Monoids of polynomial size suffice to treat inequalities as constraints, but e.g. did not allow to treat alphabetic constraints. And indeed, allowing arbitrary rational constraints in the system changes the picture drastically: A similar result about the existence of SLPs of size $p(s + \log f(s))$ cannot hold unless $\text{NP} = \text{PSPACE}$ because the “empty-intersection-problem” for regular languages is a special case which is known to be PSPACE-complete due to a classical result of Kozen [15].

In this paper we continue the research in two directions. First, we deal with extended Parikh-constraints. This is slightly more general than adding semi-linear constraints and strictly more general than alphabetic constraints, i.e., prescribing the set of letters occurring in a solution. In the setting of extended Parikh-constraints it is very natural to extend the results to finitely generated free products of abelian groups. We show that for every solution of length N there is an SLP for another solution with the same extended Parikh-image and the same length N such that the size of the SLP is logarithmic in N (if N is at least exponential in the size of the equation).

Based on the results in the first part we show in a second part that, given a Boolean formula Φ of equations over a δ -hyperbolic group with generating set Σ , for every solution of length N there is an SLP for another solution such that the size of the SLP is bounded by a polynomial in $\kappa + \|\Phi\| + \log N$, where $\|\Phi\|$ is the size of the formula and κ depends exponentially on δ and $\log |\Sigma|$ (at most double-exponentially), see Corollary 5.5. In the final part of the paper we consider systems of equations over toral relatively hyperbolic groups, and we obtain similar results with κ depending exponentially on parameters of the group.

1 Preliminaries

1.1 Words and monoids with involution

All monoids (in particular all groups) in this paper are assumed to be finitely generated. By Σ (resp. Γ^*) we denote a finite alphabet and Σ^* (resp. Γ^*) is the corresponding free monoid. (Typically, Σ is a generating set of a group G and $\Gamma = \Sigma \cup \Sigma^{-1}$.) Elements of free monoids are called *words*. A word in Σ^* can be written as $w = a_1 \cdots a_n$ with $n \geq 0$ and $a_i \in \Sigma$, where $n = |w|$ is its *length*. For $a \in \Sigma$ the a -length of w is denoted by $|w|_a$. It counts the number of a 's occurring in w . We let $\text{alph}(w) = \{a_1, \dots, a_n\}$ be the *alphabet* of w ; it is the set of letters occurring in the word. The word of length 0 is called the *empty word*; it is denoted by 1, since it is the neutral element of Σ^* . We have $\text{alph}(1) = \emptyset$.

A *factor* of a word w is a word v such that $w = w_1 v w_2$. A factor v is a *prefix* (resp. *suffix*), if we can write $w = v w_2$ (resp. $w = w_1 v$). For $0 \leq \alpha \leq \beta \leq |w|$ and $w = a_1 \cdots a_n$ we define the factor $w[\alpha, \beta]$ by

$$w[\alpha, \beta] = a_{\alpha+1} \cdots a_\beta.$$

Note that $|w[\alpha, \beta]| = \beta - \alpha$. Moreover, prefixes can be written as $w[0, \beta]$ and suffixes as $w[\alpha, |w|]$.

An *involution* on a set is a bijection $\bar{}$ such that $\overline{\overline{x}} = x$ for all elements x . If M is a monoid, then an involution $\bar{} : M \rightarrow M$ must also satisfy $\overline{xy} = \bar{y}\bar{x}$. If $1 \in M$ is the neutral element, then $\bar{1} = 1$ since neutral elements are unique in monoids. A *morphism* between monoids with involution is a homomorphism $h : M \rightarrow M'$ such that $h(\overline{x}) = \overline{h(x)}$. A group G is viewed as a monoid with involution by letting $\bar{g} = g^{-1}$ for $g \in G$. Homomorphisms between groups are morphisms of monoids with involution. In groups we do not distinguish between \bar{g} and g^{-1} .

If a group G is generated by Σ , then we may define $\Gamma = \Sigma \cup \overline{\Sigma}$, where $\overline{\Sigma} = \{\bar{a} \mid a \in \Sigma\}$ is a disjoint copy of Σ . We let $\bar{a} = a$. This defines an involution $\bar{} : \Gamma \rightarrow \Gamma$; and the involution is extended to Γ^* by $\overline{a_1 \cdots a_n} = \bar{a}_n \cdots \bar{a}_1$. Thus, Γ^* is a monoid with involution, and every mapping from Σ to another monoid M with involution extends uniquely to a morphism $\psi : \Gamma^* \rightarrow M$. Hence, Γ^* is the free monoid with involution over Σ . Every group element in G can be represented as a word over Γ . There is a canonical morphism of Γ^* onto the free group $F(\Sigma)$ over Σ . Moreover, as a set, we identify $F(\Sigma)$ with the set of *reduced words*. These are the words $w \in \Gamma^*$ without any factor $a\bar{a}$ where $a \in \Gamma$. Reduced words are unique *geodesic* normal forms for elements in $F(\Sigma)$. For $w \in \Gamma^*$ we let \hat{w} denote the reduced word such that $w = \hat{w} \in F(\Sigma)$.

1.2 Straight-line programs

By Ω we denote a finite set of variables, which is endowed with an involution $X \mapsto \overline{X}$ without fixed points. Hence we can write Ω as a disjoint union $\Omega = \Omega_+ \cup \{\overline{X} \mid X \in \Omega_+\}$. Variables occur in the context of equations and in the context of straight-line programs. For the use in straight-line programs we need to specify a partial order $<$ on them.

Straight-line programs are widely used, frequently in the context of algebraic circuits. In this paper a straight-line program is a special case of a *straight-line*

grammar which in turn is, by definition, a reduced context-free grammar which produces exactly one word. If a grammar generates only one word, then the grammar encodes the generated word. In some cases the size of the generated word can be exponentially longer than the size of the grammar; and therefore straight-line grammars can be used for data compression.

Example 1.1 Let $n \in \mathbb{N}$.

1. Consider the following grammar with axiom A_0 and rules $A_{i-1} \rightarrow A_i A_i$ for $1 \leq i \leq n$ and a single terminal rule $A_n \rightarrow a$. The grammar has linear size in n , but the axiom generates the word a^{2^n} of length 2^n .
2. (Fibonacci words) There are terminal rules $F_1 \rightarrow b$ and $F_2 \rightarrow a$, and for $n \geq 3$ there are rules $F_n \rightarrow F_{n-1} F_{n-2}$. Then each F_n generates a word $F(n)$ with length being the n -th Fibonacci number. Moreover, for $n \geq 3$ the word $F(n-1)$ is a prefix of $F(n)$, hence one can define an infinite sequence of letters where all $F(n)$ appear as prefixes.

The following example had direct impact to algorithmic group theory. The example is due to Saul Schleimer. He used it to show that the word problem for the group $\text{Aut}(F)$ of automorphisms of a free groups is decidable in polynomial time. We will come back to this later. As a matter of fact it is more convenient to consider the Schleimer's example in the setting of monoids.

Example 1.2 (Saul Schleimer) Let M be a monoid generated by Σ and A be a finite set of endomorphism of M ; e.g., M is the free group $F = F(\Sigma)$ and A any finite generating set for $\text{Aut}(F)$. Let $w = \alpha_1 \cdots \alpha_n \in A^*$ with $\alpha_i \in A$ and $a \in \Sigma$. Then the pair (w, a) defines an SLP of size $\mathcal{O}(n)$ which evaluates to $\alpha_1 \cdots \alpha_n(a)$ as a monoid element in M as follows. Variables of the SLP are denoted by $A[i, a] = A[\alpha_1 \cdots \alpha_i, a]$ for $0 \leq i \leq n$ and $a \in \Sigma$. Thus, there are exactly $|\Sigma| \cdot (n+1)$ variables. In order to define the rules consider first $i \geq 1$. If $\alpha_i(a) = b_1 \cdots b_k$ with $b_j \in \Sigma$, then we define the production

$$A[i, a] \rightarrow A[i-1, b_1] \cdots A[i-1, b_k].$$

Finally, we define terminal rules

$$A[0, a] \rightarrow a.$$

It is clear that every variable of this grammar produces exactly one word. The variable $A[n, a]$ produces a word which yields $w(a) \in M$ with the interpretation that w denotes an endomorphism of M and $a \in M$.

A straight-line program is essentially a straight-line grammar in Chomsky normal form. Formally, a *straight-line program* (SLP for short) is a set S of rules which have either form:

$$\begin{aligned} X &\rightarrow a, \\ X &\rightarrow YZ \text{ where } X < Y, X < \bar{Y}, X < Z, \text{ and } X < \bar{Z} \end{aligned}$$

Here $X \in \Omega_+$, $Y, Z \in \Omega$, and $a \in \Gamma \cup \{1\}$. Moreover, we demand that each $X \in \Omega_+$ appears exactly once on a left-hand side.

We define the *height* $h(X)$ and *evaluation* $\text{eval}(X)$ for $X \in \Omega$ inductively.

- If $X \rightarrow a$ is a rule, then $h(X) = 1$ and $\text{eval}(X) = a$.
- If $h(X)$ and $\text{eval}(X)$ are defined, then $h(\overline{X}) = h(X)$ and $\text{eval}(\overline{X}) = \text{eval}(X)$.
- If $X \rightarrow YZ$ is a rule, then $h(X) = 1 + \max\{h(Y), h(Z)\}$ and $\text{eval}(X) = \text{eval}(Y) \text{eval}(Z)$.

Example 1.3 Let M be a commutative monoid generated by Σ . Then for each word $w \in \Sigma^*$ of length n there exists an SLP with $\mathcal{O}(|\Sigma| \cdot \log n)$ variables and axiom X such that $\text{eval}(X) = w$ in G . Indeed, every w can be written in M as a product $\gamma_1^{n_1} \cdots \gamma_r^{n_r}$ with $n_i \in \mathbb{N}$ and $r = |\Sigma|$.

1.3 Interval grammars

Interval grammars have been introduced in the thesis of Hagenah [10]. They compress words in a very similar fashion as Lempel-Ziv compression. The notion of interval grammar is also very closely related to the notion of *composition system* as defined by Gasieniec, Karpinski, Plandowski, and Rytter in [8] as well as to the data structure used by Mehlhorn, Sundar, and Uhrig[20]. An SLP is a special case of a composition system, and a composition system in turn is a special case of an interval grammar. Hagenah has shown how to transform an interval grammar into an equivalent SLP with a quadratic blow-up in size, see Theorem 2.1. Thus, all three formalisms can be viewed as equivalent. As interval grammars provide a rather flexible formalism which is very intuitive, we use them here for compression.

An *interval grammar* (IG for short) is a set S of rules which have either form:

$$\begin{aligned} X &\rightarrow a, \\ X &\rightarrow Y[\alpha, \beta], \\ X &\rightarrow Y[\alpha, \beta]Z[\gamma, \delta] \end{aligned}$$

Here $X \in \Omega_+$, $Y, Z \in \Omega$, $\alpha, \beta, \gamma, \delta \in \mathbb{N}$, and $a \in \Gamma \cup \{1\}$. The other restrictions are listed below. The main idea is that if a variable X evaluates to the word w , then $X[\alpha, \beta]$ evaluates to the factor $w[\alpha, \beta]$.

In order to avoid case distinction we treat a rule $X \rightarrow Y[\alpha, \beta]$ as special case of $X \rightarrow Y[0, 0]Y[\alpha, \beta]$ whenever convenient. There are several restrictions on the rules: As for SLPs, each $X \in \Omega_+$ occurs in exactly one rule of the left hand side, and in all rules $X \rightarrow Y[\alpha, \beta]Z[\gamma, \delta]$ we must have $X < Y$, $X < \overline{Y}$, $X < Z$, and $X < \overline{Z}$. Next, we define the length $|X|$ of a variable X and the restrictions on $\alpha, \beta, \gamma, \delta$ simultaneously. If there is a rule $X \rightarrow a$, then we let $|X| = |\overline{X}| = |a| \in \{0, 1\}$. If there is a rule $X \rightarrow Y[\alpha, \beta]Z[\gamma, \delta]$, then $|X| = |\overline{X}| = \beta - \alpha + \delta - \gamma$ and we must have $0 \leq \alpha \leq \beta \leq |Y|$ and $0 \leq \gamma \leq \delta \leq |Z|$. In the following we assume that every interval grammar satisfies these restrictions.

For $w \in \Gamma^*$ we let $|w|$, $h(w) = 0$, $\text{eval}(w) = w$, and $w[\alpha, \beta]$ as above. Now, we define for $X \in \Omega \cup \Gamma$ and $0 \leq \alpha \leq \beta \leq |X|$ the terms $|X|$, $h(X)$, $\text{eval}(X)$, and $\text{eval}([\alpha, \beta])$. The general rule is $h(\overline{X}) = h(X)$, $\text{eval}(\overline{X}) = \text{eval}(X)$, and $\text{eval}(X[\alpha, \beta]) = \text{eval}(X)[\alpha, \beta]$. Moreover, $|X| = |\text{eval}(X)|$ and $|X[\alpha, \beta]| = \beta - \alpha$. Thus it is enough to define the *height* $h(X)$ and *evaluation* $\text{eval}(X)$ for $X \in \Omega_+$.

- If $X \rightarrow a$ is a rule, then $h(X) = 1$ and $\text{eval}(X) = a$.
- If $X \rightarrow Y[\alpha, \beta]Z[\gamma, \delta]$ is a rule, then $h(X) = 1 + \max\{h(Y), h(Z)\}$ and $\text{eval}(X) = \text{eval}(Y)[\alpha, \beta] \text{eval}(Z)[\delta, \gamma]$.

For $\mu = |X|$ and $X[\alpha, \beta]$ and we also define $\overline{X[\alpha, \beta]} = \overline{X}[\mu - \beta, \mu - \alpha]$. In the following it is convenient to think that for all rules $X \rightarrow Y[\alpha, \beta]Z[\gamma, \delta]$ and $X \rightarrow a$, we may also use rules $\overline{X} \rightarrow \overline{Z}[\gamma, \delta] \overline{Y}[\alpha, \beta]$ and $\overline{X} \rightarrow \overline{a}$, although our formalism does not list them explicitly.

The next proposition is used throughout in the paper. Its proof straightforward and therefore omitted.

Proposition 1.4 *The following computation can be performed in polynomial time.*

- *Input: An interval grammar S and a list of words w_1, \dots, w_m .*
- *Output for each $X \in \Omega$:*
 1. *The height $h(X)$ and the length $|X|$.*
 2. *For each w_i the answer whether w_i appears as a factor in $\text{eval}(X)$.*

2 Some polynomial time algorithms

In this section we review some polynomial time results for certain problems involving SLPs and interval grammars. We survey some known results and we give full proofs.

Theorem 2.1 ([10], **Algorithm 8.1.4**) *Let S be an interval grammar, then we can construct in polynomial time an SLP S' containing variables $X_{\alpha\beta}$ for all $X[\alpha, \beta]$ which appear in S such that $\text{eval}(X_{\alpha\beta}) = \text{eval}(X[\alpha, \beta])$. Moreover, we have $\|S'\| \in \mathcal{O}(|\Omega|^2)$.*

Proof. In order to reduce the number of case distinctions we assume that there is an ε -rule $E \rightarrow 1$. (If not, we add such a rule.) Therefore we do not treat chain rules, because a rule $X \rightarrow Y[\alpha, \beta]$ can always be written as $X \rightarrow Y[\alpha, \beta]E$. (Chain rules and ε -rule are eliminated in a final round.) Moreover, inside this proof it is convenient to assume that an interval grammar contains a rule $X \rightarrow YZ$ if and only if it contains the dual rule $\overline{X} \rightarrow \overline{Z}\overline{Y}$.

For every symbol $X[\alpha, \beta]$ which occurs in S we define its *weight* $H(X[\alpha, \beta])$ by its height $H(X[\alpha, \beta]) = h(X)$ if $\alpha = 0$ and twice its height $H(X[\alpha, \beta]) = 2h(X)$ otherwise. The weight of S is the sum of all weights. It is therefore in $\mathcal{O}(h(S) \|\Omega\|) \subseteq \mathcal{O}(|\Omega|^2)$.

We now describe a weight-reducing procedure which eliminates all symbols $X[\alpha, \beta]$. Consider a remaining $X[\alpha, \beta]$ of minimal height. For $\beta - \alpha \leq 1$ we have $\text{eval}(X[\alpha, \beta]) = a$ with $a \in \Gamma \cup \{1\}$. Without restriction there is a rule $X_a \rightarrow a$. (If not, we add such a rule.) We replace all occurrences of symbols $X[\alpha, \beta]$ by X_a . Thus, we may assume $\beta - \alpha \geq 2$.

For $\alpha > 0$ we may assume that there is a rule of the form $X \rightarrow \overline{Y}Z$, because the height is minimal and our assumption above. By some simple arithmetic we find $\gamma, \delta \in \mathbb{N}$ such that

$$\text{eval}(X[\alpha, \beta]) = \text{eval}(\overline{Y[0, \gamma]}) \text{eval}(Z[0, \delta]).$$

We introduce a new rule $X_{\alpha\beta} \rightarrow \overline{Y[0, \gamma]}Z[0, \delta]$. After that all symbols $X[\alpha, \beta]$ are replaced by the new variable $X_{\alpha\beta}$. The height of $X_{\alpha\beta}$ is $h(X)$ (but its weight is zero). Since $H(Y[0, \gamma]) + H(Z[0, \delta]) = h(Y) + h(Z) < 2h(X) = H(X[\alpha, \beta])$, this step is weight-reducing.

The remaining case is $\alpha = 0$. Without restriction we have now a rule of the form $X \rightarrow YZ$. For $\beta \leq |Y|$ we introduce a new symbol $Y[0, \beta]$ and a rule $X_\beta \rightarrow Y[0, \beta]E$ where E is the dummy symbol from above. For $\beta > |Y|$ we introduce a new symbol $Z[0, \gamma]$ with $\gamma = \beta - |Y|$ and rule $X_\beta \rightarrow YZ[\gamma]$. After that all symbols $X[0, \beta]$ are replaced by the new variable X_β . Since $H(Y[0, \beta]) = h(Y) < h(X) = H(X[0, \beta])$ and $H(Z[0, \gamma]) = h(Z) < h(X) = H(X[0, \beta])$, the step is again weight-reducing. The number of steps and the size of the new SLP is bounded by the weight of S . Thus, $\|S'\| \in \mathcal{O}(h(S) \|\Omega\|) \subseteq \mathcal{O}(|\Omega|^2)$.

The missing transformation to deal with ε - and chain rules is standard and does not further increase the size of the SLP. This proves the theorem. \square

2.1 Interval questions

The most basic question for SLPs is whether or not two variables evaluate to the same word. It can be answered in polynomial time, thus without unfolding the word in general. This fundamental result is due to Plandowski [21]. His proof uses the well-known Fine-and-Wilf-Theorem. In order to keep this section self-contained we state Fine-and-Wilf and we give its proof, which is due to Jeff Shallit.

Theorem 2.2 (Fine and Wilf, 1965) *Let $u, v \in \Sigma^*$ be non empty words, $s \in u\{u, v\}^*$ and $t \in v\{u, v\}^*$. Assume that s and t have a common prefix of length $|u| + |v| - \text{gcd}(|u|, |v|)$, then it holds $uv = vu$. In particular, $u, v \in r^*$ where r is the common prefix of u and v of length $|r| = \text{gcd}(|u|, |v|)$.*

Proof. We may assume $|u| \leq |v|$. The assertion is trivial for $|u| = 0$ or $|u| = |v|$. Hence we may assume $1 \leq |u| < |v|$. Since $\text{gcd}(|u|, |v|) \leq |v|$, we have $v = uw$. It remains to show $uw = wu$, because then $uv = u(uw) = u(wu) = (uw)u = vu$. Since $|s| \geq |u| + |v| - \text{gcd}(|u|, |v|) > |u|$, we obtain $s \in uu\{u, w\}^*$. We have $t \in vw\{u, w\}^*$, therefore $s' \in u\{u, w\}^*$ and $t' \in w\{u, w\}^*$ for the words s', t' mit $s = us'$ and $t = wt'$. Moreover $\text{gcd}(|u|, |v|) = \text{gcd}(|u|, |w|)$ and $|v| = |u| + |w|$, thus s' and t' have a common prefix of length $|u| + |w| - \text{gcd}(|u|, |w|)$. By induction we conclude $uw = wu$ and hence the claim. The standard fact that commuting words u and v are powers of a common prefix is left as an easy exercise. \square

An *interval question* for a given SLP is an expression of type

$$X[i, j] \stackrel{?}{=} Y[k, \ell].$$

It is this type of interval questions is used e.g. in the proof of Corollary 2.7.

The expression evaluates to *true* if and only if both, $0 \leq j - i = \ell - k \leq \min \{ |X|, |Y| \}$ and $\text{eval}(X)[i, j] = \text{eval}(Y)[k, \ell]$.

An interval question is of *standard type*, if it has the form $X[i, j] \stackrel{?}{=} Y[0, \ell]$ which we abbreviate as $X[i, j] \stackrel{?}{=} Y[\ell]$. It is called a *mixed question*, if it has the form $X[0, j] \stackrel{?}{=} Y[|Y| - j, |Y|]$, which we abbreviate as $X[j]_{\text{pf}} \stackrel{?}{=} Y[j]_{\text{sf}}$. The meaning is that a prefix of length j of $\text{eval}(X)$ appears as a suffix in $\text{eval}(\overline{Y})$. This explains the notation “pf” and “sf”. All mixed questions are of standard type.

Lemma 2.3 *Let $1 \leq p < q < j \leq |X|$. Then the following three mixed questions $X[j]_{\text{pf}} \stackrel{?}{=} Y[j]_{\text{sf}}$, $X[j - p]_{\text{pf}} \stackrel{?}{=} Y[j - p]_{\text{sf}}$, $X[j - q]_{\text{pf}} \stackrel{?}{=} Y[j - q]_{\text{sf}}$ evaluate to true if and only if the following two mixed questions $X[j]_{\text{pf}} \stackrel{?}{=} Y[j]_{\text{sf}}$, $X[j - g]_{\text{pf}} \stackrel{?}{=} Y[j - g]_{\text{sf}}$ evaluate to true, where $g = \gcd(p, q)$ is the greatest common divisor of p and q .*

Proof. This is direct consequence of Theorem 2.2. \square

Theorem 2.4 ([1, 8, 10, 21]) *The following computation can be performed in polynomial time.*

- *Input: SLP S and interval questions $X_m[i_m, j_m] \stackrel{?}{=} Y_m[k_m, \ell_m]$ for $1 \leq m \leq s$.*
- *Output: Those questions which evaluate to true.*

Proof. The proof follows from the next proposition. \square

Proposition 2.5 *The following problem (involving a collection of q interval questions) can be solved in $\mathcal{O}((q + \|S\|^2) \cdot h(S))$ arithmetic steps.*

- *Input: SLP S and interval questions $X_p[i_p, j_p] \stackrel{?}{=} Y_p[k_p, \ell_p]$ for $1 \leq p \leq q$.*
- *Problem: Do all questions evaluate to true?*

Proof. In a preprocessing phase we check that the requirements on indices are satisfied. We also remove variables with $|A| = 0$. The number of arithmetic operations is in $\mathcal{O}(\|S\|)$ and can be ignored.

Now, we start the transformation process on the list of questions. In the first phase we transform all interval questions into standard type $A[i, j] \stackrel{?}{=} B[\ell]$. Consider a question $A[i, j] \stackrel{?}{=} B[k, \ell]$ with $k \geq 1$, which is not standard. We may assume that the SLP contains a rule $A \rightarrow CD$, because otherwise the question had standard type. Depending on the indices there are three possibilities:

1. We can replace $A[i, j] \stackrel{?}{=} B[k, \ell]$ by some question $C[i, j] \stackrel{?}{=} B[k, \ell]$.
2. We can replace $A[i, j] \stackrel{?}{=} B[k, \ell]$ by some question $D[i', j'] \stackrel{?}{=} B[k, \ell]$.
3. We can replace $A[i, j] \stackrel{?}{=} B[k, \ell]$ by standard questions: $\overline{B[k, m]} \stackrel{?}{=} \overline{C[k']}$ and $B[m, \ell] \stackrel{?}{=} D[\ell']$.

After at most $\mathcal{O}(q \cdot h(S))$ steps we have produced a list of at most $2q$ standard questions. Thus, without restriction, all questions are of standard type at the very beginning.

Next, for each pair (A, B) we artificially introduce mixed questions $A[0]_{\text{pf}} \stackrel{?}{=} B[0]_{\text{sf}}$ and $A[|A|]_{\text{pf}} \stackrel{?}{=} A[|A|]_{\text{sf}}$ (which of course evaluate to *true*). Thus, the new number of questions is $Q = 2q + |\Omega| + |\Omega|^2$. Note that $A[i]_{\text{pf}} \stackrel{?}{=} B[i]_{\text{sf}}$ is equivalent to $\overline{B}[i]_{\text{pf}} \stackrel{?}{=} \overline{A}[i]_{\text{sf}}$. Therefore, a pair of mixed questions $A[i]_{\text{pf}} \stackrel{?}{=} B[i]_{\text{sf}}$ and $A[j]_{\text{pf}} \stackrel{?}{=} B[j]_{\text{sf}}$ can be counted as $A[i]_{\text{pf}} \stackrel{?}{=} B[i]_{\text{sf}}$ and $\overline{B}[j]_{\text{pf}} \stackrel{?}{=} \overline{A}[j]_{\text{sf}}$. In the next phases other mixed questions of type $A[i] \stackrel{?}{=} B[i]$ will be generated. However, due to Lemma 2.3 never more than two mixed questions need to be stored. We now do the counting of arithmetic operations with respect to a global sum of “euros” which are distributed over several accounts. First, each standard question $A[i, j] \stackrel{?}{=} B[k]$ obtains an account with $h(A) + h(B)$ euros. The invariant is that every question on the list has always at least $h(A) + h(B)$ euros on its account. In order to do so we need initially $\mathcal{O}(Q \cdot h(S))$ euros.

Consider a standard or mixed question $A[i, j] \stackrel{?}{=} B[k]$ on our list, where the sum of heights $h(A) + h(B)$ is maximal. If there is a rule $A \rightarrow a$ with $a \in \Gamma$, then we can evaluate this question in at most $h(B)$ arithmetic operations, and then we remove it. If the evaluation was *false*, we return *false* and stop. Thus, we may assume that the SLP contains a rule $A \rightarrow CD$. Depending on the indices there are again three possibilities:

- 1.) We can replace $A[i, j] \stackrel{?}{=} B[k]$ by some standard question $C[i, j] \stackrel{?}{=} B[k]$.
- 2.) We can replace $A[i, j] \stackrel{?}{=} B[k]$ by some standard question $D[i', j'] \stackrel{?}{=} B[k]$.
- 3.) We can replace $A[i, j] \stackrel{?}{=} B[k]$ by one mixed and one standard question:
 $B[\ell]_{\text{pf}} \stackrel{?}{=} C[\ell]_{\text{sf}}$ and $B[\ell, k] \stackrel{?}{=} D[m]$.

Note that in all three cases the sum of heights decreased. The tricky observation is that exactly two question of type $B[\ell']_{\text{pf}} \stackrel{?}{=} C[\ell']_{\text{sf}}$ and $\overline{C}[\ell'']_{\text{pf}} \stackrel{?}{=} \overline{B}[\ell'']_{\text{sf}}$ are on the list when replacing $A[i, j] \stackrel{?}{=} B[k]$, because we work top-down according to the height. Thus, the only thing that happens is that some $B[\tilde{\ell}]_{\text{pf}} \stackrel{?}{=} C[\tilde{\ell}]_{\text{sf}}$ is replaced by some $B[m]_{\text{pf}} \stackrel{?}{=} C[m]_{\text{sf}}$, where m is computed according to Lemma 2.3. In all three possibilities we need only one euro to pay the of arithmetic operations, and the rest can be transferred to the new accounts without destroying the invariant¹ If our list does not contain any question anymore without that we encountered *false*, then we can return *true*. \square

Remark 2.6 The time bound in Proposition 2.5 is not likely to be optimal. Better time complexities might be achieved by applying recompression methods in [11, 1, 20], see also [12]. We also refer to [17] for a recent survey on “Algorithmics on SLP-compressed strings”.

¹We count a gcd computation on binary numbers of polynomial length as one arithmetic operation. But this not essential because a more accurate amortized counting is possible. In any case it does not effect the polynomial time bound in Theorem 2.4.

Corollary 2.7 *The following computation can be performed in polynomial time.*

- *Input: SLP S and variables X, Y .*
- *Output: A number $p \in \mathbb{N}$ written in binary such that the length of the longest common prefix of $\text{eval}(X)$ and $\text{eval}(Y)$ has length p .*

Proof. For $X \in \Omega$ we have $|\text{eval}(X)| \leq 2^{h(X)}$, hence we can solve the problem by binary search invoking at most $h(X)$ calls to Theorem 2.4 with $s = 1$. \square

To finish the section let us go back to the situation of a free group $F(\Sigma)$ and $\Gamma = \Sigma \cup \overline{\Sigma}$. Recall that for $w \in \Gamma^*$ we denote by \widehat{w} the uniquely defined reduced word such that $w = \widehat{w} \in F(\Sigma)$.

Corollary 2.8 *The following computation can be performed in polynomial time.*

- *Input: An SLP S with constants in Γ .*
- *Output: An SPL \widehat{S} of size $\mathcal{O}(\|S\| \cdot h(S))$ such that for every variable X of S there is a variable \widehat{X} of \widehat{S} with $\text{eval}(\widehat{X}) = \widehat{\text{eval}(X)}$. This means that \widehat{X} evaluates to the reduced normal form of $\text{eval}(X)$.*

Proof. Consider a rule $X \rightarrow YZ$. By induction on the height we may assume that we have already generated variables \widehat{Y} and \widehat{Z} such that $\text{eval}(\widehat{Y}) = \widehat{\text{eval}(Y)}$ and $\text{eval}(\widehat{Z}) = \widehat{\text{eval}(Z)}$. In addition we may assume that $h(Y) = h(\widehat{Y})$ and $h(Z) = h(\widehat{Z})$. Using Corollary 2.7 we calculate the length of the longest common prefix of $\text{eval}(\widehat{Y})$ and $\text{eval}(\widehat{Z})$. Knowing the length it is straightforward how to introduce new variables Y' and Z' such that $\widehat{\text{eval}(X)} = \text{eval}(Y'Z')$. For this procedure we need at most $h(Y) + h(Z)$ new rules and additional variables. Thus, we can introduce another variable \widehat{X} and rule $\widehat{X} \rightarrow Y'Z'$. This gives us the new SLP of size $\mathcal{O}(\|S\| \cdot h(S))$. \square

The compressed word problem can be defined in arbitrary (finitely generated) monoids M . For that choose some finite generating set Σ . The input to the *compressed word problem* over M is given by two SLPs with constants in Σ and axioms A and B resp. The question is whether or not A and B evaluate to the same element in M . Changing the finite set of generators does not affect whether or not the compressed word problem can be solved in **P** or **NP**.

Proposition 2.9 ([24]) *Let M be a finitely generated monoid and N be a finitely generated submonoid of the monoid of endomorphisms $\text{End}(M)$. There is a polynomial-time reduction of the word problem of N to the compressed word problem of M .*

Proof. The reduction is explained in Example 1.2. \square

Proposition 2.10 ([16]) *Let F be a finitely generated free group. Then the compressed word problem can be solved in polynomial time.*

Proof. Compute \widehat{S} according to Corollary 2.8 and check that \widehat{X} evaluates to 1. \square

Proposition 2.9 and Proposition 2.10 show that the word problem of the automorphism group of finitely generated free groups can be decided in polynomial time [24], since their automorphism group is finitely generated. More generally, the same result holds for finitely generated right-angled Artin groups, see [18] for details.

3 Word equations with constraints

As above we let Ω be a set of variables and Γ is used as an alphabet of constants. A *word equation* is written as $L = R$ where $L, R \in \Omega^*$, a *constraint* is written as $X \in \mathcal{C}$ where $X \in \Omega_+$ and $\mathcal{C} \subseteq \Gamma^*$. A *Boolean formula of equations with constraints* \mathcal{S} is a Boolean formula where the atomic propositions are either word equations $L = R$ or constraints $X \in \mathcal{C}_j$.

A *solution* of \mathcal{S} is a morphism $\sigma : \Omega^* \rightarrow \Gamma^*$ (given by mapping $\sigma : \Omega_+ \rightarrow \Gamma^*$) such that the Boolean formula evaluates to “true”, if we substitute the atomic propositions by the corresponding truth values $\sigma(L) = \sigma(R)$ and $\sigma(X) \in \mathcal{C}$.

A *system of equations with constraints* is simply a conjunction of atomic propositions. Making non-deterministic guesses the existence of a solution of a Boolean formula can be reduced to check the existence of a solution for a system of equations. Since we allow constraints we may replace inequalities by constraints. For example, if we consider equations over a group G , an inequality $L \neq R$ can be replaced by the conjunction $L = RX \wedge X \in G \setminus \{1\}$, where X is a fresh variable. Thus, frequently it is enough to consider systems of equations with constraints. Moreover, we do not need constants. A constant $a \in \Gamma$ is replaced by a variable A and the corresponding constraint $A \in \{a\}$.

3.1 Free intervals

For the rest of the section we work with a fixed system \mathcal{S} and a fixed solution σ . In 3.2 we will define a “generic solution” specified by σ , and we show that it can be compressed by interval grammars. We write $L = X_1 \cdots X_g$ and $R = X_{g+1} \cdots X_d$ with $X_i \in \Omega$ for $1 \leq i \leq d$. Clearly, $\sigma(L) = \sigma(R)$.

For a word $w \in \Gamma^*$ we call $\{0, \dots, |w|\}$ its set of *positions*. The idea is that letters of w occur between positions. For positions α, β we call $[\alpha, \beta]$ an *interval*. If $0 \leq \alpha \leq \beta \leq m$, then it corresponds to the factor $w[\alpha, \beta]$. The involution on intervals is defined by $\overline{[\alpha, \beta]} = [\beta, \alpha]$. Accordingly, we define $w[\beta, \alpha] = \overline{w[\alpha, \beta]}$. An interval $[\alpha, \beta]$ is called *positive*, if $\alpha < \beta$.

The factorization $w = \sigma(X_1) \cdots \sigma(X_g) = \sigma(X_{g+1}) \cdots \sigma(X_d)$ along the given solution σ “cuts” the word w into pieces. To make this formal, we define for each $0 \leq i \leq d$ positions $l(i)$ and $r(i)$ such that $\sigma(X_i)$ starts in w at the left position $l(i)$ and it ends at the right position $r(i)$. Each such position is called a *cut*. Positions 0 and m are cuts and there are at most d cuts. Clearly, if $X_i = X_j = \overline{X_k}$, then

$$w[l(i), r(i)] = w[l(j), r(j)] = w[r(k), l(k)].$$

Next, we are going to define an equivalence relation \approx on the set of intervals of w . For that we start with a pair (i, j) such that $i, j \in \{1, \dots, d\}$ where $X_i = X_j$ or $X_i = \overline{X_j}$. For all $\mu, \nu \in \{0, \dots, r(i) - l(i)\}$ we define a relation between intervals \sim by:

Proof. By symmetry we may assume that $\alpha < \beta$. We show the existence of $[\gamma, \delta]$ where $[\alpha, \beta] \approx [\gamma, \delta]$ and γ is a cut. (The existence of $[\gamma', \delta']$ where $[\alpha, \beta] \approx [\gamma', \delta']$ and δ' is a cut follows analogously.)

If $\alpha = 0$, then α is a cut and we can choose $[\alpha, \beta] = [\gamma, \delta]$. Hence let $1 \leq \alpha$ and consider the positive interval $[\alpha - 1, \beta]$. Then, for some cut γ we have $[\alpha - 1, \beta] \approx [\alpha', \delta]$ with $\min\{\alpha', \delta\} < \gamma < \max\{\alpha', \delta\}$ and $|\gamma - \alpha'| = 1$. A simple reflection shows that we have $[\alpha - 1, \alpha] \approx [\alpha', \gamma]$ and $[\alpha, \beta] \approx [\gamma, \delta]$. Hence the claim. \square

Corollary 3.5 ([7], **Prop. 42**) *Let $\tilde{\Gamma}$ be the set of equivalence classes of maximal free intervals. Then $\tilde{\Gamma}$ is closed under involution and it has at most $2d - 2$ elements.*

Proof. Let $[\alpha, \beta]$ be a maximal free interval. Then $[\beta, \alpha]$ is a maximal free interval by definition. Hence $\tilde{\Gamma}$ is closed under involution. By Proposition 3.4 we may assume that α is a cut. Say $\alpha < \beta$. Then $\alpha \neq m$ and there is no other maximal free interval $[\alpha, \beta']$ with $\alpha < \beta'$ because of maximality. Hence there are at most $d - 1$ such intervals $[\alpha, \beta]$. Symmetrically, there are at most $d - 1$ maximal free intervals $[\alpha, \beta]$ where $\beta < \alpha$ and α is a cut. \square

There are two types of maximal free intervals which play a quite different role. Those of length 1 can be viewed as fixed of constants whereas maximal free intervals of length greater than 1 are specified by words which, without the presence of constraints, can be replaced by empty words in order to shorten the length of a solution.

3.2 Generic solutions

In an algebraic setting the situation is now as follows. Let $X \in \Omega$, we may assume that X appears in the equation $L = R$. Hence $\sigma(X)$ is a factor of $w = \sigma(L) = \sigma(R)$. The word w factorizes as a product $w[\alpha_0, \alpha_1] \cdots w[\alpha_{\ell-1}, \alpha_\ell]$, where $[\alpha_0, \alpha_1], \dots, [\alpha_{\ell-1}, \alpha_\ell]$ are maximal free intervals. We may read this as a factorization in a word of length ℓ over $\tilde{\Gamma}$. Thus, the solution σ defines a mapping

$$\tilde{\sigma} : \Omega_+ \rightarrow \tilde{\Gamma}^*. \quad (1)$$

Now, using the mapping $\omega : \tilde{\Gamma}^* \rightarrow \Gamma^*$ defined above by $[\alpha, \beta] \mapsto w[\alpha, \beta]$ we obtain the following factorization:

$$\sigma : \Omega_+ \xrightarrow{\tilde{\sigma}} \tilde{\Gamma}^* \xrightarrow{\omega} \Gamma^*.$$

The mapping $\tilde{\sigma} : \Omega_+ \rightarrow \tilde{\Gamma}^*$ is called the *generic solution* specified by σ .

If $\omega' : \tilde{\Gamma} \rightarrow \Gamma^*$ is any mapping which is compatible with the involution such that $\omega'(\tilde{\sigma}(X_j)) \in \mathcal{C}_j$ for all j , then the morphism $\omega' \circ \tilde{\sigma} : \Omega^* \rightarrow \Gamma^*$ is another solution. The following result is closely related to [22].

Theorem 3.6 *Let $L_i = R_i$ be a system of equations with $L_i, R_i \in \Omega^*$ where $1 \leq i \leq k$ and let $\sigma : \Omega_+ \rightarrow \Gamma^*$ be any solution. Let $d = \sum_{i=1}^k |L_i R_i|$ be the denotational length, $\tilde{\sigma} : \Omega_+ \rightarrow \tilde{\Gamma}^*$ the generic solution as defined in (1), $\tilde{N} = |\tilde{\sigma}(L)|$ its length.*

Then there is an SLP S of size $\mathcal{O}(d^2 \cdot \log^2 \tilde{N})$ such that each $X \in \Omega_+$ appears also as variable in S and satisfies $\text{eval}(X) = \tilde{\sigma}(X)$.

Proof. For the purpose of the proof we may assume that $\tilde{\Gamma} = \Gamma$ and $\tilde{\sigma} = \sigma$. We continue with the notation of above. Hence $w = \sigma(L) = \sigma(X_1 \cdots X_g) = \sigma(X_{g+1} \cdots X_d)$. Since σ is a generic solution, we know that all maximal free intervals have length 1. Therefore we do not need to compress words which correspond to long free intervals.

For all cuts γ and all $\lambda \in \mathbb{N}$ with $2^\lambda < 2m$ we introduce a new variable $C_{\gamma\lambda}$ and its dual $\overline{C}_{\gamma\lambda}$. The idea is that $C_{\gamma\lambda}$ evaluates to the word $w[\mu, \nu]$ where $\mu = \max\{0, \gamma - 2^\lambda\}$ and $\nu = \min\{m, \gamma + 2^\lambda\}$.

For $\lambda = 0$ we have $\mu, \nu \in \{\gamma - 1, \gamma, \gamma + 1\}$. The interval $[\mu, \nu]$ corresponds to a word $u_\gamma = w[\mu, \nu] \in \Gamma^*$ with $|u_\gamma| \leq 2$. In this case we introduce a rule $C_{\gamma,0} \rightarrow u_\gamma$.

Now, if $\lambda \geq 1$, then we begin with an auxiliary rule

$$C_{\gamma,0} \rightarrow [\mu, \nu]C_{\gamma, \lambda-1}[\mu', \nu']. \quad (2)$$

Here:

$$\begin{aligned} \mu &= \max\{0, \gamma - 2^\lambda\}, & \nu &= \max\{0, \gamma - 2^{\lambda-1}\}, \\ \mu' &= \min\{m, \gamma + 2^{\lambda-1}\}, & \nu' &= \min\{m, \gamma + 2^\lambda\}. \end{aligned}$$

Without restriction we have $\mu < \nu$ and $\mu' < \nu'$. Consider the interval $[\mu, \nu]$. There are two cases.

In the first case $\mu - \nu = 1$. Then $w[\mu, \nu]$ is a letter of Γ . In this case, we simply substitute in (3) the expression $[\mu, \nu]$ by that letter. Analogously, we deal with $[\mu', \nu']$, if this is a free interval.

In the second case $\mu - \nu \geq 2$ and $[\mu, \nu]$ is not free. Then however there exists a cut δ and (by symmetry and duality) $w[\mu, \nu]$ becomes the factor of some word $\text{eval}(C_{\delta, \lambda-1})[\alpha, \beta]$ for suitable values α, β with $0 \leq \alpha < \beta \leq 2^\lambda$. In this case, we substitute in (3) the expression $[\mu, \nu]$ by $C_{\delta, \lambda-1}[\alpha, \beta]$. Analogously, we deal with $[\mu', \nu']$.

For example, after these substitutions a rule in (3) might have the following form $C_{\gamma,0} \rightarrow a C_{\gamma, \lambda-1} C_{\eta, \lambda-1}[\alpha', \beta']$.

Finally, we observe that each variable X which occurs in $L = R$ is some X_i . Without restriction we have $X \in \Omega_+$. For the maximal value of λ we introduce an additional chain rule

$$X \rightarrow C_{1(i), \lambda}[0, |X|]. \quad (3)$$

After transforming all rules in Chomsky normal form we obtain an interval grammar of size $\mathcal{O}(d \cdot \log \tilde{N})$. The final step is the transformation of the interval grammar into an SLP using Theorem 2.1. This establishes the bound $\mathcal{O}(d^2 \cdot \log^2 \tilde{N})$. \square

According to Theorem 3.6 we can compress the generic solution by some SLP and then we can obtain a solution in Γ^* by substituting maximal free intervals. Say, we have a promise that a solution exists such that $|\sigma(X)|$ has at most exponential length for each variable. Then we can guess in non-deterministic polynomial time an SLP for the generic solution $\tilde{\sigma}$. But this does not mean that we can efficiently check that $\tilde{\sigma}$ corresponds to an actual solution because one still has to check that there exists a substitution respecting the constraints. In order to explain the difficulty let us consider the special case of equations with rational constraints. The family of rational subsets is defined for every monoid M . It consists of the smallest family containing the finite subsets of M and which

is closed under finite union, product and “generated submonoid”. It has been shown in [7] that the existential theory of equations with rational constraints over free groups is PSPACE complete. The PSPACE hardness follow from the classical fact that the intersection problem for regular languages in free monoids is PSPACE complete, [15]. The input to that problem is simply a collection of n finite (deterministic) automata A_1, \dots, A_n and the question is whether $L(A_1) \cap \dots \cap L(A_n) \neq \emptyset$, where $L(A_i)$ denotes the accepted language. (It is easy to encode this problem by a system of equations with rational constraints.) Now, if $L(A_1) \cap \dots \cap L(A_n) \neq \emptyset$ then a shortest word in the intersection has at most exponential length. However, in general we cannot expect that there is any SLP of polynomial size representing this shortest word. If it were then we could guess the corresponding SLP in non-deterministic polynomial time and then check in deterministic polynomial time that the SLP generates a word in the intersection. As a consequence we could deduce $\text{NP}=\text{PSPACE}$, which is widely assumed to be false.

4 Free products of abelian groups

In the following G_α denote abelian groups. We assume that each G_α is generated by a subset $\Gamma_\alpha \subseteq G_\alpha \setminus \{1\}$ which is closed under involution, i.e., $g \in \Gamma_\alpha$ implies $g^{-1} \in \Gamma_\alpha$. We let P be a finite index set and $F = \star_{\alpha \in P} G_\alpha$ be the free product. Thus, F is a finitely generated free product of abelian groups. The direct product $F^{\text{ab}} = \prod_{\alpha \in P} G_\alpha$ is the abelian quotient of F .

We let $\Gamma = \bigcup_{\alpha \in P} \Gamma_\alpha$ be the disjoint union. Then Γ is an alphabet with involution. We obtain a morphism $\psi : \Gamma^* \rightarrow F$ and elements of F can be represented as words over Γ . Words in Γ^* are split into factors according to α . To make this formal we let $\Delta_\alpha = G_\alpha \setminus \{1\}$ and $\Delta = \bigcup_{\alpha \in P} \Delta_\alpha$ be the disjoint union. Then Δ is also an alphabet with involution, but typically infinite. The inclusions $\Gamma_\alpha \subseteq \Delta_\alpha \subseteq G_\alpha$ induce canonical morphisms

$$\Gamma^* \subseteq \Delta^* \xrightarrow{\psi} F \rightarrow F^{\text{ab}}.$$

We also have a morphism $\psi_\alpha : \Delta^* \rightarrow G_\alpha$ which is induced by $\psi_\alpha(g) = g$ for $g \in G_\alpha$ and $\psi_\alpha(g) = 1$ otherwise.

A word $a_1 \cdots a_n$ with $a_i \in \Delta$ is called *reduced*, if $a_i \in \Delta_\alpha$ implies $a_{i+1} \notin \Delta_\alpha$ for all $\alpha \in P$ and $1 \leq i < n$. Every element in F has a unique normal form f_Δ as a reduced word over Δ . We identify the set F with its set of normal forms $\widehat{F} = \{f_\Delta \mid f \in F\} \subseteq \Delta^*$. For $f \in F$ we let $|f|_\Delta = |f_\Delta|$ be the length a reduced word in Δ^* representing f , whereas $|f|_\Gamma$ denotes the length of a shortest word over Γ^* representing f . Note that $|f|_\Delta \leq |f|_\Gamma$. A word $w \in \Gamma^*$ of length $|f|_\Gamma$ representing f is called a *geodesic* word for f . In contrast to f_Δ geodesics are not unique, in general.

4.1 Extended Parikh-constraints

For a word $w \in \Delta^*$ we let $|w|_\alpha$ the number of letters from Δ_α . The vector $(|w|_\alpha)_\alpha \in \mathbb{N}^P$ is called the *Parikh-image* of w . It counts how often a position $\alpha \in$

P is used as a non trivial factor in a word over Δ . We have $|w| = \sum_{\alpha} |w|_{\alpha}$. We also let $\pi_{\alpha}(w) = (|w|_{\alpha}, \psi_{\alpha}(w)) \in \mathbb{N} \times G_{\alpha}$. This extends to a unique morphism

$$\Delta^* \rightarrow \prod_{\alpha \in P} (\mathbb{N} \times G_{\alpha}) = \mathbb{N}^P \times F^{\text{ab}}.$$

However, later in the applications we need also to control the first and last positions from P , because we need that if we replace a factor by some other factor in a reduced word, the new word must be still reduced. Therefore we use two more mappings. We define $\text{first}(w) \in P \cup \{1\}$ to be 1 is empty and to be $\alpha \in P$, if the reduced form of w starts with a non empty factor in P . Symmetrically, we let $\text{last}(w)$ to be the last position. Thus, $\text{last}(w) = \text{first}(\bar{w})$.

This yields an ‘‘extended Parikh-mapping’’

$$\begin{aligned} \pi : \Delta^* &\rightarrow \mathbb{N}^P \times \prod_{\alpha \in P} G_{\alpha} \times (P \cup \{1\}) \times (P \cup \{1\}) \\ \pi(w) &= ((|w|_{\alpha})_{\alpha \in P}, \varphi(w), \text{first}(w), \text{last}(w)). \end{aligned}$$

Using $F = \widehat{F} \subseteq \Delta^*$ we can apply π to elements in the group F . The idea is to change solutions in such a way that they become compressible by SLPs, but the image under π remains invariant.

For simplicity of notation we choose for every index $\alpha \in P$ some fixed letter, called $\alpha \in \Gamma_{\alpha}$ again. Thus, we view $P \subseteq \Gamma \subseteq \Delta$ and we can speak about reduced words in P^* . Such a word is a sequence $\alpha_1 \cdots \alpha_n$ with $\alpha_i \neq \alpha_{i+1}$ for all $1 \leq i < n$. We have the following combinatorial lemma which is crucial for compression.

Proposition 4.1 *Let $w = \alpha_1 \cdots \alpha_n \in P^*$ be a reduced sequence of length $n \geq 1$ with $a = \alpha_1$, $c = \alpha_n$; and let $|\text{alph}(w)| = \ell$.*

If $\ell \leq 2$, then w has either the form $(ac)^k$ or $(ab)^k a$ for $k = \lfloor \frac{n}{2} \rfloor$. If $\ell \geq 3$, then there exists a reduced word $w' \in P^$ with $\pi(w) = \pi(w')$ and $w' \in aP^*c$ such that one of the following assertions hold.*

1. *It is $|w|_a = \lceil \frac{n}{2} \rceil$ and for $k = \ell - 1$ and some $d \in \{1, a\}$ we have*

$$w' = (a\beta_1)^{n_1} \cdots (a\beta_k)^{n_k} d.$$

2. *If $|w|_a = \frac{n}{2}$ and for $k = \ell - 1$ we have*

$$w' = (a\beta_1)^{n_1} (\beta_2 a)^{n_2} \cdots (\beta_k a)^{n_k}.$$

3. *It is $|w|_a < \frac{n}{2}$ and for some $d \in \{1, c\}$ and $k \leq \binom{\ell}{2}$ we have*

$$w' = (a\gamma_1)^{n_1} (\beta_2 \gamma_2)^{n_2} \cdots (\beta_k \gamma_k)^{n_k} d.$$

Proof. The proof is obvious for $\ell \leq 2$. Hence let $\ell \geq 3$. Note that the image of $\psi(P^*)$ lies in the abelian group F^{ab} . Thus, we it is enough to show that there exists a reduced sequence $w' \in aP^*c$ with $|w|_{\alpha} = |w'|_{\alpha}$ for all $\alpha \in \text{alph}(w)$. Since w is reduced we have $|w|_a \leq \lceil \frac{n}{2} \rceil$. For $|w|_a \geq \frac{n}{2}$ we are in situation 1 or 2. If n is odd we are in situation 1 with $a = c = d$. If n is even and $a \neq c$ we are in situation 1 with $\beta_k = c$ and $d = 1$. If n is even and $a = c$ we are in situation 2.

For the rest of the proof we may therefore assume $|w|_a < \frac{n}{2}$. If n is odd, then the assertion holds for the word $\tilde{w} = \alpha_1 \cdots \alpha_{n-1} \in P^*$ with first letter a and last letter α_{n-1} by induction. We are done in this case with $d = c$ since α_{n-1} exists and $\gamma_k = \alpha_{n-1} \neq \alpha_n = c$.

It remains to show 3 under the assumption that $|w|$ is even and $|w|_a < \frac{n}{2}$. Note that this implies $n \geq 4$. We match indices $1, \dots, n$ by defining sets $V_{ij} = \{i, j\}$ such that $\alpha_i \neq \alpha_j$ and in such a way that the collection of the sets V_{ij} yields a partition of $V = \{1, \dots, n\}$. To see that this is possible start with any partition of V into two-element subsets V_{ij} . Assume there is some V_{ij} with $\alpha_i = \alpha_j$. As w is reduced, we have $|w|_{\alpha_i} \leq n/2$. Hence there must be some V_{pq} with $\alpha_p \neq \alpha_i \neq \alpha_q$. We replace V_{ij}, V_{pq} by V_{ip}, V_{jq} . Continuing this way we achieve a partition as desired.

Consider the set $V_{i_n n}$ and let $\alpha_{i_n} = b$. Then we have $b \neq c$. We are going to construct a reduced word of the form

$$w' = (a\gamma_1)^{n_1} (\beta_2\gamma_2)^{n_2} \cdots (\beta_{k-1}\gamma_{k-1})^{n_{k-1}} (bc)^{n_k}.$$

We construct w' under the restriction that w' is reduced, it begins with a and it ends in the factor bc . The idea is to write the sets V_{ij} in a list starting with some V_{1j_1} and ending in $V_{i_n n}$; and then to replace V_{ij} by $\alpha_i\alpha_j$. We must show that the resulting word w' is reduced. We know that $V_{ij} \neq V_{pq}$ implies $\{\alpha_i, \alpha_j\} \neq \{\alpha_p, \alpha_q\}$. This shows $k \leq \binom{\ell}{2}$. As $\alpha_i \neq \alpha_j$ for all V_{ij} we have always two options how to continue until the last $V_{i_n n}$. For $a = b$ we can avoid $\gamma_{k-1} = b$ and therefore the construction of w' is straightforward. Now for $a \neq b$ we have $a \neq b \neq c$. Hence $|w|_b = n/2$ cannot happen, because w is reduced of even length. This means there is at least one set V_{ij} with $\alpha_i \neq b \neq \alpha_j$. We may assume that the replacement of V_{ij} by $\alpha_i\alpha_j$ results in a factor $\beta_m\gamma_m$ for some $m \leq k-1$ such that we can avoid $\gamma_q = b$ for all $m < q \leq k-1$. \square

4.2 Equations over free products of abelian groups

As above we continue with a free product of abelian groups F . An *equation* over F is written as $L = R$ where $L, R \in \Omega^*$. We do not need constants, because we allow extended Parikh-constraints.

We use the following well-known fact. It shows that solvability of an equation over F split into two parts. A global word equation over Δ and local equations over the G_α . Its proof is straightforward and omitted.

Lemma 4.2 *Let $u, v, w \in \Delta^*$ be reduced words. Then we have $uv = w$ in the free product F if and only if there are $\alpha \in P$, $a, b, c \in G_\alpha$, $p, q, r \in \Delta^*$ such that*

1. $u = pa\bar{q}$, $v = qbr$, and $w = pcr$ in Δ^* ,
2. $ab = c$ in the abelian group G_α .

We construct a new system of equations \mathcal{S}' such that σ solves \mathcal{S}' and such that all solutions of \mathcal{S}' are also solutions of \mathcal{S} . The construction is as follows. First, we transform all equations into triangular form, i.e., they look like $XY = Z$ where $X, Y, Z \in \Omega$.

Next, we split the triangular system of equations into two parts, a global part of word equations with solutions in Δ^* and a local part of equations of

type $AB = C$ with solutions in G_α . Now the trick is to put $ab = c$ into constraints. More concretely consider an equation $XY = Z$ of our system. Let $u = \sigma(X), v = \sigma(Y), w = \sigma(Z) \in \Delta^*$ be the reduced words given by σ . We choose $\alpha \in P, a, b, c \in G_\alpha, p, q, r \in \Delta^*$ according to Lemma 4.2. We introduce fresh symbols A, B, C, P, Q, R and we add them to Ω_+ .

In the next step we replace the equation $XY = Z$ in \mathcal{S} by three equations:

$$U = PA\overline{Q}, \quad V = QBR, \quad W = PCR.$$

We simulate the equation $AB = C$ by constraints. To do so, we introduce three additional extended Parikh-constraints:

$$A = \{a\}, \quad B = \{b\}, \quad C = \{c\}.$$

We also extend the solution by defining $\sigma(A) = a, \sigma(B) = b, \dots, \sigma(R) = r$. Moreover, we add the constraint $X \in \widehat{F}$ for all $X \in \Omega_+$ where \widehat{F} is the set of reduced words in Δ^* .

This step finishes the transformation and defines a system \mathcal{S}' with constraints. All equations are triangular and the constraints are conjunctions of extended Parikh-constraints and constraints of the form $X \in \widehat{F}$. This is not an extended Parikh-constraint!

Note that σ still solves the new system with constraints and if σ' is any other solution of the new system, then σ' solves the original system \mathcal{S} as well because $ab = c$ in the abelian group G_α .

Finally, for a solution $\sigma : \Omega_+ \rightarrow \Delta^*$ and $X \in \Omega$ we let $\sigma_\alpha(X) \in G_\alpha$ be the image of $\sigma(X)$ under the natural projection of Δ^* onto G_α . Each $\sigma_\alpha(X)$ can be written as a word over Γ_α , and a word $\sigma(X)$ can be written as a word over Γ of length $|\sigma(X)|_\Gamma$.

If Φ is a Boolean formula of equations over F with constraints and $\sigma : \Omega_+ \rightarrow \Delta^*$ is a solution then there we can extract a system of equations \mathcal{S} and subset of constraints (and their negations) and additional constraints of type $X \neq 1$ such that σ is a solution of \mathcal{S} and moreover, every solution of \mathcal{S} solves Φ , too. We define the size of the formula Φ by $\|\Phi\| = |\Gamma| + \sum_{i=1}^k |L_i R_i|$, where $L_i = R_i$ are the equations used in the formula with $L_i, R_i \in \Omega^*$. Note that the size $\|\Phi\|$ does not take the constraints into account. For the length of a solution we have to write words over DD as words over Γ . Therefore we define the length of σ by the number $N = |\Omega| + \sum_{X \in \Omega} |\sigma(X)|_\Gamma$. The term $|\Omega|$ takes care to write down $\sigma(X) = 1$.

Theorem 4.3 *There exists a polynomial $p(n)$ such that the following holds: Let F be a free product of abelian groups and Γ be a set of generators of F . Let Φ be a Boolean formula of equations over F with extended Parikh-constraints and and let $\sigma : \Omega_+ \rightarrow \Delta^*$ be a solution in reduced words of length N .*

Then there is also a solution $\sigma' : \Omega_+ \rightarrow \Delta^$ in reduced words such that the following conditions hold.*

1. *We have $\pi(\sigma(X)) = \pi(\sigma'(X))$ for all $X \in \Omega$.*
2. *There is an SLP S with constants in Γ of size at most $p(\|\Phi\| + \log N)$ such that each $X \in \Omega$ appears also as variable in S and satisfies $\text{eval}(X) = \sigma'(X)$ in the group F .*

Proof. With the help of Parikh constraints we may assume that the input Φ is given by some system \mathcal{S} of equations with extended Parikh-constraints. Next we may assume that all equations are in triangular form. According to Lemma 4.2 we transform the input into a system of word equations with extended Parikh-constraints. We do not need the constraints $X \in \hat{F}$ because σ is a solution in reduced words. Hence, σ can be extended to a solution in reduced words of the new system and the total length N is still polynomial in its original value.

In the next step, we produce the generic solution $\tilde{\sigma}$ (which belongs to σ) according to Theorem 3.6. This solution has an SLP of size which is polynomial in $|\Phi| + \log N$. In the generic solution we must substitute maximal free intervals by compressible words which respects the extended Parikh-constraints. Note that it is here that we need the control on first and last letters, because after substitution the words have to remain reduced. In order to produce short SLPs for the substitution we use Proposition 4.1. \square

Note that Theorem 4.3 does not say that every solution can be compressed using an SLP. Even if N is minimal we only state that there is another solution with good compression. However if there are no constraints at all or, more general, if we content ourselves to “alphabetic” constraints, then we can state a stronger statement.

Let $F = \prod_{\alpha \in P} G_\alpha$ be a free product as above. For an element $w \in F$ we let $\text{alph}(w) = \{ \alpha \in P \mid |w|_\alpha \geq 1 \}$ be the *alphabet* of w . The alphabet specifies which factors in the free product are used in a reduced representation of w . This allows to define an *alphabetic constraint* by

$$\{ w \in F \mid \text{alph}(w) = A, \text{first}(w) = \beta, \text{last}(w) = \gamma \},$$

where $\alpha \subseteq P$, $\beta, \gamma \in P$. Clearly, an alphabetic constraint is just a special case of an extended Parikh-constraint. There are $|P|^2 2^{|P|}$ alphabetic constraints, but in formulae it is enough to have atomic constraints of the form $\{ w \in F \mid \alpha \in \text{alph}(w), \text{first}(w) = \beta, \text{last}(w) = \gamma \}$, where $\alpha, \beta, \gamma \in P$.

Theorem 4.4 *There exists a polynomial $p(n)$ such that the following holds: Let F be a free product of abelian groups and Γ be a set of generators of F . Let Φ be a Boolean formula of equations over F with alphabetic constraints and let $\sigma : \Omega_+ \rightarrow \Delta^*$ be a solution such that its length N is minimal among all solutions. Then there is an SLP S of size $p(\|\Phi\| + \log N)$ such that each $X \in \Omega$ appears also as a variable in S and satisfies $\text{eval}(X) = \sigma(X)$ in the group F .*

Proof. The proof is almost identical to the proof of Theorem 4.3. The difference is that in order to substitute maximal free intervals of the generic solution by words we can use the words from the original solution given by σ . These words are necessarily the shortest ones which respect the alphabetic constraints. Thus they visit at most $|P|$ positions. Thus, each of them has an SLP representation of size $\mathcal{O}(|P| \cdot \log N)$. \square

Corollary 4.5 *Let F be a free product of abelian groups and Γ be a set of generators of F . Assume that the length of minimal solutions of equations over F with alphabetic constraints can be bounded by some exponential function in $2^{n^{\mathcal{O}(1)}}$. Then the question whether a given Boolean formula of equations over F with alphabetic constraints has a solution in F can be decided in NP, i.e., in non-deterministic polynomial time.*

Proof. The NP-algorithm guesses an SLP for some solution of minimal length. The size of the SLP has polynomial size. After that a deterministic polynomial-time algorithm checks that the SLP is indeed a solution in reduced words verifying the alphabetic constraints. \square

5 Hyperbolic groups

In this section G denotes a torsion-free non-elementary δ -hyperbolic group which is generated by some finite subset $\Sigma \subseteq G \setminus \{1\}$. As usual, we let $\Gamma = \Sigma \cup \bar{\Sigma}$ where $\bar{\Sigma} = \Sigma^{-1}$. We view Γ as a finite alphabet with involution and we denote by $\pi : \Gamma^* \rightarrow G$ the canonical morphism onto G . For a word $w \in \Gamma^*$ we denote by $|w|$ its length and by $|w|_G$ its *geodesic length*. It is the length of a shortest word u such that $\pi(w) = \pi(u)$. Phrased differently, $|w|_G$ is the length of a shortest path from 1 to $\pi(w)$ in the Cayley graph $\text{Cay}(G, \Sigma)$ of G with respect to the generating set Σ . As usual, a word $w \in \Gamma^*$ is called *geodesic*, if $|w| = |w|_G$. We say that a word $w \in \Gamma^*$ is (λ, d) -*quasi-geodesic*, if every factor u of w satisfies

$$|u| \leq \lambda|u|_G + d.$$

Note that a (λ, d) -quasi-geodesic of length greater than d can never represent the identity in G . A word $w \in \Gamma^*$ is called μ -*locally* (λ, d) -*quasi-geodesic*, if every factor u of w which has length at most μ is (λ, d) -quasi-geodesic. A fundamental property of a hyperbolic group is that local quasi-geodesics are global quasi-geodesics for the appropriate choice of parameters. More precisely, [2, Thm. 1.4] and [9, Rem. 7.2.B] provide for all λ, d an effective bound for μ which is polynomial in $\lambda + \delta$ such that every μ -local (λ, d) -quasi-geodesic word is (λ', d') -quasi-geodesic with $\mu > d'$. Now, being μ -locally (λ, d) -quasi-geodesic is a local property which is therefore a “rational constraint”. This fact has also been used in [5] in order to show that the existential theory of a hyperbolic group is decidable. However, we need a more precise statement than being a rational constraint. Let us have a closer look.

Lemma 5.1 *Let u be a μ -local (λ, d) -quasi-geodesic word in G . Then there is a word v of length less than $|\Gamma|^\mu$ such that for all $x, y \in \Gamma^*$ the word $z = xuy$ is μ -locally (λ, d) -quasi-geodesic if and only if $z' = xvy$ is μ -locally (λ, d) -quasi-geodesic. Moreover, if $u \neq v$ then $|v| \geq \mu - 1$.*

Proof. Within this proof we abbreviate “ μ -locally (λ, d) -quasi-geodesic” by “ μ -local”. The proof follows from pigeon hole principle. We may assume $|u| \geq |\Gamma|^\mu$ because otherwise we may choose $u = v$. The word u is longer than $\mu - 1 + |\Gamma|^{\mu-1}$ because $|\Gamma| \geq 2$. Hence there is factor r of u which has length $\mu - 1$ and which occurs at least twice. Therefore, we find factorizations $u = prs = trq$ such that p is a proper prefix of t . Now, the word $v = prq$ is μ -local and it shares the same prefix (suffix resp.) of length $\mu - 1$ as u . Thus, for all $x, y \in \Gamma^*$ the word $z = xuy$ is μ -local if and only if $z' = xvy$ is μ -local. The word v is shorter than u , but the length is at least $|r| = \mu - 1$. We continue the process until we end up in a word of length less than $|\Gamma|^\mu$. \square

In [23] Rips and Sela have shown that solvability of equations in hyperbolic groups is decidable. Their techniques rely on the notion of *canonical representative*. This is a representation of an element of G as an element over the free

group $F(\Sigma)$ (in Γ^* resp.) satisfying some “invariants”. In particular, if $\theta(g)$ is a canonical representative of $g \in G$ then $g = \pi(\theta(g))$. We do not need the explicit definition of a canonical representative, but we need some crucial properties. The following result can be deduced from [23] in a very similar way as done by Dahmani in [5, Prop. 3.4] for relatively hyperbolic groups. Since the constants are different (and as they rely on the PhD thesis [3]) we give a proof which refers to [23], only. A statement as in Lemma 5.2 is not needed in [23] since the authors study systems of equations without inequalities, only.

Lemma 5.2 *Canonical representatives (in the sense of [23]) of elements of G are (λ, d) -quasi-geodesics for some $\lambda, d \in |\Sigma|^{\mathcal{O}(\delta)}$.*

Proof. We follow the notation in [23]. For vertices x, y in the Cayley graph $\text{Cay}(G, \Sigma)$ we let $d(x, y)$ be its geodesic distance and $|x| = d(x, 1)$. We let $w \in \Gamma^*$ be some “canonical representative” of $\pi(w)$ in the sense of [23]. Moreover, let K be the 2δ neighborhood of the path in $\text{Cay}(G, \Sigma)$ defined by some geodesic γ connecting 1 and $\pi(w)$ in $\text{Cay}(G, \Sigma)$. Let $u \in \Gamma^*$ be a factor of w . Hence $w = puq$. Define vertices $\pi(p)$ and $\pi(pu)$ in $\text{Cay}(G, \Sigma)$. Then there are vertices $x, y \in K$ and so-called “slices” $S(x)$ and $S(y)$ with centers x and y such that $d(\pi(p), x) \leq 10\delta$ and $d(y, \pi(pu)) \leq 10\delta$. By definition of canonical representatives we have $|u| \leq 20\delta n + 20\delta$, where $n = |\text{diff}_w(x, y)|$, because then the number of “slices” between x and y is at most n . Here $\text{diff}_w(x, y)$ is the “difference function” applied to (x, y) . It remains to show that $n \leq \lambda d(x, y) + d$ with $\lambda, d \in |\Sigma|^{\mathcal{O}(\delta)}$. According to [23, Def. 3.3] the number $\text{diff}_w(x, y)$ is the difference between two non-negative numbers where each of these numbers is the addition of two non-negative terms. Moreover, there is some so-called “cylinder” C such that, by symmetries in x and y and in “left” and “right”, we may assume $n/2 \leq L(y) \setminus L(x)$ where

$$L(z') = \{ z \in C \mid |z| \leq |z'| \wedge d(z, z') \geq 10\delta \}. \quad (1)$$

By [23, Lem. 3.2] we have $C \subseteq K$. Hence, by (1)

$$n/2 \leq |\{ z \in K \mid |x| - 10\delta < |z| \leq |y| \}|. \quad (2)$$

Indeed, if $z \in C$ with $|x| - 10\delta \geq |z|$ then $d(z, x) \geq 10\delta$ and $z \in L(x)$. Clearly, $|z| > |y|$ implies $z \notin L(y)$ for all z . This shows (2). Since $x, y \in K$ there are $x', y' \in \gamma$ such that $d(x, x') \leq 2\delta$ and $d(y, y') \leq 2\delta$. In particular, $d(x', y') \leq d(x, y) + 4\delta$. Moreover,

$$\{ z \in K \mid |x| - 10\delta < |z| \leq |y| \} \subseteq \{ z \in K \mid |x'| - 12\delta < |z| \leq |y'| + 2\delta \}.$$

Now, let $z \in K$ and $z' \in \gamma$ such that $d(z, z') \leq 2\delta$ and $|z'| \leq |x'| - 14\delta$ or $|z'| > |y'| + 4\delta$ then $|z| \leq |x| - 10\delta$ or $|z| > |y|$. We conclude that for all $z \in L(y) \setminus L(x)$ there is some $z' \in \gamma$ with $|x'| - 14\delta \leq |z'| \leq |y'| + 4\delta$ such that $d(z, z') \leq 2\delta$. This implies

$$n/2 \leq |\Gamma|^{2\delta}(d(x', y') + 18\delta) \leq |\Gamma|^{2\delta}d(x, y) + |\Gamma|^{2\delta}22\delta. \quad (3)$$

The result follows. \square

Theorem 5.3 *There exists a polynomial $p(n)$ such that the following assertion holds: Let \mathcal{S} be a system of equations over a δ -hyperbolic group generated by Σ and let σ be a solution of length N . Then there exists another solution σ' of length in $|\Sigma|^{\mathcal{O}(\delta)}N$ and some SLP of size $p(|\Sigma|^{\delta^2 \log \delta} + \|\mathcal{S}\| + \log N)$ such that $\sigma'(X) = \text{eval}(X)$ for all variables used by σ' .*

Proof. By standard arguments we may assume that \mathcal{S} is given by n triangular equations \mathcal{S} of type $XYZ = 1$ and constraints $X = a$ where $X, Y, Z \in \Omega$ and $a \in \Gamma$. The solution σ is given by some mapping $\sigma : \Omega_+ \rightarrow \Gamma^*$ and we may assume that $\sigma(X)$ is geodesic for all $X \in \Omega$ because this cannot increase the length N . Now, [23] yields an effective constant κ depending on δ and $|\Gamma|$ and the following transformation of \mathcal{S} .

- With the help of fresh variables, each equation $XYZ = 1$ of \mathcal{S} is replaced by three equations

$$x = PA\bar{Q}, \quad y = QB\bar{R}, \quad z = RC\bar{P}.$$

- A constraint $X = a$ is replaced by the constraint $X = \theta(a)$ where $\theta(a)$ is some canonical representative of the letter a .
- The following conditions are added:
 - “ $ABC = 1$ in G and $\max\{|A|, |B|, |C|\} \leq \kappa n$ ”.

[23] shows that it is possible to choose canonical representatives $\theta(x)$ for all $x \in \sigma(\Omega) \cup \Gamma$ such that $\rho_\sigma(X) = \theta(\sigma(X))$ defines a solution $\rho_\sigma : \Omega_+ \rightarrow \Gamma^*$ for the new system over the free group $F(\Sigma)$. Moreover, if ρ' is any solution which respects the additional conditions and which solves the new system over the free group $F(\Sigma)$ then ρ' solves \mathcal{S} over G , too. By Lemma 5.2 we know that the length of the solution ρ_σ can be bounded by $|\Sigma|^{\mathcal{O}(\delta)}N$ which is the first assertion in the theorem. The new system has a size which can be bounded by $\kappa n \|\mathcal{S}\| \leq \kappa \|\mathcal{S}\|^2$. The next step is to replace ρ_σ by some minimal solution σ' for the system over the free group $F(\Sigma)$ and hence for the original system \mathcal{S} . The switch to σ' does not increase the length with respect to ρ_σ , but it allows to use Theorem 4.4. It yields a polynomial p and an SLP for σ' of size $p(\kappa + \|\mathcal{S}\| + \delta \log |\Sigma| + \log N)$ such that $\sigma'(X) = \text{eval}(X)$ for all variables used by σ' . It remains to estimate κ by some polynomial in $|\Sigma|^{\delta^2 \log \delta}$. This is done in Lemma 5.4. \square

Lemma 5.4 *The constant κ in the proof of Theorem 5.3 can be estimated by $\kappa \in |\Gamma|^{\mathcal{O}(\delta^2 \log \delta)}$.*

Proof. The constant κ appears in [23] as a product of a function $f(\delta) \in |\Gamma|^{\mathcal{O}(\delta)}$ times $\text{Ca}(\mu_0)$. Here $\text{Ca}(\mu_0)$ is an upper bound on the number of geodesics in a 2δ -neighborhood of a geodesic of length μ_0 where $\mu_0 \in \mathcal{O}(\delta^2 \log \delta)$ by [23, Def. 3.1]. Note that a geodesic contributing to $\text{Ca}(\mu_0)$ can have length at most $4\delta + \mu_0$. The size of such a neighborhood U is therefore at most $\mu_0 |\Gamma|^{2\delta}$. In [23] a doubly exponential bound for κ is used because [23] simply counts the number of all subsets of U . This number is greater than 2^{2^δ} . However, a more accurate counting is possible. Let us fix a starting point of a geodesic of length

at most $4\delta + \mu_0$. Then the geodesic can be described by a word in Γ^* of length $4\delta + \mu_0$ or its prefix of length $4\delta + \mu_0 - 1$. The number of words of length μ_0 is $|\Gamma|^{4\delta + \mu_0}$. This gives us the bound $\text{Ca}(\mu_0) \leq 2\mu_0 |\Gamma|^{2\delta} |\Gamma|^{4\delta + \mu_0}$. Hence, $\kappa = f(\delta) \cdot \text{Ca}(\mu_0) \in |\Gamma|^{\mathcal{O}(\delta^2 \log \delta)}$. \square

In Theorem 5.3 we did not treat inequalities because at present we have a worse estimation w.r.t. compression by SLPs. We obtain a parameter which is unfortunately double-exponential in δ . We can prove the following result.

Corollary 5.5 *There exists a polynomial $p(n)$ such that the following assertion holds: Let Φ be a Boolean formula of equations over a δ -hyperbolic group generated by Σ and let σ be a solution of length N . Then there exists another solution σ' of length in $|\Sigma|^{\mathcal{O}(\delta)} N$ and some SLP of size $p(2^{2^{\mathcal{O}(\delta \log \Sigma)}} + \|\Phi\| + \log N)$ such that $\sigma'(X) = \text{eval}(X)$ for all variables used by σ' .*

Proof. The proof is almost identical to the proof of Theorem 5.3. The additional difficulty is that we cannot replace the solution in canonical representatives by another solution over the free group. The problem is that $\rho(X) \neq 1$ in $F(\Sigma)$ does not transfer to $\rho(X) \neq 1$ in G . We know however by construction that $\rho_\sigma(X) \neq 1$ in G as soon as $\sigma(X) \neq 1$ in G . We also know to construct the generic solution $\widetilde{\rho}_\sigma$ as explained in Section 3.2. This solution has an SLP compression of polynomial size in $|\Sigma|^{\delta^2 \log \delta} + \|\Phi\| + \log N$ by Theorem 3.6 and (the proof of) Theorem 5.3. Our intention is to compress ρ_σ ; and for that we must consider maximal free intervals. The canonical representations $\rho_\sigma(X)$ are (λ, d) -quasi-geodesic for some $\lambda, d \in |\Sigma|^{\mathcal{O}(\delta)}$ by Lemma 5.2. Assume that we have $\rho_\sigma(X) = puq$ where u corresponds to some maximal free interval in $\widetilde{\rho}_\sigma$ and $\sigma(X) \neq 1$ in G . We know $\rho_\sigma(X) \neq 1$ in G . But the problem is that u might be long and incompressible. For some $\mu \in |\Sigma|^{\mathcal{O}(\delta)}$ every μ -local (λ, d) -quasi-geodesic is in fact a (λ', d') -geodesic with $\mu > d' + 1$. Hence, if we choose v such that pvq is μ -locally (λ', d') -quasi-geodesic and $|v| > d'$ then $pvq \neq 1$ in the group G . We care only if $|u| > |\Gamma|^\mu$. In this case we use Lemma 5.1 and we let $v \in \Gamma^*$ with $\mu - 1 \leq |v| < |\Gamma|^\mu$ such that for all x, y we have that xvy is μ -locally (λ, d) -quasi-geodesic if and only if xvy is μ -locally (λ, d) -quasi-geodesic. This allows to substitute the maximal free interval belonging to u by the word v . It might be that v is not compressible, but at least we have a length bound on v . Iterating this process we obtain a new solution σ' satisfying the following conditions for all $X \in \Omega$.

- Every factor in $\sigma'(X)$ which belongs to some maximal free interval has length less than $|\Gamma|^\mu$.
- The word $\sigma'(X)$ is μ -locally (λ, d) -quasi-geodesic.
- If $\sigma'(X) \neq \sigma(X)$ then $|\sigma'(X)| > d'$. In particular, $\sigma'(X) \neq 1 \neq \sigma(X)$ in the group G .

The SLP for σ' can be constructed from the SLP for the generic solution $\widetilde{\rho}_\sigma$ and writing all substitutions for maximal free intervals as plain words of length less than $|\Gamma|^\mu$. We have $|\Gamma|^\mu \in 2^{\Sigma^{\mathcal{O}(\delta)}}$. Hence the result. \square

Dahmani has shown that the existential theories of equations for hyperbolic groups are decidable, see [5]. He does not mention explicit complexity bounds. Therefore, we add the following result.

Proposition 5.6 *Let G be a finitely generated torsion-free δ -hyperbolic group. Then the existential theory of equations over G is in PSPACE.*

Proof. Since G is fixed all parameters in $2^{\Sigma^{O(\delta)}}$ become constants. By [23] and the methods used in the proofs of Corollary 5.5 we obtain an NP-reduction of the existential theory of equations over G to the existential theory of equations with rational constraints in a fixed free finitely generated free group $F(\Sigma)$. The later theory is in PSPACE by [7]. \square

Remark 5.7 We believe that Proposition 5.6 holds for also for hyperbolic groups with torsion. But we did not check enough details in [6] in order to make this statement rigorous. The reduction in the proof of Proposition 5.6 to the existential theory of equations with rational constraints in $F(\Sigma)$ creates only rational constraints which involve finite monoids of polynomial size of the input. This is due to the local character to test the constraint of being μ -locally (λ, d') -quasi-geodesic where (for a fixed group G) the values μ, λ, d are constants. As also supported by Corollary 5.5 we conjecture that the existential theory of equations in a fixed finitely generated (torsion-free) δ -hyperbolic group G is in NP. As soon as G contains a non-abelian free subgroup the problem is known to be NP-hard (even for systems of quadratic equations) by a recent result in [14].

6 Toral relatively hyperbolic groups

In this section we will obtain results similar to the results of the previous section for systems of equations in toral relatively hyperbolic groups using the work of Dahmani [5]. We will use the following definition of relative hyperbolicity. A f.g. group G with generating set Σ is relatively hyperbolic relative to a collection of finitely generated subgroups $\mathcal{P} = \{P_1, \dots, P_k\}$ if the Cayley graph $\text{Cay}(G, \Sigma \cup \Pi)$ (where Π is the set of all non-trivial elements of subgroups in \mathcal{P}) is a hyperbolic metric space, and the pair $\{G, \mathcal{P}\}$ has *Bounded Coset Penetration* property (BCP property for short). The pair $(G, \{P_1, P_2, \dots, P_k\})$ satisfies the *BCP property*, if for any $\lambda \geq 1$, there exists constant $a = a(\lambda)$ such that the following conditions hold. Let p, q be $(\lambda, 0)$ -quasi-geodesics without backtracking in $\text{Cay}(G, \Sigma \cup \Pi)$ such that their initial points coincide ($p_- = q_-$), and for the terminal points p_+, q_+ we have $\text{dist}_\Sigma(p_+, q_+) \leq 1$.

1) Suppose that for some i , s is a P_i -component of p such that $\text{dist}_\Sigma(s_-, s_+) \geq a$; then there exists a P_i -component t of q such that t is connected to s (there exists a path c in $\text{Cay}(G, \Sigma \cup \Pi)$ that connects some vertex of p to some vertex of q and the label of this path is a word consisting of letters from P_i).

2) Suppose that for some i , s and t are connected P_i -components of p and q respectively. Then $\text{dist}_\Sigma(s_-, t_-) \leq a$ and $\text{dist}_\Sigma(s_+, t_+) \leq a$.

A group G that is hyperbolic relative to a collection $\{P_1, \dots, P_k\}$ of subgroups is called *toral*, if P_1, \dots, P_k are all abelian and G is torsion-free. In this section we always assume that Σ contains generators of all subgroups P_1, \dots, P_k .

In [5] Dahmani has shown that the satisfiability of systems of equations and inequalities is decidable in toral relatively hyperbolic groups. He also uses the notion of canonical representatives, the canonical representatives in this case are elements of the free product $\tilde{G} = F(\Sigma) * P_1 * \dots * P_k$. In [5], Section 2.4.2 the language \mathcal{L} of so called *geometric* elements in \tilde{G} is introduced. These are elements $\tilde{\gamma} \in \tilde{G}$ that do not have any θ -detour such that $\pi(\tilde{\gamma})$ in a L -local (L_1, L_2) -quasi-geodesic in $\text{Cay}(G, \Sigma \cup \Pi)$. The constants are defined in [5], Section 2.4.2, as $L_1 = 10^4 \delta M, L_2 = 10^6 \delta^2 M$, where M is a bound on the cardinality of cones of radius and angle 50δ .

Lemma 6.1 *The cardinality of a cone of radius and angle ℓ is bounded by $C(\ell)^\ell$, where $C(\ell)$ is the number of circuits in $\text{Cay}(G, \Sigma \cup \Pi)$ of length less than ℓ . Moreover $C \leq |\Gamma|^{6(a(\ell)+1)a(\ell)}$, where $a(\ell)$ is the BCP constant for the group.*

Proof. The proof of the first statement repeats the proof of [4], Corollary 1.7.

Now we have to estimate the constant $C(\ell)$ in terms of $a(\ell)$. This can be done using [3], Proposition 1 in the Appendix. This proposition shows that each circuit of length ℓ in $\text{Cay}(G, \Sigma \cup \Pi)$ is formed by two ℓ -quasi-geodesics both belonging to a fixed ball of radius $\ell(a(\ell) + 1)$. Therefore, the number of such circuits is bounded by $|\Gamma|^{6(a(\ell)+1)\ell}$. \square

By this lemma, $C(50\delta) \leq |\Gamma|^{6(a(50\delta)+1)a(50\delta)}$. And $M \leq |\Gamma|^{300\delta(a(50\delta)+1)a(50\delta)}$. The angle θ can be taken as $10^4(D + 60\delta)$, where D is a fellow traveling constant for 1000δ -quasi-geodesics, greater than any angles at finite valency vertices. Therefore [5], Proposition 3.4 implies

Lemma 6.2 *Canonical representatives (in the sense of [5]) of elements of G are (λ, d) -quasi-geodesics for some $\lambda, d \in |\Sigma|^{\mathcal{O}(\delta(a(50\delta))^2)}$.*

Theorem 6.3 *There exist polynomials $p(n), q(n)$ such that the following assertion holds. Let \mathcal{S} be a system of equations and a toral relatively hyperbolic group with hyperbolicity constant δ for $\text{Cay}(G, \Sigma \cup \Pi)$ and BCP function $a(\ell)$, generated by Σ . Let σ be a solution of length N . Then there exists another solution σ' of length in $|\Sigma|^{\mathcal{O}(\delta a(50\delta))} N$ and some SLP of size $p(|\Sigma|^{q(\delta a(\delta^3))} + \|\mathcal{S}\| + \log N)$ such that $\sigma'(X) = \text{eval}(X)$ for all variables used by σ' .*

Proof. By standard arguments we may assume that \mathcal{S} is given by n triangular equations \mathcal{S} of type $XYZ = 1$ and constraints $X = a$ where $X, Y, Z \in \Omega$ and $a \in \Gamma$. The solution σ is given by some mapping $\sigma : \Omega_+ \rightarrow \Gamma^*$ and we may assume that $\sigma(X)$ is geodesic for all $X \in \Omega$ because this cannot increase the length N . Now, [5] yields an effective constant κ depending on δ and $|\Gamma|$ and the following transformation of \mathcal{S} .

- With the help of fresh variables, each equation $XYZ = 1$ of \mathcal{S} is replaced by three equations

$$x = PA\bar{Q}, \quad y = QB\bar{R}, \quad z = RC\bar{P}.$$

- A constraint $X = a$ is replaced by the constraint $X = \theta(a)$ where $\theta(a)$ is some canonical representative of the letter a .

- The following conditions are added:

$$- \text{“}ABC = 1 \text{ in } G \text{ and } \max \{ |A|, |B|, |C| \} \leq \kappa n \text{”}.$$

Let us show that κ is exponential in δ and $|\Sigma|$. To estimate κ we have to estimate the function $\varphi(n)$ in [4, Thm. 2.22], because κn has the order of $\varphi(n)$, size of the holes in the slice decomposition of a cylinder, see [4, Sec. 2.4]. The function $\varphi(n)$ is defined in [4, Sec. 2.3], as well as all necessary constants,

$$\varphi(n) = 24(n + 1) \text{Capa}(\mu)(2\epsilon + 1)\epsilon,$$

where $\epsilon = N_{1000\delta,\delta}$ that has the order of δ^3 , $\mu = 100N_{1000\delta,\delta} + (1000\delta)^2$ also has the order of δ^3 and $\text{Capa}(\mu)$ is the number of different channels of segments of length μ . If $g = [v_1, v_2]$ is a segment of length μ then we have to estimate the number of geodesics not shorter than $|v_2 - v_1|$ that stay in the union of the cones of radius and angle ϵ centered in the edges of g . By Lemma 6.1, the cardinality of such a cone is bounded by $C(\epsilon)^\epsilon$. The number of geodesics in one such cone is bounded by $C(\epsilon)^\epsilon$ times the bound on the number of paths of length $\leq 2\epsilon$. The number of paths is bounded by $m^{2\epsilon} = 2^{(\log m)2\epsilon}$. Therefore, the number of channels of a segment of length μ is bounded by $C(\epsilon)^{\epsilon\mu} 2^{(\log m)2\epsilon\mu}$.

Finally $\text{Capa}(\mu) \leq 2^{6(\log m)\epsilon\mu(a(\epsilon)+1)^2}$. This gives the desired estimate for $\varphi(n)$ and κ . □

References

- [1] S. Alstrup, G. S. Brodal, and T. Rauhe. Pattern matching in dynamic texts. In D. B. Shmoys, editor, *SODA*, pages 819–828. ACM/SIAM, 2000.
- [2] M. Coornaert, T. Delzant, and A. Papadopoulos. *Géométrie et théorie des groupes. Les groupes hyperboliques de M. Gromov*, volume 1441 of *Lecture Notes in Mathematics*. Springer, 1991.
- [3] F. Dahmani. *Les groupes relativement hyperboliques et leurs bords*. PhD thesis, Université Louis Pasteur, Strasbourg, 2003.
- [4] F. Dahmani. Accidental parabolics and relatively hyperbolic groups. *Israel Journal of Mathematics*, 153:93–127, 2006.
- [5] F. Dahmani. Existential questions in (relatively) hyperbolic groups. *Israel Journal of Mathematics*, 173:91–124, 2009.
- [6] F. Dahmani and V. Guirardel. Foliations for solving equations in groups: free, virtually free and hyperbolic groups. *J. of Topology*, 3:343–404, 2010.
- [7] V. Diekert, C. Gutiérrez, and Ch. Hagenah. The existential theory of equations with rational constraints in free groups is PSPACE-complete. *Information and Computation*, 202:105–140, 2005. Conference version in STACS 2001, LNCS 2010, 170–182, 2004.
- [8] L. Gasieniec, M. Karpinski, W. Plandowski, and W. Rytter. Efficient algorithms for Lempel-Zip encoding (Extended abstract). In R. G. Karlsson and A. Lingas, editors, *SWAT*, volume 1097 of *Lecture Notes in Computer Science*, pages 392–403. Springer, 1996.

- [9] M. Gromov. Hyperbolic groups. In S. M. Gersten, editor, *Essays in Group Theory*, number 8 in MSRI Publ., pages 75–263. Springer-Verlag, 1987.
- [10] Ch. Hagenah. *Gleichungen mit regulären Randbedingungen über freien Gruppen*. Ph.d.-thesis, Institut für Informatik, Universität Stuttgart, 2000.
- [11] A. Jez. Faster fully compressed pattern matching by recompression. In A. Czumaj, K. Mehlhorn, A. M. Pitts, and R. Wattenhofer, editors, *ICALP (1)*, volume 7391 of *Lecture Notes in Computer Science*, pages 533–544. Springer, 2012.
- [12] A. Jez. Recompression: Word equations and beyond. In M.-P. Béal and O. Carton, editors, *Developments in Language Theory*, volume 7907 of *Lecture Notes in Computer Science*, pages 12–26. Springer, 2013.
- [13] O. Kharlampovich, I. Lysénok, A. Myasnikov, and N. Touikan. The solvability problem for quadratic equations over free groups is NP-complete. *Theory of Computing Systems*, 47:250–258, 2010.
- [14] O. Kharlampovich, A. Mohajeri, A. Taam, and A. Vdovina. Quadratic Equations in Hyperbolic Groups are NP-complete. *ArXiv e-prints*, 2013.
- [15] D. Kozen. Lower bounds for natural proof systems. In *Proc. of the 18th Ann. Symp. on Foundations of Computer Science, FOCS’77*, pages 254–266, Providence, Rhode Island, 1977. IEEE Computer Society Press.
- [16] M. Lohrey. Word problems and membership problems on compressed words. *SIAM J. Comput.*, 35:1210–1240, 2006.
- [17] M. Lohrey. Algorithmics on SLP-compressed strings: A survey. *Groups Complexity Cryptology*, 4:241–299, 2012.
- [18] M. Lohrey and S. Schleimer. Efficient computation in groups via compression. In V. Diekert, M. V. Volkov, and A. Voronkov, editors, *CSR*, volume 4649 of *Lecture Notes in Computer Science*, pages 249–258. Springer, 2007.
- [19] M. Lothaire. *Algebraic Combinatorics on Words*, volume 90 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 2002.
- [20] K. Mehlhorn, R. Sundar, and C. Uhrig. Maintaining dynamic sequences under equality tests in polylogarithmic time. *Algorithmica*, 17(2):183–198, 1997.
- [21] W. Plandowski. Testing equivalence of morphisms on context-free languages. In J. van Leeuwen, editor, *Proc. Algorithms—ESA’94*, volume 855 of *Lecture Notes in Computer Science*, pages 460–470, Utrecht, The Netherlands, 1994. Springer.
- [22] W. Plandowski and W. Rytter. Application of Lempel-Ziv encodings to the solution of word equations. In K. G. Larsen et al., editors, *Proc. 25th International Colloquium Automata, Languages and Programming (ICALP’98), Aalborg (Denmark), 1998*, number 1443 in *Lecture Notes in Computer Science*, pages 731–742, Heidelberg, 1998. Springer-Verlag.

- [23] E. Rips and Z. Sela. Canonical representatives and equations in hyperbolic groups. *Inventiones Mathematicae*, 120:489–512, 1995.
- [24] S. Schleimer. Polynomial-time word problems. *Commentarii Mathematici Helvetici*, 83:741–765, 2008.
- [25] K. U. Schulz. Makanin’s algorithm for word equations — Two improvements and a generalization. In K. U. Schulz, editor, *Word Equations and Related Topics*, number 572 in Lecture Notes in Computer Science, pages 85–150, Heidelberg, 1991. Springer-Verlag.