The Isomorphism Problem for Finite Extensions of Free Groups Is In PSPACE

Géraud Sénizergues
LABRI, Bordeaux, France
geraud.senizergues@u-bordeaux.fr

Armin Weiß
Universität Stuttgart, FMI, Germany
armin.weiss@fmi.uni-stuttgart.de

Abstract

We present an algorithm for the following problem: given a context-free grammar for the word problem of a virtually free group G, compute a finite graph of groups $\mathcal{G}$ with finite vertex groups and fundamental group G. Our algorithm is non-deterministic and runs in doubly exponential time. It follows that the isomorphism problem of context-free groups can be solved in doubly exponential space. Moreover, if, instead of a grammar, a finite extension of a free group is given as input, the construction of the graph of groups is in NP and, consequently, the isomorphism problem in PSPACE.

2012 ACM Subject Classification Mathematics of computing → Graph theory, Theory of computation → Grammars and context-free languages, Theory of computation → Computational complexity and cryptography

Keywords and phrases virtually free groups, context-free groups, isomorphism problem, structure tree, graph of groups

Digital Object Identifier 10.4230/LIPIcs.ICALP.2018.139

Related Version A full version of the paper is available at [23], `https://arxiv.org/abs/1802.07085`.

Acknowledgements G.S. thanks the FMI for hosting him from October to the end of the year 2017. Both authors acknowledge the financial support by the DFG project DI 435/7-1 "Algorithmic problems in group theory" for this work.

1 Introduction

The study of algorithmic problems in group theory was initiated by Dehn [6] when he introduced the word and the isomorphism problem. The word problem asks whether a given word over a (finite) set of generators represents the identity of the group. It also can be viewed as a formal language, namely $\varphi^{-1}(1) \subseteq \Sigma^*$ for some surjective monoid homomorphism $\varphi : \Sigma^* \to G$. The isomorphism problem receives two finite presentations as input, the question is whether the groups they define are isomorphic. Although both these problems are undecidable in general [18, 3], there are many classes of groups where at least the word problem can be decided efficiently.

One of these classes are the finitely generated *virtually free* groups (groups with a free subgroup of finite index). It is easy to see that the word problem of a finitely generated virtually free group can be solved in linear time. Indeed, it forms a deterministic context-free language. A seminal paper by Muller and Schupp [16] shows the converse: every group

© Géraud Sénizergues and Armin Weiß;
licensed under Creative Commons License CC-BY
45th International Colloquium on Automata, Languages, and Programming (ICALP 2018).
Editors: Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella;
Article No. 139; pp. 139:1–139:14
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

with a context-free word problem is virtually free. Since then, also a wide range of other characterizations of virtually free groups have emerged – for a survey we refer to [1, 9].

The isomorphism problem of virtually free groups is also decidable as Krstić showed in [15] (indeed, later Dahmani and Guirardel showed that even the isomorphism problem for the larger class of hyperbolic groups is decidable [5]). Here the input consists of two arbitrary finite presentations with the promise that both define virtually free groups. Unfortunately, the approach in [15] does not give any bound on the complexity. For the special case where the input is given as finite extensions of free groups or as context-free grammars for the word problems, Sénizergues [21, 22] showed that the isomorphism problem is primitive recursive. Krstić's and Sénizergues' approaches both compute so-called graphs of groups, which encode groups acting on trees, and then check these graph of groups for "isomorphism". By the work of Karrass, Pietrowski and Solitar [14], a finitely generated group is virtually free if and only if it is the fundamental group of a finite graph of groups with finite vertex groups.

Contribution. We improve the complexity for the isomorphism problem by showing:

(A) Given a context-free grammar for the word problem of a context-free group G, a graph of groups for G with finite vertex groups can be computed in $\mathsf{NTIME}(2^{2^{\mathcal{O}(n^2)}})$ (Theorem 33).

(B) Given a virtually free presentation for G, a graph of groups for G with finite vertex groups can be computed in NP (Theorem 34).

(C) The isomorphism problem for context-free groups given as grammars is in $\mathsf{DSPACE}(2^{2^{\mathcal{O}(n^2)}})$ (Theorem 37).

(D) The isomorphism problem for virtually free groups given as virtually free presentations is in PSPACE (Theorem 38).

Here, a *virtually free presentation* for G consists of a free group F plus a set of representatives S for the quotient $F \backslash G$ together with relations describing pairwise multiplications of elements from F and S. Typical examples of virtually free presentations are finite extensions of free groups (i. e., where the free sugroup F is normal in G). For non-deterministic function problems we use the convention, that every accepting computation must yield a correct result; but the results of different accepting computations might differ[1].

The results C and D can be seen be to follow from A and B rather easily. Indeed, we conclude from Forester's work on deformation spaces [10] that two graphs of groups with finite vertex groups and isomorphic fundamental groups can be transformed one into each other by a sequence of slide moves (Proposition 35).

Our approach for proving A and B is as follows: in both cases the algorithm simply guesses a graph of groups together with a map and afterwards it verifies deterministically whether the map is indeed an isomorphism. The latter can be done using standard results from formal language theory. The difficult part is to show the existence of a "small" graph of groups and isomorphism (within the bounds of A and B).

For this, we introduce the *structure tree* theory by Dicks and Dunwoody [7] following a slightly different approach by Diekert and Weiß [8] based on the *optimal cuts* of the Cayley graph (Section 2.3). The optimal cuts can be seen as the edge set of some tree on which the group G acts. By Bass-Serre theory, this yields the graph of groups we are aiming

[1] Thus, B means that the graph of groups can be computed in NPMV in the sense of [20]. More precisely, it can be rephrased as follows: the multi-valued function mapping a virtually free presentation for G into a pair $(\mathcal{G}, \varphi)$, where $\mathcal{G}$ is a graph of groups and $\varphi : \pi_1(\mathcal{G}) \to G$ is an isomorphism of polynomial size, is everywhere defined and belongs to the class FNP as defined in [19].

for. Vertices in the graph of groups are defined in terms of equivalence classes of optimal cuts. The key in the proof is to bound the size of the equivalence classes. Using Muller and Schupp's [16] notion of k-triangulability, Sénizergues [22] proved bounds on the size of finite subgroups and on the number of edges in a reduced graph of groups for a context-free group, from which we derive our bounds.

Outline. After fixing our notation, we recall basic facts from Bass-Serre theory and the results from [22] and give a short review on structure trees based on [8]. Section 3, develops bounds on the size of the vertices (= equivalence classes of cuts) of the structure tree. After that, we introduce virtually free presentations formally and we derive stronger bounds for this case in Section 4. Section 5 completes the proofs of A and B. Finally, in Section 6 we derive C and D and we conclude with some open questions. Due to space constraints most of the proofs are omitted; they can be found in the full version on arXiv [23].

2 Preliminaries

Complexity. We use the following convention for non-deterministic function problems: each accepting computation path must yield a correct answer – though different accepting paths can compute different correct answers. We use this convention to define the classes NP (non-deterministic polynomial time) and $\mathsf{NTIME}(f(n))$ (non-deterministic time bounded by $f(n)$). Otherwise, we use standard complexity classes P (deterministic polynomial time), PSPACE (polynomial space) and $\mathsf{DSPACE}(f(n))$ (deterministic space bounded by $f(n)$) for both decision and function problems.

Words. An *alphabet* is a (finite) set Σ; an element $a \in \Sigma$ is called a *letter*. The set Σ^n forms the set of *words* of length n. The length of $w \in \Sigma^n$ is denoted by $|w|$. The set of all words is denoted by Σ^*. It is the free monoid over Σ – its neutral element is the empty word 1.

Context-free grammars. We use standard notation for context-free grammars: a context-free grammar is a tuple $\mathbb{G} = (V, \Sigma, P, S)$ with variables V, terminals Σ, a finite set of productions rules $P \subseteq V \times (V \cup \Sigma)^*$, and a start symbol S. We denote its size by $\|\mathbb{G}\| = |V| + |\Sigma| + \sum_{A \to \alpha \in P} |\alpha|$. It is in *Chomsky normal form* if all productions are of the form $S \to 1$, $A \to a$ or $A \to BC$ with $A, B, C \in V$, $a \in \Sigma$. For further definitions, we refer to [12].

Groups. Let Σ be an alphabet and $R \subseteq \Sigma^* \times \Sigma^*$. The monoid (or group) G presented by (Σ, R) is defined as $G = \Sigma^*/R = \Sigma^*/{=_G}$ where $=_G$ is the smallest monoid congruence over Σ^* containing R. There is a canonical projection $\pi : \Sigma^* \to G$. The *word problem* of the group G is the formal language $\mathrm{WP}(G) = \{w \in \Sigma^* \mid w =_G 1\} = \pi^{-1}(1)$. A *symmetric alphabet* is an alphabet Σ endowed with an involution $a \mapsto \overline{a}$ without fixed points (i. e., $\overline{\overline{a}} = a$ and $\overline{a} \neq a$ for all $a \in \Sigma$). If Σ is a symmetric alphabet for G (i. e., $G = \Sigma^*/{=_G}$), we always assume that $a\overline{a} =_G \overline{a}a =_G 1$ for all $a \in \Sigma$ (without writing these relations explicitly).

Let Σ be a symmetric alphabet and $w \in \Sigma^*$. We say that w is *freely reduced* if w has no factor $a\overline{a}$ for any letter $a \in \Sigma$. Given an arbitrary set X, the *free group* over X is denoted by $F(X)$. It is defined as the group presented by $(X \cup \overline{X}, \{x\overline{x} \mid x \in X \cup \overline{X}\})$.

Graphs. A *graph* $\Gamma = (V, E, s, t, \overline{\cdot})$ is given by the following data: a set of vertices $V = V(\Gamma)$, a set of edges $E = E(\Gamma)$ together with two *incidence* maps $s : E \to V$ and $t : E \to V$ and an involution $E \to E$, $e \mapsto \overline{e}$ without fixed points such that $s(e) = t(\overline{e})$. The *degree* of a vertex u is the number of incident edges. An *undirected edge* is a set $\{e, \overline{e}\}$. For the cardinality of sets of edges we usually count the number of undirected edges.

A (finite) *path* from v_0 to v_n is a pair of sequences $((v_0, \dots, v_n), (e_1, \dots, e_n))$ such that $s(e_i) = v_{i-1}$ and $t(e_i) = v_i$ for all $1 \leq i \leq n$. Similarly, a *bi-infinite path* is a pair of sequences $((v_i)_{i \in \mathbb{Z}}, (e_i)_{i \in \mathbb{Z}})$ such that $s(e_i) = v_{i-1}$ and $t(e_i) = v_i$ for all $i \in \mathbb{Z}$. A path is *simple* if the vertices are pairwise distinct. It is *closed* if $v_0 = v_n$. Depending on the situation we also denote paths simply by the sequence of edges or the sequence of vertices. The *distance* $d(u, v)$ between vertices u and v is defined as the length (i. e., the number of edges) of a shortest path connecting u and v. For $A, B \subseteq V(\Gamma)$ with $A, B \neq \emptyset$, the distance is defined as $d(A, B) = \min \{ d(u, v) \mid u \in A, v \in B \}$.

For $S \subseteq V(\Gamma)$ we define $\Gamma - S$ to be the induced subgraph of Γ with vertices $V(\Gamma) \setminus S$. For $C \subseteq V(\Gamma)$, we write $\overline{C}$ for the complement of C, i. e., $\overline{C} = V(\Gamma) \setminus C$. We call C connected, if the induced subgraph is connected. A group G acts on a graph Γ, if it acts on both $V(\Gamma)$ and $E(\Gamma)$ and the actions preserve the incidences.

Cayley graphs. Let G be a group with a symmetric alphabet Σ. The *Cayley graph* $\Gamma = \Gamma_\Sigma(G)$ of G (with respect to Σ) is defined by $V(\Gamma) = G$ and $E(\Gamma) = G \times \Sigma$, with the incidence functions $s(g, a) = g$, $t(g, a) = ga$, and involution $\overline{(g, a)} = (ga, \overline{a})$. For $r \in \mathbb{N}$ let $B(r) = \{ u \in V(\Gamma) \mid d(u, 1) \leq r \}$ denote the ball with radius r around the identity.

Cuts. For a subset $C \subseteq V(\Gamma)$ we define the *edge* and *vertex boundaries* of C as follows:

$$\vec{\delta}C = \left\{ e \in E(\Gamma) \;\middle|\; s(e) \in C, t(e) \in \overline{C} \right\} = \text{directed edge boundary,}$$

$$\partial C = \left\{ s(e) \;\middle|\; e \in \vec{\delta}C \right\} = \text{inner vertex boundary,}$$

$$\beta C = \left\{ s(e) \;\middle|\; e \in \vec{\delta}C \text{ or } \overline{e} \in \vec{\delta}C \right\} = \partial C \cup \partial \overline{C} = \text{vertex boundary.}$$

▶ **Definition 1.** A *cut* is a subset $C \subseteq V(\Gamma)$ such that C and $\overline{C}$ are both non-empty and connected and $\vec{\delta}C$ is finite. The *weight* of a cut is $|\vec{\delta}C|$. If $|\vec{\delta}C| \leq K$, we call C a *K-cut*.

2.1 Bass-Serre theory

We give a brief summary of the basic definitions and results of Bass-Serre theory [24].

▶ **Definition 2** (Graph of Groups). Let $Y = (V(Y), E(Y), s, t, \overline{\cdot})$ be a connected graph. A *graph of groups $\mathcal{G}$ over Y* is given by the following data:

(i) For each vertex $P \in V(Y)$ there is a *vertex group* G_P.

(ii) For each edge $y \in E(Y)$ there is an *edge group* G_y such that $G_y = G_{\overline{y}}$.

(iii) For each edge $y \in E(Y)$ there is an injective homomorphism from G_y to $G_{s(y)}$, which is denoted by $a \mapsto a^y$. The image of G_y in $G_{s(y)}$ is denoted by G_y^y.

Since we have $G_y = G_{\overline{y}}$, there is also a homomorphism $G_y \to G_{t(y)}$ defined by $a \mapsto a^{\overline{y}}$. The image of G_y in $G_{t(y)}$ is denoted by $G_y^{\overline{y}}$. A graph of groups is called *reduced* if $G_y^y \neq G_{s(y)}$ whenever $s(y) \neq t(y)$ for $y \in E(Y)$. Throughout we assume that all graphs of groups are connected and finite (i. e., Y is a connected, finite graph).

Fundamental group of a graph of groups. We begin with the group $F(\mathcal{G})$. It is defined as the free product of the free group $F(E(Y))$ and the groups G_P for $P \in V(Y)$ modulo the set of defining relations $\left\{ \overline{y} a^y y = a^{\overline{y}} \;\middle|\; a \in G_y, y \in E(Y) \right\}$. As an alphabet we fix the disjoint union $\Delta = \biguplus_{P \in V(Y)} (G_P \setminus \{1\}) \cup E(Y)$ throughout. Now, we have

$$F(\mathcal{G}) = F(\Delta) / \left\{ gh = [gh], \overline{y} a^y y = a^{\overline{y}} \;\middle|\; P \in V(Y),\, g, h \in G_P;\, y \in E(Y),\, a \in G_y \right\},$$

where $[gh]$ denotes the element obtained by multiplying g and h in G_P.

For $P \in V(Y)$ we define a subgroup $\pi_1(\mathcal{G}, P)$ of $F(\mathcal{G})$ by the elements $g_0 y_1 \cdots g_{n-1} y_n g_n \in F(\mathcal{G})$, such that $y_1 \cdots y_n$ is a closed path from P to P and $g_i \in G_{s(y_{i+1})}$ for $0 \leq i < n$ and $g_n \in G_P$. The group $\pi_1(\mathcal{G}, P)$ is called the *fundamental group* of $\mathcal{G}$ with respect to the base point P. Since we assumed Y to be connected, there exists a spanning tree $T = (V(Y), E(T))$ of Y. The *fundamental group* of $\mathcal{G}$ with respect to T is defined as

$$\pi_1(\mathcal{G}, T) = F(\mathcal{G}) / \{\, y = 1 \mid y \in T \,\}.$$

▶ **Proposition 3** ([24]). *The canonical homomorphism ψ from the subgroup $\pi_1(\mathcal{G}, P)$ of $F(\mathcal{G})$ to the quotient group $\pi_1(\mathcal{G}, T)$ is an isomorphism. In particular, the two definitions of the fundamental group are independent of the choice of the base point or the spanning tree.*

A word $w \in \Delta^*$ is called *reduced* if it does not contain a factor gh with $g, h \in G_P$ for some P or a factor $\overline{y} a^y y$ with $y \in E(Y)$, $a \in G_y$.

▶ **Lemma 4** ([24, Thm. I.11]). *A reduced word in $\pi_1(\mathcal{G}, P)$ represents the trivial element if and only if it is the empty word.*

The quotient of a G-tree. Graphs of groups arise in a natural way in situations where a group G acts (from the left) on some connected tree $Z = (V, E)$ without edge inversion, i. e., $\overline{e} \neq ge$ for all $e \in E$, $g \in G$. We let $Y = G \backslash Z$ be the quotient graph with vertex set $V(Y) = \{\, Gv \mid v \in V \,\}$ and edge set $E(Y) = \{\, Ge \mid e \in E \,\}$ and incidences and involution induced by Z. By choosing representatives we find embeddings $\iota : V(Y) \hookrightarrow V$ and $\iota : E(Y) \hookrightarrow E$ and we can assume that $\iota(V(Y))$ induces a connected subgraph of Z and that $\iota(\overline{y}) = \overline{\iota(y)}$ for all $y \in E(Y)$. For $P \in V(Y)$, $y \in E(Y)$, we define vertex and edge groups as the stabilizers of the respective representatives: $G_P = \mathrm{Stab}(\iota P) = \{\, g \in G \mid g\iota P = \iota P \,\}$ and $G_y = \mathrm{Stab}(\iota y) = \{\, g \in G \mid g\iota y = \iota y \,\}$. Note that as abstract groups the vertex and edge groups are independent of the choice of representatives since stabilizers in the same orbit are conjugate. Moreover, for each $y \in E(Y)$ there are $g_y, h_y \in G$ such that $s(\iota y) = g_y \iota P$ and $t(\iota y) = h_y \iota Q$ for $P = s(y)$ and $Q = t(y)$. Note that g_y and h_y are not unique; still the left cosets $g_y G_P$ resp. $h_y G_Q$ are uniquely determined. Clearly, we can choose them such that $g_y = h_{\overline{y}}$ and $h_y = g_{\overline{y}}$. This yields two embeddings:

$$G_y \to G_P, \qquad a \mapsto a^y = \overline{g}_y a g_y, \qquad \text{and} \qquad G_y \to G_Q, \qquad a \mapsto a^{\overline{y}} = \overline{h}_y a h_y. \tag{1}$$

Hence, we have obtained a well-defined graph of groups $\mathcal{G}$ over Y. Notice that the G_y^y and $G_y^{\overline{y}}$ depend on the choice of g_y and h_y (and change via conjugation when changing them).

We define a homomorphism $\varphi : \Delta^* \to G$ by $\varphi(g) = g$ for $g \in G_P$, $P \in V(Y)$. For $y \in E(Y)$, we set $\varphi(y) = \overline{g}_y h_y$. That means $\varphi(y)$ maps some edge in the preimage of y and terminating in $\iota t(y)$ to an edge in the preimage of y with source in $\iota s(y)$. By our assumption, we have $\overline{\varphi(y)} = \overline{h}_y g_y = \varphi(\overline{y})$. Since $\varphi(\overline{y} a^y y) = \varphi(\overline{y}) \varphi(a^y) \varphi(y) = \overline{h}_y g_y a^y \overline{g}_y h_y = a^{\overline{y}} = \varphi(a^{\overline{y}})$, we obtain a well-defined homomorphism $\varphi : F(\mathcal{G}) \to G$.

▶ **Theorem 5** ([24]). *The restriction $\varphi : \pi_1(\mathcal{G}, P) \to G$ is an isomorphism.*

2.2 Context-free groups and graphs

▶ **Definition 6.** A group is called *context-free*, if its word problem is a context free language.

Notice that the word problem of a context-free group is decidable in polynomial time – even if the grammar is part of the input – by applying the CYK algorithm (see e. g. [12]).

▶ **Definition 7** (k-triangulable)**.** Let Γ be a graph. Let $k \in \mathbb{N}$ and let $\gamma = v_0, v_1, \ldots, v_n = v_0$ be a sequence of vertices Γ such that $d(v_{i-1}, v_i) \leq k$ for all $i \in \{1, \ldots, n\}$ (e.g. γ can be a closed path). Let P a convex polygon in the plane whose vertices are labeled by the vertices of γ (i.e. we consider γ as a simple closed curve in the plane). A *k-triangulation* of γ is a triangulation of P which does not introduce any additional vertices (thus only consists of "diagonal" edges) and such that vertices joined by a diagonal edge are at distance at most k. If $n < 3$, we consider γ as triangulated. If every closed path γ has a k-triangulation, then Γ is called k-triangulable and we call k the triangulation constant of Γ.

▶ **Lemma 8** ([16, Thm. I])**.** *Let (V, Σ, P, S) be a context-free grammar in Chomsky normal form for the word problem of a group G where Σ is a symmetric alphabet. Then the Cayley graph Γ can be k-triangulated for $k = 2^{|P|}$.*

Note that in [16] only the existence of some k is shown; however, an easy induction shows the bound of Lemma 8. Moreover, a slightly worse bound applies if Σ is not symmetric [23].

▶ **Lemma 9** ([17, p.65])**.** *Let Γ be k-triangulable and let $r \in \mathbb{N}$. If C is a connected component of $\Gamma - B(r)$, then $\operatorname{diam}(\partial C) \leq 3k$.*

▶ **Lemma 10.** *Let Γ be connected and k-triangulable and let $C \subseteq V(\Gamma)$ be a cut. Then $\operatorname{diam}(\beta C) \leq \frac{3k}{2}|\vec{\delta}C|$.*

This lemma is asserted (without proof) in [21, Lemma 6] with a slightly worse bound on $\operatorname{diam}(\beta C)$. For a proof see [23]. The following upper-bounds will be crucial.

▶ **Proposition 11** ([22, Prop. 1.2])**.** *Let Γ be the Cayley graph of a group G on a symmetric alphabet Σ and let Γ be k-triangulable. Then $|H| \leq |\Sigma|^{12k+10}$ for every finite subgroup $H \leq G$.*

▶ **Theorem 12** ([22, Thm. 1.4])**.** *Let Γ be the Cayley graph of a group G on a symmetric alphabet Σ and let Γ be k-triangulable. Then every reduced graph of groups $\mathcal{G}$ admitting G as fundamental group has at most $|\Sigma|^{12k+11}$ undirected edges.*

2.3 Optimal cuts and structure trees

We briefly present the construction of optimal cuts and the associated structure tree from [8, 9]. While in [8], the proof was for arbitrary accessible, co-compact, locally finite graphs, here we assume that Γ is the Cayley graph of a context-free group. We are interested in bi-infinite simple paths which can be split into two infinite pieces by some cut. For a bi-infinite simple path α denote:

$$\mathcal{C}(\alpha) = \left\{ C \subseteq V(\Gamma) \;\middle|\; C \text{ is a cut and } |\alpha \cap C| = \infty = \left|\alpha \cap \overline{C}\right| \right\},$$
$$\mathcal{C}_{\min}(\alpha) = \{ C \in \mathcal{C}(\alpha) \mid |\delta C| \text{ is minimal in } \mathcal{C}(\alpha) \},$$

where we identify α with its set of vertices. If $C \in \mathcal{C}(\alpha)$, we say that C *splits* α. We define the set of *minimal cuts* $\mathcal{C}_{\min}$ as the union of the $\mathcal{C}_{\min}(\alpha)$ over all bi-infinite simple paths α. Since every bi-infinite simple path α with $\mathcal{C}(\alpha) \neq \emptyset$ can be split by a cut which is a connected component of $\Gamma - B(m)$ for some $m \in \mathbb{N}$, the next lemma follows from Lemma 9 and 10.

▶ **Lemma 13.** *Let Γ be k-triangulable and let d denote the degree of Γ. Then for every $C \in \mathcal{C}_{\min}$ we have $|\vec{\delta}C| \leq d^{3k+2}$ and $\operatorname{diam}(\beta C) \leq \frac{3k}{2} d^{3k+2}$.*

Two cuts C and D are called *nested*, if one of the four inclusions $C \subseteq D$, $C \subseteq \overline{D}$, $\overline{C} \subseteq D$ or $\overline{C} \subseteq \overline{D}$ holds. By Lemma 13, with $K = d^{3k+3}$ for every bi-infinite simple path α with

$\mathcal{C}(\alpha) \neq \emptyset$ there exists some cut $C \in \mathcal{C}(\alpha)$ with $|\vec{\delta}C| \leq K$. We fix this number K. For a cut C let $m(C)$ denote the number of K-cuts that are not nested with C. It follows from [25] that $m(C)$ is always finite, see also [8, Lem. 3.4].

▶ **Definition 14.** The set of *optimal cuts* is defined as

$$\mathcal{C}_{\text{opt}}(\alpha) = \{ C \in \mathcal{C}_{\min}(\alpha) \mid m(C) \leq m(D) \text{ for all } D \in \mathcal{C}_{\min}(\alpha) \},$$
$$\mathcal{C}_{\text{opt}} = \bigcup \{ \mathcal{C}_{\text{opt}}(\alpha) \mid \alpha \text{ is a bi-infinite simple path} \}.$$

▶ **Definition 15.** A set $\mathcal{C} \subseteq \mathcal{C}(\Gamma)$ of cuts is called a *tree set*, if $\mathcal{C}$ is pairwise nested, closed under complementation and for all $C, D \in \mathcal{C}$ the set $\{ E \in \mathcal{C} \mid C \subseteq E \subseteq D \}$ is finite.

▶ **Proposition 16** ([8])**.** *$\mathcal{C}_{\text{opt}}$ is a tree set.*

▶ **Definition 17.** Let $\mathcal{C}$ be a tree set. We can now define the following relation:

$$C \sim D :\Longleftrightarrow C = D \text{ or } (\overline{C} \subsetneqq D \text{ and } E \in \mathcal{C}, \overline{C} \subsetneqq E \subseteq D \Longrightarrow E = D).$$

Indeed, $\sim$ is an equivalence relation – see e. g. [7]. The intuition behind this definition is: We consider $\mathcal{C}$ as the edge set of a graph, and define two edges to be incident to the same vertex, if no other edge lies "between" them.

▶ **Definition 18.** Let $\mathcal{C}$ be a tree set and let $T(\mathcal{C})$ denote the graph defined by

$$V(T(\mathcal{C})) = \{ [C] \mid C \in \mathcal{C} \}, \qquad E(T(\mathcal{C})) = \mathcal{C}.$$

The incidence maps are defined by $s(C) = [C]$ and $t(C) = [\overline{C}]$. The involution $\overline{C}$ is defined by the complementation $\overline{C} = V(\Gamma) \setminus C$; hence, we do not need to change notation.

The directed edges are in canonical bijection with the pairs $([C], [\overline{C}])$. Indeed, let $C \sim D$ and $\overline{C} \sim \overline{D}$. It follows $C = D$ because otherwise $C \subsetneqq \overline{D} \subsetneqq C$. Thus, $T(\mathcal{C})$ is an undirected graph without self-loops and multi-edges. Indeed, $T(\mathcal{C})$ is a tree [7].

▶ **Theorem 19** ([8, Thm. 5.9])**.** *Let Γ be a connected, k-triangulable, locally finite graph. Let a group G act on Γ such that $G\backslash\Gamma$ is finite and each node stabilizer G_v is finite. Then G acts on the tree $T(\mathcal{C}_{\text{opt}})$ such that all vertex and edge stabilizers are finite and $G\backslash T(\mathcal{C}_{\text{opt}})$ is finite.*

Complete cut sets. By Proposition 16 and Theorem 19, $\mathcal{C}_{\text{opt}}$ is a tree set on which G acts with finitely many orbits such that the vertex stabilizers $G_{[C]} = \{ g \in G \mid gC \sim C \}$ of the structure tree are finite. We shall call a set of cuts with these properties a *complete cut set.*

3 Bounds on the structure tree

In order to prove our main result, we have to show that there exists a "small" graph of groups together with a "small" isomorphism. We start with the structure tree and bound the size of the equivalence classes and the diameter of the boundaries of the cuts in one equivalence class. As before Γ is the Cayley graph of a context-free group G.

Avoiding edge inversion. Let $\mathcal{C}$ be a tree set (e. g. $\mathcal{C} = \mathcal{C}_{\mathrm{opt}}$). We aim to construct a graph of groups as described in Section 2.1 from the structure tree $T(\mathcal{C})$. However, if the action of G on $T(\mathcal{C})$ is with edge inversion, the construction cannot be applied directly. Instead, we switch to a subdivision $\widetilde{T}(\mathcal{C})$ of $T(\mathcal{C})$ by putting a new vertex in the middle of every edge which is inverted (in particular, $V(T(\mathcal{C})) \subseteq V(\widetilde{T}(\mathcal{C}))$). Formally, $\widetilde{T}(\mathcal{C})$ is defined as follows: for every edge C of $T(\mathcal{C})$ with $gC = \overline{C}$ for some $g \in G$ we remove C and $\overline{C}$ and instead add a new vertex $v_{\{C,\overline{C}\}}$ together with edges $C_1, \overline{C}_1, C_2, \overline{C}_2$ with $gC_1 = \overline{C}_2$, $gC_2 = \overline{C}_1$ and $s(C_1) = [C]$, $t(C_1) = v_{\{C,\overline{C}\}}$, $s(C_2) = v_{\{C,\overline{C}\}}$, and $t(C_2) = [\overline{C}]$. We extend this in a G-equivariant way to the whole tree. From now on we work with the tree $\widetilde{T}(\mathcal{C})$.

Reduced cut sets. Given a tree set $\mathcal{C}$ with a finite quotient graph $G\backslash\widetilde{T}(\mathcal{C})$, we obtain a graph of groups as in in Section 2.1. We aim to apply Theorem 12, to bound the number of edges in this graph of groups. However, the graph of groups might not be reduced. In terms of the set of cuts $\mathcal{C}$ this means that $G_{[C]} = G_C$ for some $C \in \mathcal{C}$ and either $[C]$ and $[\overline{C}]$ are not in the same G-orbit or there is some $g \in G$ with $gC = \overline{C}$. (Note that the latter case implies that the action on $T(\mathcal{C})$ is with edge inversion. Thus, the new vertex $v_{\{C,\overline{C}\}}$ is introduced and the condition of being reduced is violated for the vertex $G \cdot [C]$ of the corresponding graph of groups.) Nevertheless, in this case we can switch to a subset $\mathcal{C}' \subseteq \mathcal{C}$ such that the corresponding graph of groups is reduced: if there is some cut $C \in \mathcal{C}$ with $G_{[C]} = G_C$ and either $[C] \notin G \cdot \{[\overline{C}]\}$ or $\overline{C} \in G \cdot \{C\}$, then we can replace $\mathcal{C}$ by $\mathcal{C} \setminus G \cdot \{C, \overline{C}\}$ (in terms of the structure tree this means we collapse the respective edges). If no such $C \in \mathcal{C}$ remains, we have obtained a *reduced* set of cuts $\mathcal{C}'$. Since the number of G-orbits of cuts is finite, this procedure terminates. The following lemmas are straightforward to verify.

▶ **Lemma 20.** *Let $\mathcal{C}$ be a complete cut set and let $\mathcal{C}'$ be the reduced cut set obtained by the above procedure. Then $\mathcal{C}'$ is also complete (i. e., all vertex stabilizers are still finite).*

▶ **Lemma 21.** *Let $\mathcal{C}$ be a reduced cut set and let $\widetilde{T}(\mathcal{C})$ be the associated subdivision of the structure tree without edge inversion. Then the graph of groups built on $G\backslash\widetilde{T}(\mathcal{C})$ is reduced.*

Let Ξ be an upper bound on the order of finite subgroups of G and let Θ be a bound on the number of undirected edges in a reduced graph of groups for G. Notice that by Proposition 11, we have $\Xi \leq d^{12k+10}$ and by Theorem 12 we have $\Theta \leq d^{12k+11}$ where k is the triangulation constant and d the degree of Γ. The following lemma is straightforward to prove using the fact that every orbit of cuts in $\mathcal{C}$ yields an edge in the graph of groups.

▶ **Lemma 22.** *Let $\mathcal{C}$ be a reduced complete set of cuts and let $C \sim D \in \mathcal{C}$. Then we have $|\{\, g \in G \mid gD \sim C \,\}| \leq \Xi$ and $|[C]| \leq 2 \cdot \Theta \cdot \Xi$.*

▶ **Lemma 23.** *Let $\mathcal{C}$ be a tree set of cuts and let G act on $\mathcal{C}$. Let $C \in \mathcal{C}$ and $C \in P \subseteq [C]$. Then $P \neq [C]$ if and only if there is some $E \in [C] \setminus P$ with $d(\partial E, \bigcup_{D \in P} \partial D) \leq 1$.*

Proof. The if-part is clear. Thus, let $P \neq [C]$. Then there is some $E \in [C] \setminus P$. Since $\overline{E} \subsetneq D$ for all $D \in P$, we have $\emptyset \neq \overline{E} \subseteq \bigcap_{D \in P} D$. Now, if $\partial \overline{E} \subseteq \bigcup_{D \in P} \partial D$, we are done. Otherwise, there is some vertex $u \in \overline{E} \subseteq \bigcap_{D \in P} D$ with $d(u, \bigcup_{D \in P} \partial D) \geq 1$. Since $\partial\big(\bigcap_{D \in P} D\big) \subseteq \bigcup_{D \in P} \partial D$, we also find a vertex $v \in \bigcap_{D \in P} D$ with $d(v, \bigcup_{D \in P} \partial D) = 1$ by following a path from u to $\partial\big(\bigcap_{D \in P} D\big)$ inside $\bigcap_{D \in P} D$. Notice that, in particular, we have

$$v \notin \beta D \cup \overline{D} \qquad \text{for all } D \in P. \tag{2}$$

Now since Γ is vertex-transitive, we can find some cut $\widetilde{E} \in \mathcal{C}$ such that $v \in \beta\widetilde{E}$. After possibly exchanging $\widetilde{E}$ with its complement, we can assume that $\widetilde{E} \subsetneq C$ or $\widetilde{E} \subseteq \overline{C}$. The latter would imply $v \in \beta\widetilde{E} \subseteq \beta C \cup \overline{C}$ contradicting (2). Moreover, for any other $D \in P$, we have $\widetilde{E} \subseteq D$ because all other possibilities for $\widetilde{E}$ and D being nested lead to a contradiction:

- if $\overline{D} \subsetneq \widetilde{E}$, then $\overline{D} \subsetneq \widetilde{E} \subsetneq C$ contradicting $D \sim C$,
- if $D \subsetneq \widetilde{E}$, then $D \subseteq \widetilde{E} \subseteq C$ and $\overline{D} \subseteq C$ contradicting $\overline{C} \neq \emptyset$,
- if $\widetilde{E} \subseteq \overline{D}$, then $v \in \beta\widetilde{E} \subseteq \beta D \cup \overline{D}$ contradicting (2).

Thus, $\widetilde{E} \subseteq \bigcap_{D\in P} D$. Let $E \in \mathcal{C}$ be minimal with respect to inclusion such that $\widetilde{E} \subseteq \overline{E} \subsetneq C$. Then $E \sim C$, but $E \notin P$ because $v \in \beta\widetilde{E} \subseteq \beta E \cup \overline{E}$.

It remains to verify that $d(\partial E, \bigcup_{D\in P} \partial D) \leq 1$. Let $w \in \partial D$ for some $D \in P$ a vertex with $d(w, v) = 1$. Then, we have $w \in \beta D \cup \overline{D} \subseteq \beta E \cup E$. Consider the two cases: $v \in \overline{E}$ and $v \in \partial E$. If $v \in \overline{E}$, then $w \in \beta E \cap \partial D$ and hence $d(\partial E, \bigcup_{D\in P} \partial D) \leq 1$. If $v \in \partial E$, then $d(\partial E, \bigcup_{D\in P} \partial D) \leq d(v, w) = 1$. ◀

Now, an easy inductive argument shows the next lemma.

▶ **Lemma 24.** *Let $\mathcal{C}$ be a complete set of cuts and $R \in \mathbb{N}$ such that $\operatorname{diam} \beta C \leq R$ for all $C \in \mathcal{C}$. Let $C \in \mathcal{C}$, then* $\operatorname{diam}\Big(\bigcup_{C\sim D} \beta C\Big) \leq (R+1) \cdot |[C]|$.

▶ **Lemma 25.** *Let Γ be the Cayley graph of G. Moreover, assume that*

- *R bounds the diameter of the boundary of minimal cuts,*
- *Θ bounds the number of undirected edges of a reduced graph of groups for G,*
- *Ξ is an upper bound on the size of finite subgroups of G.*

Then there exists a graph of groups $\mathcal{G}$ over Y and an isomorphism $\varphi : \pi_1(\mathcal{G}, T) \to G$ with

(i) $|V(Y)| \leq \Theta + 1$,
(ii) $|G_P| \leq \Xi$ *for all* $P \in V(Y)$,
(iii) $|\varphi(a)| \leq 4(R+1) \cdot (\Theta+1)^2 \cdot \Xi$ *for every* $a \in \bigcup_{P\in V(Y)} G_P \cup E(Y)$.

Points i and ii of Lemma 25 are immediate. The proof iii starts with the set of optimal cuts. As described at the beginning of this section, one can switch to a reduced, complete subset $\mathcal{C}$ yielding a reduced graph of groups over $G\backslash\widetilde{T}(\mathcal{C})$ by Lemmas 20 and 21. Following the construction of the graph of groups in Section 2.1, we can choose representatives for $V(Y) \subseteq V(\widetilde{T}(\mathcal{C}))$ for $G\backslash V(\widetilde{T}(\mathcal{C}))$ such that $\partial C \subseteq B(\Lambda)$ for any $C \in P$ and $P \in V(Y)$ with $\Lambda = 2(R+1) \cdot (\Theta+1) \cdot \Theta \cdot \Xi + \Theta$. This bound follows from Lemmas 24 and 22. Thus, we have a graph of groups and it remains to bound the size of the isomorphism φ. Consider the action of G on its Cayley graph Γ: every $g \in G_P$ for $P \in V(Y)$ maps a vertex from $B(\Lambda)$ to another vertex in $B(\Lambda)$ (namely all vertices in $\bigcup_{D\in P} \partial D$). Likewise the image of an edge D maps $\partial\overline{D} \subseteq B(\Lambda+1)$ into $B(\Lambda)$. Since the action is free, iii follows. For details, see [23].

4 Stronger bounds for virtually free presentations

Let us start with a virtually free group G given as a free subgroup $F(X)$ of finite index and a system of representatives S of $F\backslash G$. That means every group element can be written in a unique way as xs with $x \in F(X)$ and $s \in S$. Moreover, this normal form can be computed in linear time from an arbitrary word by successively applying "commutation rules" of letters from S and $X \cup \overline{X} \cup S$ to the word. This gives a virtually free presentation. For this special case, we can derive stronger bounds on the triangulation constant k and other parameters.

Formally, a *virtually free presentation* $\mathcal{V}$ for G is given by the following data:

- finite sets $X, \overline{X}, S$, where $X \cup \overline{X}$ is a symmetric alphabet and $(X \cup \overline{X}) \cap S = \emptyset$,
- for all $y \in X \cup \overline{X}$, $r, t \in S$, there are words $x_{r,y}, x_{r,t} \in (X \cup \overline{X})^*$, $s_{r,y}, s_{r,t} \in S \cup \{1\}$ with

$$ry =_G x_{r,y}s_{r,y} \qquad rt =_G x_{r,t}s_{r,t} \tag{3}$$

fulfilling two properties:

(i) for all $r \in S$ there is some $r' \in S$ such that $s_{r',r} = 1$ (i. e., G is a group),

(ii) the equations (3), oriented from left to right, together with the free reductions $x\overline{x} = 1$ for $x \in X \cup \overline{X}$ form a *confluent* rewriting system (for a definition, see e. g. [2, 13]).

We write S^1 for $S \cup \{1\}$. Clearly the associated rewriting system is terminating (noetherian), $F(X)$ is a subgroup of G, $G = F(X) \cdot S^1$, and $F(X) \cap S = \emptyset$ (hence S^1 is a system of right-representatives for $F(X)$). Note that properties i and ii can be checked in polynomial time. Using this confluent rewriting system, every $g \in G$ can be uniquely written in its *normal form* $g = xs$ where $x \in (X \cup \overline{X})^*$ is a freely reduced word and $s \in S^1$. Given any word in $(X \cup \overline{X} \cup S)^*$, the normal form can be computed in linear time from left to right by applying the identities (3) and reducing freely. This is the computation of a deterministic pushdown automaton for the word problem of G:

▶ **Lemma 26.** *Let G be the group defined by a virtually free presentation $\mathcal{V}$. Then a deterministic pushdown automaton for* $\mathrm{WP}(G)$ *can be computed in polynomial time.*

Notice that a finite extension of a free group is a special case of a virtually free presentation where $F(X)$ is a normal subgroup of G (i. e., $s_{r,y} = r$ for all $r \in S$, $y \in X \cup \overline{X}$). We assume that $\mathcal{V}$ is written down in a naive way as input for algorithms: there is a table which for all $a \in X \cup \overline{X} \cup S$ and $r \in S$ contains a word $x_{r,a}$ and some $s_{r,a} \in S$. The size (number of letters) of this table is $|S| \cdot (2\,|X| + |S|) \cdot \max\left\{\, |x_{r,a}| + 1 \;\middle|\; a \in X \cup \overline{X} \cup S,\, r \in S \,\right\}$. Up to logarithmic factors, this is the number of bits required to write down $\mathcal{V}$ this way.

We can always add a disjoint copy of formal inverses $\overline{S}$ of S representing S^{-1} in G. Note that for $\overline{s} \in \overline{S}$ this yields the rule $r\overline{s} = x_{r,\overline{s}}\, s_{r,\overline{s}}$ for some $s_{r,\overline{s}} \in S$ where $x_{r,\overline{s}} = x^{-1}_{s_{r,\overline{s}},s}$. In particular, $|x_{r,\overline{s}}| \le \max\left\{\, |x_{r,a}| \;\middle|\; a \in X \cup \overline{X} \cup S,\, r \in S \,\right\}$. We define the size of $\mathcal{V}$ as $\|\mathcal{V}\| = \left|S^1\right| \cdot (2\,|X| + 2\,|S|) \cdot \max\left\{\, |x_{r,a}| + 1 \;\middle|\; a \in X \cup \overline{X} \cup S,\, r \in S \,\right\}$.

Whenever we talk about a group G given as a virtually free presentation, we denote $\Sigma = X \cup \overline{X} \cup S \cup \overline{S}$. The Cayley graph $\Gamma = \Gamma_\Sigma(G)$ is defined with respect to this alphabet. In particular, its degree is bounded by $\|\mathcal{V}\|$. The following lemma is easy to prove by considering the sequence of normal forms $x_0 s_0, \ldots, x_n s_n$ of a closed path: then $x_0, \ldots, x_n$ is $2\,\|\mathcal{V}\|$-triangulable in the Cayley graph of $F(X)$.

▶ **Lemma 27.** *Let G be the group defined by a virtually free presentation $\mathcal{V}$ and let Γ be its Cayley graph. Then Γ is k-triangulable for $k = 2\,\|\mathcal{V}\| + 2$.*

▶ **Lemma 28.** *Let G be the group defined by a virtually free presentation $\mathcal{V}$. Then for every finite subgroup $H \le G$, we have $|H| \le \left|S^1\right|$. Hence, in particular, $|H| \le \|\mathcal{V}\|$.*

▶ **Lemma 29.** *Let G be the group defined by a virtually free presentation $\mathcal{V}$. Then the number of edges of a reduced graph of groups for G with finite vertex groups is at most $\|\mathcal{V}\|$.*

The proof of Lemma 29 is almost a verbatim repetition of the proof of [22, Thm. 1.4].

▶ **Lemma 30.** *Let G be the group defined by a virtually free presentation $\mathcal{V}$ and let Γ be its Cayley graph. Then every minimal cut in Γ is a K-cut for $K = \|\mathcal{V}\|^2$.*

The proof of Lemma 30 is based on the fact that all cuts of the form $C_x = \left\{\, ys \;\middle|\; s \in S, x \text{ is a prefix of } y \,\right\}$ for $x \in (X \cup \overline{X})^*$ satisfy $|\vec{\delta} C_x| \le \|\mathcal{V}\|^2$ and that all bi-infinite simple paths which are split by some cut also are split by some cut of the form C_x.

5 Main results: computing graphs of groups

▶ **Lemma 31.** *The uniform rational subset membership problem for virtually free groups given as virtually free presentation or as context-free grammar for the word problem can be decided in polynomial time. More precisely, the input is given as*

- *either a virtually free presentation $\mathcal{V}$ or a context-free grammar $\mathbb{G} = (V, \Sigma, P, S)$ for the word problem of a group G,*
- *a rational subset of G given as non-deterministic finite automaton or regular expression over Σ (in the case of a virtually free presentation Σ is defined as in Section 4),*
- *a word $w \in \Sigma^*$.*

The question is whether w is contained in the rational subset of G.

The proof of Lemma 31 is straightforward (for details see [23]): let $p : \Sigma^* \to G$ denote the canonical projection. Then $p(w) \in p(L) \iff 1 \in p(w^{-1}L) \iff \mathrm{WP}(G) \cap w^{-1}L \neq \emptyset$. The latter can be tested in polynomial time by standard facts from formal language theory.

▶ **Proposition 32.** *The following problem is in* P*: Given a virtually free group G either as virtually free presentation $\mathcal{V}$ or as context-free grammar $\mathbb{G}$ for its word problem and a graph of groups $\mathcal{G}$ over the graph Y (with vertex groups as multiplication tables, i. e., for all $g, h \in G_P$ the product gh is written down explicitly) together with a homomorphism $\varphi : \Delta^* \to \Sigma^*$ (where $\Delta = \bigcup_{P \in V(Y)} G_P \cup E(Y)$ and Σ is the alphabet for G defined by $\mathcal{V}$ (resp. $\mathbb{G}$)), decide whether φ induces an isomorphism $\pi_1(\mathcal{G}, T) \to G$.*

Proof. We verify that φ induces a homomorphism $\tilde{\varphi} : \pi_1(\mathcal{G}, T) \to G$ and that $\tilde{\varphi}$ is injective and surjective.

Testing that φ really induces a *homomorphism* reduces to the word problem for the group G, which can be solved in polynomial time: for every relation $a_1 \cdots a_m = 1$ of $\pi_1(\mathcal{G}, T)$ test whether $\varphi(a_1) \cdots \varphi(a_m) = 1$ in G. Testing that $\tilde{\varphi}$ is *surjective* reduces to polynomially many membership-problems for rational subsets of G: for all $a \in \Sigma$ test whether a is contained in the rational subset $\{ \tilde{\varphi}(g) \mid g \in \Delta \}^*$. By Lemma 31 this can be done in polynomial time.

It remains to test whether $\tilde{\varphi}$ is *injective*. Let $\pi : \Delta^* \to F(\mathcal{G})$ and $\psi : F(\mathcal{G}) \to \pi_1(\mathcal{G}, T)$ denote the canonical projections (note that ψ induces an isomorphism $\pi_1(\mathcal{G}, P) \xrightarrow{\sim} \pi_1(\mathcal{G}, T)$). Let $\mathcal{R} \subseteq \Delta^*$ denote the set of reduced words. With slight abuse of notation we use $\pi_1(\mathcal{G}, P)$ also to denote the set of words $g_0 y_1 \cdots g_{n-1} y_n g_n \in \Delta^*$ where $y_1 \cdots y_n$ is a closed path based at P and $g_i \in G_{s(y_{i+1})}$ for $0 \leq i < n$ and $g_n \in G_P$. Testing that $\tilde{\varphi}$ is injective amounts to test whether the language $L = \left(\pi^{-1}(\psi^{-1}(\tilde{\varphi}^{-1}(1))) \cap \pi_1(\mathcal{G}, P) \cap \mathcal{R}\right) \setminus \{1\} \subseteq \Delta^*$ is empty because 1 is the only reduced word in $\pi_1(\mathcal{G}, P)$ representing the identity, by Lemma 4.

Notice that $\pi^{-1}(\psi^{-1}(\tilde{\varphi}^{-1}(1))) = \varphi^{-1}(\mathrm{WP}(G))$. Since $\mathrm{WP}(G)$ is context-free (for virtually free presentations, see Lemma 26) and since context-free languages are closed under inverse homomorphism, $\varphi^{-1}(\mathrm{WP}(G))$ is a context-free language – and a pushdown automaton for it can be computed in polynomial time from the pushdown automaton for $\mathrm{WP}(G)$ (see e. g. [12]). Thus, L is a context-free language and we obtain a pushdown automaton for L, which can be tested for emptiness in polynomial time (see e. g. [12]). ◀

▶ **Theorem 33.** *The following problem is in* $\mathsf{NTIME}(2^{2^{\mathcal{O}(N)}})$*:*
Input: *a context-free grammar $\mathbb{G} = (V, \Sigma, P, S)$ in Chomsky normal form with $\|\mathbb{G}\| \leq N$ which generates the word problem of a group G,*
Compute *a graph of groups with finite vertex groups and fundamental group G.*

Note that if $\mathbb{G}$ is not in Chomsky normal form, it can be transformed into Chomsky normal form in quadratic time. In this case the graph of groups can be computed in $\mathsf{NTIME}(2^{2^{\mathcal{O}(N^2)}})$.

Table 1 Summary of appearing constants. The third column shows a bound in terms of the size of a context-free grammar $\mathbb{G}$ for the word problem (due to Lemma 8, Lemma 13, Proposition 11, and Theorem 12), the fourth column shows a bound in terms of the size of a virtually free presentation $\mathcal{V}$ (due to Lemma 27, Lemma 30, Lemma 10, Lemma 28, and Lemma 29).

N	size of the input	$\lVert\mathbb{G}\rVert$	$\lVert\mathcal{V}\rVert$
d	degree of Γ	N	N
k	triangulation constant	2^{N+2}	$2N+2$
K	maximal weight of a minimal cut	d^{3k+3}	N^2
$R = \frac{3kK}{2}$	maximal diameter of the boundary of a minimal cut	$\frac{3k}{2}d^{3k+3}$	$3(N+1)N^2$
Ξ	maximum cardinality of a finite subgroup	d^{12k+10}	N
Θ	maximum number of edges in a reduced graph of groups	d^{12k+11}	N

▶ **Theorem 34.** *The following problem is in* NP*:*
Input: *a group G as a virtually free presentation,*
Compute *a graph of groups with finite vertex groups and fundamental group G.*

The proofs of Theorem 34 and Theorem 33 are now straightforward: guess a graph of groups and a map $\varphi : \Delta \to \Sigma^*$ and use Proposition 32 to check that it induces an isomorphism. By Lemma 25 and Table 1, such a guess can be made within the time bounds of the theorems.

6 Slide moves and the isomorphism problem

Given two groups G_1 and G_2 one can calculate the respective graph of groups and then check with Krstić's algorithm by ([15]) whether their fundamental groups are isomorphic. A closer analysis shows that this algorithm runs in polynomial space. As the description is quite involved, we follow a different approach based on Forester's theory of deformation spaces [10, 4].

Let $\mathcal{G}$ be a graph of groups over Y. A *slide move* is the following operation on $\mathcal{G}$: let G_P be a vertex group and G_x, G_y edge groups with $s(x) = s(y) = P$. If G_x^x (the image of G_x in G_P) can be conjugated by an element of G_P into G_y^y i. e., there is some $g \in G_P$ such that $g^{-1}G_x^x g \leq G_y^y$, then x can be slid along y to $Q = t(y)$, i. e., $s(x)$ is changed to Q. The new inclusion of $G_x \to G_Q$ is then given by $\iota_{\overline{y}} \circ \iota_y^{-1} \circ c_g \circ \iota_x$ where ι_x is the inclusion $G_x \to G_P$ (likewise for $\iota_{\overline{y}}, \iota_y$) and c_g is the conjugation with g (i. e., $h \mapsto g^{-1}hg$). A slide move induces an isomorphism φ of the fundamental groups of the two graph of groups by $\varphi(h) = h$ for $h \in G_R$, $R \in V(Y)$, and $\varphi(z) = z$ for $z \in E(Y) \setminus \{x, \overline{x}\}$ and $\varphi(x) = gyx$. The following result is an immediate consequence of [10, Thm. 1.1] and [11, Thm. 7.2] (resp. [4, Cor. 3.5]). Since we are not aware of an explicit reference, we give the details in [23].

▶ **Proposition 35.** *Let $\mathcal{G}_1$ and $\mathcal{G}_2$ be reduced finite graphs of groups with finite vertex groups. Then $\pi_1(\mathcal{G}_1, P_1) \cong \pi_1(\mathcal{G}_2, P_2)$ if and only if $\mathcal{G}_1$ can be transformed into $\mathcal{G}_2$ by a sequence of slide moves.*

Clearly, any sequence of slide moves can be performed in linear space. By guessing a sequence of slide moves transforming $\mathcal{G}_1$ into $\mathcal{G}_2$, we obtain the following corollary.

▶ **Corollary 36.** *Given two graph of groups $\mathcal{G}_1$ and $\mathcal{G}_2$ where all vertex groups are given as full multiplication tables, it can be checked in* $\mathsf{NSPACE}(\mathcal{O}(n))$ *whether $\pi_1(\mathcal{G}_1, P_1) \cong \pi_1(\mathcal{G}_2, P_2)$.*

In combination with Theorem 33 (and Savitch's theorem) and Theorem 34 this gives an algorithm to solve the isomorphism problem for virtually free groups:

▶ **Theorem 37.** *The isomorphism problem for context-free groups is in* DSPACE($2^{2^{\mathcal{O}(N)}}$)*. More precisely, the input is given as two context-free grammars of size at most N which are guaranteed to generate word problems of groups.*

▶ **Theorem 38.** *The isomorphism problem for virtually free groups given as a virtually free presentation is in* PSPACE*.*

7 Conclusion and open questions

We have shown that the isomorphism problem for virtually free groups is in PSPACE (resp. DSPACE($2^{2^{\mathcal{O}(N)}}$)) depending on the type of input – thus, improving the previous bound (primitive recursive) significantly. The following questions remain open:

- What is the complexity of the isomorphism problem for virtually free groups given as an arbitrary presentation?
- Is the doubly exponential bound $n^{12\cdot 2^n+10}$ on the size of finite subgroups tight or is there a bound $2^{p(n)}$ for some polynomial p? This is closely related to another question:
- What is the minimal size of a context-free grammar of the word problem of a finite group? Can it be $\log\log(n)$ where n is the size of the group?
- Is there a polynomial bound on the number of slide moves necessary to transform two graphs of groups with isomorphic fundamental groups into each? This would lead to an NP algorithm for the isomorphism problem with virtually free presentations as input. We conjecture, however, that this is not true.

References

1 Yago Antolin. On Cayley graphs of virtually free groups. *Groups – Complexity – Cryptology*, 3:301–327, 2011. doi:10.1515/gcc.2011.012.
2 Ron Book and Friedrich Otto. *String-Rewriting Systems.* Springer-Verlag, 1993.
3 W. W. Boone. The Word Problem. *Ann. of Math.*, 70(2):207–265, 1959.
4 Matt Clay and Max Forester. Whitehead moves for G-trees. *Bull. Lond. Math. Soc.*, 41(2):205–212, 2009. doi:10.1112/blms/bdn118.
5 François Dahmani and Vincent Guirardel. The isomorphism problem for all hyperbolic groups. *Geom. Funct. Anal.*, 21(2):223–300, 2011. doi:10.1007/s00039-011-0120-0.
6 Max Dehn. Ueber unendliche diskontinuierliche Gruppen. *Math. Ann.*, 71:116–144, 1911.
7 Warren Dicks and Martin J. Dunwoody. *Groups acting on graphs.* Cambridge University Press, 1989.
8 Volker Diekert and Armin Weiß. Context-Free Groups and Their Structure Trees. *International Journal of Algebra and Computation*, 23:611–642, 2013. doi:10.1142/S0218196713500124.
9 Volker Diekert and Armin Weiß. Context-Free Groups and Bass-Serre Theory. In Juan González-Meneses, Martin Lustig, and Enric Ventura, editors, *Algorithmic and Geometric Topics Around Free Groups and Automorphisms*, Advanced Courses in Mathematics - CRM Barcelona. Birkhäuser, Basel, Switzerland, 2017. doi:10.1007/978-3-319-60940-9.
10 Max Forester. Deformation and rigidity of simplicial group actions on trees. *Geom. Topol.*, 6:219–267, 2002. doi:10.2140/gt.2002.6.219.
11 Vincent Guirardel and Gilbert Levitt. Deformation spaces of trees. *Groups Geom. Dyn.*, 1(2):135–181, 2007. doi:10.4171/GGD/8.
12 John E. Hopcroft and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages and Computation.* Addison-Wesley, 1979.

13 Matthias Jantzen. *Confluent String Rewriting*, volume 14 of *EATCS Monographs on Theoretical Computer Science*. Springer-Verlag, 1988.

14 A. Karrass, A. Pietrowski, and D. Solitar. Finite and infinite cyclic extensions of free groups. *Journal of the Australian Mathematical Society*, 16(04):458–466, 1973. doi:10.1017/S1446788700015445.

15 Sava Krstic. Actions of finite groups on graphs and related automorphisms of free groups. *Journal of Algebra*, 124:119–138, 1989. doi:10.1016/0021-8693(89)90154-3.

16 David E. Muller and Paul E. Schupp. Groups, the theory of ends, and context-free languages. *J. Comput. Syst. Sci.*, 26:295–310, 1983. doi:10.1016/0022-0000(83)90003-X.

17 David E. Muller and Paul E. Schupp. The theory of ends, pushdown automata, and second-order logic. *Theor. Comput. Sci.*, 37(1):51–75, 1985. doi:10.1016/0304-3975(85)90087-8.

18 P. S. Novikov. On the algorithmic unsolvability of the word problem in group theory. *Trudy Mat. Inst. Steklov*, pages 1–143, 1955. In Russian.

19 Christos H. Papadimitriou. *Computational Complexity*. Addison Wesley, 1994.

20 Alan L. Selman. A taxonomy of complexity classes of functions. *J. Comput. Syst. Sci.*, 48(2):357–381, 1994. doi:10.1016/S0022-0000(05)80009-1.

21 Géraud Sénizergues. An effective version of Stalling's theorem in the case of context-free groups. In Andrzej Lingas, Rolf G. Karlsson, and Svante Carlsson, editors, *Proc. 20th International Colloquium Automata, Languages and Programming (ICALP 93), Lund (Sweden)*, volume 700 of *Lecture Notes in Computer Science*, pages 478–495. Springer-Verlag, 1993.

22 Géraud Sénizergues. On the finite subgroups of a context-free group. In Gilbert Baumslag, David Epstein, Robert Gilman, Hamish Short, and Charles Sims, editors, *Geometric and Computational Perspectives on Infinite Groups*, number 25 in DIMACS series in Discrete Mathematics and Theoretical Computer Science, pages 201–212. Amer. Math. Soc., 1996.

23 Géraud Sénizergues and Armin Weiß. The isomorphism problem for finite extensions of free groups is in PSPACE. *CoRR*, To appear soon in arXiv, 2018. URL: https://arxiv.org/abs.

24 Jean-Pierre Serre. *Trees*. Springer, 1980. French original 1977.

25 Carsten Thomassen and Wolfgang Woess. Vertex-transitive graphs and accessibility. *J. Comb. Theory Ser. B*, 58(2):248–268, 1993. doi:10.1006/jctb.1993.1042.