Intersection problem for Droms RAAGs

Jordi Delgado∗1, Enric Ventura†2, and Alexander Zakharov‡3

1,3Centre of Mathematics, University of Porto
2Department de Matemàtiques, Universitat Politècnica de Catalunya

February 5, 2022

Abstract

We solve the subgroup intersection problem (SIP) for any RAAG G of Droms type (i.e., with defining graph not containing induced squares or paths of length 3): there is an algorithm which, given finite sets of generators for two subgroups H, K ⩽ G, decides whether H ∩ K is finitely generated or not, and, in the affirmative case, it computes a set of generators for H ∩ K. Taking advantage of the recursive characterization of Droms groups, the proof consists in separately showing that the solvability of SIP passes through free products, and through direct products with free-abelian groups. We note that most of RAAGs are not Howson, and many (e.g. \( \mathbb{F}_2 \times \mathbb{F}_2 \)) even have unsolvable SIP.

1 Introduction

In group theory, the study of intersections of subgroups has been recurrently considered in the literature. Roughly speaking, the problem is “given subgroups H, K ⩽ G, find H ∩ K”. However, in the context of Geometric Group Theory, where groups may be infinite, or even non finitely generated, one needs to be more precise about the word find, specially if one is interested in the computational point of view.

A group is said to satisfy Howson’s property — or to be Howson, for short — if the intersection of any two (and so, finitely many) finitely generated subgroups is again finitely generated.

Classical examples of Howson groups include free-abelian, and free groups. In \( \mathbb{Z}^m \) Howson’s property is trivial, whereas for free groups it was proved by Howson himself in [15], where he also gave an algorithm to compute generators for the intersection.

Not far from these groups one can find examples without the Howson property: consider the group \( \mathbb{F}_2 \times \mathbb{Z} = \langle a, b \mid - \rangle \times \langle t \mid - \rangle \) and the subgroups H = \( \langle a, b \rangle \) and K = \( \langle ta, b \rangle \); both

* jdelgado@crm.cat
† enric.ventura@upc.edu
‡ zakhar.sasha@gmail.com
are clearly 2-generated but \( H \cap K = \{ w(a, b) : |w|_a = 0 \} = \langle b \rangle \leqslant F_2 \), which is not finitely generated. In this context, it is natural to consider the following decision problems.

**Subgroup intersection problem, SIP(\( G \)).** Given words \( u_1, \ldots, u_n, v_1, \ldots, v_m \) in the generators of \( G \), decide whether the subgroup intersection \( \langle u_1, \ldots, u_n \rangle \cap \langle v_1, \ldots, v_m \rangle \) is finitely generated or not; and, in the affirmative case, compute a generating set for this intersection.

**Coset intersection problem, CIP(\( G \)).** Given a finite set of words \( w, w', u_1, \ldots, u_n, v_1, \ldots, v_m \) in the generators of \( G \), decide whether the coset intersection \( w\langle u_1, \ldots, u_n \rangle \cap w'\langle v_1, \ldots, v_m \rangle \) is empty or not; and in the negative case, compute a coset representative.

In [8], Delgado–Ventura prove that direct products of free-abelian and free groups have both SIP and CIP solvable. The goal of the present paper is to extend the algebraic arguments given there, in order to achieve similar properties for a much wider family of groups. To this end it is convenient to consider the following variations for a general finitely presented group \( G \).

**Twofold intersection problem, TIP(\( G \)).** Solve both SIP(\( G \)) and CIP(\( G \)).

**Extended subgroup intersection problem, ESIP(\( G \)).** Given a finite set of words \( u_1, \ldots, u_n, v_1, \ldots, v_m, w, w' \) in the generators of \( G \), decide whether the intersection of the subgroups \( H = \langle u_1, \ldots, u_n \rangle \) and \( K = \langle v_1, \ldots, v_m \rangle \) is finitely generated or not; and in the affirmative case: (i) compute a generating set for \( H \cap K \), and (ii) decide whether the coset intersection \( wH \cap w'K \) is empty or not (denoted by \( \text{CIP}_{\text{fg}} \)), computing a coset representative in case it is not.

The main result in this paper is about *finitely generated* PC-groups (a.k.a. right-angled Artin groups, or RAAGs). This prominent class of groups is closely linked to some crucial examples of groups, notably Bestvina and Brady’s example of a group which is homologically finite (of type \( \text{FP} \)) but not geometrically finite (in fact not of type \( F_2 \)), and Mihailova’s example of a group with unsolvable subgroup membership problem. More recently D.Wise et al. developed a method of showing that a vast amount of groups are virtual subgroups of RAAGs. Wise used this method to solve some well-known problems in group theory, like Baumslag’s conjecture on residual finiteness of one-relator groups with torsion. Furthermore, building on the work of Wise and Kahn–Markovic, I. Agol proved the famous virtually fibred conjecture (the last main open problem in 3-manifold theory due to Thurston), by showing that fundamental groups of closed, irreducible, hyperbolic 3-manifolds are virtual subgroups of RAAGs.

**Definition 1.1.** A group \( G \) is said to be *partially commutative* (a PC-group, for short) if it admits a presentation of the form

\[
\langle X \mid [x_i, x_j] = 1, \text{ whenever } \{x_i, x_j\} \in E \rangle,
\]

for some (not necessarily finite) simple graph \( \Gamma = (X, E) \). In this case, we say that \( G \) is presented by the *commutation graph* \( \Gamma \) and write \( G = G_{\Gamma} \). Then, we say that (1) is a *graphical presentation* for \( G \), and \( X \) is a *graphical generating set* (or *basis*) for \( G \). In the f.g. case (i.e., when \( X \) is finite) we shall refer to \( G_{\Gamma} \) as a *right-angled Artin group* (a RAAG, for short).
A subgraph $\Delta$ of a graph $\Gamma = (X,E)$ is said to be full if it has exactly the edges that appear in $\Gamma$ over its vertex set, say $Y \subseteq X$; in this case, $\Delta$ is called the full subgraph of $\Gamma$ spanned by $Y$ and we write $\Delta \leq \Gamma$. When none of the graphs belonging to a certain family $\mathcal{F}$ appear as a full subgraph of $\Gamma$, we say that $\Gamma$ is $\mathcal{F}$-free.

PC-groups can be thought as a family of groups interpolating between two extreme cases: free-abelian groups (presented by complete graphs), and free groups (presented by edgeless graphs); having as graphical generating set precisely the standard free-abelian and free bases, respectively. More generally, the PC-group presented by the disjoint union of graphs $\Gamma \sqcup \Delta$ is the free product $G_{\Gamma} \ast G_{\Delta}$, and the PC-group presented by the join of graphs $\Gamma \vee \Delta$ (obtained by adding to $\Gamma \sqcup \Delta$ every edge joining a vertex in $\Gamma$ to a vertex in $\Delta$) is the direct product $G_{\Gamma} \times G_{\Delta}$.

Despite the extreme (free and free-abelian) cases being subgroup-closed, this is not the case for PC-groups. Droms characterized the finitely generated PC-groups having this property in the following well known result.

**Theorem 1.2** (C. Droms, 1987, [11]). Let $\Gamma$ be a finite graph. Then, every subgroup of $G_{\Gamma}$ is again a (possibly non finitely generated) PC-group if and only if $\Gamma$ is $\{P_4,C_4\}$-free.

Here, $P_n$ stands for the path graph on $n$ vertices; and $C_n$ stands for the cycle graph on $n$ vertices. Accordingly, we say that a graph is a Droms graph if it is finite and $\{P_4,C_4\}$-free, and a PC-group is a Droms group if it is presented by a Droms graph.

**Remark 1.3.** We note that finite $\{P_4,C_4\}$-free graphs have received diverse denominations throughout the literature, including comparability graphs of forests (in [29]), transitive forests (in [26]), trivially perfect graphs (in [4]), and quasi-threshold graphs (in [19]).

### 1.1 Results

The main result in the present paper is the following theorem.

**Theorem 1.4.** Every Droms group has solvable ESIP (and, in particular, solvable SIP).

The strategy of the proof arises from the following crucial lemma given by Droms on the way of proving Theorem 1.2: Every nonempty Droms graph is either disconnected, or it contains a central vertex (i.e., one vertex adjacent to any other vertex).

This easily provides the following recursive definition of the Droms family (with both the graphical and the algebraic counterparts):

**Corollary 1.5** (Droms, [11]). The family of Droms graphs (resp., Droms groups) can be recursively defined as the smallest family $\mathcal{D}$ (resp., $\mathcal{D}$) satisfying the following rules:

[D1] $K_0 \in \mathcal{D}$; 
[D2] $\Gamma_1, \Gamma_2 \in \mathcal{D} \Rightarrow \Gamma_1 \sqcup \Gamma_2 \in \mathcal{D}$; 
[D3] $\Gamma \in \mathcal{D} \Rightarrow K_1 \vee \Gamma \in \mathcal{D}$.

[D1] $(1) \in \mathcal{D}$; 
[D2] $G_1, G_2 \in \mathcal{D} \Rightarrow G_1 \ast G_2 \in \mathcal{D}$; 
[D3] $G \in \mathcal{D} \Rightarrow \mathbb{Z} \times G \in \mathcal{D}$.

3
Our proof of Theorem 1.4 is based on the following preservability results for the intersection properties we are interested in.

**Theorem 1.6.** Let $G$ be a Droms group. If $G$ has solvable SIP, then $\mathbb{Z}^m \times G$ also has solvable SIP.

**Theorem 1.7.** Let $G$ be a Droms group. If $G$ has solvable ESIP, then $\mathbb{Z}^m \times G$ also has solvable ESIP.

**Theorem 1.8.** If two finitely presented groups $G_1$ and $G_2$ have solvable ESIP, then their free product $G_1 \ast G_2$ also has solvable ESIP.

**Theorem 1.9.** If two finitely presented groups $G_1$ and $G_2$ have solvable TIP, then their free product $G_1 \ast G_2$ also has solvable TIP.

To prove Theorems 1.6 and 1.7 we extend the techniques in [8] from free groups to RAAGs; and to prove Theorems 1.8 and 1.9 we use Ivanov’s techniques to understand and work with subgroups of free products (see [16, 17]). Both are relatively long and technical arguments, each requiring several pages of preliminary considerations. For the sake of clarity, we decided to include them instead of writing a shorter preprint but harder to read.

Our main result (Theorem 1.4) can be seen as a partial generalization of Theorem 2.7(i) by Kapovich–Weidmann–Myasnikov in the sense that we prove a stronger thesis than them (namely, ESIP instead of MP, see Figure 1) for a smaller class of groups (Droms instead of coherent PC-groups). In this situation, it is interesting to ask the following questions.

**Question 1.** Does the group $G_{P_{4}}$ have solvable SIP? Is it true that a RAAG have solvable SIP if and only if it is Droms?

Particularly suggestive for us is the result from Aalbersberg–Hoogeboom [1] stating that the intersection problem for a partially commutative monoid is solvable if and only if its commutation graph is Droms. The situation is intriguingly similar to that for the MP.

The paper is organized as follows. In Section 2 we establish the necessary background and references about Droms groups and algorithmic issues, and we prove the main result, Theorem 1.4, modulo Theorems 1.6, 1.7, 1.9, and 1.8. Then, in Section 3, we study the direct product case, proving Theorems 1.6 and 1.7; and finally, in Section 4, we consider the free product situation proving Theorems 1.9 and 1.8. See [6, Part III] for a more detailed version of these results.

## 2 Preliminaries

Below we present the necessary preliminaries on algorithmicity and PC-groups.
2.1 Algorithmic aspects

Similar preserving properties concerning free and direct products were studied for the Membership Problem (MP) by K. A. Mikhailova. In [25] she proved that MP is preserved under free products; whereas in [24], she showed that $F_2 \times F_2$ has unsolvable membership problem, proving that MP (and thus SIP and CIP) do not pass to direct products.

Several obvious relations among the already introduced algorithmic problems are summarized in the diagram below:

![Diagram of dependencies between algorithmic problems]

where the starred implication is true with the extra assumption that the involved group is torsion-free, and has solvable word problem. We recall that WP and MP stand for the classical word problem, and the subgroup membership problem stated below.

(Subgroup) membership problem, $MP(G)$. Given a finite set of words $w, u_1, \ldots, u_n$ in the generators of $G$, decide whether $w$ represents an element in the subgroup generated by $u_1, \ldots, u_n$; and in the affirmative case compute an expression of $w$ as a word in the $u_i$’s.

Lemma 2.1. If a torsion-free group satisfies SIP and WP, then it also satisfies MP.

Proof. Let $G = \langle X \mid R \rangle$. Given words $u, v_1, \ldots, v_m$ in $X$, apply SIP to $H = \langle u \rangle$, and $K = \langle v_1, \ldots, v_m \rangle$: since $H \cap K$ is cyclic (and so, finitely generated), SIP will always answer yes, and return a finite set of words $w_1, \ldots, w_p$ in $X$ such that $H \cap K = \langle w_1, \ldots, w_p \rangle = \langle u^r \rangle$, for some unknown $r \in \mathbb{Z}$.

Now, since each $w_i$ must be a power of $u^r$ (say $w_i = u^{r_i}$), we can compute the exponents $r_1, \ldots, r_p \in \mathbb{Z}$ by brute force enumeration (even without using WP). Once we have obtained the integers $r_1, \ldots, r_p \in \mathbb{Z}$, we can effectively compute the greatest common divisor $r = \gcd(r_1, \ldots, r_p)$ and get $H \cap K = \langle w_1, \ldots, w_p \rangle = \langle u^r \rangle$.

Now, it is clear that $u \in K$ if and only if $u \in H \cap K = \langle u^r \rangle$; i.e., if and only if $u = u^{rs}$, for some $s \in \mathbb{Z}$. To decide whether such an $s$ exists, first apply WP to the input word $u$ in order to decide whether $u = 1$ or not. In the affirmative case the answer is obviously yes; otherwise, $u \neq 1$ and torsion-freeness of $G$ tells us that $u \in K$ (and the answer is yes) if and only if $r = \pm 1$.

Remark 2.2. Note that $CIP_{fg} \Rightarrow MP$ without any further condition, since $g \in \langle h_1, \ldots, h_k \rangle$ if and only if $g \cdot 1 \cap 1 \cdot \langle h_1, \ldots, h_k \rangle \neq \emptyset$.

Corollary 2.3. For PC-groups, both SIP and CIP imply MP. In particular, SIP and CIP are unsolvable for $F_2 \times F_2 = G_{C_4}$, and hence for any PC-group $G_\Gamma$ with $C_4 \leq \Gamma$. 

Remark 2.4. Note that the difference between properties TIP and ESIP is that the second one says nothing about \(wH \cap w'K\) in the case when \(H \cap K\) is not finitely generated, while TIP is required to answer about emptiness even in this case; this is a subtlety that will become important along the paper.

Finally, note that \((Hw)^{-1} = w^{-1}H\) and \(w_1Hw_2 = w_1w_2(w_2^{-1}Hw_2) = w_1w_2Hw_2\). Therefore, the variants of CIP for right, left, and two-sided cosets are equivalent problems.

2.2 PC-groups

Below, we recall some well-known results about PC-groups we will need throughout the paper; we refer the reader to [5, 12, 13, 21] for detailed surveys, and further reference.

Theorem 2.5. (i) Let \(\Gamma_1, \Gamma_2\) be simple graphs. Then, the groups \(G_{\Gamma_1}, G_{\Gamma_2}\) are isomorphic if and only if the graphs \(\Gamma_1, \Gamma_2\) are isomorphic (\([10]\)). In particular, the isomorphism problem is solvable within RAAGs.

(ii) The abelianization of the PC-group \(G_{\Gamma}\) is the free-abelian group of rank \(|V_{\Gamma}|\).

(iii) The word and conjugacy problems are solvable for RAAG's (\([14, 22, 28, 30, 31]\)).

(iv) PC-groups are torsion-free (\([2]\)).

(v) RAAGs are residually finite; in particular, they are Hopfian (\([21]\)).

(vi) The maximum rank of a free-abelian subgroup of a RAAG \(G_{\Gamma}\) is the size of a largest clique in \(\Gamma\) (\([21]\)).

(vii) Disjoint union and graph join correspond, respectively, to free product and direct product of groups: \(G_{X\sqcup Y} \simeq G_X \ast G_Y\) and \(G_{X\vee Y} \simeq G_X \times G_Y\).

(viii) A PC-group \(G_{\Gamma}\) splits as a nontrivial free product if and only if its defining graph \(\Gamma\) is disconnected.

(ix) A PC-group \(G_{\Gamma}\) splits as a nontrivial direct product if and only if its defining graph \(\Gamma\) is a join.

(x) The center of a PC-group \(G_{\Gamma}\) is the (free-abelian) subgroup generated by the set of central vertices in \(\Gamma\).

(xi) Let \(\Gamma\) be an arbitrary simple graph, and \(Y\) a subset of vertices of \(\Gamma\). Then, the subgroup of \(G_{\Gamma}\) generated by \(Y\) is again a PC-group, presented by the corresponding full subgraph, \(\langle Y \rangle \simeq G_{\Gamma[Y]}\). □

Besides Droms groups, other subfamilies of PC-groups naturally arise as directly related with the intersection problem. For example, in [7], Delgado characterized the PC-groups satisfying the Howson property precisely as those being fully residually free (or free products of free-abelian groups).

Another interesting subfamily of PC-groups is that of chordal groups, that is the PC-groups presented by a finite chordal graph (i.e., one with no induced cycles of length strictly greater than three).
Clearly, Droms graphs are chordal and not the other way around. From Theorem 1.2, it is clear that Droms groups are coherent (every finitely generated subgroup is finitely presented). However, this last class was proved to be bigger, corresponding precisely to chordal groups, which turn out to have some nice algorithmic properties as well.

**Theorem 2.6** (Droms, [9]). Let $\Gamma$ be a finite graph. Then, the RAAG $G_{\Gamma}$ is coherent if and only if $\Gamma$ is chordal.

**Theorem 2.7** (Kapovich–Weidmann–Myasnikov, [20]). Let $\Gamma$ be a finite chordal graph (i.e., $G_{\Gamma}$ is a coherent RAAG). Then, (i) $G_{\Gamma}$ has solvable membership problem; (ii) given a finite subset $S \subseteq G_{\Gamma}$, we can algorithmically find a presentation for the subgroup $\langle S \rangle \leq G_{\Gamma}$.

We remark the pertinacious absence of $G_{C_4} = F_2 \times F_2$ from any family of algorithmically well behaved groups. However, the exact boundary of the class of RAAGs having solvable MP is not known: chordal groups have it, and $G_{C_4}$ does not. Which of the groups $G_{C_n}$, for $n \geq 5$, have solvable MP? Is it possible to find a characterization of the RAAG’s with solvable MP?

Finally, we recall that for submonoids, the exact border for the corresponding membership problem is already known: In [23] is proved that the submonoid membership problem is solvable in a PC-group if and only if it is Droms. Note that this implies, in particular, that $G_{P_4}$ is a group with solvable subgroup membership problem (it is chordal), but unsolvable submonoid membership problem (it is not Droms).

### 2.3 Droms groups

Due to the recursive description in Corollary 1.5, any Droms graph $\Gamma$ decomposes as the join of its central part $Z(\Gamma) \cong K_m$, and the full subgraph $\Gamma_0 = \Gamma \setminus Z(\Gamma) \leq \Gamma$, that is $\Gamma = K_m \vee \Gamma_0$, where $m \geq 0$ and $\Gamma_0$ being either empty or a disconnected Droms graph; this is called the primary decomposition of $\Gamma$. In particular: (i) $G_{\Gamma}$ is free-abelian $\iff$ $\Gamma$ is complete $\iff Z(\Gamma) = \Gamma \iff \Gamma_0 = \emptyset$. (ii) $\Gamma$ is connected $\iff Z(\Gamma) \neq \emptyset \iff m \geq 1 \iff \Gamma$ is a cone. (iii) $G_{\Gamma}$ is centerless $\iff \Gamma$ is disconnected $\iff Z(\Gamma) = \emptyset \iff m = 0$.

**Remark 2.8.** However, all subgroups of Droms groups (including the non finitely generated ones) are again $\{P_4, C_4\}$-free PC-groups. In particular, every finitely generated subgroup of a Droms group is again a Droms group.

Finally, we need the following algorithmic result for later use.

**Proposition 2.9.** Let $G_{\Gamma} = \langle X \mid R \rangle$ be a Droms group. Then, there exists an algorithm which, given words $w_1(X), \ldots, w_p(X)$ in the generators of $G_{\Gamma}$, (i) computes a basis for the subgroup $H = \langle w_1, \ldots, w_p \rangle \leq G_{\Gamma}$; (ii) writes the basis elements in terms of the original generators, and vice versa.

**Proof.** Since Droms graphs are chordal, by Theorem 2.7, we can effectively compute a finite presentation for $H$, say $H = \langle Y \mid S \rangle$. Then, one can exhaustively explore the tree of all possible Tietze transformations applied to $\langle Y \mid S \rangle$ until getting one, say $\langle Z \mid Q \rangle$, in graphical
form (namely, with all relators being commutators of certain pairs of generators); this will be achieved in finite time because we know in advance that $H$ is indeed a RAAG.

At this point, we know that $H = \langle w_1, \ldots, w_p \rangle \cong \langle Y \mid S \rangle \cong \langle Z \mid Q \rangle$, and need to compute expressions for the basis elements $Z = \{z_1, \ldots, z_r\}$ in terms of the $w_i$’s and vice versa. We start a brute force search, using the following two parallel procedures:

1. enumerate all homomorphisms $\langle Z \mid Q \rangle \to H = \langle w_1, \ldots, w_p \rangle$; this can be done enumerating all possible $r$-tuples $(v_j)^r_{j=1}$ of words in $\{w_1, \ldots, w_p\}^\pm$, and checking whether they determine a well-defined homomorphism.

2. for each such homomorphism $z_j \mapsto v_j$, $j = 1, \ldots, r$, analyze whether it is onto $H$ by enumerating all words in $\{v_1, \ldots, v_r\}^\pm$, and checking whether each of $w_1, \ldots, w_p$ appear in the list.

Since we know that $\langle Z \mid Q \rangle \cong H$, there exists a surjective homomorphism from $\langle Z \mid Q \rangle$ onto $H$ and so, the above procedure will eventually find and output one of them, $z_j \mapsto v_j$, $j = 1, \ldots, r$. Finally, since RAAG’s are Hopfian (see Theorem 2.5(v)), such a surjective homomorphism is indeed an isomorphism. We then have the $z_i$’s written as words on the $w_i$’s, and the $w_i$’s as words on the $z_i$’s from the stopping criteria at step (2).

### 2.4 Proof of the main result

The main result in the present paper easily reduces to Theorems 1.7 and 1.8.

**Proof of Theorem 1.4.** Let $\Gamma$ be a Droms graph, and let $G_\Gamma$ be the corresponding Droms PC-group. We will prove Theorem 1.4 by induction on the number of vertices $|V\Gamma|$. If $|V\Gamma| = 0$, then $G_\Gamma = 1$, and obviously has solvable ESIP.

Now, consider a nonempty Droms graph $\Gamma$, and assume that every Droms PC-group with strictly less than $|V\Gamma|$ vertices has solvable ESIP. Consider the primary decomposition of $\Gamma$, say $\Gamma = K_m \vee \Gamma_0$. If $\Gamma_0$ is empty then $\Gamma$ is complete, $G_\Gamma \cong Z^m$ is free-abelian and so, it has solvable ESIP. Otherwise, $\Gamma_0$ is disconnected, say $\Gamma_0 = \Gamma_1 \vee \Gamma_2$ with $\Gamma_1$ and $\Gamma_2$ being Droms again. By induction, both $G_{\Gamma_1}$ and $G_{\Gamma_2}$ have solvable ESIP, by Theorem 1.8 $G_{\Gamma_0} = G_{\Gamma_1} * G_{\Gamma_2}$ has solvable ESIP, and by Theorem 1.7 $G_\Gamma = Z^m \times G_{\Gamma_0}$ also has solvable ESIP.

### 3 The direct product case

This section is devoted to proving Theorems 1.6 and 1.7. To this end, we analyze the Droms groups presented by connected graphs.
3.1 Preparation

For all this subsection, we fix an arbitrary connected non-complete Droms graph $\Gamma$ and its primary decomposition $\Gamma = \Gamma_m \sqcup \Gamma_0$, where $m \geq 1$ and $\Gamma_0$ is a disconnected Droms graph. Algebraically, $G_{\Gamma} = Z^m \times G_{\Gamma_0}$, $Z(G_{\Gamma}) = Z^m$, and $G_{\Gamma_0}$ is a nontrivial free product. Let $\forall \Gamma_0 = \{ x_1, \ldots, x_n \}$ and $\forall K_m = T = \{ t_1, \ldots, t_m \}$.

Every element in $G_{\Gamma}$ can be written as a word on $(t_1, \ldots, t_m, x_1, \ldots, x_n)$, where the $t_i$’s are free to move to any position. We will systematically write all these $t_i$’s ordered on the left, and we will abbreviate them as a vectorial power of a formal symbol ‘$t$’. This way, every element in $G_{\Gamma}$ can be written in the form $t^{a_1} \cdots t^{a_m} u(x_1, \ldots, x_n) = t^a u(x_1, \ldots, x_n)$, where $a = (a_1, \ldots, a_m) \in Z^m$, and $u = u(x_1, \ldots, x_n)$ is a word on the $x_i$’s. Clearly, the product of elements is then given by the rule $(t^{a} u) \cdot (t^{b} v) = t^{a+b} uv$.

Quotienting by the center of $G_{\Gamma}$ gives rise to the short exact sequence

$$1 \rightarrow Z^m \rightarrow G_{\Gamma} \xrightarrow{\pi_0} G_{\Gamma_0} \rightarrow 1,$$

where $\pi_0$ just erases the occurrences of letters in $T^\pm = \{ t_1, \ldots, t_m \}^\pm$.

**Definition 3.1.** For a given subgroup $H \leq G_{\Gamma}$, and an element $u \in G_{\Gamma_0}$, we define the (abelian) completion of $u$ in $H$ (the H-completion of $u$, for short) to be the set $\mathcal{C}_H(u) = \{ a \in Z^m \mid t^a u \in H \}$.

**Lemma 3.2.** The completion $\mathcal{C}_H(u)$ is either empty (when $u \notin H\pi_0$), or a coset of $Z^m \cap H$. More precisely, if $u_1, \ldots, u_n \in H\pi_0$, and $\omega(u_1, \ldots, u_n)$ is an arbitrary word on them, then $\mathcal{C}_H(\omega(u_1, \ldots, u_n)) = \sum_{i=1}^n \omega_i \mathcal{C}_H(u_i)$, where $\omega_i = |\omega_i|$ is the total exponent of the variable $u_i$ in $\omega$.

**Lemma 3.3.** Let $G_{\Gamma} = Z^m \times G_{\Gamma_0}$ be the primary decomposition of a connected Droms group. Then, any subgroup $H \leq G_{\Gamma}$ splits as $H = (Z^m \cap H) \times H\pi_0$, where $\pi_0: G_{\Gamma} \rightarrow G_{\Gamma_0}$ is the natural projection killing the center of $G_{\Gamma}$, and $\sigma: H\pi_0 \rightarrow H$ is a section of $\pi_0|_H$.

**Proof.** Let $X = \{ x_1, \ldots, x_n \}$ be the (finite) set of vertices of $\Gamma_0$ (i.e., let $G_{\Gamma_0} = \langle X \mid R \rangle$, where $R \subseteq [X, X]$); and let $Z^m = \langle t_1, \ldots, t_m \mid [t_i, t_j] \forall i, j \rangle$. Now, consider the restriction to $H \leq Z^m \times G_{\Gamma_0}$ of the natural short exact sequence (2):

$$1 \rightarrow Z^m \rightarrow Z(G_{\Gamma}) \rightarrow G_{\Gamma} \rightarrow G_{\Gamma_0} \rightarrow 1 \quad \text{(3)}$$

$$1 \rightarrow Z^m \cap H \rightarrow \forall \rightarrow H \rightarrow H\pi_0 \rightarrow 1, \quad \text{(4)}$$

Since $G_{\Gamma_0}$ is Droms, we know that $H\pi_0 \leq G_{\Gamma_0}$ is again a PC-group. Thus, there exists a (not necessarily finite) subset $Y = \{ y_1 \} \subseteq G_{\Gamma_0}$ such that $H\pi_0 \simeq \langle Y \mid S \rangle$, where $S$ is a certain collection of commutators of the $y_j$’s.

Now, observe that any map $\sigma: Y \rightarrow H$ sending each $y_j \in Y$ back to any of its $\pi_0$-preimages in $H$ will necessarily respect the relations in $S$: indeed, for each commutator $[y_i, y_j] \in S$,
we have \([y_1\sigma, y_1\sigma] = [t^a y_1, t^a y_1] = [y_1, y_1]\) (for certain abelian completions \(a_i, a_j \in \mathbb{Z}^m\)). Therefore, any such map \(\sigma\) defines a (injective) section of the restriction \(\tau|_H\). Thus, the short exact sequence (4) splits and, for any such section \(\sigma\), \(H\pi_0 \simeq H\pi_0\sigma \leq H\). Moreover, since the kernel of the subextension (4) lies in the center of \(G\), the conjugation action is trivial, and the claimed result follows.  

**Corollary 3.4.** In the above situation, the subgroup \(H \leq G\) is finitely generated if and only if \(H\pi_0 \leq G_0\) is finitely generated.  

**Remark 3.5.** For any \(H = \langle b_1, \ldots, b_s \rangle \leq G = \mathbb{Z}^m \times G_0\), where \(a_1 \neq 1\) for all \(i = 1, \ldots, s\), we have \(\langle t^b_1, \ldots, t^b_s \rangle \leq \mathbb{Z}^m \cap H = \mathbb{Z}(G)\cap H \leq \mathbb{Z}(H)\), but these two inclusions are, in general, not equalities: for the first one, a nontrivial product of the last \(s\) generators could, in principle, be equal to \(t^e\) for some element \(e \notin \langle b_1, \ldots, b_s\rangle\); and for the second we could have, for example, \(u_1 = \cdots = u_s \neq 1\) so that \(t^u u_1\) belongs to \(Z(H)\) but not to \(Z(G)\).

Let us consider now two finitely generated subgroups \(H_1, H_2 \leq G\) and analyze when the intersection \(H_1 \cap H_2\) is again finitely generated. We will see that the behaviour of the embedding \((H_1 \cap H_2)\pi_0 \leq (H_1)\pi_0 \cap (H_2)\pi_0\) is crucial to this end.

**Lemma 3.6.** Let \(H_1, H_2 \leq G\). Then,

(i) \((H_1 \cap H_2)\pi_0 \leq H_1\pi_0 \cap H_2\pi_0\), sometimes with strict inclusion;

(ii) \((H_1 \cap H_2)\pi_0 \leq H_1\pi_0 \cap H_2\pi_0\);

(iii) \([(H_1)\pi_0 \cap (H_2)\pi_0, (H_1)\pi_0 \cap (H_2)\pi_0] \leq (H_1 \cap H_2)\pi_0\).

**Proof.** (i). This is clear.

(ii). To see normality, consider \(u \in (H_1 \cap H_2)\pi_0\) and \(v \in (H_1)\pi_0 \cap (H_2)\pi_0\); then, there exist elements \(t^a u \in H_1 \cap H_2\), and \(t^b v \in H_1\), for \(i = 1, 2\). Now observe that \(t^a (v^{-1} uv) = v^{-1} (t^a u) v = (t^b v)^{-1} (t^a u) (t^b v) \in H_1\), for \(i = 1, 2\). Thus, \(t^a v^{-1} uv \in H_1 \cap H_2\) and so, \(v^{-1} uv \in (H_1 \cap H_2)\pi_0\).

(iii). Take \(u, v \in (H_1)\pi_0 \cap (H_2)\pi_0\); then, there exist elements \(t^a u \in H_1\), \(t^b v \in H_1\), for \(i = 1, 2\). Now, observe that \([u, v] = u^{-1} v^{-1} uv = (t^a u)^{-1} (t^b v)^{-1} (t^a u) (t^b v) \in H_1\), for \(i = 1, 2\). Thus, \([u, v]\) belongs to \(H_1 \cap H_2\), and to \((H_1 \cap H_2)\pi_0\), as claimed.  

**Lemma 3.7.** If \((H_1 \cap H_2)\pi_0\) (and so, \(H_1 \cap H_2\)) is finitely generated, then \((H_1)\pi_0 \cap (H_2)\pi_0\) is also finitely generated.

**Proof.** Let us assume that \((H_1)\pi_0 \cap (H_2)\pi_0\) is not finitely generated and find a contradiction.

By Remark 2.8, \((H_1)\pi_0 \cap (H_2)\pi_0\) is again a PC-group with infinite \(\{P_4, C_4\}\)-free commutation graph, say \(\Delta\), and \((H_1 \cap H_2)\pi_0 \leq G_{\Delta'} \leq G_\Delta = (H_1)\pi_0 \cap (H_2)\pi_0\), where \(\Delta'\) is the full subgraph of \(\Delta\) determined by the vertices appearing in the reduced expressions of elements in \((H_1 \cap H_2)\pi_0\). Note that the assumption of finite generability for \((H_1 \cap H_2)\pi_0\) implies that \(\Delta'\) is finite. Note also that, by construction, \(\Delta'\) is minimal, i.e., for any \(x \in \mathbb{V}\Delta'\), there exists and element \(g \in (H_1 \cap H_2)\pi_0\) such that \(g \notin G_{\Delta' \setminus \{x\}}\).
In this situation, \((H_1)\pi_0 \cap (H_2)\pi_0\) cannot be abelian since, if so, we would have a non-finitely generated free-abelian group embedded in the finitely generated PC-group \(G_{\Gamma}\), which is not possible (see Theorem 2.5(vi)). So, \(\Delta\) is not complete. Take two non-adjacent vertices, say \(u, v\) from \(\Delta\); since \(F_2 \simeq \langle u, v \rangle \leq G_{\Delta}\), Lemma 3.6(iii) tells us that \(F_{\infty} \simeq \langle (u, v), (u, v) \rangle \leq [G_{\Delta}, G_{\Delta}] \leq (H_1 \cap H_2)\pi_0\) and thus, \((H_1 \cap H_2)\pi_0\) is not abelian either. Accordingly, neither the infinite graph \(\Delta\), nor the finite graph \(\Delta'\) are complete.

Suppose now there is a missing edge between some vertex \(x \in V\Delta'\) and some vertex \(y \in V\Delta \setminus V\Delta'\). Take an element \(g \in (H_1 \cap H_2)\pi_0\) with \(g \notin G(\Delta' \setminus \{x\})\) and Lemma 3.6(ii) would tell us that \(y^{-1}gy \in G_{\Delta'}\), which is a contradiction.

Hence, in \(\Delta\), every vertex from \(\Delta'\) is connected to every vertex outside \(\Delta'\). But now, take two non-adjacent vertices \(x_1, x_2\) from \(\Delta'\) and two non-adjacent vertices \(y_1, y_2\) from \(\Delta \setminus \Delta'\) (there must also be some since \(\Delta \setminus \Delta'\) is infinite and \(Z^\infty\) does not embed into \(G_{\Gamma}\)). Then, the full subgraph of \(\Delta\) with vertex set \(\{x_1, x_2, y_1, y_2\}\) form a copy of \(C_4\), a contradiction with \(\Delta\) being Droms.

\[\square\]

### 3.2 Proofs of Theorems 1.6 and 1.7

**Proof of Theorem 1.6.** First of all, observe that we can restrict ourselves to the case where \(G\) is a disconnected Droms group.

Therefore, we consider \(\Gamma = K_m \vee \Gamma_0\) (where \(m \geq 1\) and \(\Gamma_0\) is a disconnected Droms graph), we will assume SIP is solvable for \(G_{\Gamma'}\), and we will prove it solvable for \(G_{\Gamma} = Z^m \times G_{\Gamma_0}\).

Let \(X = V\Gamma_0 = \{x_1, \ldots, x_n\}\) and \(T = VK_m = \{t_1, \ldots, t_m\}\).

Given a finite set of generators for a subgroup, say \(H_1 \leq G_{\Gamma}\), the first step is to improve them: project them to \(G_{\Gamma_0}\), and then apply Proposition 2.9 to compute a basis for \(H_1\pi_0\), say \(\{u_1, \ldots, u_n\}\) with commutation graph \(G_{\Delta}\). The respective completions, say \(t_1 u_1, \ldots, t_n u_n \in H_1\) can be computed from the words expressing the \(u_i\)'s in terms of the projected generators, and recomputing them on the original generators for \(H_1\).

Now, for each of the original generators of \(H_1\), say \(t^c v\), we can write \(v \in H_1\pi_0\) in terms of the basis \(u_1, \ldots, u_n\), say \(v = v(u_1, \ldots, u_n)\) and compute \(v(t_1^c u_1, \ldots, t_n^c u_n) = t^d v(u_1, \ldots, u_n) = t^d v\). Since \(t^c v, t^d v \in H_1\), we get \(t^c d^{-1} \in H_1 \cap Z^m\). Repeating this operation for each generator of \(H_1\), we get a generating set for \(H_1 \cap Z^m\) which is easily reducible to a free-abelian basis, say \(\{t_1, \ldots, t_{m_1}\}\). In this way, we can compute bases for \(H_1\) and \(H_2\):

\[
\{t_1, \ldots, t_{m_1}, t^{a_1} u_1, \ldots, t^{a_n} u_n\} \quad \text{and} \quad \{t_1', \ldots, t_2', t^{a_1'} u_1', \ldots, t^{a_n'} u_n'\},
\]

where \(\{t^{b_1}, \ldots, t^{b_{m_1}}\}\) and \(\{t^{b_1'}, \ldots, t^{b_{m_2}}\}\) are free-abelian bases of \(L_1 = H_1 \cap Z^m\) and \(L_2 = H_2 \cap Z^m\), respectively; and \(\{u_1, \ldots, u_n\}\) and \(\{u_1', \ldots, u_n'\}\) are basis of \(H_1\pi_0\) and \(H_2\pi_0\), with commutation graphs \(\Delta_1, \Delta_2\). That is, \(H_1\pi_0 \simeq G_{\Delta_1}\) and \(H_2\pi_0 \simeq G_{\Delta_2}\).

Now, the solvability of SIP in \(G_{\Gamma_0}\) (assumed by hypothesis) allows us to decide whether \(H_1\pi_0 \cap H_2\pi_0\) is finitely generated or not. If not, then (by Lemma 3.7) neither is \(\langle H_1 \cap H_2 \rangle\pi_0\),
and we are done. Thus, we can assume that $H_1 \pi_0 \cap H_2 \pi_0$ is finitely generated. Then, the hypothesis provides a finite set of generators and hence a basis — say $W = \{w_1, \ldots, w_{n_3}\}$ with commutation graph $\Delta_3$ — for $H_1 \pi_0 \cap H_2 \pi_0$. That is, $H_1 \pi_0 \cap H_2 \pi_0 = G_{\Delta_3} = \langle w_1, \ldots, w_{n_3}\rangle \leq G$, where the $w_i$’s are words on $X$. After writing each $w_i \in W$ as a word on $U$ and $U'$ respectively — say $w_i = \omega_i(u_1, \ldots, u_{n_1})$ and $w_i = \omega'_i(u'_1, \ldots, u'_{n_2})$ — we obtain a description of the inclusions $\iota_1 : H_1 \pi_0 \cap H_2 \pi_0 \rightarrow H_1 \pi_0$, and $\iota_2 : H_1 \pi_0 \cap H_2 \pi_0 \rightarrow H_2 \pi_0$ in terms of the corresponding bases.

Abelianizing $\iota_1$ and $\iota_2$, we get the integral matrices $P_1$ (of size $n_3 \times n_1$), and $P_2$ (of size $n_3 \times n_2$) and complete the upper half of Figure 2, where the $\rho_i$’s are the corresponding abelianization maps. Note that, even though $\iota_1$ and $\iota_2$ are injective, their abelianizations $P_1$ and $P_2$ need not be $(n_3$ could very well be bigger than $n_1$ or $n_2)$. 

$$
\begin{array}{c}
\{H_1 \cap H_2\} \pi_0 \\
\cap
\\
\uparrow \iota_1 \\
\downarrow \rho_1 \\
\rho_3 \\
\rho_2 \\
Z^{n_1} \\
\cap
\\
P_1 \\
\cap
\\
Z^{n_3} \\
\cap
\\
P_2 \\
\cap
\\
Z^{n_2} \\
\cap
\\
Z^m \\
L_1 \\
L_2 \\
\end{array}
$$

Figure 2: Intersection diagram for subgroups of Droms groups

Now, we can recompute the words $\omega_i$ (resp., $\omega'_i$) as words on the $(t^n u_i)$’s (resp., on the $(t^n u'_i)$’s) to get particular preimages of the $w_i$’s in $H_1$ (resp., $H_2$). Namely,

$$
\begin{align*}
\omega_i(t^{n_1} u_1, \ldots, t^{n_1} u_{n_1}) &= \iota^{\omega_i A_1} \omega_i(u_1, \ldots, u_{n_1}) = \iota^{\omega_i A_1} w_i \in H_1, \\
\omega_i(t^{n_2} u'_1, \ldots, t^{n_2} u'_{n_2}) &= \iota^{\omega_i A_2} \omega_i(u'_1, \ldots, u'_{n_2}) = \iota^{\omega_i A_2} w_i \in H_2,
\end{align*}
$$

where $\omega_i = (\omega_i)^{ab}$, $\omega'_i = (\omega'_i)^{ab}$; and $A_1, A_2$ are the integral matrices (of sizes $n_1 \times m$ and $n_2 \times m$) having as rows $\{a_1, \ldots, a_{n_1}\}$ and $\{a'_1, \ldots, a'_{n_2}\}$ respectively. Hence, the abelian completions of $w_i \in H_1 \pi_0 \cap H_2 \pi_0$ in $H_1$ and $H_2$ are the linear varieties:

$$
\begin{align*}
\tilde{c}_{H_1}(w_i) &= \omega_i A_1 + L_1 = w_i \iota_1 \rho_1 A_1 + L_1 = w_i \rho_3 R_1 + L_1, \\
\tilde{c}_{H_2}(w_i) &= \omega'_i A_2 + L_2 = w_i \iota_2 \rho_2 A_2 + L_2 = w_i \rho_3 R_2 + L_2,
\end{align*}
$$

where $L_j = Z^m \cap H_j$, and we have used the commutation $\iota_1 \rho_1 = \rho_3 P_1$ together with the definition $R_j := P_1 A_j$, for $j = 1, 2$; see Figure 2. Note that all maps and matrices involved in Figure 2 are explicitly computable from the data.
To finish our argument, it suffices to understand which elements of \( H_1 \pi_0 \cap H_2 \pi_0 \) belong to \((H_1 \cap H_2) \pi_0 \). They are, precisely, those whose \( H_1 \)-completion and \( H_2 \)-completion intersect:

\[
(H_1 \cap H_2) \pi_0 = \{ w \in H_1 \pi_0 \cap H_2 \pi_0 \mid (wp_3 P_1 A_1 + L_1) \cap (wp_3 P_2 A_2 + L_2) \neq \emptyset \} \\
= \{ (d \in \mathbb{Z}^{n_3} \mid (d R_1 + L_1) \cap (d R_2 + L_2) \neq \emptyset) \} \rho_3^{-1} \\
= \{ (d \in \mathbb{Z}^{n_3} \mid d (R_1 - R_2) \in L_1 + L_2) \} \rho_3^{-1} \\
= (L_1 + L_2) (R_1 - R_2)^{-1} \rho_3^{-1} = M \rho_3^{-1},
\]

where \( M = (L_1 + L_2) (R_1 - R_2)^{-1} \) denotes the full preimage of \( L_1 + L_2 \) by the matrix \( R_1 - R_2 \) for which a basis is clearly computable using linear algebra. At this point, we can decide whether \((H_1 \cap H_2) \pi_0 \) is finitely computable or not by distinguishing two cases.

If \( \Delta_3 \) is complete (this includes the case where \( \Delta_3 \) is empty and \( n_3 = 0 \)), then \( H_1 \pi_0 \cap H_2 \pi_0 \simeq \mathbb{Z}^{n_3} \) is abelian, \( \rho_3 \) is the identity, and \((H_1 \cap H_2) \pi_0 = (L_1 + L_2) (R_1 - R_2)^{-1} = M \) is always finitely computable and computable.

So, assume \( \Delta_3 \) is not complete. Since it is a Droms graph, it will have a primary decomposition, say \( \Delta_3 = K_{n_4} \vee \Delta_5 \), where \( n_4 \geq 0 \), and \( \Delta_5 \) is Droms again, disconnected, and with \( |\Delta_5| = n_5 = n_3 - n_4 \geq 2 \). Let us rename the vertices \( \{w_1, \ldots, w_{n_3}\} \) of \( \Delta_3 \) as \( VK_{n_4} = \{z_1, \ldots, z_{n_4}\} \), and \( V\Delta_5 = \{y_1, \ldots, y_{n_5}\} \), depending on whether they belong to \( K_{n_4} \) or \( \Delta_5 \). This means that \( H_1 \pi_0 \cap H_2 \pi_0 \simeq G_{\Delta_3} = \mathbb{Z}^{n_4} \times G_{\Delta_5} \), where \( n_4 \geq 0 \), and \( G_{\Delta_5} \neq 1 \) decomposes as a nontrivial free product. Furthermore, the normal subgroup \((H_1 \cap H_2) \pi_0 \leq G_{\Delta_3} \) is not contained in \( \mathbb{Z}^{n_4} \) (taking two vertices, say \( y_{i_1}, y_{j_1} \), in different components of \( \Delta_5 \), Lemma 3.6(iii) tells us that \( 1 \neq [y_{i_1}, y_{j_1}] \in (H_1 \cap H_2) \pi_0 \)). In this situation, the abelianization map \( \rho_3 : G_{\Delta_3} \to \mathbb{Z}^{n_3} \) is the identity on the center \( \mathbb{Z}^{n_4} \) of \( G_{\Delta_3} \) and so, can be decomposed in the form \( \rho_3 = id \times \rho_5 : G_{\Delta_3} = \mathbb{Z}^{n_4} \times G_{\Delta_5} \to \mathbb{Z}^{n_4} \times \mathbb{Z}^{n_5} = \mathbb{Z}^{n_3} \), \( (c, v) \to (c, v) \). where \( v \) denotes the abelianization \( v = v^{ab} \in \mathbb{Z}^{n_5} \). Of course, if \( n_4 = 0 \) then \( \rho_5 = \rho_3 \).

Now consider the image of \((H_1 \cap H_2) \pi_0 \) under the projection \( \pi_i : \mathbb{Z}^{n_4} \times G_{\Delta_5} \to G_{\Delta_i} \), which is nontrivial since \((H_1 \cap H_2) \pi_0 \not\simeq \mathbb{Z}^{n_4} \). We have \( 1 \neq (H_1 \cap H_2) \pi_0 \pi_i \leq G_{\Delta_5} \), a nontrivial normal subgroup in a group which decomposes as a nontrivial free product. Therefore,

\[
H_1 \cap H_2 \text{ is f.g. } \iff (H_1 \cap H_2) \pi_0 \pi_i \text{ is f.g. } \iff (H_1 \cap H_2) \pi_0 \pi_i \leq fi G_{\Delta_5} \\
\iff M \rho_3^{-1} \pi_i \leq fi G_{\Delta_5} \iff M \pi_i^{ab} \rho_3^{-1} \leq fi G_{\Delta_5} \\
\iff M \pi_i^{ab} \leq fi \mathbb{Z}^{n_5} \iff \text{rk} (M \pi_i^{ab}) = n_5.
\]

The first of these equivalences is a (double) application of Corollary 3.4. The second one is an application of the following theorem in [3, Section 6] by B. Baumslag: Let \( G \) be the free product of two nontrivial groups. Let \( H \) be a finitely generated subgroup containing a nontrivial normal subgroup of \( G \). Then \( H \) is of finite index in \( G \). The fourth equivalence is correct since \( \pi_0 \rho_5 = \rho_3 \pi_i^{ab} \) and all of them are surjective maps. Finally, the fifth equivalence is correct because following backwards the epimorphism \( \rho_5 \), a subgroup \( M \pi_i^{ab} \leq \mathbb{Z}^{n_5} \) is of finite index if and only if its full preimage \( M \pi_i^{ab} \rho_3^{-1} \) is of finite index in \( G_{\Delta_5} \), in which case the two indices do coincide, namely, \( \mathbb{Z}^{n_5} : M \pi_i^{ab} = [G_{\Delta_5} : M \pi_i^{ab} \rho_3^{-1}] \); see Figure 3.
\[(H_1 \cap H_2)\pi_6 \simeq M \rho_3^{-1} \leq G_{\Delta_3} \xrightarrow{\pi_1} G_{\Delta_3} \geq M \pi_{1b} \rho_5^{-1}
\]
\[
(L_1 + L_2)(R_1 - R_2)^{-1} = M \leq Z^{n_3} \xrightarrow{\pi_{1b}} Z^{n_5} \geq M \pi_{1b}
\]

Figure 3: The map \(\pi_1\) and its abelianization

Since the map \(\pi_{1b}\) is computable, the last condition in (7) can be effectively checked. Hence, we can algorithmically decide whether \(H_1 \cap H_2\) is finitely generated or not (ultimately, in terms of some integral matrix having the correct rank). This solves the decision part of SIP.

It only remains to compute a finite set of generators for \(H_1 \cap H_2\) assuming it is finitely generated, i.e., assuming the equivalent conditions in (7) are satisfied.

We first use linear algebra to compute a finite family \(C\) of coset representatives of \(Z^{n_5}\) modulo \(M \pi_{1b}\). Then, choose arbitrary \(\rho_5\)-preimages in \(G_{\Delta_3}\), say \(\{v_1, \ldots, v_r\}\), where \(r = |G_{\Delta_3} : M \pi_{1b} \rho_5^{-1}| = |Z^{n_5} : M \pi_{1b}|\) (we can take, for example, \(y_1^{a_1} \cdots y_{n_5}^{a_{n_5}} \in G_{\Delta_3}\) for each vector \(a = (a_1, \ldots, a_{n_5}) \in Z^{n_5}\)). Now, construct the Schreier graph of the subgroup \((H_1 \cap H_2)\pi_6\pi_1 = M \rho_3^{-1} \pi_1 = M \pi_{1b} \rho_5^{-1} \leq_G G_{\Delta_3}\) with respect to \(V \Delta_5 = \{y_1, \ldots, y_{n_5}\}\), in the following way: draw as vertices the cosets \([v_1], \ldots, [v_r]\); then, for every \([v_i]\) (\(i = 1, \ldots, r\)), and every \(y_j\) (\(j = 1, \ldots, n_5\)), draw an edge labelled \(y_j\) from \([v_i]\) to \([v_i y_j]\). Here, we need to algorithmically recognize which is the coset \([v_1 y_j]\) from our list of vertices, but this is easy since: \([v_1 y_j] = [v_k] \iff v_1 y_j v_k^{-1} \in M \pi_{1b} \rho_5^{-1} \iff (v_1 y_j v_k^{-1}) \rho_5 \in M \pi_{1b}\).

From the Schreier graph of \((H_1 \cap H_2)\pi_6\pi_1 \leq_G G_{\Delta_3}\), we can obtain a finite set of generators for \((H_1 \cap H_2)\pi_6\pi_1\) just reading the labels of the closed paths \(T[e]\) corresponding to the arcs, e, outside a chosen maximal tree \(T\). These will be words on \(V \Delta_5 = \{y_1, \ldots, y_{n_5}\}\), i.e., elements of \(G_{\Delta_3}\) not using the central vertices \((z_1, \ldots, z_{n_3})\).

The next step is to lift the obtained generators to generators of \((H_1 \cap H_2)\pi_6\), pulling them back through \(\pi_1\). For each one of them, say \(g(y_1, \ldots, y_{n_3})\), we look for its preimages in \((H_1 \cap H_2)\pi_6\); they all are of the form \(z_1^{\lambda_1} \cdots z_{n_3}^{\lambda_{n_3}} g(y_1, \ldots, y_{n_3})\), where the unknowns \(\lambda_1, \ldots, \lambda_{n_3} \in Z\) can be found by solving the system of linear equations coming from the fact \(z_1^{\lambda_1} \cdots z_{n_3}^{\lambda_{n_3}} g(y_1, \ldots, y_{n_3}) \in M \rho_3^{-1}\). That is, \((\lambda_1, \ldots, \lambda_{n_3}, |g|, 0, \ldots, 0)(R_1 - R_2) \in L_1 + L_2\).

For each such \(g(y_1, \ldots, y_{n_3})\), we compute a particular preimage of the previous form and put them all, together with a free-abelian basis for

\[
\ker \pi_1 \cap (H_1 \cap H_2)\pi_6 = \left\{ z_1^{\lambda_1} \cdots z_{n_3}^{\lambda_{n_3}} : (\lambda_1, \ldots, \lambda_{n_3}, 0, \ldots, 0)(R_1 - R_2) \in L_1 + L_2 \right\},
\]

to constitute a set of generators for \((H_1 \cap H_2)\pi_6\).

Finally, we have to lift these generators for \((H_1 \cap H_2)\pi_6\) to a set of generators for \(H_1 \cap H_2\): for each such generator, say \(h_j\), write it as a word \(h_j = \omega_j(u_1, \ldots, u_{n_1})\) and as a word \(h_j = \omega_j'(u_1', \ldots, u_{n_2}')\) in the original bases \(U\) for \(H_1\pi_6\) and \(U'\) for \(H_2\pi_6\), respectively. Now,
reevaluate each $\omega_j$ and $\omega'_j$ in the corresponding basis elements from (5) for $H_1$ and $H_2$ respectively, to obtain vectors $c_j, c'_j \in \mathbb{Z}^m$ such that:

$$\omega_j(t^{a_1}u_1, \ldots, t^{a_n}u_n) = t^v \omega_j(u_1, \ldots, u_n) = t^v h_j \in H_1,$$

$$\omega'_j(t^{a_1}u'_1, \ldots, t^{a_n}u'_n) = t^v \omega'_j(u'_1, \ldots, u'_n) = t^v h_j \in H_2.$$

Finally, for each $j$, compute a vector $d_j \in (c_j + L_1) \cap (c'_j + L_2)$ (note that these intersections of linear varieties must be nonempty because $h_j \in (H_1 \cap H_2)\pi_0$, and consider the element $t^d h_j \in H_1 \cap H_2$. All these elements $t^d h_j$, together with a free-abelian basis for $H_1 \cap H_2 \cong \mathbb{Z}^m = (H_1 \cap H_2) \pi_0 \cap (H_1 \cap H_2) \pi_0 = L_1 \cap L_2$, constitute the desired set of generators for $H_1 \cap H_2$, and the proof is completed.

Below, we extend the previous arguments to prove Theorem 1.7.

**Proof of Theorem 1.7.** By exactly the same argument as before, we can reduce to a non-complete connected Droms graph $\Gamma$ with primary decomposition $\Gamma = K_m \vee \Gamma_0$ (where $m \geq 1$ and $\Gamma_0$ is a disconnected Droms graph), we assume ESIP to be solvable for $G_{\Gamma_0}$, and we have to solve it for $G_{\Gamma} = \mathbb{Z}^m \times G_{\Gamma_0}$.

We are given finite sets of generators for two subgroups $H_1, H_2 \subseteq G_{\Gamma}$, and two extra elements $t^a u, t^a' u' \in G_{\Gamma}$. Since the solvability of ESIP implies that of SIP, we can apply Theorem 1.6 to effectively decide whether $H_1 \cap H_2$ is finitely generated or not, and in the affirmative case compute a basis for $H_1 \cap H_2$. We assume all the notation developed along the proof of Theorem 1.6.

Now, if $H_1 \cap H_2$ is not finitely generated there is nothing else to do; otherwise, we can compute a basis, say $\{v_1, \ldots, v_p\}$, for $H_1 \cap H_2$ and we have to decide whether the coset intersection $(t^a u)H_1 \cap (t^a' u')H_2$ is empty or not. Note that $(t^a u)H_1 \cap (t^a' u')H_2 = \emptyset$ if and only if $(t^a u)H_1 \cap (t^a' u')H_2)\pi_0 = \emptyset$, and that $((t^a u)H_1 \cap (t^a' u')H_2)\pi_0 \subseteq ((t^a u)H_1)\pi_0 \cap ((t^a' u')H_2)\pi_0 = u(H_1\pi_0) \cap u'(H_2\pi_0)$. Then, since $H_1 \cap H_2$ is finitely generated, we know from Lemma 3.7 that $H_1\pi_0 \cap H_2\pi_0$ is finitely generated as well. Hence, an application of the ESIP solvability hypothesis for $G_{\Gamma_0}$ tells us whether the coset intersection $u(H_1\pi_0) \cap u'(H_2\pi_0)$ is empty or not. If it is empty, then $((t^a u)H_1 \cap (t^a' u')H_2)\pi_0 = \emptyset$, and we are done.

Otherwise, $u(H_1\pi_0) \cap u'(H_2\pi_0) \neq \emptyset$, and the hypothesis gives us an element $v_0 \in u(H_1\pi_0) \cap u'(H_2\pi_0)$ as a word on $Vf_0 = \{x_1, \ldots, x_n\}$; further, $u(H_1\pi_0) \cap u'(H_2\pi_0) = v_0(H_1\pi_0 \cap H_2\pi_0)$.

Observe that $((t^a u)H_1 \cap (t^a' u')H_2)\pi_0$ consists precisely of those elements $v_0 w$, with $w \in H_1\pi_0 \cap H_2\pi_0$, for which there exists a vector $c \in \mathbb{Z}^m$ such that $t^c v_0 w \in (t^a u)H_1 \cap (t^a' u')H_2$; that is, such that $t^{-a} u^{-1} v_0 w \in H_1$, and $t^{-a'}(u')^{-1} v_0 w \in H_2$. That is, $c - a \in \bar{c}_{H_1}(u^{-1} v_0 w)$, and $c - a' \in \bar{c}_{H_2}((u')^{-1} v_0 w)$. Hence, $((t^a u)H_1 \cap (t^a' u')H_2)\pi_0 = \emptyset$ if and only if, for all $w \in H_1\pi_0 \cap H_2\pi_0$, $(a + c_{H_1}(u^{-1} v_0 w)) \cap (a' + c_{H_2}((u')^{-1} v_0 w)) = \emptyset$. 

15
Fix an arbitrary word \( w = \omega(w_1, \ldots, w_{n_3}) \in H_1 \pi_0 \cap H_2 \pi_0 \) (for its abelianization, write \( |w|_1 = \lambda_i \), for \( i = 1, \ldots, n_3 \)). Choose vectors \( c \in \mathbb{C} H_1 (u^{-1} v_0), c' \in \mathbb{C} H_2 ((u')^{-1} v_0), \) and \( d_i \in \mathbb{C} H_1 (w_l), d'_i \in \mathbb{C} H_2 (w_l), \) for \( i = 1, \ldots, n_3 \). By Lemma 3.2:

\[
(a + \mathbb{C} H_1 (u^{-1} v_0)) \cap (a' + \mathbb{C} H_2 ((u')^{-1} v_0)) = \\
= (a + c + \mathbb{C} H_1 (w)) \cap (a' + c' + \mathbb{C} H_2 (w)) \\
= (a + c + \sum_{i=1}^{n_3} \lambda_i \mathbb{C} H_1 (w_l)) \cap (a' + c' + \sum_{i=1}^{n_3} \lambda_i \mathbb{C} H_2 (w_l)) \\
= (a + c + \sum_{i=1}^{n_3} \lambda_i d_i + L_1) \cap (a' + c' + \sum_{i=1}^{n_3} \lambda_i d'_i + L_2).
\]

Hence, the coset intersection \( ([t^u] H_1 (t^u') H_2) \pi_0 \) is empty if and only if for every integer \( \lambda_1, \ldots, \lambda_{n_3} \in \mathbb{Z}, (a - a' + c - c') + \sum_{i=1}^{n_3} \lambda_i (d_i - d'_i) \notin L_1 + L_2; \) or equivalently, if and only if

\[
((a - a' + c - c') + \langle d_1 - d'_1, \ldots, d_{n_3} - d'_{n_3} \rangle) \cap (L_1 + L_2) = \emptyset.
\]

This can be effectively decided using linear algebra. And furthermore, in case it is not empty, one can compute an explicit element from \( ([t^u] H_1 (t^u') H_2), \) just following the computations back. This completes the proof. \( \square \)

4 The free product case

In this section, we shall consider the free product case and prove Theorems 1.8 and 1.9. We follow the graph-theoretical approach developed by S. Ivanov in [16].

The classical theory of Stallings foldings (see [27]) provides a bijection between the subgroups of a given free group and certain kind of labelled directed graphs (the so-called Stallings automata) which when restricted to finitely generated subgroups (corresponding to finite automata) is fully constructive. This geometric approach allowed to solve many algorithmic problems about free groups in a very nice and intuitive way.

S. Ivanov (in [16]) generalized this machinery to free products. Among other applications, this allowed him to give a modern proof of the Kurosh Subgroup Theorem, and of B. Baumslag theorem stating that free products of Howson groups are again Howson (see [16–18]). However, he did not consider algorithmic issues in his approach.

More recently, Kapovich–Weidmann–Myasnikov extended further these folding techniques (see [20]) to fundamental groups of graphs of groups. Their method produces the automaton corresponding to a given subgroup, under some conditions on the edge groups (which automatically hold in the case of free products), and leads to the solution of the membership problem in some cases. However, they do not analyze subgroup intersections.
In order to treat algorithmically intersections of subgroups of free groups, we use Ivanov’s approach with the necessary technical adaptations to make it fully algorithmic. In the sake of clarity, we offer here a self-contained exposition.

The idea is to use generalized folding techniques to algorithmically represent any finitely generated subgroup \( H \leq G_1 \ast G_2 \) by a finite graph of certain kind, called a reduced wedge automaton, denoted by \( \Gamma_H \). We note that from such an object \( \Gamma_H \) one can already deduce, algorithmically, a Kurosh decomposition for the subgroup \( H \).

Note also that following Ivanov’s argument it is possible to define \( \Gamma_H \) for arbitrary subgroups \( H \leq G_1 \ast G_2 \). On the other hand, Ivanov gives in [16] a generalization of the classical “pullback” technique for free groups: given two subgroups \( H_1, H_2 \leq G_1 \ast G_2 \), and having at hand corresponding reduced wedge automata \( \Gamma_{H_1} \) and \( \Gamma_{H_2} \), he describes a reduced wedge automaton \( \Gamma_{H_1} \wedge \Gamma_{H_2} \) for \( H_1 \cap H_2 \), in terms of \( \Gamma_{H_1} \) and \( \Gamma_{H_2} \). Note a substantial difference with the free situation: \( \Gamma_{H_1} \wedge \Gamma_{H_2} \) may very well be an infinite object even with \( \Gamma_{H_1} \) and \( \Gamma_{H_2} \) being finite (corresponding to the possible non-Howson situation; i.e., \( H_1 \cap H_2 \) may very well be non finitely generated, even with \( H_1 \) and \( H_2 \) being finitely generated).

The main argument in the present section is the following: given finite generating sets for \( H_1 \) and \( H_2 \), we are able to construct \( \Gamma_{H_1} \) and \( \Gamma_{H_2} \), and then start constructing \( \Gamma_{H_1} \wedge \Gamma_{H_2} \) (even with the possibility of this being infinite). The crucial point is that, in finite time while the construction is running, we are able to either detect that \( H_1 \cap H_2 \) is not finitely generated, or to complete the construction of \( \Gamma_{H_1} \wedge \Gamma_{H_2} \); in the first case we have algorithmically deduced that the intersection \( H_1 \cap H_2 \) is not finitely generated, and in the second case we have effectively constructed \( \Gamma_{H_1} \wedge \Gamma_{H_2} \) from which we shall be able to extract a finite set of generators for \( H_1 \cap H_2 \). In order to check whether \( H_1 \cap H_2 \) is not finitely generated the hypothesis of SIP or ESIP in the factor groups \( G_1 \) and \( G_2 \) will be crucial.

4.1 Wedge automata

We assume the reader familiar with standard Stallings automata (representing subgroups of free groups, say \( \mathbb{F}_2 = \langle a \rangle \ast \langle b \rangle \), as involutive \( \{a, b\}\)-automata recognizing exactly the elements in the corresponding subgroup).

To cover the more general situation of \( G_1 \ast G_2 \), we need to encode more information into the arcs. A classical \( a \)-labelled arc would correspond to what we call here a \( G_1 \)-wedge: an arc subdivided in two halves, admitting a (possibly trivial) label from \( G_1 \) on each side, and also admitting a (possibly trivial) subgroup \( A \leq G_1 \) as a label of the middle (special) vertex between the two halves. Doing the same with the \( b \)-arcs (and subgroups of \( G_2 \)) we get an automaton with two types of vertices, primary (the original ones), and secondary (the new ones). See Figure 4.
In these new automata, walks are going to spell subsets of $G_1 * G_2$ (instead of words in $(a,b)^\pm$), by picking all the elements from the label of a secondary vertex when traversing it. Allowing, in addition, vertices to have any degree, we get the new notion of wedge automaton.

**Definition 4.1.** Let $G_1, G_2$ be two arbitrary groups. A (wedge) $(G_1, G_2)$-automaton is a septuple $\Gamma = (V\Gamma, E\Gamma, \imath, \kappa, \ell, \ell^{-1}, \ast)$, where:

(i) $\Gamma = (V\Gamma, E\Gamma, \imath, \kappa, \ell, \ell^{-1})$ is an involutive digraph (called *underlying digraph* of $\Gamma$) with three disjoint types of vertices, $V\Gamma = V_0\Gamma \sqcup V_1\Gamma \sqcup V_2\Gamma$; namely, primary (those in $V_0\Gamma$, denoted by $\bullet$), 1-secondary (those in $V_1\Gamma$, denoted by $\bigcirc$), and 2-secondary (those in $V_2\Gamma$, denoted by $\bigtriangleup$); and with all arcs in $E\Gamma$ joining (in either direction) a primary vertex with a secondary one; i.e., $E\Gamma = E_1\Gamma \sqcup E_2\Gamma$, where the arcs in $E_j$ are called j-arcs and connect primary vertices with j-secondary vertices (for $j = 1, 2$).

(ii) $\ell$ is a twofold *label map*: for $\nu = 1, 2$, $\ell: E_{\nu}\Gamma \to G_{\nu}$, $e \mapsto \ell(e)$ compatible with the involution in $\Gamma$; and $\ell: V_\nu\Gamma \to Sgp(G_{\nu})$, $q \mapsto \ell(q)$.

(iii) $\ast$ is a distinguished primary vertex called the *basepoint* of $\Gamma$.

We say that a wedge automaton $\Gamma$ is connected (resp., finite) if the underlying undirected graph is so; note that, by definition, it is always primary-secondary bipartite. We will also say that a wedge automaton $\Gamma$ is of *finite type* if the underlying digraph is finite, and the subgroups labelling the secondary vertices are all finitely generated. This will always be the situation when we consider computational issues; in this case, the labels of vertices will usually be given by finite sets of generators. We say that a vertex label is trivial if $\ell(q) = \{1\}$. If not stated otherwise all the wedge automata appearing from this point will be assumed to be finite.

Recall that wedge automata are involutive (as automata): for every arc $e \equiv p \longrightarrow q$ reading $g$, there exists a unique inverse arc $e^{-1} \equiv q \longrightarrow p$ reading $g^{-1}$. Hence, a wedge automaton $\Gamma$ can always be represented by one of its (say positive, denoted by $E^+\Gamma$) arc orientations. Then $E\Gamma = E^\pm\Gamma := E^+\Gamma \sqcup E^-\Gamma$, where $E^-\Gamma$ is the set of inverses of the arcs in $E^+\Gamma$. A *walk* in $\Gamma$ is a sequence of alternating and successively incident vertices and arcs, starting and ending at primary vertices, $\gamma = p_0(e_1^{-1}q_1e'_1)p_1(e_2^{-1}q_2e'_2)p_2 \cdots p_{r-1}(e_r^{-1}q_r e'_r)p_r$, where $p_0, \ldots, p_r$ are (not necessarily distinct) primary vertices, $q_1, \ldots, q_r$ are (not necessarily distinct) secondary vertices, and for every $k = 1, \ldots, r$, all three of $e_k, q_k, e'_k$ are simultaneously of the same type $\nu_k$ ($\nu_k = 1, 2$). A $\nu$-elementary walk is a walk of length 2 visiting a secondary vertex of $\nu$-type, $\nu = 1, 2$; it is degenerate if it consists of two mutually inverse arcs; otherwise it is called non-degenerate. Every walk $\gamma$ decomposes as a product of elementary walks (either
degenerate or nondegenerate, and with possible repetitions) in a unique way, corresponding to the brackets in the expression above: this is called the elementary decomposition of \( \gamma \) (for convention, we take \( r = 0 \) when the walk \( \gamma \) is trivial). We say that a walk \( \gamma \) is alternating if its elementary decomposition sequence \( \gamma_1, \gamma_2, \ldots, \gamma_r \) alternates between types 1 and 2.

The length of a walk is the number of arcs in the sequence defining it, i.e., twice the number of elementary walks \( r \) in its elementary decomposition.

**Remark 4.2.** Note that \( \gamma \) involves no backtracking if and only if the \( \gamma_i \)'s in its elementary decomposition \( \gamma = \gamma_1 \cdot \gamma_2 \cdots \gamma_r \) are all nondegenerate, and there is no backtracking in the consecutive products \( \gamma_i \cdot \gamma_{i+1} \).

**Definition 4.3.** The label of a walk \( \gamma = p_0(e_1^{-1}q_1e_1^r)p_1(e_2^{-1}q_2e_2^r)p_2 \cdots p_{r-1}(e_r^{-1}q_re_r^r)p_r \), denoted by \( \ell(\gamma) \), is the subset \( \ell(\gamma) = (\ell_{e_1}^{-1}(q_1\ell_{e_1}^r))(\ell_{e_2}^{-1}(q_2\ell_{e_2}^r)) \cdots (\ell_{e_r}^{-1}(q_r\ell_{e_r}^r)) \subseteq G_1 \ast G_2 \). That is, while travelling along \( \gamma \), when we traverse an arc \( e \), we pick its label \( \ell_e \), and when we traverse a secondary vertex \( q \) we take all labels \( c \in \ell_q \) (primary vertices have no contribution to \( \ell(\gamma) \)). Picking always the trivial element when visiting a secondary vertex, we obtain the so-called basic label of \( \gamma \), \( \ell^b(\gamma) = (\ell_{e_1}^{-1}\ell_{c_1}^1)(\ell_{e_2}^{-1}\ell_{c_2}^2) \cdots (\ell_{e_r}^{-1}\ell_{c_r}^r) \in \ell(\gamma) \subseteq G_1 \ast G_2 \).

It is clear that \( \ell^{-1}(\alpha) = \ell^{-1}(\alpha) \) and \( \ell(\alpha) \cdot \ell(\beta) = \ell(\alpha \cdot \beta) \).

**Remark 4.4.** Note also that if \( \gamma = p_0(e_1^{-1}q_1e_1^r)p_1(e_2^{-1}q_2e_2^r)p_2 \cdots p_{r-1}(e_r^{-1}q_re_r^r)p_r \), and we take elements \( c_i \in \ell_{q_i} \) such that \( \ell_{e_i}^{-1}c_i\ell_{e_i}^r \not= 1 \) for all \( i = 1, \ldots, r \), then the brackets in the expression \( (\ell_{e_1}^{-1}c_1\ell_{e_1}^r)(\ell_{e_2}^{-1}c_2\ell_{e_2}^r) \cdots (\ell_{e_r}^{-1}c_r\ell_{e_r}^r) \) indicate, precisely, the syllable decomposition of the element in \( G_1 \ast G_2 \) read by \( \gamma \); otherwise, some consecutive pairs of brackets may merge into the same syllable. This remark will be crucial later.

Let \( \Gamma \) be a \((G_1, G_2)\)-automaton and let \( p, p' \) be two primary vertices. We define the coset recognized by \( \Gamma \) relative to \((p, p')\) to be the set \( \langle \Gamma \rangle_{(p, p')} = \cup \ell \), where the union runs over all walks in \( \Gamma \) from \( p \) to \( p' \). When \( p = p' \), then we abbreviate \( \langle \Gamma \rangle_p \) := \( \langle \Gamma \rangle_{(p, p)} \). Moreover, if \( p = p' = \hat{e} \) then we simply write \( \langle \Gamma \rangle = \langle \Gamma \rangle_{\hat{e}} \) and we call it the subgroup recognized by \( \Gamma \). The lemma below, which is straightforward to prove, justifies this terminology.

**Lemma 4.5.** Let \( \Gamma \) be a \((G_1, G_2)\)- wedge automaton, and let \( p, p' \in V_0 \Gamma \). Then,

(i) \( \langle \Gamma \rangle_p \) is a subgroup of \( G_1 \ast G_2 \);

(ii) \( \langle \Gamma \rangle_p \) and \( \langle \Gamma \rangle_p' \) are conjugate to each other; viz. \( \langle \Gamma \rangle_p' = (\langle \Gamma \rangle_p)^g \), for every \( g \in \langle \Gamma \rangle_{(p, p')} \);

(iii) \( \langle \Gamma \rangle_{(p, p')} \) is a right coset of \( \langle \Gamma \rangle_{(p, p)} \) viz. \( \langle \Gamma \rangle_{(p, p')} = \langle \Gamma \rangle_{(p, p)} \cdot g \), for every \( g \in \langle \Gamma \rangle_{(p, p')} \) •

**Proposition 4.6.** For every subgroup \( H \subseteq G_1 \ast G_2 \), there exists a \((G_1, G_2)\)-automaton \( \Gamma \) recognizing \( H \). Furthermore, if \( H \) is finitely generated, one such \( \Gamma \) is of finite type and algorithmically constructible from a finite set of generators for \( H \) given in normal form.

**Proof.** Let \( H = \langle W \rangle \), where \( W = \{w_1, w_2, \ldots \} \) is a set of generators for \( H \). For every non-trivial generator in \( W \), say \( w \), consider its normal form as an element of \( G_1 \ast G_2 \), say \( w = a_1 b_1 \cdots a_s b_s \), with \( s \geq 1 \), \( a_i \in G_1 \), \( b_i \in G_2 \), \( a_i \not= 1 \) for all \( i = 2, \ldots, s \), and \( b_i \not= 1 \) for all \( i = 1, \ldots, s-1 \). Let \( \text{Fl}(w) \) denote the \((G_1, G_2)\)-automaton depicted in Fig. 5, and called the petal automaton corresponding to \( w \). Clearly, \( \langle \text{Fl}(w) \rangle = \langle w \rangle \).
Now consider $\text{Fl}(W)$ the disjoint union of all the $\text{Fl}(W_i)$’s identifying the basepoints into a single primary vertex (declared as basepoint); the resulting object is a $(G_1, G_2)$-automaton called the *flower automaton* corresponding to $W$. Clearly, $\langle \text{Fl}(W) \rangle = \langle W \rangle = H$. Moreover, if $|W| < \infty$ then $\text{Fl}(W)$ is of finite type and constructible.

4.2 Reduced wedge automata

In the same vein as in the classical Stallings graphs, we will ask our wedge graphs to be ‘deterministic’ (in a precise sense specified below). Similar constructions are called ‘irreducible graphs’ by Ivanov in [16], and are particular cases of the so-called ‘folded graphs’ in Kapovich–Weidman–Miasnikov [20].

**Definition 4.7.** Let $G_1, G_2$ be two groups, and let $\Gamma$ be a finite $(G_1, G_2)$-automaton. We say that $\Gamma$ is reduced if the following conditions are satisfied: (i) $\Gamma$ is connected; (ii) every primary vertex of $\Gamma$ is incident with at most one arc from $E_1 \Gamma$, and at most one arc from $E_2 \Gamma$; (iii) no nondegenerate elementary walk reads the trivial element; that is, for $\nu = 1, 2$, every $\nu$-secondary vertex $q \in V_\nu \Gamma$, and every pair of different $\nu$-arcs $e_1, e_2$ with $\iota_{e_1} = \iota_{e_2} = q$, we have that $1 \not\in \ell_{e_1}^{-1} q \ell_{e_2}$ (equivalently, $\ell_{e_1} \ell_{e_2}^{-1} \not\in \ell_q$).

In a reduced wedge automaton $\Gamma$, remark 4.2 can be restated in the following way: $\gamma$ presents no backtracking if and only if it is alternating and the elementary walks on its elementary decomposition are all nondegenerate. In this case, additionally, property (iii) from Definition 4.7 ensures that the elementary decomposition of $\gamma$ gives the syllable decomposition of every $g \in \ell_\gamma$ as element from $G_1 \ast G_2$ (since nondegenerate elementary walks do not admit the trivial element as a label). However, this is not the whole story: even with some of the $\gamma_i$’s being degenerate, we can still get the syllable decomposition of $g \in \ell_\gamma$ assuming that the elements picked from the labels of the backtracking vertices (if any) are nontrivial. This motivates the following lemma, definition and the subsequent important technical lemma.

**Lemma 4.8.** For any reduced $(G_1, G_2)$-automaton $\Gamma$, and any nontrivial walk $\gamma$ between two primary vertices and without backtracking, $1 \not\in \ell_\gamma$.

**Proof.** This is clear since, by the previous remark, the label of $\gamma$ gives its syllable decomposition as element of $G_1 \ast G_2$. □
Definition 4.9. Let \( \gamma \) be a walk in a wedge automaton \( \Gamma \), with elementary decomposition \( \gamma = \gamma_1 \gamma_2 \cdots \gamma_r = p_0(e_1^{-1}q_1e'_1)p_1(e_2^{-1}q_2e'_2)p_2 \cdots p_{r-1}(e_r^{-1}q_re'_r)p_r \). We define the reduced label of \( \gamma \) as

\[
\widetilde{\ell}_\gamma = \left\{ (\ell_{e_i}^{-1}c_1\ell_{e'_j})(\ell_{e_2}^{-1}c_2\ell_{e'_2}) \cdots (\ell_{e_r}^{-1}c_r\ell_{e'_r}) \mid c_i \in \ell_{q_i}, c_i \neq 1 \text{ if } \gamma_i \text{ is degenerate} \right\} \subseteq \ell_\gamma.
\]

Lemma 4.10. For a reduced \( (G_1, G_2) \)-automaton \( \Gamma \), we have \( \langle \Gamma \rangle = \bigcup_\gamma \ell_\gamma = \bigcup_\gamma \widetilde{\ell}_\gamma \), where the first union runs over all \( \ast \)-walks \( \gamma \) of \( \Gamma \), and the second one only over the alternating \( \ast \)-walks \( \hat{\gamma} \) of \( \Gamma \).

Proof. The inclusion ‘\( \supseteq \)’ is clear, since the first union is over more sets than the second one, and \( \ell_\gamma \supseteq \widetilde{\ell}_\gamma \). To see ‘\( \subseteq \)’, fix a \( \ast \)-walk \( \gamma = \gamma_1 \cdots \gamma_r \), and an element \( g \in \ell_\gamma \), and let us find an alternating \( \ast \)-walk \( \hat{\gamma} \) such that \( g \in \widetilde{\ell}_{\hat{\gamma}} \).

In fact, if \( \gamma \) is not alternating then there are \( \nu = 1, 2 \) and \( i = 1, \ldots, r-1 \) such that \( \gamma_i = p_{i-1}e_i^{-1}q_ie_ip_i \) and \( \gamma_{i+1} = p_ie_{i+1}^{-1}q_{i+1}e_{i+1}'p_{i+1} \) are both of type \( \nu \); so, by condition (ii) in Definition 4.7, \( q_i = q_{i+1} \) and \( e_i = e_{i+1}' \). Replacing \( \gamma_i \gamma_{i+1} = p_{i-1}e_i^{-1}q_ie_{i+1}'p_{i+1} \), we get a new \( \ast \)-walk \( \gamma_{i, r} \), with shorter elementary decomposition and such that \( g \in \ell_{\gamma_{i, r}} \) as well, since \( (\ell_{e_i}^{-1}c_1\ell_{e'_j})(\ell_{e_{i+1}}^{-1}c_{i+1}\ell_{e'_{i+1}}) = \ell_{e_i}^{-1}(c_i c_{i+1}')\ell_{e'_{i+1}} \), for all \( c_i, c_{i+1} \in \ell_{q_i} \subseteq G_\gamma \). Repeating this operation a finite number of times, say \( k \), we can assume that \( \hat{\gamma} = \gamma_{i, k} \) is alternating and \( g \in \ell_{\hat{\gamma}} \).

It remains to prove that, maybe simplifying \( \hat{\gamma} \), \( g \in \widetilde{\ell}_{\hat{\gamma}} \) if \( q_j \) is the secondary vertex in the degenerate elementary walk \( \gamma_j = p_{j-1}e_j^{-1}q_je_jp_{j-1} \), and the corresponding \( c_j \) picked in the formation of \( g \) is trivial, then just ignore \( (\ell_{e_j}^{-1}c_j\ell_{e'_j}) = 1 \), and realize \( g \) in the label of \( \hat{\gamma}_{(i)} = \gamma_1 \cdots \gamma_{j-1} \gamma_{j+1} \cdots \gamma_s \), a \( \ast \)-walk with shorter elementary decomposition, which will be again alternating after repeating the operation in the above paragraph. Repeating this operations a finite number of times, until no trivial choices are made at the degenerate vertices, we obtain the desired result. \( \square \)

Remark 4.11. The usefulness of the previous lemma is the following: when realizing an element \( g \in \langle \Gamma \rangle \) from the subgroup recognized by a reduced automaton \( X \) as \( g \in \widetilde{\ell}_\gamma \) for some alternating \( \ast \)-walk \( \hat{\gamma} \), the elementary decomposition of \( \hat{\gamma} \) automatically provides the syllable decomposition of \( g \) as element of \( G_1 \ast G_2 \). This is a crucial bridge between the algebraic and the geometric aspects of the theory.

One of the most useful applications of reduced \((G_1, G_2)\)-automata is that they naturally encode the Kurosh free product decomposition (the induced splitting) of their recognized subgroups as subgroups of \( G_1 \ast G_2 \). With some technical differences, our exposition follows [16, 20], but with special emphasis on the algorithmic point of view. The theorem below appears as Lemma 4 in Ivanov’s [18] and, in a more general setting, as Proposition 4.3 in Kapovich–Weidmann–Myasnikov [20]. A detailed proof (including the algorithmicity in the the finite type case) can be found in [6].

Notation 4.12. Let \( \Gamma \) be a reduced \((G_1, G_2)\)-wedge automaton. Fix a maximal subtree \( T \) in \( \Gamma^+ \), and let \( E = \Gamma^+ \setminus ET \) be the set of arcs of \( \Gamma^+ \) outside \( T \). For every two vertices \( u, v \in \Gamma T \), let \( T[u, v] \) denote the unique walk without backtracking from \( u \) to \( v \) along the tree \( T \). Now,
for every vertex \( u \in V \), let \( z_u \) denote the basic label 
\[ z_u = \ell_u \in G_1 * G_2 \]
and, for every arc \( e \in E \), let \( x_e \) denote the element 
\[ x_e = z_e^{-1} \ell_e z_e \in G_1 * G_2. \]

**Theorem 4.13.** Let \( \Gamma \) be a reduced \((G_1, G_2)\)-automaton. Then, with the above notations, the subgroup recognized by \( \Gamma \) is

\[ \langle \Gamma \rangle = \mathbb{F} \ast \left( \bigstar_{q \in V_1} z_q^{-1} \ell_q z_q \right) \ast \left( \bigstar_{q \in V_2} z_q^{-1} \ell_q z_q \right), \]

where \( \mathbb{F} \) is the free subgroup of \( \langle \Gamma \rangle \) freely generated by the set \( \{ x_e \mid e \in E \} \). Moreover, if \( \Gamma \) is of finite type, then the subgroup \( \langle \Gamma \rangle \) is finitely generated, and we can algorithmically compute a Kurosh decomposition like (8) for \( \langle \Gamma \rangle \leq G_1 * G_2. \)

**Corollary 4.14.** For a reduced \((G_1, G_2)\)-automaton \( \Gamma \), the group \( \langle \Gamma \rangle \) is finitely generated if and only if \( \Gamma \) is of finite type (i.e., the underlying graph of \( \Gamma \) has finite rank, and all vertex labels of \( \Gamma \) are finitely generated). In this case, a set of generators (in the form of the Kurosh decomposition theorem) for \( \langle \Gamma \rangle \) is computable.

Theorem 4.13 and Corollary 4.14 are easily seen to be false if we substitute reduced automata by general wedge automata.

### 4.3 Effective reduction of wedged automata

Following [16] and [20], the next step is to show that every finitely generated subgroup \( H \leq G_1 * G_2 \) is the subgroup recognized by some reduced \((G_1, G_2)\)-automaton of finite type. We introduce several elementary operations on wedge automata which will not change their recognized subgroup (to simplify notation, we shall work with positive arcs and assume that everything done to an arc \( e \) will also be done accordingly to \( e^{-1} \)).

**Definition 4.15.** Let us consider the following elementary transformations on a \((G_1, G_2)\)-wedge automata:

(i) **Adjustment:** replacing the label of any arc \( e \) leaving a secondary vertex \( q \), by \( c \cdot \ell_e \), for any \( c \in \ell_q \); see Figure 6.

(ii) **Conjugation:** (for \( v = 1, 2 \)) replacing, given \( g \in G_v \), the label \( \ell_q \) of a \( v \)-secondary vertex \( q \), by \( (\ell_q)^{g^{-1}} = g\ell_q g^{-1} \); and replacing the label \( \ell_{e_i} \) of every arc \( e_i \) incident from \( q \), by the respective \( g\ell_{e_i} \); see Figure 7.
Figure 7: Conjugation

(iii) **Isolation**: removing from $\Gamma$ all the connected components not containing the basepoint; see Figure 8.

![Figure 8: Isolation](image)

(iv) **Primary open folding**: for $\nu = 1, 2$, given two $\nu$-secondary vertices $q_1, q_2$ adjacent to the same primary vertex through respective arcs $e_1, e_2$ with the same label $g \in G\nu$, identify $q_1$ and $q_2$ into a new secondary vertex with label $\langle \ell_{q_1}, \ell_{q_2} \rangle$, and identify the arcs $e_1, e_2$ into a new arc with the same label $g$; see Figure 9.

![Figure 9: Primary open folding, where $C = \langle C_1 \cup C_2 \rangle$](image)

(v) **Secondary open folding**: for $\nu = 1, 2$, given a $\nu$-secondary vertex $q$ adjacent to two different primary vertices $p_1, p_2$ through arcs $e_1, e_2$ having the same label $g \in G\nu$, identify the vertices $p_1$ and $p_2$, and the arcs $e_1, e_2$ into an arc maintaining the label $g$; see Figure 10.

![Figure 10: Secondary open folding](image)

(vi) **Closed folding**: for $\nu = 1, 2$, given a primary vertex $p$ adjacent to a $\nu$-secondary vertex $q$ by two non mutually inverse arcs $e_1, e_2$, consists in identifying $e_1$ and $e_2$ into a single arc with label $\ell_{e_1}$, and change the label of $q$ from $\ell_q$ to $\langle \ell_q, \ell_{e_1}\ell_{e_2}^{-1} \rangle \leq G\nu$; see Figure 11.
Figure 11: Closed folding, where $C' = \langle C, gh^{-1} \rangle$

Note that the (folding) transformations (iv), (v), and (vi) in Definition 4.15 decrease the number of arcs in the automata exactly by 1. The following result is also straightforward.

**Lemma 4.16.** Let $\Gamma$ be a reduced $(G_1, G_2)$-automaton, and let $\Gamma \sim \Gamma'$ be any of the elementary transformations in Definition 4.15. Then, the recognized subgroups of $\Gamma$ and $\Gamma'$ coincide. $\square$

**Theorem 4.17** (S. V. Ivanov, 1999, [16]). For any groups $G_1, G_2,$ and any finitely generated subgroup $H \leq G_1 * G_2,$ there exists a reduced $(G_1, G_2)$-automaton recognizing $H.$ Moreover, if both $G_1$ and $G_2$ have solvable membership problem, then given a finite set of generators of a subgroup $H \leq G_1 * G_2,$ one can algorithmically obtain a reduced $(G_1, G_2)$-automaton of finite type recognizing $H.$

**Proof.** Since the trivial automaton recognizes the trivial subgroup, we can assume $H \neq 1.$ Starting with the corresponding flower automaton for $H,$ successively do the following:

(I) if, for $\nu = 1, 2,$ $\Gamma$ has a primary vertex adjacent to two different $\nu$-secondary vertices, then apply a suitable conjugation move to one of the $\nu$-vertices and then a primary open folding.

(II) if, for $\nu = 1, 2,$ $\Gamma$ has a primary vertex and a $\nu$-secondary vertex connected to each other by two non mutually inverse arcs then apply a closed folding.

(III) if $\Gamma$ has a secondary vertex $q$ adjacent to two different primary vertices through arcs $e_1$ and $e_2,$ $te_1 = te_2 = q,$ such that $\ell_{e_1}\ell_{e_2}^{-1} \in \ell_q,$ apply a suitable adjustment move to one of the two arcs, and then an elementary open folding.

Note that this can be done algorithmically (we use MP in $G_1$ and $G_2$ for (III)). Since the number of edges decreases at each step, this process will eventually stop. It is clear that the resulting automaton is reduced, of finite type, and it recognizes $H$ by Lemma 4.16. $\square$

### 4.4 A reduced automaton for the intersection

Recall that if $G_1$ and $G_2$ are Howson, then $G_1 * G_2$ is Howson as well; see [3] and [16], the second proof being essentially the one we present here.

The goal of this section is to describe, following [16], a reduced automaton $\Gamma_{H \∩ K}$ recognizing the intersection $H \cap K,$ in terms of given reduced automata $\Gamma_H$ and $\Gamma_K,$ with $\langle \Gamma_H \rangle = H$ and $\langle \Gamma_K \rangle = K,$ where $H, K \leq G_1 * G_2$ are finitely generated subgroups. This construction is
not algorithmic, in general, since $\Gamma_H \times \Gamma_K$ may very well be not of finite type, even when both $\Gamma_H$ and $\Gamma_K$ are so (corresponding to the case where $H, K$ are both finitely generated but $H \cap K$ is not, see Corollary 4.14).

Later, in Section 4.6, we shall give an effective procedure which starts constructing, locally, the aforementioned automaton $\Gamma_H \times \Gamma_K$. While running, there will be an alert observing the construction: if some (algorithmically checkable) specific situation occurs, then the intersection $H \cap K$ is not finitely generated. We shall achieve our goal by proving that, in finite time, either the alert sounds or the procedure terminates providing the finite type reduced automaton $\Gamma_H \times \Gamma_K$ as output.

So, suppose we are given two finite type reduced $(G_1, G_2)$-wedge automata $\Gamma_H$ and $\Gamma_K$ with recognized subgroups $\langle \Gamma_H \rangle = H$ and $\langle \Gamma_K \rangle = K$, where $H, K \leq G_1 \ast G_2$ are finitely generated subgroups. Along the following paragraphs, we will first define the product automaton $\tilde{\Gamma}$ of $\Gamma_H$ and $\Gamma_K$ whose main connected component will be the junction $\Gamma_H \times \Gamma_K$.

Define the set of primary vertices of the product $\tilde{\Gamma}$ as the cartesian product $V_0\tilde{\Gamma} = V_0\Gamma_H \times V_0\Gamma_K$, and the basepoint $*$ of $\tilde{\Gamma}$ to be the pair of basepoints, i.e., $* = (\ast_H, \ast_K)$. Now, for $\nu = 1, 2$, we consider the subsets of primary vertices (in $\Gamma_H$ and $\Gamma_K$) adjacent to some $\nu$-secondary vertex, i.e., $V_{0-\nu} \Gamma_H = \{ p \in V_0\Gamma_H \mid p \text{ is adj. to a } \nu\text{-secondary in } \Gamma_H \} \subseteq V_0\Gamma_H$ (and idem for $V_{0-\nu} \Gamma_K$) and define the relation $\equiv_\nu$ on the set $V_{0-\nu} \Gamma_H \times V_{0-\nu} \Gamma_K \subseteq V_0\Gamma$ to be: $(p_1, p_1') \equiv_\nu (p_2, p_2')$ if and only if there exist two $\nu$-elementary walks: $\gamma$ from $p_1$ to $p_2$ in $\Gamma_H$ (say $\gamma = p_1 e_1 e_2 \cdots e_{25}$, with $q \in V_{0-\nu} \Gamma_H$), and $\gamma'$ from $p_1'$ to $p_2'$ in $\Gamma_K$ (say $\gamma' = p_1' e_1' e_2' \cdots e_{25}'$ with $q' \in V_{0-\nu} \Gamma_K$), such that the intersection of their labels is nonempty, $\ell_\nu \cap \ell_{\nu'} \neq \emptyset$.

The following result is not hard to see with the natural reasoning.

**Lemma 4.18.** The relations $\equiv_1$ and $\equiv_2$ are equivalence relations. □

Now, define the $\nu$-secondary vertices of $\tilde{\Gamma}$ to be the equivalence classes modulo $\equiv_\nu$, namely $V_\nu \tilde{\Gamma} = (V_{0-\nu} \Gamma_H \times V_{0-\nu} \Gamma_K) / \equiv_\nu$. Finally, define the $\nu$-arcs of $\tilde{\Gamma}$ as $E_\nu \tilde{\Gamma} = \{ q \to p \mid q \in V_\nu \tilde{\Gamma}, p \in q \}$, i.e., for each secondary vertex $q \in V_\nu \tilde{\Gamma}$, and each primary vertex $p = (p, p') \in q$, add a $\nu$-arc $q \to p$. This finishes the definition of the underlying digraph of $\tilde{\Gamma}$. Observe that $\tilde{\Gamma}$ may not be connected in general, even with $\Gamma_H$ and $\Gamma_K$ being so.

By construction, it is clear that, for $\nu = 1, 2$, every primary vertex of $\tilde{\Gamma}$ is adjacent to at most one $\nu$-secondary vertex of $\tilde{\Gamma}$ through at most one arc. We define now natural projections (digraph homomorphisms) from $\tilde{\Gamma}$ to $\Gamma_H$ and $\Gamma_K$. Define $\pi : \tilde{\Gamma} \to \Gamma_H$ as follows: for primary vertices, take the projection to the first coordinate, $\pi : V_0\tilde{\Gamma} \to V_0\Gamma_H$, $(p, p') \mapsto p$; for $\nu$-secondary vertices, assign to every vertex $q \in V_\nu \tilde{\Gamma}$ the only $\nu$-secondary vertex in $\Gamma_H$ adjacent to every $p = (p, p') \pi \in V_0 \Gamma_H$ for $(p, p') \in q$; finally, for $\nu$-arcs, assign to every $q \to p$ (for $q \in V_\nu \tilde{\Gamma}$ and $(p, p') \in q$) the unique $\nu$-arc in $\Gamma_H$ from $q = q_{\pi}$ to $p = (p, p') \pi$. Clearly, $\pi$ is a well defined digraph homomorphism, called the projection to $\Gamma_H$. The projection to $\Gamma_K$, denoted by $\pi' : \tilde{\Gamma} \to \Gamma_K$, is defined in an analogous way.

It remains to establish the labels for its vertices and arcs. For $\nu = 1, 2$, and for every $\nu$-secondary vertex $q \in V_\nu \tilde{\Gamma}$, choose a distinguished primary vertex $p_q = (p_q, p_q') \in q$, and let $e_q$ be the only arc in $\tilde{\Gamma}$ form $q$ to the representative $p_q$. This means that, in $\Gamma_H$, there
is a $\nu$-arc $e_q := (e_q)\pi \in E_\nu \Gamma_{1\nu}$ from $q = q\pi \in V_\nu \Gamma_{1\nu}$ to $p_q$, and, in $\Gamma_{1\nu}$, there is a $\nu$-arc $e'_q := (e_q)\pi' \in E_\nu \Gamma_{1\nu}$ from $q' = q\pi' \in V_\nu \Gamma_{1\nu}$ to $p'_q$. Then, we define the label of vertex $q$ as
\[
\ell_q := \ell_{e_q}^{-1} \ell_q \ell_{q'}^{-1} \ell_{q'} \equiv \ell_q \ell_{q'} \leq G_\nu.
\]
Finally, for any $\nu$-arc $e \in E_\nu \bar{\Gamma}$ from $q$ to a primary vertex $p = (p, p') \in q$, call $e := e\pi$, $e' := e\pi'$, and define the label of $e$ as an arbitrary element from the coset intersection $\ell_e \in \ell_{e_q}^{-1} \ell_q \ell_{e'}^{-1} \ell_{e'}$, which is nonempty since $(p_q, p'_q) \equiv (p, p')$ by construction; see Figure 12. Note that, in particular, we can take $\ell_{e_q} = 1$.

![Figure 12: Secondary vertex and related labels in $\bar{\Gamma}$](image)

This completes the definition of the product automaton $\bar{\Gamma}$ (technically depending on some choices made on the way). The main property of the labels defined for $\bar{\Gamma}$ is expressed in the following lemma.

**Lemma 4.19.** Let $\bar{\Gamma}$ be the product of two reduced wedge automata $\Gamma_{1\nu}$ and $\Gamma_{1\nu}$. Then, for every $\nu$-elementary walk $\gamma$ in $\bar{\Gamma}$, the projected walks $\gamma \pi$ and $\gamma \pi'$ are $\nu$-elementary in $\Gamma_{1\nu}$ and $\Gamma_{1\nu}$, respectively, and we have $\ell_\gamma = \ell_{\gamma \pi} \cap \ell_{\gamma \pi'}$. Furthermore, $\gamma$ is degenerate if and only if both $\gamma \pi$ and $\gamma \pi'$ are degenerate.

**Proof.** Clearly, the projections by $\pi$ and $\pi'$ of $\nu$-elementary walks in $\bar{\Gamma}$ are also $\nu$-elementary walks (in $\Gamma_{1\nu}$ and $\Gamma_{1\nu}$, respectively), with the original one being degenerate if and only if both projections are degenerate (note that $\gamma$ may be nondegenerate with one (and only one) of $\gamma \pi$ and $\gamma \pi'$ being degenerate; see Figure 12).

To see the equality in labels, let $\gamma = p_1 e_1^{-1} q e_2 p_2$ be a $\nu$-elementary walk in $\bar{\Gamma}$, where $p_1 = (p_1, p'_1)$ and $p_2 = (p_2, p'_2)$, and let $\gamma \pi = p_1 e_1^{-1} q e_2 p_2$ and $\gamma \pi' = p'_1 e_1^{-1} q e_2 p'_2$ be the corresponding $\nu$-elementary walks in $\Gamma_{1\nu}$ and $\Gamma_{1\nu}$, respectively. We have $\ell_{\gamma \pi} = \ell_{e_1}^{-1} \ell_q \ell_{e_2}$ and $\ell_{\gamma \pi'} = \ell_{e'_1}^{-1} \ell_q \ell_{e'_2}$.
Now, consider the distinguished \( v \)-arc \( e_q \) incident to \( q \) (possibly equal to \( e_1 \) and/or \( e_2 \)). According to the above definitions, we have \( \ell_q = \ell_q^{e_q} \cap \ell_q^{e'_q} \) and \( \ell_i \in \ell_i^{e_i} \ell_q \cap \ell_i^{e'_i} \ell_q \) for \( i = 1, 2 \); see Figure 12. Therefore,

\[
\ell_y = \ell_{e_1}^{e_1} \ell_{e_2} = \ell_{e_1}^{e_1} \left( \ell_q^{e_q} \cap \ell_q^{e'_q} \right) \cdot \ell_{e_2} = \ell_{e_1}^{e_1} \ell_q \ell_{e_2} \cap \ell_{e_i}^{e'_i} \ell_q \ell_{e'_i} = \ell_{y\pi} \cap \ell_{y\pi'}.
\]

(10)

This completes the proof.

We can now state the definition of junction automaton \( \Gamma_{H \land K} \) and show that it is a reduced \((G_1, G_2)\)-wedge automaton such that \( \langle \Gamma_{H \land K} \rangle = H \cap K \).

**Definition 4.20.** With the above notation, we define the junction automaton of \( \Gamma_{H} \) and \( \Gamma_{K} \), denoted by \( \Gamma_{H \land K} \), as the connected component of \( \tilde{\Gamma} \) containing the basepoint. (Recall that there are some arbitrary choices made on the way so, \( \Gamma_{H \land K} \) is not canonically associated to \( \Gamma_{H} \) and \( \Gamma_{K} \).

**Proposition 4.21.** The junction automaton \( \Gamma_{H \land K} \) is a \((G_1, G_2)\)-reduced automaton recognizing the subgroup \( H \cap K \).

**Proof.** It is clear that the junction automaton satisfies the properties (i) and (ii) in Definition 4.7.

To see property (iii), take \( v = 1, 2 \), let \( q \in V, \tilde{\Gamma} \) be a \( v \)-secondary vertex of \( \tilde{\Gamma} \), let \( e_1, e_2 \) be two different \( v \)-arcs from \( q \) to \( p_1 = (p_1, p'_1) \) and \( p_2 = (p_2, p'_2) \), respectively, and consider the nondegenerate elementary walk \( y = p_1 e_1^{q_1} q_2 e_2 p_2 \). By symmetry, we can assume \( p_1 \neq p_2 \), i.e., that \( y\pi = p_1 e_1^{q_1} q_2 e_2 p_2 \) is a nondegenerate elementary walk in \( \Gamma_{H} \). Since \( \Gamma_{H} \) is reduced, \( 1 \not\in e_1^{-1} e_q e_2 = e_{y\pi} \) and by Lemma 4.19, \( 1 \not\in e_{y\pi} = e_{\pi y} \cap e_{y\pi'} \).

It remains to show that \( \langle \Gamma_{H \land K} \rangle = H \cap K \). Indeed, let \( y \) be an arbitrary \( \pi \)-walk in \( \tilde{\Gamma} \), and let \( y = y_1 \cdots y_r \) be its elementary decomposition. Clearly, the elementary decompositions of \( y\pi \) in \( \Gamma_{H} \), and \( y\pi' \) in \( \Gamma_{K} \), are \( y\pi = (y_1\pi) \cdots (y_r\pi) \) and \( y\pi' = (y_1\pi') \cdots (y_r\pi') \), respectively. Then, by Lemma 4.19, \( e_y \cdot e_{y\pi} \subseteq e_{y\pi} \cdots e_{y\pi'} = e_{(y_1\pi) \cdots (y_r\pi)} = e_{y\pi} \subseteq H \). Since this is true for every \( y \), we deduce \( \langle \Gamma_{H \land K} \rangle \subseteq H \); and, by the symmetric argument, also \( \langle \Gamma_{H \land K} \rangle \subseteq K \).

For the other inclusion, take an element \( g \in H \cap K \), and let \( g = g_1 \cdots g_r \) be its syllable decomposition in \( G_1 \times G_2 \). Since \( \Gamma_{H} \) is a reduced automaton and \( \langle \Gamma_{H} \rangle = H \), Lemma 4.10 ensures us that \( g \in e_{y} \) for some alternating \( \pi \)-walk \( y \) from \( \Gamma_{H} \); in this situation, its elementary decomposition, \( y = y_1 \cdots y_r \), corresponds to the syllable decomposition \( g = g_1 \cdots g_r \), i.e., \( g_i \in e_{y_i} \), for \( i = 1, \ldots, r \). Similarly, there exists an alternating \( \pi \)-walk \( y' \) from \( \Gamma_{K} \), whose elementary decomposition \( y' = y'_1 \cdots y'_r \) again corresponds to the syllable decomposition \( g = g_1 \cdots g_r \), i.e., \( g_i \in e_{y'_i} \), for \( i = 1, \ldots, r \).

Write \( y_i = p_i e_i^{q_i} f_i p_i \) and \( y'_i = p'_i e'_i^{q'_i} f'_i p'_i \). Then, for each \( i = 1, \ldots, r \), we have \( g_i \in e_{y_i} = e_i^{q_i} e_i f_i e_i \) and \( g_i \in e_{y'_i} = e_i^{q'_i} e_i f'_i e_i \); so, \( g \not\in e_i^{q_i} e_i f_i e_i \cup e_i^{q'_i} e_i f'_i e_i \subseteq G_v \). This means that \( p_{i-1} = (p_{i-1}, p'_{i-1}) \equiv_{v_i} (p_i, p'_i) = p_i \), where \( v_i \) is the common type of the vertices \( q_i \) (in \( \Gamma_{H} \)) and \( q'_i \) (in \( \Gamma_{K} \)). Therefore, \( p_{i-1} \) and \( p_i \) are both incident to a common \( v_i \)-secondary
vertex in $\Gamma_H \land \Gamma_K$. In other words, there is a $v_i$-elementary walk in $\Gamma_H \land \Gamma_K$, say $\gamma_i$, from $p_{i-1}$ to $p_i$. Finally, by Lemma 4.19, $g_i \in \ell_{y_1} \cap \ell_{y'_1} = \ell_{y_i}$. Therefore, $g = g_1 \cdots g_r \in \ell_{y_1} \cdots \ell_{y_r} = \ell_{y_1} \cdots y_r \subseteq (\Gamma_H \land \Gamma_K)$, concluding the proof.

**Corollary 4.22.** In the above situation, $H \cap K$ is finitely generated if and only if all the vertex labels of $\Gamma_H \land \Gamma_K$ are finitely generated.

### 4.5 Understanding intersections of cosets

According to Lemma 4.5, given a wedge automaton $\Gamma_H$, the union of labels of all the walks in $\Gamma_H$ from the basepoint to a primary vertex $p \in V_0 \Gamma_H$, denoted by $\langle \Gamma_H \rangle_{(\phi,p)}$, constitutes a coset of the recognized subgroup $\langle \Gamma_H \rangle = H$. In general, though, this does not reflect all the cosets of $H$ (consider, for example the cases when $\Gamma_H$ has only finitely many primary vertices, but $H$ has infinite index in $G_1 \ast G_2$). We can slightly modify the automaton $\Gamma_H$ to fix this issue.

Let $u = a_1 b_1 \cdots a_s b_s \in G_1 \ast G_2$, written in normal form. Consider the $(G_1, G_2)$-wedge automaton (also denoted by $u$) consisting on a chain spelling the normal form for $g$, and having trivial vertex label; let us call it the *thread* for $u$. Attach this thread to $\Gamma_H$ by identifying the basepoint $*_H$ with $u$, and then apply the folding process until no more foldings are possible (see the proof of Theorem 4.17). Observe that operation (II) will not be used, and the triviality of the vertex labels in the thread implies that the vertex groups already present in $\Gamma_H$ will not be changed along the process. So, the output is the exact same graph $\Gamma_H$ with a terminal segment of the thread attached somewhere and sticking out; denote this new automaton by $\Gamma_{Hu}$. Clearly, $\Gamma_{Hu}$ is a reduced automaton, like $\Gamma_H$, and furthermore, since the new secondary vertices out of $\Gamma_H$ have trivial label, $\langle \Gamma_{Hu} \rangle = \langle \Gamma_H \rangle = H$.

By Lemma 4.5 (iii), $\langle \Gamma_{Hu} \rangle_{(\phi, t_u)} = \langle \Gamma_H \rangle \cdot u$ (the situation where this coset could already be represented by a vertex in $\Gamma_H$ corresponds to the fact that the thread happens to fold completely and so, $\Gamma_{Hu} = \Gamma_H$).

Now let us go back to the graph $\Gamma_H \land \Gamma_K$. It is useful to understand the intersection of $H$ and $K$ but also, adding the corresponding hairs, it will be useful to understand the intersection of two arbitrary cosets $Hu$ and $Kv$.

Given elements $u, v \in G_1 \ast G_2$, consider the reduced automata $\Gamma_{Hu}$ and $\Gamma_{Kv}$, and consider the junction automaton $\Gamma_{Hu} \land \Gamma_{Kv}$.

**Lemma 4.23.** With the above notation,

(i) $\Gamma_H \land \Gamma_K$ is a reduced subgraph of $\Gamma_{Hu} \land \Gamma_{Kv}$;

(ii) $Hu \cap Kv \neq \emptyset$ if and only if the vertex $(t_u, t_v)$ belongs to $\Gamma_{Hu} \land \Gamma_{Kv}$;

(iii) for any walk $\gamma$ in $\Gamma_{Hu} \land \Gamma_{Kv}$ from $(\phi_H, \phi_K)$ to $(t_u, t_v)$, and any $g \in \ell_\gamma$, we have $Hu \cap Kv = \langle \Gamma_{Hu} \land \Gamma_{Kv} \rangle_{(\phi_H, \phi_K), (\tau_u, \tau_v)} = (H \cap K)g$. 

28
Proof. Note that the initial set of primary vertices for $\Gamma_{HU} \wedge \Gamma_{KV}$, namely $V_0 \Gamma_{HU} \times V_0 \Gamma_{KV}$, contains as a subset $V_0 \Gamma_H \times V_0 \Gamma_K$, the initial set of primary vertices for $\Gamma_H \wedge \Gamma_K$. And two old vertices $(p_1, p_1'), (p_2, p_2') \in \Gamma_H \wedge \Gamma_K$ are $\equiv_v$-equivalent in $\Gamma_H \wedge \Gamma_K$ if and only if they are $\equiv_v$-equivalent as vertices in $\Gamma_{HU} \wedge \Gamma_{KV}$ (since vertices of $\Gamma_{HU} \wedge \Gamma_{KV}$ outside $\Gamma_H \wedge \Gamma_K$ have always trivial labels). This proves (i).

Suppose first that $(\tau u, \tau v) \in V \Gamma_{HU} \wedge \Gamma_{KV}$, let $\gamma$ be a walk in $\Gamma_{HU} \wedge \Gamma_{KV}$ from $(\ast_H, \ast_K)$ to $(\tau u, \tau v)$, and consider its basic label $\ell_\gamma \in G_1 \ast G_2$. By the same argument as in Proposition 4.21, $\ell_\gamma \in \ell_\gamma' \cap \ell_\gamma''$. But $\gamma \gamma'$ is a walk in $\Gamma_{HU}$ from $\ast_H$ to $\tau u$ (resp., a walk in $\Gamma_{KV}$ from $\ast_K$ to $\tau v$) hence, by Lemma 4.5 (iii), $\ell_\gamma \in (\Gamma_{HU})(\ast_H, \tau u) \cap (\Gamma_{KV})(\ast_K, \tau v) = \Gamma_H \cup \Gamma_K$, concluding that $\Gamma_H \cup \Gamma_K \neq \emptyset$.

Conversely, suppose that $\Gamma_H \cup \Gamma_K \neq \emptyset$ and let $g \in \Gamma_H \cup \Gamma_K$. Again by Lemma 4.5 (iii), there exist walks $\gamma$ in $\Gamma_{HU}$ from $\ast_H$ to $\tau u$, and $\gamma'$ in $\Gamma_{KV}$ from $\ast_K$ to $\tau v$, such that $g \in \ell_\gamma \cap \ell_\gamma'$. Again, with an argument like in the proof of Proposition 4.21, there exists a walk $\gamma$ in $\Gamma_{HU} \wedge \Gamma_{KV}$ from $(\ast_H, \ast_K)$ to $(\tau u, \tau v)$ such that $g \in \ell_\gamma$. In particular, $(\tau u, \tau v) \in \Gamma_{HU} \wedge \Gamma_{KV}$. This proves (ii) and (iii).

4.6 Proofs of Theorems 1.8 and 1.9

Let us now address the algorithmic aspects of this construction. For all the present section, assume the two starting reduced $(G_1, G_2)$-automata $\Gamma_H$ and $\Gamma_K$ to be of finite type (namely, $H$ and $K$ are finitely generated groups of $G_1 \ast G_2$). Recall that although the underlying graph of $\Gamma_H \wedge \Gamma_K$ is finite, the labels of the vertices in $\Gamma_H \wedge \Gamma_K$ may very well be non finitely generated as a result of intersections of finitely generated subgroups of $G_1$ and of $G_2$.

A first easy observation is that, under the assumption that both $G_1$ and $G_2$ are Howson, then $\Gamma_H \wedge \Gamma_K$ will always be of finite type. This proves that the free product of two Howson groups is again Howson, recovering a classical result originally proved by B. Baumslag in [3].

Proof of Theorem 1.8. Assume that both $G_1$ and $G_2$ have solvable ESIP; and suppose we are given two finitely generated subgroups $H, K \leq G_1 \ast G_2$ by finite sets of generators, and two extra elements $u, v \in G_1 \ast G_2$, all of them in normal form. By Remark 2.2, both $G_1$ and $G_2$ also have solvable membership problem; and by Theorem 4.17, we can compute reduced $(G_1, G_2)$-automata $\Gamma_H$ and $\Gamma_K$ such that $\langle \Gamma_H \rangle = H$, and $\langle \Gamma_K \rangle = K$.

Now let us keep constructing $\Gamma_H \wedge \Gamma_K$: we start looking at the basepoint $\ast = (\ast_H, \ast_K)$, with the whole set $V_0 \tilde{\Gamma} = V_0 \Gamma_H \times V_0 \Gamma_K$ in the background. We have to keep adding $v$-secondary vertices (with their labels), and $v$-arcs (with their labels too) connecting them to certain primaries, until getting $\Gamma_H \wedge \Gamma_K$, the full connected component of $\tilde{\Gamma}$ containing the basepoint $\ast$.

We start checking whether there exists $v \in \{1, 2\}$, such that both $\ast_H$ and $\ast_K$ have nonempty $v$-neighborhoods. If not, then the basepoint $\ast$ is not adjacent to any secondary vertex in $\tilde{\Gamma}$, and we are done (namely, the product is the trivial automaton, and $H \cap K = 1$). Otherwise,
let \( q \in V_v \Gamma_H \), and \( e \in E_v \Gamma_H \) with \( te = q, \tau e = \ast_H \); and let \( q' \in V_v \Gamma_K \) and \( e' \in E_v \Gamma_K \) with \( te' = q', \tau e' = \ast_K \), and enlarge our picture by drawing a new \( \nu \)-secondary vertex, say \( q \), and a new \( \nu \)-arc, say \( e = (e,e') \), from \( q \) to \( \ast \). According to (9) — and with respect to the choice \((p_q,p_q') = (\ast_H,\ast_K)\) — we know that the label of \( q \) is \( \ell_q = \ell_q^{t_e} \cap \ell_q^{t_{e'}} \leq G_v \).

Applying SIP for \( G_v \) to the (finitely generated) subgroups \( \ell_q^{t_e} \) and \( \ell_q^{t_{e'}} \), we can decide whether \( \ell_q \) is finitely generated or not. In case it is not, kill the whole process and declare \( H \cap K \) to be non finitely generated. Otherwise, compute a finite set of generators for \( \ell_q \), assign \( \ell_e = 1 \), and check which other primary vertices from \( \tilde{\Gamma} \) are adjacent to \( q \): \( p = (p,p') \in V_0 \tilde{\Gamma} \) is adjacent to \( q \) if and only if \( (p,p') \equiv_v (p_q,p_q') \), which happens if and only if there exists \( f \in E_v \Gamma_H \) from \( q \) to \( p \), and \( f' \in E_v \Gamma_K \) from \( q' \) to \( p' \), such that \( \ell_e^{-1} \ell_q \ell_f \cap \ell_{e'}^{-1} \ell_q \ell_{f'} = \emptyset \). So, run over every \( p \in V_0 \Gamma_H \) adjacent to \( q \), and every \( p' \in V_0 \Gamma_K \) adjacent to \( q' \) and, for each such pair, check whether the intersection of (right) cosets

\[
\ell_q^{\ell_{t_e}} \cdot (\ell_e^{-1} \ell_f) \cap \ell_q^{\ell_{t_{e'}}} \cdot (\ell_{e'}^{-1} \ell_{f'})
\]

is empty or not; this can be done using the above call to ESIP from \( G_v \), since they are right cosets of \( \ell_q^{\ell_{t_e}}, \ell_q^{\ell_{t_{e'}}} \leq G_v \), whose intersection happens to be finitely generated.

In case this intersection is not empty, add a \( \nu \)-arc, say \( f = (f,f') \), from \( q \) to \( p \), and \( \ell_{(f,f')} \) arbitrarily chosen from that nonempty intersection. After this procedure, we have a complete picture of the 1-elementary and 2-elementary walks in \( \Gamma_H \cap \Gamma_K \) starting at the basepoint \( \ast \).

Now, for every \( \nu = 1,2 \), and every primary vertex \( p = (p,p') \) added to the picture and not yet explored, repeat the same process (with \( p \) in place of \( \ast \)). Since the underlying graph of \( \tilde{\Gamma} \) is finite, this procedure will either find a non finitely generated vertex label, or will finish the complete construction of \( \Gamma_H \cap \Gamma_K \) in finite time. In the first case, we deduce the non finitely generated type of \( H \cap K \); in the second case, we can compute generators for \( H \cap K \) (in fact, a Kurosh decomposition) applying Theorem 4.13.

Hence, so far, we have solved SIP(\( G_1 \ast G_2 \)). To finish the proof, let us place ourselves in the case where \( H \cap K \) is finitely generated (and so, with the junction automaton \( \Gamma_H \cap \Gamma_K \) fully constructed), and let us decide whether the intersection of right cosets \( H_u \cap K_v \) is empty or not. We can extend the computation of \( \Gamma_H \cap \Gamma_K \) to that of \( \Gamma_{H_u} \cap \Gamma_{K_v} \); or alternatively construct directly \( \Gamma_{H_u} \cap \Gamma_{K_v} \) from the beginning.

It only remains to check whether the vertex \((\tau u, \tau v)\) appears in \( \Gamma_{H_u} \cap \Gamma_{K_v} \), or not. Namely, using Lemma 4.23 (ii): \( (\tau u, \tau v) \) is connected to \((\ast_H, \ast_K)\) if and only if the intersection \( H_u \cap K_v \) is nonempty; and, if so, any element \( g \) from the label of any walk from \((\ast_H, \ast_K)\) to \((\tau u, \tau v)\) belongs to such intersection, \( g \in H_u \cap K_v = (H \cap K)g \). This concludes the proof. \( \square \)

Finally, we complement the arguments in the last proof to prove that TIP also passes through free products.

Proof of Theorem 1.9. Since solvability of TIP implies that of ESIP, Theorem 1.8 already gives us ESIP for \( G_1 \ast G_2 \). It remains to solve CIP for \( G_1 \ast G_2 \) in the case where the given finitely generated subgroups \( H, K \) have a non finitely generated intersection.
Given $H, K \leq G_1 \ast G_2$ finitely generated, and $u, v \in G_1 \ast G_2$, run the same algorithm as in the proof of Theorem 1.8: construct $\Gamma_{Hu}$ and $\Gamma_{Kv}$ and start building the junction $\Gamma_{Hu} \wedge \Gamma_{Kv}$; when we encounter a secondary vertex $q$ whose label $\ell_q = \ell_q^e \cap \ell_q^{e'} \leq G_\nu$ is not finitely generated, instead of computing a set of generators for it (which is not possible), we just put the trivial subgroup as a label in place of $\ell_q$. Then, when analyzing which other primary vertices are adjacent to $q$, we need to decide if the intersection of cosets from equation (11) are empty or not: even though $\ell_q^e \cap \ell_q^{e'} \leq G_\nu$ is not finitely generated, the decision can be made effective using CIP from $G_\nu$. This way, we can algorithmically complete the description of $\Gamma_{Hu} \wedge \Gamma_{Kv}$ except that, for some secondary vertices $q$, instead of having generators for $\ell_q$, we just have the trivial element labelling them.

Of course, this is not enough information for computing a set of generators for $H \cap K$. But it suffices for deciding whether the vertices $\langle s_H, s_K \rangle$ and $\langle \tau_w, \tau_{w'} \rangle$ belong to the same connected component of $\Gamma_{Hu} \wedge \Gamma_{Kv}$. By Lemma 4.23, this allows us to decide whether the intersection of cosets $Hu \cap Kv$ is empty or not; and in case it is not, we can compute an element from it, just choosing a walk $\gamma$ from $\langle s_H, s_K \rangle$ to $\langle \tau_w, \tau_{w'} \rangle$, and then picking an element from $\ell_\gamma$ (if $\gamma$ traverses some secondary vertex with a non finitely generated label, we just recorded the trivial element from it for this purpose). This completes the proof. □

Acknowledgments

The first two named authors acknowledge financial support from the Spanish Agencia Estatal de Investigación, through grant MTM2017-82740-P (AEI/FEDER, UE), and also the “María de Maeztu” Programme for Units of Excellence in R&D (MDM-2014-0445). The third named author acknowledges support by CMUP (UID/MAT/00144/2013), which is funded by FCT (Portugal) with national (MEC) and European structural funds (FEDER), under the partnership agreement PT2020; he was also partially supported by the ERC Grant 336983, by the Basque Government grant IT974-16, by the grant MTM2014-53810-C2-2-P of the Ministerio de Economía y Competitividad of Spain, and by the Russian Foundation for Basic Research (project no. 15-01-05823).

References

[1] I. J. Aalbersberg and H. J. Hoogeboom. “Characterizations of the decidability of some problems for regular trace languages”. Mathematical systems theory 22.1 (Dec. 1989), pp. 1–19 (cit. on p. 4).
[2] A. Baudisch. “Subgroups of semifree groups”. Acta Mathematica Academiae Scientiarum Hungaricae 38.1 (Mar. 1981), pp. 19–28 (cit. on p. 6).
[3] B. Baumslag. “Intersections of Finitely Generated Subgroups in Free Products”. Journal of the London Mathematical Society s1-41 (Jan. 1, 1966), pp. 673–679 (cit. on pp. 13, 24, 29).
[4] A. Brandstädt, V. B. Le, and J. P. Spinrad. Graph Classes: A Survey. n edition. Philadelphia: Society for Industrial and Applied Mathematics, Jan. 1, 1987. 316 pp. (cit. on p. 3).
[5] R. Charney. “An introduction to right-angled Artin groups”. Geometriae Dedicata 125.1 (2007), pp. 141–158 (cit. on p. 6).

[6] J. Delgado Rodríguez. “Extensions of free groups: algebraic, geometric, and algorithmic aspects”. PhD thesis. Universitat Politècnica de Catalunya, 2017 (cit. on pp. 4, 21).

[7] J. Delgado. “Some characterizations of Howson PC-groups”. Reports@SCM 1.1 (Jan. 10, 2014), pp. 33–38 (cit. on p. 6).

[8] J. Delgado and E. Ventura. “Algorithmic problems for free-abelian times free groups”. Journal of Algebra 391 (Oct. 1, 2013), pp. 256–283 (cit. on pp. 2, 4).

[9] C. Droms. “Graph groups, coherence, and three-manifolds”. Journal of Algebra 106.2 (1987), pp. 484–489 (cit. on p. 7).

[10] C. Droms. “Isomorphisms of graph groups”. Proceedings of the American Mathematical Society 100.3 (Mar. 1987), pp. 407–407 (cit. on p. 6).

[11] C. Droms. “Subgroups of graph groups”. Journal of Algebra 110.2 (Oct. 15, 1987), pp. 519–522 (cit. on p. 3).

[12] E. S. Esyp, I. V. Kazachkov, and V. N. Remeslennikov. “Divisibility Theory and Complexity of Algorithms in Free Partially Commutative Groups”. arXiv:math/0512401 (Dec. 16, 2005) (cit. on p. 6).

[13] E. R. Green. “Graph products of groups”. PhD thesis. 1990 (cit. on p. 6).

[14] S. Hermiller and J. Meier. “Algorithms and Geometry for Graph Products of Groups”. Journal of Algebra 171.1 (1995), pp. 230–257 (cit. on p. 6).

[15] A. G. Howson. “On the Intersection of Finitely Generated Free Groups”. Journal of the London Mathematical Society s1-29.4 (Oct. 1, 1954), pp. 428–434 (cit. on p. 1).

[16] S. V. Ivanov. “On the intersection of finitely generated subgroups in free products of groups”. International Journal of Algebra and Computation 09.5 (Oct. 1, 1999), pp. 521–528 (cit. on pp. 4, 16, 17, 20–22, 24).

[17] S. V. Ivanov. “Intersecting free subgroups in free products of groups”. International Journal of Algebra and Computation 11.3 (June 1, 2001), pp. 281–290 (cit. on pp. 4, 16).

[18] S. V. Ivanov. “On the Kurosh rank of the intersection of subgroups in free products of groups”. Advances in Mathematics 218.2 (2008), pp. 465–484 (cit. on pp. 16, 21).

[19] Y. Jing-Ho, C. Jer-Jeong, and G. J. Chang. “Quasi-threshold graphs”. Discrete Applied Mathematics 69.3 (1996), pp. 247–255 (cit. on p. 3).

[20] I. Kapovich, R. Weidmann, and A. Myasnikov. “Foldings, graphs of groups and the membership problem”. International Journal of Algebra and Computation 15.1 (Feb. 2005), pp. 95–128 (cit. on pp. 7, 16, 20–22).

[21] T. Koberda. “Right angled Artin groups and their subgroups”. Lecture notes Yale University (Mar. 7, 2013), pp. 1–50 (cit. on p. 6).

[22] H.-N. Liu, C. Wrathall, and K. Zeger. “Efficient solution of some problems in free partially commutative monoids”. Information and Computation 89.2 (Dec. 1990), pp. 180–198 (cit. on p. 6).
[23] M. Lohrey and B. Steinberg. “The submonoid and rational subset membership problems for graph groups”. Journal of Algebra. Computational Algebra 320.2 (July 15, 2008), pp. 728–755 (cit. on p. 7).

[24] K. A. Mikhailova. “The occurrence problem for direct products of groups”. Doklady Akademii Nauk SSSR 119 (1958), pp. 1103–1105 (cit. on p. 5).

[25] K. A. Mikhailova. “The occurrence problem for free products of groups”. Mathematics of the USSR-Sbornik 4.2 (Feb. 28, 1968), pp. 181–190 (cit. on p. 5).

[26] E. Rodaro, P. V. Silva, and M. Sykiotis. “Fixed points of endomorphisms of graph groups”. Journal of Group Theory 16.4 (Jan. 5, 2013), pp. 573–583 (cit. on p. 3).

[27] J. R. Stallings. “Topology of finite graphs”. Inventiones Mathematicae 71 (Mar. 1983), pp. 551–565 (cit. on p. 16).

[28] L. Van Wyk. “Graph groups are biautomatic”. Journal of Pure and Applied Algebra 94.3 (July 8, 1994), pp. 341–352 (cit. on p. 6).

[29] E. S. Wolk. “The Comparability Graph of a Tree”. Proceedings of the American Mathematical Society 13.5 (Oct. 1, 1962), pp. 789–795 (cit. on p. 3).

[30] C. Wrathall. “The word problem for free partially commutative groups”. J. Symb. Comput. 6.1 (Aug. 1988), pp. 99–104 (cit. on p. 6).

[31] C. Wrathall. Free partially commutative groups. 1989 (cit. on p. 6).