Solution sets for equations over free groups are EDT0L languages*

Laura Ciobanu[1], Volker Diekert[2], and Murray Elder[3]

[1] Institut de mathématiques, Université de Neuchâtel, Switzerland
[2] Institut für Formale Methoden der Informatik, Universität Stuttgart, Germany
[3] School of Mathematical & Physical Sciences, The University of Newcastle, Australia

Dedicated to Manfred Kudlek[4] (1940 – 2012)

Abstract We show that, given a word equation over a finitely generated free group, the set of all solutions in reduced words forms an EDT0L language. In particular, it is an indexed language in the sense of Aho. The question of whether a description of solution sets in reduced words as an indexed language is possible has been been open for some years [9, 10], apparently without much hope that a positive answer could hold. Nevertheless, our answer goes far beyond: they are EDT0L, which is a proper subclass of indexed languages. We can additionally handle the existential theory of equations with rational constraints in free products $\star_{1\leq i\leq s}F_i$, where each F_i is either a free or finite group, or a free monoid with involution. In all cases the result is the same: the set of all solutions in reduced words is EDT0L. This was known only for quadratic word equations by [8], which is a very restricted case. Our general result became possible due to the recent recompression technique of Jeż. In this paper we use a new method to integrate solutions of linear Diophantine equations into the process and obtain more general results than in the related paper [5]. For example, we improve the complexity from quadratic nondeterministic space in [5] to quasi-linear nondeterministic space here. This implies an improved complexity for deciding the existential theory of non-abelian free groups: $\mathsf{NSPACE}(n \log n)$. The conjectured complexity is NP, however, we believe that our results are optimal with respect to space complexity, independent of the conjectured NP.

Introduction

The first algorithmic description of all solutions to a given equation over a free group is due to Razborov [17, 18]. His description became known as a *Makanin-Razborov diagram*. This concept plays a major role in the positive solution of Tarski's conjectures about the elementary theory in free groups [12, 21].

* Research supported by the Australian Research Council FT110100178 and the University of Newcastle G1301377. The first author was supported by a Swiss National Science Foundation Professorship FN PP00P2-144681/1. The first and third authors were supported by a University of Neuchâtel Overhead grant in 2013.

[4] Manfred Kudlek has the distinction of being the only person to have attended all ICALP conferences during his lifetime. He worked on Lindenmayer systems, visited Kyoto several times, and taught the second author that bikes are the best means of transport inside Kyoto.

It was however unknown that there is an amazingly simple formal language description for the set of all solutions of an equation over free groups in reduced words: they are EDT0L. An EDT0L language L is given by a nondeterministic finite automaton (NFA), where transitions are labeled by endomorphisms in a free monoid which contains a symbol #. Such an NFA defines a rational language $\mathcal{R}$ of endomorphisms, and the condition on L is that $L = \{h(\#) \mid h \in \mathcal{R}\}$. The NFA we need for our result can be computed effectively in nondeterministic quasi-linear space, i.e., by some $\mathsf{NSPACE}(n \log n)$ algorithm. As a consequence, the automaton has singly exponential size $2^{\mathcal{O}(n \log n)}$ in the input size n.

A description of solution sets as EDT0L languages was known before only for quadratic word equations by [8]; the recent paper [5] did not aim at giving such a structural result. There is also a description of all solutions for a word equation by Plandowski in [14]. His description is given by some graph which can computed in singly exponential time, but without the aim to give any formal language characterization. Plandowski claimed in [14] that his method applies also to free groups with rational constraints, but he found a gap [15].

The technical results are as follows. Let $\mathrm{F}(A_+)$ be the free group over a finite generating set A_+ of (positive) letters. We let $A_\pm = A_+ \cup \left\{a^{-1} \;\middle|\; a \in A_+\right\} \subseteq \mathrm{F}(A_+)$. We view $A_\pm$ as a finite alphabet (of *constants*) with the involution $\overline{a} = a^{-1}$. The involution is extended to the free monoid $A_\pm^*$ by $\overline{a_1 \cdots a_k} = \overline{a_k} \cdots \overline{a_1}$. We let $\pi : A_\pm^* \to \mathrm{F}(A_+)$ be the canonical morphism. As a set, we identify $\mathrm{F}(A_+)$ with the rational (i.e., regular) subset of reduced words inside $A_\pm^*$. A word is *reduced* if it does not contain any factor $a\overline{a}$ where $a \in A_\pm$. Thus, $w \in A_\pm^*$ is reduced if and only if $\pi(w) = w$. We emphasize that $\mathrm{F}(A_+)$ is realized as a subset of $A_\pm^*$. Let Ω be a set of *variables* with involution. An *equation* over $\mathrm{F}(A_+)$ is given as a pair (U, V), where $U, V \in (A_\pm \cup \Omega)^*$ are words over constants and variables. A *solution* of (U, V) is a mapping $\sigma : \Omega \to A_\pm^*$ which respects the involution such that $\pi\sigma(U) = \pi\sigma(V)$ holds in $\mathrm{F}(A_+)$. As usual, σ is extended to a morphism $\sigma : (A_\pm \cup \Omega)^* \to A_\pm^*$ by leaving constants invariant. Throughout we let # denote a special symbol, whose main purpose is to encode a tuple of words $(w_1, \ldots, w_k)$ as a single word $w_1\# \cdots \#w_k$.

Theorem 1. *Let (U, V) be an equation over $F(A_+)$ and $\{X_1, \ldots, X_k\}$ be any specified subset of variables. Then the solution set* $\mathrm{Sol}(U, V)$ *is EDT0L where* $\mathrm{Sol}(U, V) = \{\sigma(X_1)\# \cdots \#\sigma(X_k) \mid \sigma$ *solves* (U, V) *in reduced words*$\}$.

Moreover, there is a nondeterministic algorithm which takes (U, V) as input and computes an NFA $\mathcal{A}$ such that $\mathrm{Sol}(U, V) = \{\varphi(\#) \mid \varphi \in L(\mathcal{A})\}$ *in quasi-linear space.*

The statement of Theorem 1 shifts the perspective on how to solve equations. Instead of solving an equation, we focus on an effective construction of some NFA producing the EDT0L set. Once the NFA is constructed, the existence of a solution, or whether the number of solutions is zero, finite or infinite, become graph properties of the NFA.

Theorem 1 is a special case of a more general result involving the existential theory with rational constraints over free products. The generalization is done

in several directions. First, we can replace $\mathrm{F}(A_+)$ by any finitely generated free product $\mathbb{F} = \star_{1\leq i\leq s} F_i$ where each F_i is either a free or finite group, or a free monoid with arbitrary involutions (including the identity). Thus, for example we may have $\mathbb{F} = \{a,b\}^* \star \mathbb{Z} \star \mathrm{PSL}(2,\mathbb{Z}) = \{a,b\}^* \star \mathbb{Z} \star (\mathbb{Z}/3\mathbb{Z}) \star (\mathbb{Z}/2\mathbb{Z})$ where $\overline{a} = a$ and $\overline{b} = b$. Second, we allow arbitrary rational constraints. We consider Boolean formulae Φ, where each atomic formula is either an equation or a *rational constraint*, written as $X \in L$, where $L \subseteq \mathbb{F}$ is a rational subset.

Theorem 2. *Let $\mathbb{F}$ be a free product as above, Φ a Boolean formula over equations and rational constraints, and $\{X_1, \ldots, X_k\}$ any subset of variables. Then $\mathrm{Sol}(\Phi) = \{\sigma(X_1)\# \cdots \#\sigma(X_k) \mid \sigma$ solves Φ in reduced words$\}$ is EDT0L.*

Moreover, there is an algorithm which takes Φ as input and produces an NFA $\mathcal{A}$ such that $\mathrm{Sol}(\Phi) = \{\varphi(\#) \mid \varphi \in L(\mathcal{A})\}$. The algorithm is nondeterministic and uses quasi-linear space in the input size $\|\Phi\|$.

For lack of space we present the main steps used to show Theorem 1, only. However, this covers the essential ideas to prove the more general result in Theorem 2 as well. All missing proofs and details are in our paper on the arXiv.

Preliminaries

The notion of a *rational set* is defined in any monoid, and a rational set can be specified by some NFA with arcs labeled by monoid elements, see [7]. Traditionally, rational sets in finitely generated free monoids are also called *regular*. If M is a monoid and $u, v \in M$, then we write $u \leq v$ if u is a *factor* of v, which means we can write $v = xuy$ for some $x, y \in M$. We denote the neutral element in M by 1, thus, the empty word is also 1. The length of word w is denoted by $|w|$, and $|w|_a$ counts how often a letter a appears in w. An *involution* of a set A is a mapping $x \mapsto \overline{x}$ such that $\overline{\overline{x}} = x$ for all $x \in A$. For example, the identity map is an involution. A *morphism* between sets with involution is a mapping respecting the involution. A *monoid with involution* has to additionally satisfy $\overline{xy} = \overline{y}\,\overline{x}$. A *morphism* between monoids with involution is a homomorphism $\varphi : M \to N$ such that $\varphi(\overline{x}) = \overline{\varphi(x)}$. It is a *$\Delta$-morphism* if $\varphi(x) = x$ for all $x \in \Delta$ where $\Delta \subseteq M$. In this article, whenever the term "morphism" is used it refers to a mapping which respects the underlying structure including the involution. All groups are monoids with involution given by $\overline{x} = x^{-1}$, and all group-homomorphisms are morphisms. Any involution on a set A extends to A^*: for a word $w = a_1 \cdots a_m$ we let $\overline{w} = \overline{a_m} \cdots \overline{a_1}$. If $\overline{a} = a$ for all $a \in A$ then $\overline{w}$ is simply the word w read from right-to-left. The monoid A^* is called a *free monoid with involution*.

The notion of an *EDT0L system* refers to **E***xtended,* **D***eterministic,* **T***able,* **0** *interaction, and* **L***indenmayer*. There is a vast literature on Lindenmayer systems, see [19], with various acronyms such as D0L, DT0L, ET0L, and HDT0L. The subclass EDT0L is equal to HDT0L (see e.g. [20, Thm. 2.6]), and has received particular attention. We content ourselves to define EDT0L through a characterization (using rational control) due to Asveld [2]. The class of EDT0L languages is a proper subclass of indexed languages in the sense of [1], see [6]. For more background we refer to [20].

Definition 1. *Let A be an alphabet and $L \subseteq A^*$. We say that L is* EDT0L *if there is an alphabet C with $A \subseteq C$, a rational set of endomorphisms $\mathcal{R} \subseteq \mathrm{End}(C^*)$, and a symbol $\# \in C$ such that $L = \{\varphi(\#) \mid \varphi \in \mathcal{R}\}$.*

Note that for a set $\mathcal{R}$ of endomorphisms of C^* we have $\{\varphi(\#) \mid \varphi \in \mathcal{R}\} \subseteq C^*$, in general. Our definition implies that $\mathcal{R}$ must guarantee that $\varphi(\#) \in A^*$ for all $\varphi \in R$. The set $\mathcal{R}$ is the *rational control*, and C is the *extended alphabet*.

Example 1. Let $A = \{a, b\}$ and $C = \{a, b, \#, \$\}$. We let H be the set of four endomorphisms f, g_a, g_b, h satisfing $f(\#) = \$\$$, $g_a(\$) = \$a$, $g_b(\$) = \$b$, and $h(\$) = 1$, and on all other letters the f, g_a, g_b, h behave like the identity. Consider the rational language $\mathcal{R} \subseteq H^*$ defined by $\mathcal{R} = h\{g_a, g_b\}^* f$ (where endomorphisms are applied right-to-left). A simple inspection shows that $\{\varphi(\#) \mid \varphi \in R\} = \{vv \mid v \in A^*\}$, which is not context-free.

Proof of Theorem 1

Preprocessing. We start by adding the special symbol # to the alphabet $A_\pm$, and define $A = A_\pm \cup \{\#\}$. We let $\overline{\#} = \#$; this will be the only self-involuting letter in this article. We must make sure that no solution uses # and every solution is in reduced words. We do so by introducing a finite monoid N with involution which plays the role of (a specific) rational constraint. Let $N = \{1, 0\} \cup A_\pm \times A_\pm$. We define a multiplication on N by $1 \cdot x = x \cdot 1 = x$, $0 \cdot x = x \cdot 0 = 0$, and

$$(a, b) \cdot (c, d) = \begin{cases} (a, d) & \text{if } b \neq \overline{c} \\ 0 & \text{otherwise.} \end{cases}$$

The monoid N has an involution by $\overline{1} = 1$, $\overline{0} = 0$, and $\overline{(a, b)} = (\overline{b}, \overline{a})$. Consider the morphism $\mu_0 : A^* \to N$ given by $\mu_0(\#) = 0$ and $\mu_0(a) = (a, a)$ for $a \in A_\pm$. It is clear that μ_0 respects the involution and $\mu_0(w) = 0$ if and only if either w contains # or w is not reduced. If, on the other hand, $1 \neq w \in A_\pm^*$ is reduced, then $\mu_0(w) = (a, b)$, where a is the first and b the last letter of w. Thus, if σ is a solution in reduced words, then for each variable $X \in \Omega$ there exists some element $\mu(X) \in N$ with $0 \neq \mu(X) = \overline{\mu(\overline{X})} \in N$ and $\mu(X) = \mu_0\sigma(X)$. Note that while rational constraints are not explicitly mentioned in Theorem 1, they play an essential role in ensuring that solutions are in reduced words.

Since EDT0L is closed under finite union, and since there are only finitely many choices for $\mu(X)$, we may assume that our input equation is specified together with a fixed morphism $\mu : \Omega \longrightarrow N$. A solution σ is now given by a mapping $\sigma : \Omega \to A^*$ satisfying three properties: $\pi\sigma(U) = \pi\sigma(V)$ (the equation holds in $\mathrm{F}(A_+)$), $\overline{\sigma(X)} = \sigma(\overline{X})$, ($\sigma$ respects the involution), and $\mu(X) = \mu_0\sigma(X)$ for all $X \in \Omega$.

The next steps are standard, see [4]. With the help of additional variables we produce a system of equations (X_i, V_i), $1 \leq i \leq s$, such that each X_i is a variable and each V_i is a word of length 2. (The number s of equations is in $\mathcal{O}(|UV|)$ after this transformation.) Thus, we obtain a *triangular* system of equations. We may

still assume that each variable X comes with a value $0 \neq \mu(X) = \overline{\mu(\overline{X})}$ and we extend μ to a morphism μ_{init} by $\mu_{\text{init}}(X) = \mu(X)$ for $X \in \Omega$ and $\mu_{\text{init}}(a) = \mu_0(a)$ for each $a \in A$. Next, by the following lemma, we switch to free monoids with involution. Lemma 1 is well-known and easy to see. Its geometric interpretation is the fact that the Cayley graph of a free group is a tree.

Lemma 1. *Let x, y, z be reduced words in $A_{\pm}^*$. Then $x = yz$ in the group $F(A_+)$ if and only if there are reduced words P, Q, R in $A_{\pm}^*$ such that $x = PR$, $y = PQ$, and $z = \overline{Q}R$ hold in the free monoid $A_{\pm}^*$.*

By Lemma 1 we content ourselves to prove the analogue of Theorem 1 for free monoids with involution, systems of equations $(U_i, V_i)_{1 \leq i \leq s}$, and a morphism $\mu_{\text{init}} : (A \cup \Omega)^* \to N$ such that $0 \neq \mu_{\text{init}}(X)$ for all $X \in \Omega$, where Ω is an enlarged set of variables. We return to a single equation (U', V') over $A \cup \Omega$ by letting $U' = U_1 \# \cdots \# U_s$ and $V' = V_1 \# \cdots \# V_s$. Notice that $\mu_{\text{init}}(X) \neq 0$ for all X and $|U_i|_{\#} = |V_i|_{\#} = 0$ for all i. A solution $\sigma : \Omega \to A_{\pm}^*$ must satisfy $\sigma(U') = \sigma(V')$, $\overline{\sigma(X)} = \sigma(\overline{X})$, and $\mu_{\text{init}}(X) = \mu_0\sigma(X)$ for all $X \in \Omega$. The set of variables $\{X_1, \ldots, X_k\}$, specified in Theorem 1, is a subset of Ω, and the original solution set is a finite union of solution sets with respect to different choices for μ_{init}. In order to achieve our result we *protect* each variable X_i by defining a factor $\#X_i\#$ as follows. We assume $A_{\pm} \cup \Omega = \{x_1, \ldots, x_\ell\}$ with $x_i = X_i$ for $1 \leq i \leq k$ where $\{X_1 \ldots, X_k\}$ is the specified subset in the statement of Theorem 1. The word W_{init} over $(A \cup \Omega)^*$ is then defined as:

$$W_{\text{init}} = \#x_1\# \ldots \#x_\ell\#U'\#V'\#\overline{U'}\#\overline{V'}\#\overline{x_\ell}\# \ldots \#\overline{x_1}\#.$$

Observe that W_{init} is longer than (but still linear in) $|A| + |\Omega| + |UV|$. The number of #'s in W_{init} is odd; and if $\sigma : (A \cup \Omega)^* \to A^*$ is a morphism with $\sigma(X)$ reduced for all $X \in \Omega$, then: $\pi\sigma(U) = \pi\sigma(V) \iff \sigma(U') = \sigma(V') \iff \sigma(W_{\text{init}}) = \sigma(\overline{W_{\text{init}}})$. Here (U, V) is the equation in Theorem 1 and (U', V') is the intermediate word equation over $A \cup \Omega$. Therefore Theorem 1 follows by showing that the following language is EDT0L:

$$\left\{ \sigma(X_1)\# \cdots \#\sigma(X_k) \;\middle|\; \sigma(W_{\text{init}}) = \sigma(\overline{W_{\text{init}}}) \wedge \mu_{\text{init}} = \mu_0\sigma \wedge \forall X : \sigma(\overline{X}) = \overline{\sigma(X)} \right\}.$$

Partial commutation and extended equations. Partial commutation is an important concept in our proof. It pops up where traditionally the unary case (solving a linear Diophantine equation) is used as a black box, as is done in [5]. At first glance it might seem like an unneccesary complication, but in fact the contrary holds. Using partial commutation allows us to encode all solutions completely in the edges of a graph, which we can construct in quasi-linear space, and is one of the major differences to [5]. As a (less important) side effect, results on linear Diophantine equations come for free as this is the special case $\mathrm{F}(A_+) = \mathbb{Z}$: solving linear Diophantine equations becomes part of a more general process.

We fix $n = n_{\text{init}} = |W_{\text{init}}|$ and some $\kappa \in \mathbb{N}$ large enough, say $k = 100$. We let C be an alphabet with involution (of constants) such that $|C| = \kappa n$ and

$A \subseteq C$. We define $\Sigma = C \cup \Omega$ and assume that $\#$ is the only self-involuting symbol of Σ. In the following $x, y, z, \ldots$ refer to words in Σ^* and $X, Y, Z, \ldots$ to variables in Ω. Throughout we let B, B' and $\mathcal{X}, \mathcal{X}'$ denote subsets which are closed under involution and satisfy $\mathcal{X}' \subseteq \mathcal{X} \subseteq \Omega$ and either $A \subseteq B \subseteq B' \subseteq C$ or $A \subseteq B' \subseteq B \subseteq C$. In particular, B and B' are always comparable.

We encode partial commutation by *types*. Let $\theta \subseteq (B \cup \mathcal{X}) \times B$ denote an irreflexive and antisymmetric relation. It is called a *type* if $(x, y) \in \theta$ implies:

- $(\overline{x}, \overline{y}) \in \theta$,
- $x, y \notin A$,
- $|\theta(x)| \leq 1$, where $\theta(x) = \{y \in B^* \mid (x, y) \in \theta\}$.

The type relation θ can be stored in quasi-linear space. Given θ and $\mu : B \cup \mathcal{X} \to N$ such that $\mu(xy) = \mu(yx)$ for all $(x, y) \in \theta$, we define a free partially commutative monoid with involution by $M(B, \mathcal{X}, \theta, \mu) = (B \cup \mathcal{X})^* / \{xy = yx \mid (x, y) \in \theta\}$ with a morphism $\mu : M(B, \mathcal{X}, \theta, \mu) \to N$. By $M(B, \theta, \mu)$ we denote the submonoid generated by B with the corresponding restrictions of θ and μ. Note that $M(A, \theta, \mu) = M(A, \emptyset, \mu_0)$ is the free monoid A^*.

If w is a factor of $W \in M(B, \mathcal{X}, \theta, \mu)$, then w is called a *proper factor* if $1 \neq w \neq W$ and $|w|_{\#} = 0$. The numbers $|u|$ and $|u|_a$ are well defined for every $u \in M(B, \mathcal{X}, \theta, \mu)$ since if two words represent the same monoid element then the number of occurences of each letter is the same. Typically we represent w, W by words $w, W \in (B \cup \mathcal{X})^*$, but their interpretation is always in $M(B, \mathcal{X}, \theta, \mu)$.

Definition 2. *We call* $W \in M(B, \mathcal{X}, \theta, \mu)$ well-formed *if* $|W| \leq \kappa n$, $|W|_{\#} = |W_{\text{init}}|_{\#}$, *and every proper factor* x *of* W *and every* $x \in B \cup \mathcal{X}$ *satisfies* $\mu(x) \neq 0$. *In addition, if* x *is a proper factor then* $\overline{x}$ *is also a proper factor and for each* $a \in A_{\pm}$ *there must be a factor* $\#a\#$ *in* W.

An extended equation *is a tuple* $V = (W, B, \mathcal{X}, \theta, \mu)$ *where* $W \in M(B, \mathcal{X}, \theta, \mu)$ *is well-formed. A* B-solution *of* V *is a* B*-morphism* $\sigma : M(B, \mathcal{X}, \theta, \mu) \to M(B, \theta, \mu)$ *such that* $\sigma(W) = \sigma(\overline{W})$ *and* $\sigma(X) \in y^*$ *whenever* $(X, y) \in \theta$. *A* solution *of* V *is a pair* (α, σ) *such that* $\alpha : M(B, \theta, \mu) \to A^*$ *is an* A*-morphism satisfying* $\mu_0 \alpha = \mu$ *and* σ *is a* B*-solution.*

W	= equation, where the solution is a "palindrome" $\sigma(W) = \overline{\sigma(W)} \in A^*$.
B	= alphabet of constants with $\# \in A \subseteq B = \overline{B} \subseteq C$.
$\mathcal{X}$	= variables appearing in W. Hence, $\mathcal{X} = \overline{\mathcal{X}} \subseteq \Omega$.
μ	= morphism to control the constraint that the solution is reduced.
θ	= partial commutation.

During the process of finding a solution, we change these parameters, and we describe the process in terms of a diagram (directed graph) of states and arcs between them.

The directed labeled graph $\mathcal{G}$. We are now ready to define the directed labeled graph $\mathcal{G}$ which will be the core of the NFA defining the EDT0L language $\text{Sol}(U, V) = \{\sigma(X_1)\# \cdots \#\sigma(X_k) \mid \sigma \text{ solves } (U, V) \text{ in reduced words}\}$.

Define the *vertex set* for $\mathcal{G}$ to be the set of all extended equations $V = (W, B, \mathcal{X}, \theta, \mu)$. The *initial vertices* are of the form $(W_{\text{init}}, A, \Omega, \emptyset, \mu_{\text{init}})$. Due to the different possible choices for μ_{init} there are exponentially many initial vertices. We define the set of *final vertices* by $\left\{(W, B, \emptyset, \emptyset, \mu) \;\middle|\; W = \overline{W}\right\}$. By definition every final vertex trivially has a B-solution $\sigma = \mathrm{id}_B$. (Note that in a final vertex there are no variables.) The arcs in $\mathcal{G}$ are labeled and are of the form $(W, B, \mathcal{X}, \theta, \mu) \xrightarrow{h} (W', B', \mathcal{X}', \theta', \mu')$. Here $h : C^* \to C^*$ is an endomorphism given by a morphism $h : B' \to B^*$ such that h induces a well-defined morphism $h : M(B' \cup \mathcal{X}', \theta', \mu') \to M(B \cup \mathcal{X}, \theta, \mu)$. Note that the direction of the morphism is opposite to the direction of the arc. There will be further restrictions on arcs. For example, we will have $|h(b')| \leq 2$ for all b'. The main idea is as follows. Suppose $(W, B, \mathcal{X}, \theta, \mu) \xrightarrow{h} (W', B', \mathcal{X}', \theta', \mu')$ is an arc, $\alpha : M(B, \theta, \mu) \to M(A, \emptyset, \mu_0)$ is an A-morphism, and $(W', B', \mathcal{X}', \theta', \mu')$ has a B'-solution σ'; then there exists a solution (α, σ) of the vertex $(W, B, \mathcal{X}, \theta, \mu)$. Moreover, for the other direction if (α, σ) solves $V = (W, B, \mathcal{X}, \theta, \mu)$ and V is not final then we can follow an outgoing arc and recover (α, σ) from a solution at the target node. We will make this more precise below.

Compression arcs. These arcs transform the sets of constants. Let $V = (W, B, \mathcal{X}, \theta, \mu)$ and $V' = (W', B', \mathcal{X}, \theta', \mu')$ be two vertices in $\mathcal{G}$. The compression arcs have the form $V \xrightarrow{h} V'$, where either $h = \mathrm{id}_{C^*}$ is the identity on C^* and we write $h = \varepsilon$ in this case, or h is defined by a mapping $c \mapsto h(c)$ where $c \in B'$. Recall that if a morphism h is defined by $h(c) = u$ for some letter c then, automatically, $h(\overline{c}) = \overline{u}$ and $h(x) = x$ for all $x \in \Sigma$ which are different from c and $\overline{c}$. We assume $0 \neq \mu'(c) = \mu(h(c)) \neq 1$ and $\mu(x) = \mu'(x)$ for all $x \in (B \cap B') \cup \mathcal{X}$ (if not explicitly stated otherwise).

We define compression arcs $(h(W'), B, \mathcal{X}, \theta, \mu) \xrightarrow{h} (W', B', \mathcal{X}, \theta', \mu')$ of the following three types.

1. (**Renaming.**) The morphism h is defined by $h(c) = a$ such that $B \subseteq B' = B \cup \{c, \overline{c}\}$, and $\theta \subseteq \theta'$. Thus, possibly, $\theta \subsetneq \theta'$.
2. (**Compression.**) We have $h(c) = u$ with $1 \neq |u| \leq 2$ and either $B = B'$ and $\theta' = \theta$ or $B \subsetneq B' = B \cup \{c, \overline{c}\}$ and $\theta = \theta' = \emptyset$.
3. (**Alphabet reduction.**) We have $B' \subsetneq B$, $\theta' = \emptyset$, and h is induced by the inclusion $B' \subseteq B$ which leads to an arc label $h = \varepsilon = \mathrm{id}_{C^*}$.

For the proof of Theorem 1 it is enough to compress words of length at most 2 into a single letter. For Theorem 2 we need additionally arcs of type **2** where $u = a\overline{a}c$ with either $a = c$ (and $\overline{a} = \overline{c}$) or $(a\overline{a}, c\overline{c}) \in \theta$. In particular, the type relation has to be defined in slightly more complicated way. The purpose of arcs of type **3** is to remove letters in B that do not appear in the word W. This allows us to reduce the size of B and also to "kill" partial commutation.

Lemma 2. *Let $(W, B, \mathcal{X}, \theta, \mu) \xrightarrow{h} (W', B', \mathcal{X}', \theta', \mu')$ be a compression arc with $W = h(W')$. Let $\alpha : M(B, \theta, \mu) \to M(A, \emptyset, \mu_0)$ be an A-morphism at the vertex*

$V = (h(W'), B, \mathcal{X}, \theta, \mu)$ and let σ' be a B'-solution to $V' = (W', B', \mathcal{X}', \theta', \mu')$. Define a B-morphism $\sigma : M(B, \mathcal{X}, \theta, \mu) \to M(B, \theta, \mu)$ by $\sigma(X) = h\sigma'(X)$. Then (α, σ) is a solution at V, $(\alpha h, \sigma')$ is a solution at V' and $\alpha\sigma(W) = \alpha h\sigma'(W')$.

Substitution arcs. These arcs transform variables. Let $V = (W, B, \mathcal{X}, \theta, \mu)$ and $V' = (W', B, \mathcal{X}', \theta', \mu')$ be vertices in $\mathcal{G}$ and $X \in \mathcal{X}$. We assume that $\mathcal{X} = \mathcal{X}' \cup \{X, \overline{X}\}$ and $\mu(x) = \mu'(x)$, as well as $\theta(x) = \theta'(x)$ for all $x \in (B \cup \mathcal{X}) \setminus \{X, \overline{X}\}$. The set of constants is the same on both sides, but $\mathcal{X}'$ might have fewer variables. Substitution arcs are defined by a morphism $\tau : \{X\} \to BX \cup \{1\}$ such that we obtain a B-morphism $\tau : M(B, \mathcal{X}, \theta, \mu) \to M(B, \mathcal{X}', \theta', \mu')$. We let $\varepsilon = \mathrm{id}_{C^*}$ as before. We define substitution arcs $(W, B, \mathcal{X}, \theta, \mu) \xrightarrow{\varepsilon} (\tau(W), B, \mathcal{X}', \theta', \mu')$ if one of the following conditions apply.

4. (**Removing a variable.**) Let $\mathcal{X}' = \mathcal{X} \setminus \{X, \overline{X}\}$. The B-morphism $\tau : M(B, \mathcal{X}, \theta, \mu) \to M(B, \mathcal{X}', \theta', \mu')$ is defined by $\tau(X) = 1$.
5. (**Variable typing.**) The purpose of this arc is to introduce some type for variables without changing anything else, so $\mathcal{X}' = \mathcal{X}$ and $\mu' = \mu$. Suppose that $\theta(X) = \emptyset$ and $c \in B$ is a letter with $\mu(Xc) = \mu(cX)$ and such that $\theta' = \theta \cup \{(X, c), (\overline{X}, \overline{c})\}$. The B-morphism $\tau : M(B, \mathcal{X}, \theta, \mu) \to M(B, \mathcal{X}, \theta', \mu)$ is defined by the identity on $B \cup \mathcal{X}$. Note that the condition $\mu(Xc) = \mu(cX)$ implies that if $\mu : M(B, \mathcal{X}, \theta, \mu) \to N$ is well-defined, then $\mu : M(B, \mathcal{X}, \theta', \mu) \to N$ is well-defined, too. The other direction is trivial.
6. (**Substitution of a variable.**) We have $(B, \mathcal{X}, \theta) = (B', \mathcal{X}', \theta')$. Let $a \in B$ be such that $\theta(X) \subseteq \{a\}$. (For $\theta(X) = \emptyset$ this is true for any $a \in B$.) We suppose that $\mu(X) = \mu(a)\mu'(X)$ (hence, automatically $\mu(\overline{X}) = \mu'(\overline{X})\mu(\overline{a})$) and that $\tau(X) = aX$ defines a morphism $\tau : M(B, \mathcal{X}, \theta, \mu) \to M(B, \mathcal{X}, \theta, \mu')$.

Lemma 3. *Let $V = (W, B, \mathcal{X}, \theta, \mu) \xrightarrow{\varepsilon} (W', B, \mathcal{X}', \theta', \mu') = V'$ with $\varepsilon = \mathrm{id}_{C^*}$ be a substitution arc with $W' = \tau(W)$. Let $\alpha : M(B, \theta, \mu) \to M(A, \emptyset, \mu_0)$ be an A-morphism at vertex V and σ' be a B-solution to V'. Define a B-morphism $\sigma : M(B, \mathcal{X}, \theta, \mu) \to M(B, \theta, \mu)$ by $\sigma(X) = \sigma'\tau(X)$. Then (α, σ) is a solution at V and (α, σ') is a solution at V'. Moreover, $\alpha\sigma(W) = \alpha h\sigma'(W')$ where $h = \varepsilon$ is viewed as the identity on $\mathrm{id}_{M(B,\theta,\mu)}$.*

Proof. Since σ' is a B-solution to V' we have $\sigma(W) = \sigma'(\tau(W)) = \sigma'(\overline{\tau(W)}) = \overline{\sigma'\tau(W)} = \overline{\sigma(W)}$. Hence, (α, σ) is a solution at V. Since $M(B, \theta, \mu) = M(B, \theta', \mu')$ (a possible change in μ or θ concerns variables, only), (α, σ') is a solution at V'. The assertion $\alpha\sigma(W) = \alpha h\sigma'(W')$ is trivial since $W' = \tau(W)$, $\sigma = \sigma'\tau$, and $h = \varepsilon$ induces the identity on $M(B, \theta, \mu)$. □

Proposition 1. *Let $V_0 \xrightarrow{h_1} V_1 \cdots \xrightarrow{h_t} V_t$ be a path in $\mathcal{G}$ of length t, where $V_0 = (W_{\mathrm{init}}, A, \Omega, \emptyset, \mu_{\mathrm{init}})$ is an initial and $V_t = (W', B, \emptyset, \emptyset, \mu)$ is a final vertex. Then V_0 has a solution (id_A, σ) with $\sigma(W_{\mathrm{init}}) = h_1 \cdots h_t(W')$. Moreover, we have $W' \in \#u_1\# \cdots \#u_k\#B^*$ such that $|u_i|_\# = 0$ and we can write:*

$$h_1 \cdots h_t(u_1\# \cdots \#u_k) = \sigma(X_1)\# \cdots \#\sigma(X_k), \tag{1}$$

Proof. By definition of final vertices we have $\overline{W'} = W'$ and no variables occur in W'. Hence, id_{B^*} defines the (unique) B-solution of W'. By definition of the arcs, $h = h_1 \cdots h_t : M(B, \emptyset, \mu) \to A^* = M(A, \emptyset, \mu_{\mathrm{init}})$ is an A-morphism which shows that (h, id_{B^*}) solves W'. There is only one A-morphism at V_0, namely id_{A^*}. Using Lemma 2 and Lemma 3 we see first that V_0 has some solution $(\mathrm{id}_{A^*}, \sigma)$ and second, that

$$\mathrm{id}_{A^*}\sigma(W_{\mathrm{init}}) = \mathrm{id}_{A^*}h_1 \cdots h_t \mathrm{id}_{B^*}(W') = h_1 \cdots h_t(W'). \tag{2}$$

Finally, for $1 \leq j \leq t$ we have $h_j(\#) = \#$ and $|h_j(x)|_{\#} = 0$ for all other symbols. Hence the claim $h_1 \cdots h_t(u_1 \# \cdots \# u_k) = \sigma(X_1) \# \cdots \# \sigma(X_k)$. □

Compression[1] Consider an initial vertex $V_0 = (W_{\mathrm{init}}, A, \Omega, \emptyset, \mu_{\mathrm{init}})$ with a solution (α, σ). We will show below that $\mathcal{G}$ contains a path $V_0 \xrightarrow{h_1} V_1 \cdots \xrightarrow{h_t} V_t$ to some final vertex $V_t = (W', B, \emptyset, \emptyset, \mu)$ such that $\sigma(W_{\mathrm{init}}) = h_1 \cdots h_t(W')$, and so $\mathcal{G}$ contains all solutions to W_{init}. Let us show why then, indeed, we are almost done with Theorem 1. We augment the graph $\mathcal{G}$ by one more vertex which is just the symbol #. Recall that $\{X_1, \ldots, X_k\}$ has been the set of specified variables. Every final vertex $(W', B, \emptyset, \emptyset, \mu)$ has a unique factorization $W' = \# w' \# w''$ with $|w'|_{\#} = k$. Let us add arcs $(W', B, \emptyset, \emptyset, \mu) \xrightarrow{g_{w'}} \#$ where $g_{w'} : C^* \to C^*$ is the homomorphism (not necessarily respecting the involution) defined by $g_{w'}(\#) = w'$. If we define the NFA $\mathcal{A}$ as $\mathcal{G}$ with this augmentation and if we let # be the exclusive accepting vertex, then by Proposition 1 we obtain Theorem 1. The construction of $\mathcal{A}$ can easily be implemented by an $\mathsf{NSPACE}(n \log n)$ procedure in such a way that the NFA $\mathcal{A}$ becomes *trim*. This means that every vertex is on some path from an initial to a final vertex. Trimming is important to derive the complexity bounds announced in the abstract[2].

We show the existence of the path corresponding to the solution (α, σ) using an alternation between "block compression" and "pair compression", repeated until we reach a final vertex. The procedures use knowledge of the solution being aimed for. We proceed along arcs in $\mathcal{G}$ of the form $V = (W, B, \mathcal{X}, \theta, \mu) \xrightarrow{h} V' = (W', B', \mathcal{X}', \theta', \mu')$ thereby transforming a solution (α, σ) to V into a solution (α', σ') to V'. However, this is not allowed to be arbitrary: we must keep the invariant $\alpha\sigma(W) = \alpha' h \sigma'(W')$. For example, consider the alphabet reduction where $B' \subsetneq B$ and $W = W' \in (B' \cup \mathcal{X})^*$. In this case we have $h = \mathrm{id}_{C^*}$, which induces the inclusion $\varepsilon : M(B', \emptyset, \mu') \to M(B, \theta, \mu)$. If σ does not use letters outside B' there is no obstacle. In the other case, fortunately, we will need alphabet reduction only when the type relation is empty on both sides. Then we can define $\beta(b) = \alpha(b) \in A^*$ for $b \in B \setminus B'$ and $\beta(b) = b$ for $b \in B'$. We let $\sigma'(X) = \beta\sigma(X)$. This defines a B'-solution at V'. In some sense this is a huge "decompression".

[1] Compression became a main tool for solving word equations thanks to [16].

[2] The possibility to trim $\mathcal{A}$ in $\mathsf{NSPACE}(n \log n)$ uses the result of Immerman and Szelepcsényi that nondeterministic space complexity classes are closed under complementation. For a proof of the Immerman-Szelepcsényi Theorem, see e.g. [13].

A word $w \in \Sigma^*$ is a sequence of *positions*, say $1, 2, \ldots, |w|$, and each position is labeled by a letter from Σ. If $W = u_0 x_1 u_1 \cdots x_m u_m$, with $u_i \in C^*$ and $x_i \in \Omega$, then $\sigma(W) = u_0 \sigma(x_1) u_1 \cdots \sigma(x_m) u_m$ and the positions in $\sigma(W)$ corresponding to the u_i's are henceforth called *visible*.

Block compression. Let $V = (W, B, \mathcal{X}, \emptyset, \mu)$ be some current non-final vertex with an empty type relation and a solution (α, σ). We start a block compression only if $B \le |W| \le 29n$. Since $|C| = 100n$, there will be sufficiently many "fresh" letters in $C \setminus B$ at our disposal.

1. Follow substitution arcs to remove all variables with $|\sigma(X)| \le 2$. If V became final, we are done and we stop. Otherwise, for each X we have $\sigma(X) = bw$ for some $b \in B$ and $w \in B^+$. Following a substitution arc we replace X by bX. Of course, we also replace $\overline{X}$ by $\overline{X}\,\overline{b}$, changing $\mu(X)$ to $\mu(X) = \overline{\mu(\overline{X})} = \mu(w)$ (from now on we will do this without comment). If $bX \le W$ and $b'X \le W$ are factors with $b, b' \in B$, then $\# \ne b = b'$ due to the previous substitution $X \mapsto bX$. For each $b \in B \setminus \{\#\}$ define sets $\Lambda_b \subseteq \mathbb{N}$ which contain those $\lambda \ge 2$ such that there is an occurrence of a factor $db^\lambda e$ in $\sigma(W)$ with $d \ne b \ne e$, where at least one of the b's is visible. We also let $\mathcal{X}_b = \{X \in \mathcal{X} \mid bX \le W \wedge \sigma(X) \in bB^*\}$. Note that $\sum_b |\Lambda_b| + |\mathcal{X}_b| \le |W|$.
2. Since W is well-formed we have $\Lambda_b = \Lambda_{\overline{b}}$. Fix some subset $B_+ \subseteq B$ such that for each $\# \ne b \in B$ we have $b \in B_+ \iff \overline{b} \ne B_+$. For each $b \in B_+$, where $\Lambda_b \ne \emptyset$, run the following *b-compression*:
3. ***b*-compression.** (This step removes all factors b^ℓ and $\overline{b}^\ell$, $\ell \ge 2$, from W.)
   (a) Introduce fresh letters $c_b, \overline{c_b}$ with $\mu(c_b) = \mu(b)$. In addition, for each $\lambda \in \Lambda_b$ introduce fresh letters $c_{\lambda,b}, \overline{c_{\lambda,b}}$ with $\mu(c_{\lambda,b}) = \mu(b)$. We abbreviate $c = c_b$, $\overline{c} = \overline{c_b}$, $c_\lambda = c_{\lambda,b}$, and $\overline{c_\lambda} = \overline{c_{\lambda,b}}$. We let $h(c_\lambda) = h(c) = b$ and we introduce a type by letting $\theta = \{(c_\lambda, c) \mid \lambda \in \Lambda_b\}$. Renaming arcs (type **1**) realize this transformation.
   So far we did not change W, but we enlarged the alphabet B to B', and introduced partial commutation between the fresh letters c_λ and c. The next steps change W and its solution.
   (b) Replace in $\sigma(W) \in B^*$ every factor $db^\lambda e$ (resp. $d\overline{b}^\lambda e$), where $d \ne b \ne e$ and $\lambda \in \Lambda_b$, by $dc^\lambda e$ (resp. $d\overline{c}^\lambda e$). This yields a new word $W' \in B'^*$, which was obtained via the renaming arc $h(c) = b$. Recall that for every $X \in \mathcal{X}_b$ we had $bX \le W$ and for some positive ℓ we had $\sigma(X) = b^\ell w$ with $w \notin bB^*$. In the new word W' we have $cX \le W'$ and for the new solution σ' we have $\sigma'(X) = c^\ell w'$ with $w' \notin cB'^*$. We rename $W', B', \alpha' = \alpha h, \sigma'$ as W, B, α, σ.
   (c) Enlarge θ by $\{(X, c) \mid X \in \mathcal{X}_b \wedge \sigma(X) \in c^*\}$ using a substitution arc.
   (d) The solution $\sigma(W)$ is still a word $\sigma(W) \in B^*$. Scan this word from left to right. Stop at each factor $dc^\lambda e$ with $d \ne c \ne e$ and $\lambda \in \Lambda_b$. If in this factor some position of the c's is visible then choose exactly one of these visible positions and replace that c by c_λ. If no c is visible, they are all inside some $\sigma(X)$; then choose any c and replace it by c_λ. Recall that c and

c_λ commute, hence $dc^\lambda e$ became $dc_\lambda c^{\lambda-1}e = dc^{\ell_1}c_\lambda c^{\ell_2}e \in M(B,\theta,\mu)$ for all $\ell_1+\ell_2 = \lambda-1$. In parallel we run the same steps for $\overline{c}$. The whole transformation can be realized by renaming arcs defined by $h(c_\lambda) = c$. There is a crucial observation: if $X \in \mathcal{X}_b$ and we had $\sigma(X) = c^\ell w$ with $w \notin cB^*$ before the transformation then now still $\sigma'(X) = c^\ell w'$. It is not clear which position has been occupied by c_λ, but due to commutation $c_\lambda\sigma'(X)$ is a factor in $\sigma'(W') \in M(B,\theta,\mu)$.

(e) Rename $W', B', \alpha' = \alpha h, \sigma'$ as W, B, α, σ. Perform the following loop 3(e)i – 3(e)iv until no c and no $X \in \mathcal{X}_b$ with $\sigma(X) \in c^*$ occurs in W.
   i. If $X \in \mathcal{X}_b$ and if the maximal ℓ is odd where $\sigma(X) \in c^\ell B^*$, then follow a substitution arc $X \mapsto cX$. Do the same for $\overline{c}$.
   ii. For all λ where there is some odd ℓ with $dc_\lambda c^\ell e \le \sigma'(W')$ follow a compression arc defined by $h(c_\lambda) = cc_\lambda$. This is possible since for each such factor $dc_\lambda c^\ell e$ either none of the positions in $c_\lambda c^\ell$ is visible or $c_\lambda c$ is visible. Thus, $dc_\lambda c^\ell e \le \sigma'(W')$ implies that ℓ is even.
   iii. Follow a compression arc defined by $h(c) = c^2$, after which $|W'|_c$ and $|W'|_{\overline{c}}$ are divided by 2. We obtain a new W'' with solution σ''.
   iv. Remove all X with $\sigma''(X) = 1$ by following a substitution arc (type **4**); rename all parameters back to $W, B, \mathcal{X}, \theta, \mu, \alpha, \sigma$.

(f) Let $B' = B \setminus \{c, \overline{c}\}$ and μ' be induced by μ. Observe that no letter c or $\overline{c}$ appears in $\sigma(W)$: they have all been consumed by c_λ. Thus, the type relation is empty again. Hence we can follow an alphabet reduction arc $(W, B, \mathcal{X}, \theta, \mu) \xrightarrow{\varepsilon} (W, B', \mathcal{X}, \emptyset, \mu')$. The new solution to $(W, B', \mathcal{X}', \emptyset, \mu')$ is the pair (α', σ) where $\alpha' = \alpha\varepsilon$ is defined by the restriction of α to $M(B', \emptyset, \mu')$.

Having performed b-compressions for all $b \in B_+$, we have increased the length of W. But it is not difficult to see that the total increase can be bounded in $\mathcal{O}(n)$. Actually, we have $|W| \le 31n$ at the end because we started with $|W| \le 29n$ and step 1 of block compression increases $|W|$ by at most $2n$. Now we use alphabet reduction in a final step of block compression in order to reduce the alphabet B such that $|B| \le |W|$. We end up at a vertex again named $V = (W, B, \mathcal{X}, \emptyset, \mu)$, which has a solution (α, σ). The new situation is that no proper factor b^2 appears in W anymore. The price is $|B| \le |W| \le 31n$.

We now run Jeż's procedure *pair compression*, which brings us back to $|B| \le |W| \le 29n$ and allows us to start another block compression. This keeps the length in $\mathcal{O}(n)$. Since our pair compression is very close to Jeż's presentation as published in [11] we content ourselves with the basic idea. For pair compression we begin with a partition $B \setminus \{\#\} = L \cup R$ such that $b \in L \iff \overline{b} \in R$. In general, there are many such partitions, but we choose with care a "good" partition, see below. Next, for all X, if $b \in R$ and $\sigma(X) \in bB^*$ then replace X by bX and $\overline{X}$ by $\overline{X}\,\overline{b}$. After that, no factor $ab \in LR$ is "crossing", i.e., is consisting of a visible and invisible letter, anymore. Moreover, $ab \in LR \iff \overline{b}\overline{a} \in LR$. Thus, we can follow compression arcs labeled by $h(c) = ab$, where c is a fresh letter. Let $s(i)$ denote the length of W after the i-th iteration of pair-compression. For at least one partition $B \setminus \{\#\} = L \cup R$, the "good" partition, it can be guaranteed

that $s(i+1) \in \frac{5s(i)}{6} + \mathcal{O}(n)$, see [11]. Together with $s(1) \in \mathcal{O}(n)$ this shows $s(i) \in \mathcal{O}(n)$ for all i. Another fact is crucial. We restricted ourselves to solutions in reduced words, which implies that whenever ab is a proper factor of $\sigma(W)$, then $b \neq \overline{a}$.

References

[1] A. V. Aho. Indexed grammars—an extension of context-free grammars. *J. Assoc. Comput. Mach.*, 15:647–671, 1968.

[2] P. R. Asveld. Controlled iteration grammars and full hyper-AFL's. *Information and Control*, 34(3):248 – 269, 1977.

[3] M. Benois. Parties rationelles du groupe libre. *C. R. Acad. Sci. Paris, Sér. A*, 269:1188–1190, 1969.

[4] 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.

[5] V. Diekert, A. Jeż, and W. Plandowski. Finding all solutions of equations in free groups and monoids with involution. *Proc. CSR 2014* **LNCS** 8476: 1–15, 2014.

[6] A. Ehrenfeucht and G. Rozenberg. On some context free languages that are not deterministic ET0L languages. *RAIRO Theor. Inform. Appl.*, 11:273–291, 1977.

[7] S. Eilenberg. *Automata, Languages, and Machines*, Vol A. Acad. Press, 1974.

[8] J. Ferté, N. Marin, and G. Sénizergues. Word-mappings of level 2. *Theory Comput. Syst.*, 54:111–148, 2014.

[9] R. H. Gilman. Personal communication, 2012.

[10] S. Jain, A. Miasnikov, and F. Stephan. The complexity of verbal languages over groups. *Proc. LICS 2012*, pages 405–414. IEEE Computer Society, 2012.

[11] A. Jeż. Recompression: a simple and powerful technique for word equations. *Proc. STACS.* **LIPIcs**, 20:233–244, 2013. Journal version to appear in JACM.

[12] O. Kharlampovich and A. Myasnikov. Elementary theory of free non-abelian groups. *J. of Algebra*, 302:451–552, 2006.

[13] Ch. H. Papadimitriou. *Computational Complexity.* Addison Wesley, 1994.

[14] W. Plandowski. An efficient algorithm for solving word equations. *Proc. STOC'06*: 467–476. ACM Press, 2006.

[15] W. Plandowski. Personal communication, 2014.

[16] W. Plandowski and W. Rytter. Application of Lempel-Ziv encodings to the solution of word equations. *Proc. ICALP'98.* **LNCS** 1443: 731–742, 1998.

[17] A. A. Razborov. *On Systems of Equations in Free Groups.* PhD thesis. 1987.

[18] A. A. Razborov. On systems of equations in free groups. In *Combinatorial and Geometric Group Theory*, pages 269–283. Cambridge University Press, 1994.

[19] G. Rozenberg and A. Salomaa. *The Book of L.* Springer, 1986.

[20] G. Rozenberg et al. (Eds.) *Handbook of Formal Languages*, Vol 1. Springer, 1997.

[21] Z. Sela. Diophantine geometry over groups VIII: Stability. *Annals of Math.*, 177:787–868, 2013.