On the isomorphism problem for generalized Baumslag-Solitar groups: invariants and flexible configurations

Dario Ascari

Department of Mathematics, University of the Basque Country,

Barrio Sarriena, Leioa, 48940, Spain

e-mail: ascari.maths@gmail.com

Montserrat Casals-Ruiz

Ikerbasque - Basque Foundation for Science and Matematika Saila,

UPV/EHU, Sarriena s/n, 48940, Leioa - Bizkaia, Spain

e-mail: montsecasals@gmail.com

Ilya Kazachkov

Ikerbasque - Basque Foundation for Science and Matematika Saila,

UPV/EHU, Sarriena s/n, 48940, Leioa - Bizkaia, Spain

e-mail: ilya.kazachkov@gmail.com

1 Introduction

As the name suggests, GBSs are a generalization of the classical Baumslag-Solitar groups, which are traditionally a significant source of counterexamples in geometric group theory (representing a key obstruction to hyperbolicity). P. Kropholler [Kro90] gives a characterization of GBSs as the only groups of cohomological dimension 2 containing an infinite cyclic subgroup that intersects all of its conjugates. We also point out that they have been classified up to quasi-isometry [FM98, Why01], and we refer the reader to [McC91] for a comparison of some classes of GBSs with one-relator groups.

Theorem A (Theorem 4.9)**.** *Let $(\Gamma, \psi), (\Gamma', \psi')$ be two GBS graphs with a bijection $V(\Gamma) = V(\Gamma')$. Then the following are equivalent:*

1. *There is a sequence of slides, swaps, connections going from (Γ, ψ) to (Γ', ψ') inducing the given bijection.*
2. *For every quasi-conjugacy class Q, there is a sequence of swaps, connections, internal slides, external slides (Lemma 4.8) - each of them involving only edges in Q - going from the configuration in (Γ, ψ) to the configuration in (Γ', ψ').*

Theorem B (Corollary 5.27)**.** *There is an algorithm that, given two GBS graph $(\Gamma, \psi), (\Delta, \phi)$ with one qc-class and full-support gaps, decides whether the corresponding GBS groups are isomorphic or not.*

2 GBS graphs and quasi-conjugacy classes

In this section we set up the notation for the rest of the paper. We review the notions of conjugacy and quasi-conjugacy, and we show how they can be described by means of a finite amount of data. We provide explicit algorithms for dealing with them, and examples to illustrate these notions.

2.1 Graphs of groups

We consider graphs as combinatorial objects, following the notation of [Ser77]. A **graph** is a quadruple $\Gamma = (V, E, \bar{\cdot}, \iota)$ consisting of a set $V = V(\Gamma)$ of *vertices*, a set $E = E(\Gamma)$ of *edges*, a map $\bar{\cdot} : E \to E$ called *reverse* and a map $\iota : E \to V$ called *initial endpoint*; we require that, for every edge $e \in E$, we have $\bar{e} \neq e$ and $\bar{\bar{e}} = e$. For an edge $e \in E$, we denote with $\tau(e) = \iota(\bar{e})$ the *terminal endpoint* of e. A **path** in a graph Γ, with *initial endpoint* $v \in V(\Gamma)$ and *terminal endpoint* $v' \in V(\Gamma)$, is a sequence $\sigma = (e_1, ..., e_\ell)$ of edges $e_1, ..., e_\ell \in E(\Gamma)$ for some integer $\ell \geq 0$, with the conditions $\iota(e_1) = v$ and $\tau(e_\ell) = v'$ and $\tau(e_i) = \iota(e_{i+1})$ for $i = 1, ..., \ell - 1$. A graph is **connected** if for every couple of vertices, there is a path going from one to the other. For a connected graph Γ, we define its **rank** $\operatorname{rk}\Gamma \in \mathbb{N} \cup \{+\infty\}$ as the rank of its fundamental group (which is a free group).

Definition 2.1. *A **graph of groups** is a quadruple*

$$\mathcal{G} = (\Gamma, \{G_v\}_{v \in V(\Gamma)}, \{G_e\}_{e \in E(\Gamma)}, \{\psi_e\}_{e \in E(\Gamma)})$$

consisting of a connected graph Γ, a group G_v for each vertex $v \in V(\Gamma)$, a group G_e for every edge $e \in E(\Gamma)$ with the condition $G_e = G_{\bar{e}}$, and an injective homomorphism $\psi_e : G_e \to G_{\tau(e)}$ for every edge $e \in E(\Gamma)$.

Let $\mathcal{G} = (\Gamma, \{G_v\}_{v \in V(\Gamma)}, \{G_e\}_{e \in E(\Gamma)}, \{\psi_e\}_{e \in E(\Gamma)})$ be a graph of groups. Define the **universal group** $\mathrm{FG}(\mathcal{G})$ as the quotient of the free product $(*_{v \in V(\Gamma)} G_v) * F(E(\Gamma))$ by the relations

$$\bar{e} = e^{-1} \qquad\qquad \psi_{\bar{e}}(g) \cdot e = e \cdot \psi_e(g)$$

for $e \in E(\Gamma)$ and $g \in G_e$.

Define the **fundamental group** $\pi_1(\mathcal{G}, \widetilde{v})$ of a graph of group $\mathcal{G}$ with basepoint $\widetilde{v} \in V(\Gamma)$ to be the subgroup of $\mathrm{FG}(\mathcal{G})$ of the elements that can be represented by words such that, when going along the word, we read a path in the graph Γ from $\widetilde{v}$ to $\widetilde{v}$. The fundamental group $\pi_1(\mathcal{G}, \widetilde{v})$ doesn't depend on the chosen basepoint $\widetilde{v}$, up to isomorphism. Given an element g inside a vertex group G_v, we can take a path $(e_1, ..., e_\ell)$ from $\widetilde{v}$ to v: we define the conjugacy class $[g] := [e_1 ... e_\ell g \bar{e}_\ell ... \bar{e}_1] \in \pi_1(\mathcal{G}, \widetilde{v})$, and notice that this doesn't depend on the chosen path.

2.2 Generalized Baumslag-Solitar groups

Definition 2.2. *A **GBS graph of groups** is a finite graph of groups*

$$\mathcal{G} = (\Gamma, \{G_v\}_{v \in V(\Gamma)}, \{G_e\}_{e \in E(\Gamma)}, \{\psi_e\}_{e \in E(\Gamma)})$$

such that each vertex group and each edge group is $\mathbb{Z}$.

A **Generalized Baumslag-Solitar group** is a group G isomorphic to the fundamental group of some GBS graph of groups.

Definition 2.3. *A **GBS graph** is a couple (Γ, ψ) where Γ is a finite graph and $\psi : E(\Gamma) \to \mathbb{Z} \setminus \{0\}$ is a function.*

Given a GBS graph of groups $\mathcal{G} = (\Gamma, \{G_v\}_{v \in V(\Gamma)}, \{G_e\}_{e \in E(\Gamma)}, \{\psi_e\}_{e \in E(\Gamma)})$, the map $\psi_e : G_e \to G_{\tau(e)}$ is an injective homomorphism $\psi_e : \mathbb{Z} \to \mathbb{Z}$, and thus coincides with multiplication by a unique non-zero integer $\psi(e) \in \mathbb{Z} \setminus \{0\}$. We define the GBS graph associated to $\mathcal{G}$ as (Γ, ψ) associating to each edge e the factor $\psi(e)$ characterizing the homomorphism ψ_e, see Figure 1. Giving a GBS graph of groups is equivalent to giving the corresponding GBS graph. In fact, the numbers on the edges are enough to reconstruct the injective homomorphisms and thus the graph of groups.

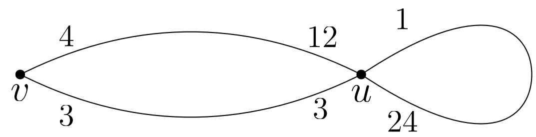

Figure 1: In the figure we can see a GBS graph (Γ, ψ) with two vertices v, u and three edges e_1, e_2, e_3 (and their reverses). The edge e_1 goes from v to u and has $\psi(\overline{e}_1) = 4$ and $\psi(e_1) = 12$. The edge e_2 goes from v to u and has $\psi(\overline{e}_2) = \psi(\overline{e}_2) = 3$. The edge e_3 goes from u to u and has $\psi(\overline{e}_3) = 1$ and $\psi(e_3) = 24$.

Let $\mathcal{G}$ be a GBS graph of groups and let (Γ, ψ) be the corresponding GBS graph. The universal group $\mathrm{FG}(\mathcal{G})$ has a presentation with generators $V(\Gamma) \cup E(\Gamma)$, the generator $v \in V(\Gamma)$ representing the element 1 in $\mathbb{Z} = G_v$. The relations are given by $\overline{e} = e^{-1}$ and $u^{\psi(\overline{e})} e = e v^{\psi(e)}$ for every edge $e \in E(\Gamma)$ with $\iota(e) = u$ and $\tau(e) = v$.

2.3 Reduced affine representation of a GBS graph

Definition 2.4. *For a GBS graph* (Γ, ψ)*, define its* ***set of primes***

$$\mathcal{P}(\Gamma, \psi) := \{ r \in \mathbb{N} \text{ prime } : r \mid \psi(e) \text{ for some } e \in E(\Gamma) \}$$

Given a GBS graph (Γ, ψ), consider the finitely generated abelian group

$$\mathbf{A} := \mathbb{Z}/2\mathbb{Z} \oplus \bigoplus_{r \in \mathcal{P}(\Gamma, \psi)} \mathbb{Z}.$$

We denote with $\mathbf{0} \in \mathbf{A}$ the neutral element. For an element $\mathbf{a} = (a_0, a_r : r \in \mathcal{P}(\Gamma, \psi)) \in \mathbf{A}$ (with $a_0 \in \mathbb{Z}/2\mathbb{Z}$ and $a_r \in \mathbb{Z}$ for $r \in \mathcal{P}(\Gamma, \psi)$), we denote $\mathbf{a} \geq \mathbf{0}$ if $a_r \geq 0$ for all $r \in \mathcal{P}(\Gamma, \psi)$; notice that we aren't requiring any condition on a_0. We define the positive cone $\mathbf{A}^+ := \{ \mathbf{a} \in \mathbf{A} : \mathbf{a} \geq \mathbf{0} \}$.

Definition 2.5. *Let* (Γ, ψ) *be a GBS graph. Define its* ***(reduced) affine representation*** *to be the graph* $\Lambda = \Lambda(\Gamma, \psi)$ *given by:*

1. $V(\Lambda) = V(\Gamma) \times \mathbf{A}^+$ *is the disjoint union of copies of* $\mathbf{A}^+$*, one for each vertex of* Γ.
2. $E(\Lambda) = E(\Gamma)$ *is the same set of edges as* Γ*, and with the same reverse map.*
3. *For an edge* $e \in E(\Lambda)$ *we write the unique factorization* $\psi(e) = (-1)^{a_0} \prod_{r \in \mathcal{P}(\Gamma, \psi)} r^{a_r}$ *and we define the terminal endpoint* $\tau_\Lambda(e) = (\tau_\Gamma(e), (a_0, a_r, ...))$*, see Figure 2.*

For a vertex $v \in V(\Gamma)$ *we denote* $\mathbf{A}_v^+ := \{v\} \times \mathbf{A}^+$ *the corresponding copy of* $\mathbf{A}^+$.

If Λ contains an edge going from p to q, then we denote $p \text{ --- } q$. If Λ contains edges from p_i to q_i for $i = 1, ..., m$, then we denote

$$\begin{cases} p_1 \text{ --- } q_1 \\ \dots \\ p_m \text{ --- } q_m \end{cases}$$

This doesn't mean that $p_1 \text{ --- } q_1, ..., p_m \text{ --- } q_m$ are all the edges of Λ, but only that in a certain situation we are focusing on those edges. If we are focusing on a specific copy $\mathbf{A}_v^+$ of $\mathbf{A}^+$, for some $v \in V(\Gamma)$, and we have edges $(v, \mathbf{a}_i) \text{ --- } (v, \mathbf{b}_i)$ for $i = 1, ..., m$, then we say that $\mathbf{A}_v^+$ contains edges

$$\begin{cases} \mathbf{a}_1 \text{ --- } \mathbf{b}_1 \\ \dots \\ \mathbf{a}_m \text{ --- } \mathbf{b}_m \end{cases}$$

omitting the vertex v.

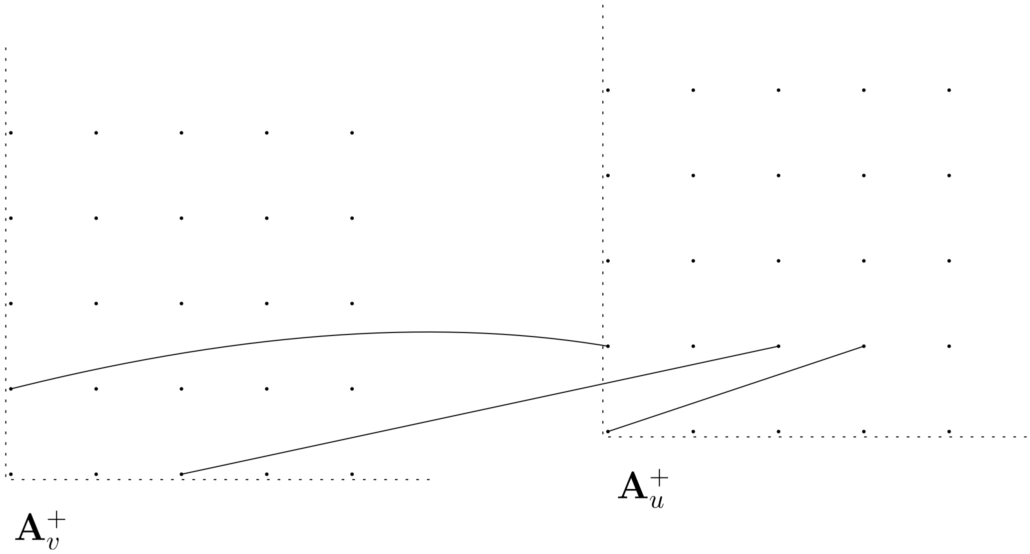

Figure 2: The affine representation Λ of the GBS graph (Γ, ψ) of Figure 1. The set of vertices consists of two copies $\mathbf{A}_v^+$ and $\mathbf{A}_u^+$ of the positive affine cone $\mathbf{A}^+$, associated to the two vertices v and u respectively. The edge e_1 going from v to u was labeled with $(\psi(\overline{e}_1), \psi(e_1)) = (4, 12) = (2^2 3^0, 2^2 3^1)$, and thus now it goes from the point $(2, 0)$ in $\mathbf{A}_v^+$ to the point $(2, 1)$ in $\mathbf{A}_u^+$. Similarly for e_2 and e_3.

2.4 Support and control of vectors

The following notions for elements of $\mathbf{A}$ will be widely used along the paper.

Definition 2.6. *For $\mathbf{x} \in \mathbf{A}$ define its **support** as the set*

$$\text{supp}(\mathbf{x}) := \{r \in \mathcal{P}(\Gamma, \psi) : x_r \neq 0\}$$

Remark 2.7. Note that we omit the $\mathbb{Z}/2\mathbb{Z}$ component from the definition of support.

Definition 2.8. *Let $\mathbf{a}, \mathbf{b}, \mathbf{w} \in \mathbf{A}^+$. We say that $\mathbf{a}, \mathbf{w}$ **controls** $\mathbf{b}$ if any of the following equivalent conditions holds:*

1. *We have $\mathbf{a} \leq \mathbf{b} \leq \mathbf{a} + k\mathbf{w}$ for some $k \in \mathbb{N}$.*
2. *We have $\mathbf{b} - \mathbf{a} \geq \mathbf{0}$ and $\text{supp}(\mathbf{b} - \mathbf{a}) \subseteq \text{supp}(\mathbf{w})$.*

2.5 Affine paths and conjugacy classes

Let (Γ, ψ) be a GBS graph and let Λ be its affine representation. Given a vertex $p = (v, \mathbf{a}) \in V(\Lambda)$ and an element $\mathbf{w} \in \mathbf{A}^+$, we define the vertex $p + \mathbf{w} := (v, \mathbf{a} + \mathbf{w}) \in V(\Lambda)$. For two vertices $p, p' \in V(\Lambda)$ we denote $p' \geq p$ if $p' = p + \mathbf{w}$ for some $\mathbf{w} \in \mathbf{A}^+$; in particular this implies that both p, p' belong to the same $\mathbf{A}_v^+$ for some $v \in V(\Gamma)$.

Definition 2.9. *An **affine path** in Λ, with initial endpoint $p \in V(\Lambda)$ and terminal endpoint $p' \in V(\Lambda)$, is a sequence $(e_1, ..., e_\ell)$ of edges $e_1, ..., e_\ell \in E(\Lambda)$ for some $\ell \geq 0$, such that there exist $\mathbf{w}_1, ..., \mathbf{w}_\ell \in \mathbf{A}^+$ satisfying the conditions $\iota(e_1) + \mathbf{w}_1 = p$ and $\tau(e_\ell) + \mathbf{w}_\ell = p'$ and $\tau(e_i) + \mathbf{w}_i = \iota(e_{i+1}) + \mathbf{w}_{i+1}$ for $i = 1, ..., \ell - 1$.*

The elements $\mathbf{w}_1, ..., \mathbf{w}_\ell$ are called *translation coefficients* of the path; if they exist, then they are uniquely determined by the path and by the endpoints, and they can be computed algorithmically. They mean that an edge $e \in E(\Lambda)$ connecting p to q allows us also to travel from $p + \mathbf{w}$ to $q + \mathbf{w}$ for every $\mathbf{w} \in \mathbf{A}^+$.

Definition 2.10. *Let $p, q \in V(\Lambda)$.*

1. *We denote $p \sim_{\mathrm{c}} q$, and we say that p, q are* ***conjugate****, if there is an affine path going from p to q.*

2. *We denote $p \preceq_{\mathrm{c}} q$ if $p \leq q'$ for some $q' \sim_{\mathrm{c}} q$.*

3. *We denote $p \sim_{\mathrm{qc}} q$, and we say that p, q are* ***quasi-conjugate****, if $p \preceq_{\mathrm{c}} q$ and $q \preceq_{\mathrm{c}} p$.*

The relation $\sim_{\mathrm{c}}$ is an equivalence relations on the set $V(\Lambda)$. The relation $\preceq_{\mathrm{c}}$ is a pre-order on $V(\Lambda)$, and $\sim_{\mathrm{qc}}$ is the equivalence relation induced by the pre-order. Note that if $p \sim_{\mathrm{c}} p'$ then $p + \mathbf{w} \sim_{\mathrm{c}} p' + \mathbf{w}$ for all $\mathbf{w} \in \mathbf{A}^+$. Similarly, if $p \sim_{\mathrm{qc}} p'$ then $p + \mathbf{w} \sim_{\mathrm{qc}} p' + \mathbf{w}$ for all $\mathbf{w} \in \mathbf{A}^+$.

2.6 Minimal points, quasi-conjugacy support, linear algebra

We now show how conjugacy classes and quasi-conjugacy classes can be completely described by means of three pieces of data: the *minimal points*, the *quasi-conjugacy support* and the *linear algebra*.

Definition 2.11. *Let (Γ, ψ) be a GBS graph and let p be a vertex of its affine representation Λ.*

1. *Define the* ***minimal points***

$$\min_{\mathrm{qc}}(p) = \{q \in V(\Lambda) : q \sim_{\mathrm{qc}} p \text{ and } q \preceq_{\mathrm{c}} q' \text{ for all } q' \sim_{\mathrm{qc}} p\}$$

2. *Define the* ***quasi-conjugacy support***

$$\mathrm{supp}_{\mathrm{qc}}(p) = \bigcup \{\mathrm{supp}(\mathbf{u}) : \mathbf{u} \in \mathbf{A}^+ \text{ and } p + \mathbf{u} \sim_{\mathrm{qc}} p\} \subseteq \mathcal{P}(\Gamma, \psi).$$

3. *Define the* ***linear algebra***

$$\mathrm{lin}_{\mathrm{qc}}(p) = \{\mathbf{r} \in \mathbf{A} : \mathbf{r} = \mathbf{w} - \mathbf{w}' \text{ for some } \mathbf{w}, \mathbf{w}' \in \mathbf{A}^+ \text{ such that } p + \mathbf{w} \sim_{\mathrm{c}} p + \mathbf{w}' \sim_{\mathrm{c}} p\}.$$

Lemma 2.12. *The set $\min_{\mathrm{qc}}(p)$ is finite.*

Proof. Fix a vertex $u \in V(\Gamma)$: since Γ has finitely many vertices, it's enough to prove that $\min_{\mathrm{qc}}(p) \cap \mathbf{A}_u^+$ is finite. Consider a set of variables $\{x_i\}_{i \in \mathcal{P}(\Gamma,\psi)}$ and consider the ring of polynomials $\mathbb{C}[x_i : i \in \mathcal{P}(\Gamma, \psi)]$. For each $(u, \mathbf{q}) \sim_{\mathrm{qc}} p$ consider the monomial $x^{\mathbf{q}} = \prod x_i^{q_i} \in \mathbb{C}[x_i, \dots]$. Consider the ideal $I = (x^{\mathbf{q}} : (u, \mathbf{q}) \sim_{\mathrm{qc}} p) \subseteq \mathbb{C}[x_i, \dots]$ generated by these monomials. We have that I is a monomial ideal, i.e. it contains a polynomial P if and only if it contains all the monomial with non-zero coefficients of P. By Hilbert's basis theorem, I must be finitely generated, and thus we can find we find finitely many $\mathbf{q}_1, \dots \mathbf{q}_\ell \in \mathbf{A}^+$ such that $I = (x^{\mathbf{q}_1}, \dots, x^{\mathbf{q}_\ell})$. It follows that $\min_{\mathrm{qc}}(p) \cap \mathbf{A}_u^+ \subseteq \{(u, \mathbf{q}_1), \dots, (u, \mathbf{q}_\ell)\}$ must be finite, as desired. □

Lemma 2.13. *If $p \sim_{\mathrm{qc}} q$ then $\mathrm{supp}_{\mathrm{qc}}(p) = \mathrm{supp}_{\mathrm{qc}}(q)$.*

Proof. If $p + \mathbf{u} \sim_{\mathrm{qc}} p$ for some $\mathbf{u} \in \mathbf{A}^+$, then $q + \mathbf{u} \sim_{\mathrm{qc}} p + \mathbf{u} \sim_{\mathrm{qc}} p \sim_{\mathrm{qc}} q$. □

Lemma 2.14. *For all $p \in V(\Lambda)$ there is $\mathbf{w} \in \mathbf{A}^+$ such that $p + \mathbf{w} \sim_{\mathrm{c}} p$ and $\mathrm{supp}(\mathbf{w}) = \mathrm{supp}_{\mathrm{qc}}(p)$.*

Proof. Realize the quasi-conjugacy support as a finite union $\mathrm{supp}_{\mathrm{qc}}(p) = \mathrm{supp}(\mathbf{u}_1) \cup \dots \cup \mathrm{supp}(\mathbf{u}_\ell)$ for $\mathbf{u}_i \in \mathbf{A}^+$ with $p + \mathbf{u}_i \sim_{\mathrm{qc}} p$. By definition of quasi-conjugacy, we can find $\mathbf{v}_i \in \mathbf{A}^+$ such that $p + \mathbf{u}_i + \mathbf{v}_i \sim_{\mathrm{c}} p$. Now set $\mathbf{w} = (\mathbf{u}_1 + \mathbf{v}_1) + \dots + (\mathbf{u}_\ell + \mathbf{v}_\ell)$ and the conclusion follows. □

Lemma 2.15. *We have that $\mathrm{lin}_{\mathrm{qc}}(p)$ is a subgroup of $\mathbf{A}$.*

Proof. Suppose that we are given $\mathbf{r}, \mathbf{s} \in \mathrm{lin}_{\mathrm{qc}}(p)$. Thus we have $\mathbf{r} = \mathbf{w} - \mathbf{w}'$ and $\mathbf{s} = \mathbf{u} - \mathbf{u}'$ for $\mathbf{w}, \mathbf{w}', \mathbf{u}, \mathbf{u}' \in \mathbf{A}^+$ with $p \sim_{\mathrm{c}} p + \mathbf{w} \sim_{\mathrm{c}} p + \mathbf{w}' \sim_{\mathrm{c}} p + \mathbf{u} \sim_{\mathrm{c}} p + \mathbf{u}'$. In particular we have that $p + \mathbf{w} + \mathbf{u} \sim_{\mathrm{c}} p + \mathbf{w}' + \mathbf{u} \sim_{\mathrm{c}} p + \mathbf{w}' + \mathbf{u}'$ and $(\mathbf{w} + \mathbf{u}) - (\mathbf{w}' + \mathbf{u}') = \mathbf{r} + \mathbf{s}$. Thus $\mathbf{r} + \mathbf{s} \in \mathrm{lin}_{\mathrm{qc}}(p)$. The conclusion follows. □

Lemma 2.16 (Equivalent characterization of linear algebra)**.** *For $p \in V(\Lambda)$ and $\mathbf{r} \in \mathbf{A}$, we have that $\mathbf{r} \in \mathrm{lin}_{\mathrm{qc}}(p)$ if and only it satisfies the following two conditions:*

1. *Λ contains vertices $(u, \mathbf{a}) \sim_{\mathrm{c}} (u, \mathbf{a}+\mathbf{r}) \sim_{\mathrm{qc}} p$ for some $u \in V(\Gamma)$ and $\mathbf{a} \in \mathbf{A}$ with $\mathbf{a}, \mathbf{a}+\mathbf{r} \geq \mathbf{0}$.*
2. $\mathrm{supp}(\mathbf{r}) \subseteq \mathrm{supp}_{\mathrm{qc}}(p)$

Proof. The definition of $\mathrm{lin}_{\mathrm{qc}}(p)$ obviously implies the two conditions. Suppose now that $\mathrm{supp}(\mathbf{r}) \subseteq \mathrm{supp}_{\mathrm{qc}}(p)$ and there are vertices $(u, \mathbf{a}) \sim_{\mathrm{c}} (u, \mathbf{a}+\mathbf{r}) \sim_{\mathrm{qc}} p$. Since $\mathrm{supp}(\mathbf{r}) \subseteq \mathrm{supp}_{\mathrm{qc}}(p)$, we can find $\mathbf{w} \in \mathbf{A}^+$ such that $p + \mathbf{w} \sim_{\mathrm{c}} p$ and $\mathbf{w} + \mathbf{r} \geq \mathbf{0}$. Since $p \sim_{\mathrm{qc}} (u, \mathbf{a})$ we can find $p' \sim_{\mathrm{c}} p$ with $(u, \mathbf{a}) \leq p'$; in particular we also have $p + \mathbf{w} \sim_{\mathrm{c}} p' + \mathbf{w}$ and $p + (\mathbf{w} - \mathbf{r}) \sim_{\mathrm{c}} p' + (\mathbf{w} - \mathbf{r})$. But since $(u, \mathbf{a}) \leq p' + \mathbf{w}$, the condition $(u, \mathbf{a}) \sim_{\mathrm{c}} (u, \mathbf{a}+\mathbf{r})$ implies $p' + \mathbf{w} \sim_{\mathrm{c}} p + \mathbf{w} + \mathbf{r}$. Therefore we have $p + \mathbf{w} \sim_{\mathrm{c}} p' + \mathbf{w} \sim_{\mathrm{c}} p' + (\mathbf{w} - \mathbf{r}) \sim_{\mathrm{c}} p + (\mathbf{w} - \mathbf{r})$. The conclusion follows. □

Remark 2.17. Condition 2 of Lemma 2.16 can't be waived. In fact, one might try to define the linear algebra of a point p as the set of elements $\mathbf{r} \in \mathbf{A}$ satisfying Condition 1 of Lemma 2.16. However, in general this definition gives a set which is different from the set $\mathrm{lin}_{\mathrm{qc}}(p)$ of Definition 2.11, and which might not be a subgroup.

It follows that if $p \sim_{\mathrm{qc}} q$ then $\mathrm{min}_{\mathrm{qc}}(p) = \mathrm{min}_{\mathrm{qc}}(q)$ and $\mathrm{supp}_{\mathrm{qc}}(p) = \mathrm{supp}_{\mathrm{qc}}(q)$ and $\mathrm{lin}_{\mathrm{qc}}(p) = \mathrm{lin}_{\mathrm{qc}}(q)$; thus, for a quasi-conjugacy class Q, the objects $\mathrm{min}_{\mathrm{qc}}(Q), \mathrm{supp}_{\mathrm{qc}}(Q), \mathrm{lin}_{\mathrm{qc}}(Q)$ are well-defined. The following Proposition 2.18 explains how the minimal points and the quasi-conjugacy support give a complete description of a quasi-conjugacy class. The subsequent Proposition 2.19 explains how the linear algebra gives a complete description of the conjugacy classes contained in the quasi-conjugacy class.

Proposition 2.18. *Let (Γ, ψ) be a GBS graph and let p be a vertex of its affine representation Λ. Then the following are equivalent:*

1. *$q \sim_{\mathrm{qc}} p$*
2. *$q = m + \mathbf{w}$ for some $m \in \mathrm{min}_{\mathrm{qc}}(p)$ and for some $\mathbf{w} \in \mathbf{A}^+$ with $\mathrm{supp}(\mathbf{w}) \subseteq \mathrm{supp}_{\mathrm{qc}}(p)$.*

Proof. Immediate from the definitions. □

Proposition 2.19. *Let (Γ, ψ) be a GBS graph and let p be a vertex of its affine representation Λ. Then we have the following:*

1. *For all $\mathbf{w}, \mathbf{w}' \in \mathbf{A}^+$ with $\mathrm{supp}(\mathbf{w}), \mathrm{supp}(\mathbf{w}') \subseteq \mathrm{supp}_{\mathrm{qc}}(p)$, we have that*
$$p + \mathbf{w} \sim_{\mathrm{c}} p + \mathbf{w}' \Leftrightarrow \mathbf{w} - \mathbf{w}' \in \mathrm{lin}_{\mathrm{qc}}(p).$$
2. *There is a natural inclusion of subgroups $\mathrm{lin}_{\mathrm{qc}}(p) \leq \mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}^{\mathrm{supp}_{\mathrm{qc}}(p)} \leq \mathbf{A}$.*
3. *There is a bijection between the following two sets:*
   (i) The set of conjugacy classes contained in the quasi-conjugacy class of p.
   (ii) The quotient $(\mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}^{\mathrm{supp}_{\mathrm{qc}}(p)})/\mathrm{lin}_{\mathrm{qc}}(p)$.

Remark 2.20. The bijection in Item 3 of Proposition 2.19 is not canonic. It's canonic only up to translations in $(\mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}^{\mathrm{supp}_{\mathrm{qc}}(p)})/\mathrm{lin}_{\mathrm{qc}}(p)$.

Proof. Item 1 follows from Lemma 2.16 and Item 2 is trivial.

Take a conjugacy class Q contained in the quasi-conjugacy class of p. This means that we can find $q \in Q$ such that $q = p + \mathbf{w}$ for some $\mathbf{w} \in \mathbf{A}^+$ with $\mathrm{supp}(\mathbf{w}) \subseteq \mathrm{supp}_{\mathrm{qc}}(p)$. By Lemma 2.16, a different choice of $q \in Q$ will change $\mathbf{w}$ by adding an element of $\mathrm{lin}_{\mathrm{qc}}(p)$. Thus from Q we obtain a well-defined coset $\mathbf{w} + \mathrm{lin}_{\mathrm{qc}}(p)$. This defines a map from the set of conjugacy classes contained in the quasi-conjugacy class of p to $(\mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}^{\mathrm{supp}_{\mathrm{qc}}(p)})/\mathrm{lin}_{\mathrm{qc}}(p)$.

To prove surjectivity, take $\mathbf{x} \in \mathbf{A}$ with $\mathrm{supp}(\mathbf{x}) \subseteq \mathrm{supp}_{\mathrm{qc}}(p)$ and consider the coset $\mathbf{x} + \mathrm{lin}_{\mathrm{qc}}(p)$. Since $\mathrm{supp}(\mathbf{x}) \subseteq \mathrm{supp}_{\mathrm{qc}}(p)$, we can find $\mathbf{w} \in \mathbf{A}^+$ such that $p + \mathbf{w} \sim_{\mathrm{c}} p$ and $\mathbf{x} + \mathbf{w} \geq \mathbf{0}$. In particular $\mathbf{x} + \mathrm{lin}_{\mathrm{qc}}(p) = (\mathbf{x} + \mathbf{w}) + \mathrm{lin}_{\mathrm{qc}}(p)$ and this coset is realized by the conjugacy class of $p + \mathbf{x} + \mathbf{w}$.

To prove injectivity, suppose that we are given $\mathbf{w}, \mathbf{w}' \in \mathbf{A}^+$ with $\mathrm{supp}(\mathbf{w}), \mathrm{supp}(\mathbf{w}') \subseteq \mathrm{supp}_{\mathrm{qc}}(p)$, and such that $p + \mathbf{w}$ and $p + \mathbf{w}'$ define the same coset $\mathbf{w} + \mathrm{lin}_{\mathrm{qc}}(p) = \mathbf{w}' + \mathrm{lin}_{\mathrm{qc}}(p)$. But then $\mathbf{w} - \mathbf{w}' \in \mathrm{lin}_{\mathrm{qc}}(p) = \mathrm{lin}_{\mathrm{qc}}(p)$ and we find $\mathbf{u}, \mathbf{u}' \in \mathbf{A}^+$ such that $\mathbf{w} - \mathbf{w}' = \mathbf{u} - \mathbf{u}'$ and $p + \mathbf{u} \sim_{\mathrm{c}} p + \mathbf{u}' \sim_{\mathrm{c}} p$. This implies that $p + \mathbf{w} \sim_{\mathrm{c}} p + \mathbf{w} + \mathbf{u}' = p + \mathbf{w}' + \mathbf{u} \sim_{\mathrm{c}} p + \mathbf{w}'$, and injectivity follows. □

2.7 The graph of families and the modular map

Let (Γ, ψ) be a GBS graph and let Λ be its affine representation.

Definition 2.21. *We say that $p, q \in V(\Lambda)$ are **in the same family** if $p \sim_{\mathrm{qc}} q$ and they are of the form $p = (v, \mathbf{a})$ and $q = (v, \mathbf{b})$ with $\operatorname{supp}(\mathbf{b} - \mathbf{a}) \subseteq \operatorname{supp}_{\mathrm{qc}}(p)$, for some $v \in V(\Gamma)$ and $\mathbf{a}, \mathbf{b} \in \mathbf{A}^+$.*

Being in the same family is an equivalence relation on $V(\Lambda)$; an equivalence class is called **family** of vertices. Every family of vertices is contained in a quasi-conjugacy class (by definition), and it always contains at least one minimal point of the quasi-conjugacy class (by Proposition 2.19). In particular, each quasi-conjugacy class is partitioned into finitely many families.

Definition 2.22. *Define the **graph of families** $\mathcal{F} = \mathcal{F}(Q)$ of a quasi-conjugacy class Q as follows:*

1. *$V(\mathcal{F})$ is the finite set of families contained in the quasi-conjugacy class Q.*
2. *$E(\mathcal{F})$ is the set of triples $(F, e, F') \in V(\mathcal{F}) \times E(\Lambda) \times V(\mathcal{F})$ such that there is $\mathbf{w} \in \mathbf{A}^+$ with $\iota(e) + \mathbf{w} \in F$ and $\tau(e) + \mathbf{w} \in F'$.*
3. *We set $\overline{(F, e, F')} = (F', \overline{e}, F)$ and $\iota((F, e, F')) = F$ and $\tau((F, e, F')) = F'$.*

Lemma 2.23. *Suppose that the endpoints of $e \in E(\Lambda)$ lie in Q. Then there are unique families $F, F' \in V(\mathcal{F})$ such that $(F, e, F') \in E(\mathcal{F})$, and they are the families containing $\iota(e)$ and $\tau(e)$ respectively.*

Proof. If for some $\mathbf{w} \in \mathbf{A}^+$ the vertex $\iota(e) + \mathbf{w}$ lies in Q, then by definition of quasi-conjugacy support we have $\operatorname{supp}(\mathbf{w}) \subseteq \operatorname{supp}_{\mathrm{qc}}(Q)$, and by definition of family we have that $\iota(e) + \mathbf{w}$ lies in the same family as $\iota(e)$. Similarly for $\tau(e)$. □

If the endpoints of $e \in E(\Lambda)$ don't lie in Q, then the graph of families can contain several edges of the form (F, e, F'), corresponding to different translates of e. However, it will always contain finitely many edges, as there are finitely many possible triples in $V(\mathcal{F}) \times E(\Lambda) \times V(\mathcal{F})$. Note also that the graph of families is connected: for every two families in the same quasi-conjugacy class, there must be an affine path connecting two points in the two families.

Definition 2.24. *Define the **modular map** $q : E(\Gamma) \to \mathbf{A}$ given by $q(e) = \mathbf{b} - \mathbf{a}$ where $\iota_\Lambda(e) = (v, \mathbf{a})$ and $\tau_\Lambda(e) = (u, \mathbf{b})$ for some $v, u \in V(\Gamma)$ and $\mathbf{a}, \mathbf{b} \in \mathbf{A}^+$.*

Note that $q(\overline{e}) = -q(e)$. In particular, we obtain an homomorphism $q : \pi_1(\Gamma) \to \mathbf{A}$, which we call the **modular homomorphism**.

Let Q be a quasi-conjugacy class and let $\mathcal{F} = \mathcal{F}(Q)$ be the corresponding graph of families. Then we have a modular map $q_Q : E(\mathcal{F}) \to \mathbf{A}$ defined by $q_Q((F, e, F')) = q(e)$. Thus we can define a homomorphism $q_Q : \pi_1(\mathcal{F}) \to \mathbf{A}$, which we call **modular homomorphism of** Q .

Proposition 2.25. *Let Q be a quasi-conjugacy class and let $\mathcal{F}(Q)$ be the corresponding graph of families. Then the image of $q_Q : \pi_1(\mathcal{F}(Q)) \to \mathbf{A}$ is exactly $\operatorname{lin}_{\mathrm{qc}}(Q)$.*

Proof. Fix $p \in Q$ and take $\mathbf{r} \in \operatorname{lin}_{\mathrm{qc}}(p)$. Then there is an affine path $(e_1, \ldots, e_\ell)$, with translation coefficients $\mathbf{w}_1, \ldots, \mathbf{w}_\ell$, going from $\iota(e_1) + \mathbf{w}_1 = p + \mathbf{w}$ to $\tau(e_\ell) + \mathbf{w}_\ell = p + \mathbf{w}'$ with $\mathbf{w}, \mathbf{w}' \in \mathbf{A}^+$ and $\mathbf{w}' - \mathbf{w} = \mathbf{r}$. For all $i = 1, \ldots, \ell - 1$ we have that $\tau(e_i) + \mathbf{w}_i = \iota(e_{i+1}) + \mathbf{w}_{i+1}$ must lie in some family $F_i \subseteq Q$. Thus we obtain a path $\sigma = ((F_1, e_1, F_2), \ldots, (F_{\ell-1}, e_\ell, F_1))$ in $\mathcal{F}(Q)$. It's easy to check that $q_Q(\sigma) = \mathbf{w}' - \mathbf{w} = \mathbf{r}$.

Conversely, suppose that we are given a closed path $\sigma = ((F_1, e_1, F_2), \ldots, (F_{\ell-1}, e_\ell, F_1))$ in $\mathcal{F}(Q)$, for some families $F_i \subseteq Q$ with $i = 1, \ldots, \ell$. Then we can find $\mathbf{w}_i \in \mathbf{A}^+$ such that $\iota(e_i) + \mathbf{w}_i \in F_i$ and $\tau(e_i) + \mathbf{w}_i \in F_{i+1}$. It's easy to see that we can adjust $\mathbf{w}_i$ in such a way that $\tau(e_i) + \mathbf{w}_i = \iota(e_{i+1}) + \mathbf{w}_{i+1}$, obtaining an affine path $(e_1, \ldots, e_\ell)$ from $\iota(e_1) + \mathbf{w}_1$ to $\tau(e_\ell) + \mathbf{w}_\ell$. In particular, we can check that $q_Q(\sigma) = \mathbf{w}_\ell - \mathbf{w}_1 \in \operatorname{lin}_{\mathrm{qc}}(p)$ as $\iota(e_1) + \mathbf{w}_1$ and $\tau(e_\ell) + \mathbf{w}_\ell$ lie in the same family F_1.

This proves that the image of the homomorphism $q_Q : \pi_1(\mathcal{F}(Q)) \to \mathbf{A}$ is exactly $\operatorname{lin}_{\mathrm{qc}}(p)$, as desired. □

Remark 2.26. Applying the modular map q to a single edge isn't usually very meaningful. If the two endpoints of e are different vertices of Γ, then the difference $\mathbf{b} - \mathbf{a}$ doesn't make much sense, as we are comparing powers of generators of different vertices. It only makes sense when applied to paths that begine and terminate at the same vertex.

However, if we are interested in studying a certain quasi-conjugacy class Q, then we need to refine the modular homomorphism even more. For example, if the two endpoints of e are the same vertex of Γ, but lie in different families of Q, then the difference $\mathbf{b}-\mathbf{a}$ still needs to be ignored. This is because it will give some linear algebra which is in some sense "external" to the quasi-conjugacy class Q. This is why we consider the modular homomorphism $q_Q : \pi_1(\mathcal{F}(Q)) \to \mathbf{A}$ instead.

For example, in [ACRK25b] we describe a procedure to make a GBS graph into a one-vertex GBS graph. The procedure will add a lot of loops to $\pi_1(\Gamma)$, adding noise to the modular homomorphism $q : \pi_1(\Gamma) \to \mathbf{A}$. However, the procedure will preserve the graphs of families and the modular homomorphisms associated with the quasi-conjugacy classes. Considering the graph of families makes the noise invisible to us.

2.8 Algorithmic computation of conjugacy classes

All the discussion of the previous Section 2.6 can be made algorithmic, as follows.

Proposition 2.27. *There is an algorithm that, given a GBS graph (Γ, ψ) and a vertex p of its affine representation Λ, computes the following:*

1. $\min_{\mathrm{qc}}(p)$.
2. $\mathrm{supp}_{\mathrm{qc}}(p)$.
3. *For every $m \in \min_{\mathrm{qc}}(p)$, an affine path from m to a point $\geq p$.*
4. *An affine path from p to $p + \mathbf{w}$ for some $\mathbf{w} \in \mathbf{A}^+$ with $\mathrm{supp}(\mathbf{w}) = \mathrm{supp}_{\mathrm{qc}}(p)$.*

Description of the algorithm. In order to run the algorithm, it's not restrictive to ignore the component $\mathbb{Z}/2\mathbb{Z}$, just remember to add it to all minimal points at the end of the procedure. If the GBS graph (Γ, ψ) has negative labels, just change them with their absolute values. So here we work with $\mathbf{A} = \mathbb{Z}^{\mathcal{P}(\Gamma,\psi)}$. In each moment, the algorithm keeps in memory

1. A finite set of vertices $M \subseteq V(\Lambda)$.
2. A finite set of components $S \subseteq \mathcal{P}(\Gamma, \psi)$.
3. For every $m \in M$, an affine path σ_m from m to $p+\mathbf{x}_m$ for some $\mathbf{x}_m \in \mathbf{A}^+$ with $\mathrm{supp}(\mathbf{x}_m) \subseteq S$.
4. An affine path σ from p to $p + \mathbf{z}$ for some $\mathbf{z} \in \mathbf{A}^+$ with $\mathrm{supp}(\mathbf{z}) = S$.

CLEAN UP ROUTINE. For every $m, m' \in M$ distinct we check whether $m' \geq m$. If so, we write $m' = m + \mathbf{r}$ for some $\mathbf{r} \in \mathbf{A}^+$ and we perform the following operations:

1. We change S into $S \cup \mathrm{supp}(\mathbf{r})$
2. We change the path σ into $\sigma^d \overline{\sigma}_{m'} \sigma_m$ for some natural number d big enough.
3. We remove m' from M, and we forget the path $\sigma_{m'}$.

We reiterate until we find no couple $m, m' \in M$ distinct with $m' \geq m$.

MAIN ALGORITHM. We initialize $M = \{p\}$ with σ_p trivial. We initialize $S = \emptyset$ with σ trivial. Then we repeat the following iteration until the two sets M, S don't remain unchanged. **Iteration:** We write a list of all the couples $(m, e) \in M \times E(\Lambda)$. For each couple (m, e), we look for the minimum $\mathbf{w} \in \mathbf{A}^+$ such that $\iota(e) + \mathbf{w} = m + \mathbf{s}$ for some $\mathbf{s} \in \mathbf{A}^+$ with $\mathrm{supp}(\mathbf{s}) \subseteq S$. If it doesn't exist, then we just ignore that couple, and go on examining the other couples (m, e). If it exists, then it's unique, and we proceed as follows:

1. Suppose that there is no $m' \in M$ such that $\tau(e) + \mathbf{w} \geq m'$. In this case we change M into $M \cup \{\tau(e) + \mathbf{w}\}$ and we set $\sigma_{\tau(e)+\mathbf{w}} = \overline{e}\sigma_m$. Then we run the clean up routine, and we start a new iteration (i.e. we write again the list of all couples $(m, e) \in M \times E(\Lambda)$ and so on).

2. Suppose that there is $m' \in M$ such that $\tau(e) + \mathbf{w} = m' + \mathbf{r}$ with $\mathbf{r} \in \mathbf{A}^+$ and $\mathrm{supp}(\mathbf{r}) \not\subseteq S$. Then we change S into $S \cup \mathrm{supp}(\mathbf{s'})$, and we change the path σ into $\sigma^d \overline{\sigma}_m e \sigma_{m'}$ for some natural number d big enough. Then we run the clean up routine, and we start a new iteration (i.e. we write again the list of all couples $(m, e) \in M \times E(\Lambda)$ and so on).

3. Suppose that there is $m' \in M$ such that $\tau(e) + \mathbf{w} \geq m'$, and for each such m' we have that $\tau(e) + \mathbf{w} = m' + \mathbf{r}$ with $\mathrm{supp}(\mathbf{r}) \subseteq S$. Then we just ignore that couple, and go on examining the other couples (m, e).

When we run a full iteration without changing the sets M, S, then we quit the cycle. The algorithm terminates and outputs $\mathrm{min}_{\mathrm{qc}}(p) = M$, $\mathrm{supp}_{\mathrm{qc}}(p) = S$, the affine paths σ_m for $m \in M$, the affine path σ. $\square$

Proof that the algorithm works. Observe that we can add elements to the set S only finitely many times (since S is a subset of a finite set). As in the proof of Lemma 2.12, we can consider the ring of polynomials $\mathbb{C}[x_i : i \in \mathcal{P}(\Gamma, \psi)]$ and the ideal $(x^{\mathbf{m}} : (u, \mathbf{m}) \in M)$. This ideal increases strictly every time that the algorithm adds a new element to M; since rings of polynomial are Noetherian, this can happen only finitely many times. This proves that we can change the set M only finitely many times, and thus that the algorithm terminates.

It's easy to prove by induction that, in each moment while running the algorithm, M is contained in the quasi-conjugacy class of p, $S \subseteq \mathrm{supp}_{\mathrm{qc}}(p)$, the paths σ_m for $m \in M$ satisfy the poperty of Item 3, the path σ satisfies the property of Item 4. In particular, when during the algorithm we substitute σ with $\sigma^d \overline{\sigma}_m e \sigma_{m'}$, we are always able to (algorithmically) find an integer d such that the substitution gives another affine path starting at p.

When the algorithm terminates, consider the set of vertices $M^{+S} = \{m + \mathbf{w} : m \in M$ and $\mathbf{w} \in \mathbf{A}^+$ with $\mathrm{supp}(\mathbf{w}) \subseteq S\}$ and observe that M^{+S} is contained in the quasi-conjugacy class of p. But since no couple (m, e) changes the sets M, S, we have that no translate of any edge e can be used to move from a point in M^{+S} to a point outside M^{+S}. This proves that M^{+S} is closed under quasi-conjugacy, and thus M^{+S} is the quasi-conjugacy class of p. Since every time we add a vertex to M we also run the clean up routine, this implies that $M = \mathrm{min}_{\mathrm{qc}}(p)$. This also implies that $S = \mathrm{supp}_{\mathrm{qc}}(p)$. It follows that the output of the algorithm is correct. $\square$

Proposition 2.28. *There is an algorithm that, given a GBS graph (Γ, ψ) and a vertex p of its affine representation Λ, computes the following:*

1. *A finite set of generators for $\mathrm{lin}_{\mathrm{qc}}(p)$.*

2. *For each generator $\mathbf{r}$, an affine path $\sigma_{\mathbf{r}}$ satisfying the condition of Item 1 of Lemma 2.16.*

Proof. Let Q be the quasi-conjugacy class of p. By Proposition 2.27 we can algorithmically compute $\mathrm{supp}_{\mathrm{qc}}(Q)$ and the finite set $M = \mathrm{min}_{\mathrm{qc}}(Q)$. We can also algorithmically compute the graph of families $\mathcal{F}(Q)$ (see Definition 2.22), as $V(\mathcal{F}(Q))$ is a computable quotient of the finite set M and the set of edges is a computable subset of the finite set $V(\mathcal{F}(Q)) \times E(\Gamma) \times V(\mathcal{F}(Q))$. Moreover, we can give an algorithmic description of the modular homomorphism $q_Q : \pi_1(\mathcal{F}(Q)) \to \mathbf{A}$: given a loop θ in $\mathcal{F}(Q)$, we can algorithmically compute $q_Q(\theta) \in \mathbf{A}$. In particular we can compute a set of generators for $\pi_1(\mathcal{F}(Q))$ and their images through the homomorphism q_Q. According to Proposition 2.25, the images are the desired set of generators for $\mathrm{lin}_{\mathrm{qc}}(Q)$ and the loops in $\pi_1(\mathcal{F}(Q))$ are the corresponding affine paths (for some suitable translation coefficients), as desired. $\square$

Corollary 2.29. *There is an algorithm that, given a GBS graph (Γ, ψ) and two vertices p, q of its affine representation Λ, does the following:*

- *Decides whether $p \sim_{\mathrm{qc}} q$, and in case they are, computes two affine paths, one from each of them to a vertex above the other.*

- *Decides whether $p \sim_{\mathrm{c}} q$, and in case they are, computes an affine path from one to the other.*

2.9 Examples

We show a couple of examples, to help the reader understand how conjugacy classes and quasi-conjugacy classes work. In the examples along the paper, for simplicity of notation, we will only deal with GBS graphs (Γ, ψ) with positive labels. For this reason we will use $\mathbf{A} = \mathbb{Z}^{\mathcal{P}(\Gamma,\psi)}$ omitting the $\mathbb{Z}/2\mathbb{Z}$ summand.

Example 2.30. Later in the paper (see Example 4.22), we will describe the isomorphism problem for this example. Consider the GBS graph (Γ, ψ) with a single vertex v and nine edges, with labels as in Figure 3. We call g the generator of $G_v \cong \mathbb{Z}$. Since the labels only use the prime factors 2 and 3, we have that $\mathcal{P}(\Gamma, \psi) = \{2, 3\}$ and we use $\mathbf{A} = \mathbb{Z}^2$. The number $2^a 3^b$ will be associated to the element $(a, b) \in \mathbf{A}^+$ for $a, b \in \mathbb{N}$.

Consider the point $p = (v, (1, 3))$ corresponding to the element $g^{2^1 3^3} = g^{54}$. The quasi-conjugacy class of p consists of only two points $(v, (1, 3))$, $(v, (3, 2))$, corresponding to the elements g^{54}, g^{72} respectively, see Figure 4. In this case $\mathrm{supp}_{\mathrm{qc}}(p) = \emptyset$ and $\min_{\mathrm{qc}}(p)$ is the set of the two points of the quasi-conjugacy class, which is also a conjugacy class. This means that g^n for $n \geq 1$ is conjugate to g^{54} if and only if $n = 54, 72$. In this case $\mathrm{lin}_{\mathrm{qc}}(p) = 0$.

Consider the point $p = (v, (4, 1))$ corresponding to the element $g^{2^4 3^1} = g^{48}$. The quasi-conjugacy class of p consists of the points $(v, (k, 1))$ for $k \geq 4$, corresponding to the elements $g^{2^k \cdot 3}$ for $k \geq 4$. In this case $\mathrm{supp}_{\mathrm{qc}}(p) = \{2\}$ and $\min_{\mathrm{qc}}(p) = \{(v, (4, 1))\}$. This quasi-conjugacy class is partitioned into two conjugacy classes, depending on the parity of k. This means that g^n for $n \geq 1$ is conjugate to g^{48} if and only if $n = 2^k \cdot 9$ for $k \geq 4$ even. In this case $\mathrm{lin}_{\mathrm{qc}}(p) = \langle (2, 0) \rangle$.

Consider the point $p = (v, (2, 3))$ corresponding to the element g^{108}. In this case $\mathrm{supp}_{\mathrm{qc}}(p) = \{2, 3\}$ and $\min_{\mathrm{qc}}(p) = \{(2, 3), (4, 2)\}$ and $\mathrm{lin}_{\mathrm{qc}}(p) = \langle (2, 0), (0, 1) \rangle$. The quasi-conjugacy class is partitioned into two conjugacy classes, as $(\mathbb{Z}^{\mathrm{supp}}_{\mathrm{qc}}(p))/\mathrm{lin}_{\mathrm{qc}}(p) \cong \mathbb{Z}/2\mathbb{Z}$ which has cardinality 2.

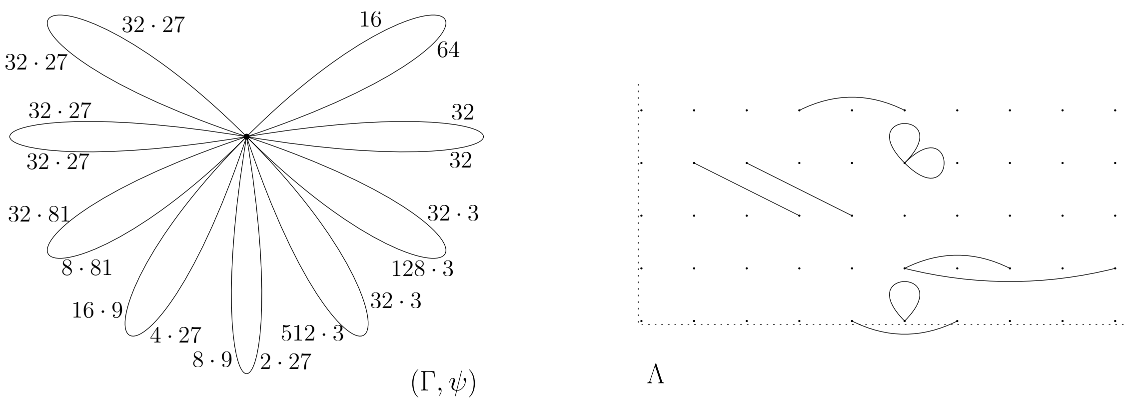

Figure 3: On the left a GBS graph (Γ, ψ) with one vertex and nine edges. On the right the corresponding affine representation Λ. On the horizontal axis the number of factors 2 and on the vertical axis the number of factors 3.

Example 2.31. This example is taken from [Wan23]. Later in the paper (see Example 4.21), we will describe the isomorphism problem for this example. Consider the GBS graph with a single vertex v and four edges, with labels as in Figure 5. The prime numbers of this GBS graph are $\mathcal{P}(\Gamma, \psi) = \{2, 3, 5, 7\}$. The number $2^a 3^b 5^c 7^d$ will be associated to the element $(a, b, c, d) \in \mathbf{A}^+$ for $a, b, c, d \in \mathbb{N}$. We call Λ the affine representation of (Γ, ψ): this consists of a single copy $\mathbf{A}_v^+$ of $\mathbf{A}^+$ containing four edges

$$\begin{cases} (3,0,0,0) \text{ —— } (1,0,1,0) \\ (1,1,0,0) \text{ —— } (0,1,1,0) \\ (0,1,0,1) \text{ —— } (1,1,1,0) \\ (1,0,0,1) \text{ —— } (1,1,1,0) \end{cases}$$

Consider the point $p = (v, (3, 0, 0, 0))$. Its quasi-conjugacy class is finite, with $\mathrm{supp}_{\mathrm{qc}}(p) = \emptyset$ and minimal points $(3, 0, 0, 0), (1, 0, 1, 0)$ inside $\mathbf{A}_v^+$. It coincides with a conjugacy class and $\mathrm{lin}_{\mathrm{qc}}(p) = 0$.

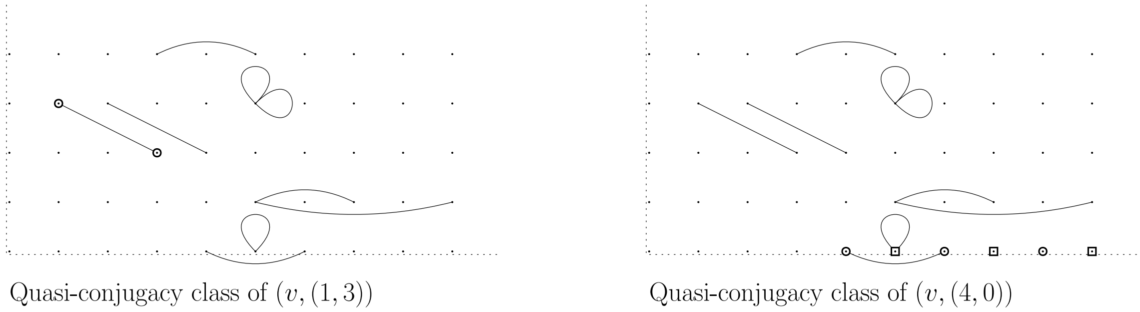

Figure 4: In the figure we see four copies of the same affine representation Λ associated with Example 2.30. In each copy, we put in evidence a different quasi-conjugacy class. We can see the quasi-conjugacy class of $(1, 3)$ (top left), of $(4, 0)$ (top right), of $(4, 1)$ (bottom left), of $(2, 3)$ (bottom right). To distinguish different conjugacy classes inside a common quasi-conjugacy class, we mark the vertices with different symbols (circles, squares).

Consider the point $p = (v, (1, 1, 0, 0))$. Its quasi-conjugacy class is finite, with $\mathrm{supp}_{\mathrm{qc}}(p) = \emptyset$ and minimal points $(1, 1, 0, 0), (0, 1, 1, 0)$ inside $\mathbf{A}_v^+$. It coincides with a conjugacy class and $\mathrm{lin}_{\mathrm{qc}}(p) = 0$.

Consider the point $p = (v, (0, 1, 0, 1))$. Its quasi-conjugacy class has $\mathrm{supp}_{\mathrm{qc}}(p) = \{2, 3, 5, 7\}$ and minimal points $(2, 1, 0, 0), (1, 1, 1, 0), (0, 1, 2, 0), (1, 0, 0, 1), (0, 1, 0, 1)$ inside $\mathbf{A}_v^+$. It coincides with a conjugacy class since $\mathrm{lin}_{\mathrm{qc}}(p) = \mathbb{Z}^{\mathrm{supp}_{\mathrm{qc}}(p)}$.

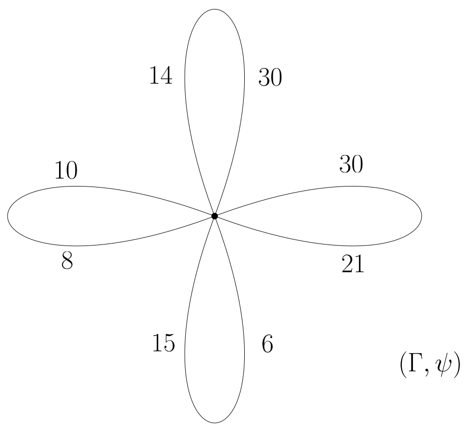

Figure 5: In the picture a GBS graph (Γ, ψ) with one vertex and four edges.

3 Moves on GBS graphs

In this section we define a family of moves that, given a GBS graph, produce another GBS graph with isomorphic fundamental group; these are called slide, induction, swap, connection (see [For06] [ACRK25b]). We recall some literature about the topic.

3.1 Sign-change move

Let (Γ, ψ) be a GBS graph. Let $v \in V(\Gamma)$ be a vertex. Define the map $\psi' : E(\Gamma) \to \mathbf{Z} \setminus \{0\}$ such that $\psi'(e) = -\psi(e)$ if $\tau(e) = v$ and $\psi'(e) = \psi(e)$ otherwise. We say that the GBS graph (Γ, ψ') is obtained from (Γ, ψ) by means of a **vertex sign change**. If $\mathcal{G}$ is the GBS graph of groups associated to (Γ, ψ), then the vertex sign change move corresponds to changing the chosen generator for the vertex group G_v; in particular it induces an isomorphism at the level of the universal group $\mathrm{FG}(\mathcal{G})$ and of the fundamental group $\pi_1(\mathcal{G})$.

Let (Γ, ψ) be a GBS graph. Let $d \in E(\Gamma)$ be an edge. Define the map $\psi' : E(\Gamma) \to \mathbf{Z} \setminus \{0\}$ such that $\psi'(e) = -\psi(e)$ if $e = d, \overline{d}$ and $\psi'(e) = \psi(e)$ otherwise. We say that the GBS graph (Γ, ψ') is obtained from (Γ, ψ) by means of an **edge sign change**. If $\mathcal{G}$ is the GBS graph of groups associated to (Γ, ψ), then the vertex sign change move corresponds to changing the chosen generator for the edge group G_d; in particular it induces an isomorphism at the level of the universal group $\mathrm{FG}(\mathcal{G})$ and of the fundamental group $\pi_1(\mathcal{G})$.

3.2 Slide move

Let (Γ, ψ) be a GBS graph. Let d, e be distinct edges with $\tau(d) = \iota(e) = u$ and $\tau(e) = v$; suppose that $\psi(\overline{e}) = n$ and $\psi(e) = m$ and $\psi(d) = \ell n$ for some $n, m, \ell \in \mathbb{Z} \setminus \{0\}$ (see Figure 6). Define the graph Γ' by replacing the edge d with an edge d'; we set $\iota(d') = \iota(d)$ and $\tau(d') = v$; we set $\psi(\overline{d'}) = \psi(\overline{d})$ and $\psi(d') = \ell m$. We say that the GBS graph (Γ', ψ) is obtained from (Γ, ψ) by means of a **slide**.

At the level of the affine representation, we have an edge p —— q and we have another edge with an endpoint at $p + \mathbf{a}$ for some $\mathbf{a} \in \mathbf{A}^+$. The slide has the effect of moving the endpoint from $p + \mathbf{a}$ to $q + \mathbf{a}$ (see Figure 6).

$$\begin{cases} p \text{ —— } q \\ r \text{ —— } p + \mathbf{a} \end{cases} \xrightarrow{\text{slide}} \begin{cases} p \text{ —— } q \\ r \text{ —— } q + \mathbf{a} \end{cases}$$

3.3 Induction move

Let (Γ, ψ) be a GBS graph. Let e be an edge with $\iota(e) = \tau(e) = v$; suppose that $\psi(\overline{e}) = 1$ and $\psi(e) = n$ for some $n \in \mathbb{Z} \setminus \{0\}$, and choose $\ell \in \mathbb{Z} \setminus \{0\}$ and $k \in \mathbb{N}$ such that $\ell \mid n^k$. Define the map ψ' equal to ψ except on the edges $d \neq e, \overline{e}$ with $\tau(e) = v$, where we set $\psi'(d) = \ell \cdot \psi(d)$. We say that the GBS graph (Γ, ψ') is obtained from (Γ, ψ) by means of an **induction**.

At the level of the affine representation, we have an edge $(v, \mathbf{0})$ —— $(v, \mathbf{w})$. We choose $\mathbf{w}_1 \in \mathbf{A}^+$ such that $\mathbf{w}_1 \leq k\mathbf{w}$ for some $k \in \mathbb{N}$; we take all the endpoints of other edges lying in $\mathbf{A}_v^+$, and we translate them up by adding $\mathbf{w}_1$ (see Figure 7).

3.4 Swap move

Let (Γ, ψ) be a GBS graph. Let e_1, e_2 be distinct edges with $\iota(e_1) = \tau(e_1) = \iota(e_2) = \tau(e_2) = v$; suppose that $\psi(\overline{e}_1) = n$ and $\psi(e_1) = \ell_1 n$ and $\psi(\overline{e}_2) = m$ and $\psi(e_2) = \ell_2 m$ and $n \mid m$ and $m \mid \ell_1^{k_1} n$ and $m \mid \ell_2^{k_2} n$ for some $n, m, \ell_1, \ell_2 \in \mathbb{Z} \setminus \{0\}$ and $k_1, k_2 \in \mathbb{N}$ (see Figure 8). Define the graph Γ' by substituting the edges e_1, e_2 with two edges e_1', e_2'; we set $\iota(e_1') = \tau(e_1') = \iota(e_2') = \tau(e_2') = v$; we set $\psi(\overline{e_1'}) = m$ and $\psi(e_1') = \ell_1 m$ and $\psi(\overline{e_2'}) = n$ and $\psi(e_2') = \ell_2 n$. We say that the GBS graph (Γ', ψ) is obtained from (Γ, ψ) by means of a **swap move**.

Let $\mathbf{w}_1, \mathbf{w}_2 \in \mathbf{A}^+$ and $p, q \in V(\Lambda)$ be such that $p \leq q \leq p + k_1\mathbf{w}_1$ and $p \leq q \leq p + k_2\mathbf{w}_2$ for some $k_1, k_2 \in \mathbb{N}$. At the level of the affine representation, we have an edge e_1 going from p to $p + \mathbf{w}_1$ and an edge e_2 going from q to $q + \mathbf{w}_2$. The swap has the effect of substituting them with e_1' from q to $q + \mathbf{w}_1$ and with e_2' from p to $p + \mathbf{w}_2$ (see Figure 8).

$$\begin{cases} p \text{ —— } p + \mathbf{w}_1 \\ q \text{ —— } q + \mathbf{w}_2 \end{cases} \xrightarrow{\text{swap}} \begin{cases} q \text{ —— } q + \mathbf{w}_1 \\ p \text{ —— } p + \mathbf{w}_2 \end{cases}$$

3.5 Connection move

Let (Γ, ψ) be a GBS graph. Let d, e be distinct edges with $\iota(d) = u$ and $\tau(d) = \iota(e) = \tau(e) = v$; suppose that $\psi(\overline{d}) = m$ and $\psi(d) = \ell_1 n$ and $\psi(\overline{e}) = n$ and $\psi(e) = \ell n$ and $\ell_1 \ell_2 = \ell^k$ for some $m, n, \ell_1, \ell_2, \ell \in \mathbb{Z} \setminus \{0\}$ and $k \in \mathbb{N}$ (see Figure 9). Define the graph Γ' by substituting the edges d, e with two edges d', e'; we set $\iota(d') = v$ and $\tau(d') = \iota(e') = \tau(e') = u$; we set $\psi(\overline{d'}) = n$ and $\psi(d') = \ell_2 m$ and $\psi(\overline{e'}) = m$ and $\psi(e') = \ell m$. We say that the GBS graph (Γ', ψ) is obtained from (Γ, ψ) by means of a **connection move**.

Let $\mathbf{w}, \mathbf{w}_1, \mathbf{w}_2 \in \mathbf{A}^+$ and $k \in \mathbb{N}$ be such that $\mathbf{w}_1 + \mathbf{w}_2 = k \cdot \mathbf{w}$. At the level of the affine representation, we have a two edges q —— $p + \mathbf{w}_1$ and p —— $p + \mathbf{w}$. The connection move has the effect of replacing them with two edges p —— $p + \mathbf{w}_2$ and q —— $q + \mathbf{w}$ (see Figure 9).

$$\begin{cases} q \text{ —— } p + \mathbf{w}_1 \\ p \text{ —— } p + \mathbf{w} \end{cases} \xrightarrow{\text{connection}} \begin{cases} p \text{ —— } q + \mathbf{w}_2 \\ q \text{ —— } q + \mathbf{w} \end{cases}$$

Remark 3.1. In the definition of connection move, we also allow for the two vertices u, v to coincide.

3.6 Self-slide and reverse slide

The following Lemmas 3.2 and 3.3 introduce two additional moves that will be used in the next sections.

Lemma 3.2 (Self-slide)**.** *Let* $\mathbf{a}, \mathbf{b}, \mathbf{w} \in \mathbf{A}^+$ *and* $\mathbf{x} \in \mathbf{A}$. *Suppose* $\mathbf{a}, \mathbf{w}$ *controls* $\mathbf{b}$ *and* $\mathbf{b} + 2\mathbf{x}$. *Then we can change*

$$\begin{cases} \mathbf{a} \text{ —— } \mathbf{a} + \mathbf{w} \\ \mathbf{b} \text{ —— } \mathbf{b} + \mathbf{x} \end{cases} \quad \textit{into} \quad \begin{cases} \mathbf{a} \text{ —— } \mathbf{a} + \mathbf{w} \\ \mathbf{b} + \mathbf{x} \text{ —— } \mathbf{b} + 2\mathbf{x} \end{cases}$$

by means of a sequence of slides and swaps.

Proof. See [ACRK25b]. □

Lemma 3.3 (Reverse slide)**.** *Let* $\mathbf{a}, \mathbf{b}, \mathbf{w} \in \mathbf{A}^+$ *and* $\mathbf{x} \in \mathbf{A}$ *with* $\mathbf{w} + \mathbf{x} \geq \mathbf{0}$. *Suppose* $\mathbf{a}, \mathbf{w}$ *controls* $\mathbf{b}, \mathbf{b} + \mathbf{x}$ *and suppose* $\mathbf{a}, \mathbf{w} + \mathbf{x}$ *controls* $\mathbf{b}, \mathbf{b} + \mathbf{x}$. *If* $\mathbf{A}_v^+$ *contains edges* $\mathbf{a}$ —— $\mathbf{a} + \mathbf{w}$ *and* $\mathbf{b}$ —— $\mathbf{b} + \mathbf{x}$, *then we can change*

$$\begin{cases} \mathbf{a} \text{ —— } \mathbf{a} + \mathbf{w} \\ \mathbf{b} \text{ —— } \mathbf{b} + \mathbf{x} \end{cases} \quad \textit{into} \quad \begin{cases} \mathbf{a} \text{ —— } \mathbf{a} + \mathbf{w} + \mathbf{x} \\ \mathbf{b} \text{ —— } \mathbf{b} + \mathbf{x} \end{cases}$$

by means of a sequence of slides and swaps.

Proof. See [ACRK25b]. □

3.7 Sequences of moves

In this section we recall how the isomorphism problem for generalized Baumslag-Solitar groups is reduced to checking whether two GBS graphs can be related by means of a sequence of moves. We need to introduce the notion of *totally reduced* GBS graph (see [ACRK25b]); we will only need the fact that every GBS graph (Γ, ψ) can be algorithmically changed to a totally reduced one.

Definition 3.4. *A GBS graph* (Γ, ψ), *with affine representation* Λ, *is called* ***totally reduced*** *if the following conditions hold:*

1. *For every vertex* $v \in V(\Gamma)$, *if* $(v, \mathbf{0}) \sim_c (u, \mathbf{b})$ *for* $u \in V(\Gamma)$ *and* $\mathbf{b} \in \mathbf{A}^+$, *then* $u = v$.

2. *For every vertex* $v \in V(\Gamma)$, *in* Λ *there is an edge* $(v, \mathbf{0})$ —— $(v, \mathbf{w})$ *with* $\mathbf{w} \in \mathbf{A}^+$ *such that, if* $(v, \mathbf{0}) \sim_c (v, \mathbf{b})$ *for* $\mathbf{b} \in \mathbf{A}^+$, *then* $\mathbf{0}, \mathbf{w}$ *controls* $\mathbf{b}$.

Remark 3.5. Condition 1 in the above Definiton 3.4 coincides with the notion of *fully reduced* GBS graph introduced in [For06].

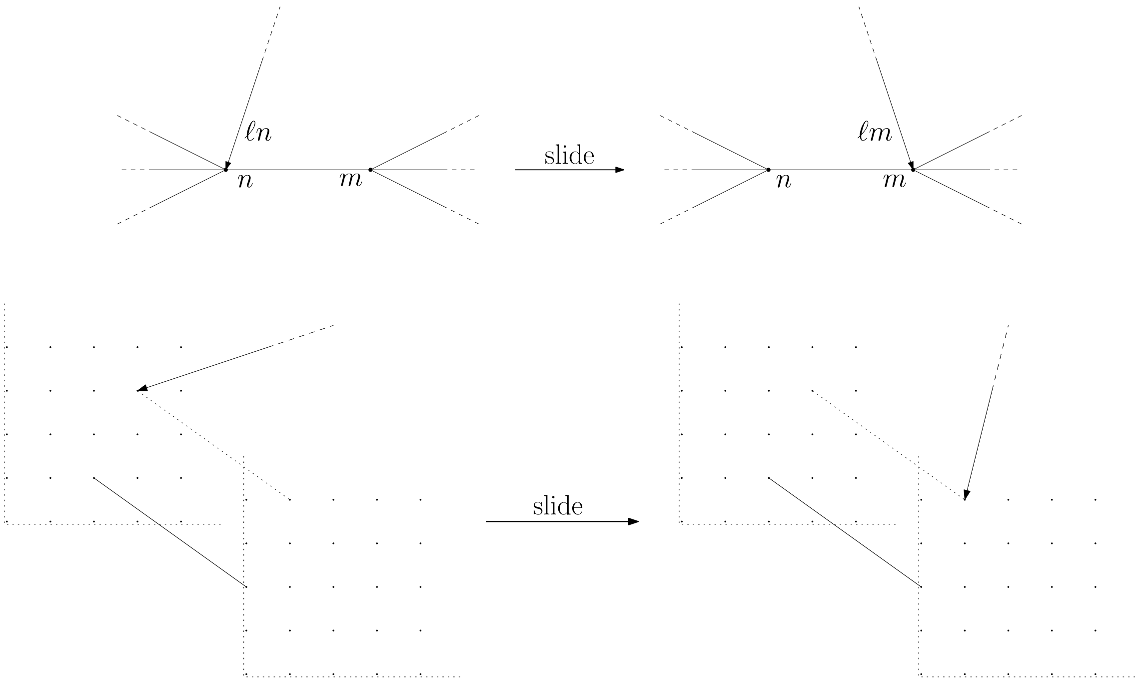

Figure 6: An example of a slide move. Above you can see the GBS graphs. Below you can see the corresponding affine representations.

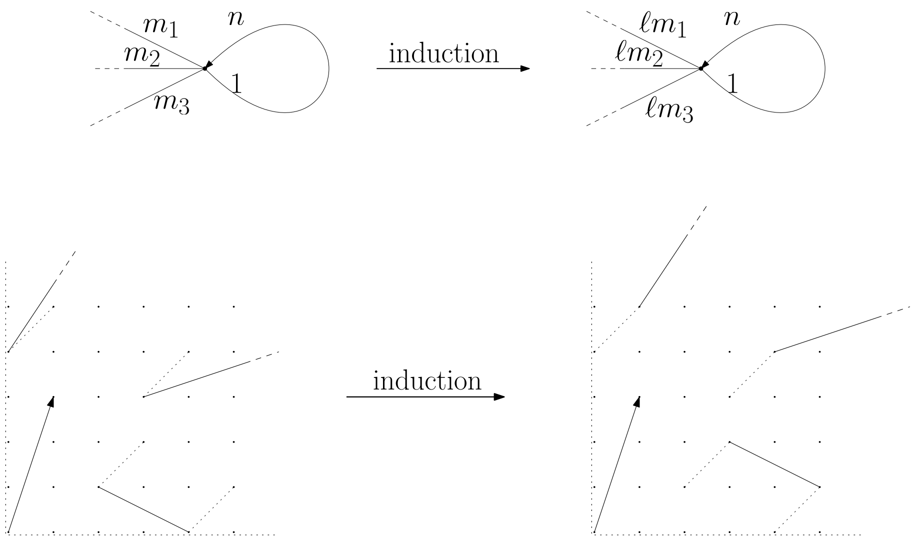

Figure 7: An example of an induction move. Above you can see the GBS graphs; here $\ell \mid n^k$ for some integer $k \geq 0$. Below you can see the corresponding affine representations.

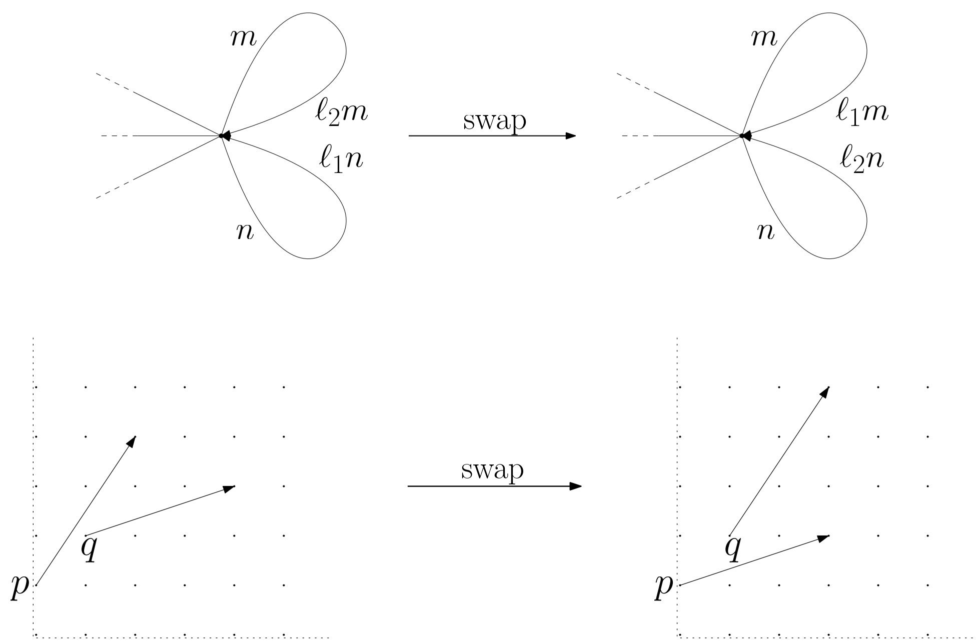

Figure 8: An example of a swap move. Above you can see the GBS graphs; here $n \mid m \mid \ell_1^{k_1} n$ and $n \mid m \mid \ell_2^{k_2} n$ for some integers $k_1, k_2 \geq 0$. Below you can see the corresponding affine representations.

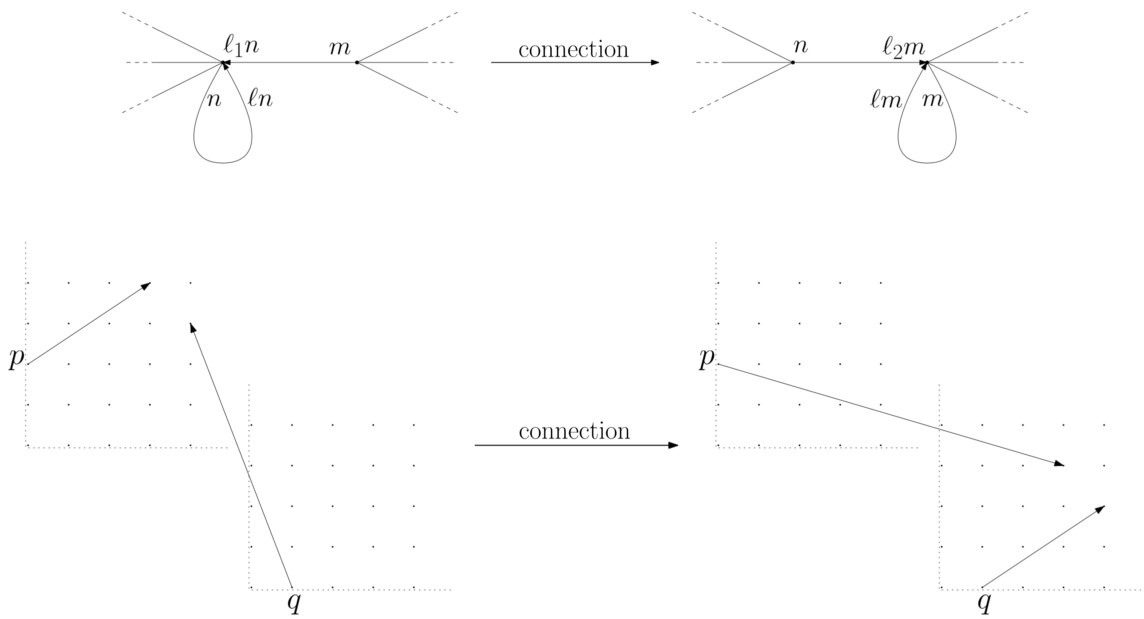

Figure 9: An example of a connection move. Above you can see the GBS graphs; here $\ell_1 \ell_2 = \ell^k$ for some integer $k \geq 0$. Below you can see the corresponding affine representations.

Proposition 3.6. *There is an algorithm that, given a GBS graph (Γ, ψ), computes a totally reduced GBS graph (Γ', ψ') such that the corresponding graphs of groups have isomorphic fundamental group.*

Proof. See [ACRK25b]. $\square$

Here's one of the main results from [ACRK25b], which will be the starting point for all the results of this paper.

Theorem 3.7. *Let $(\Gamma, \psi), (\Delta, \phi)$ be totally reduced GBS graphs and suppose that the corresponding graphs of groups have isomorphic fundamental group. Then $|V(\Gamma)| = |V(\Delta)|$ and there is a sequence of slides, swaps, connections, sign-changes and inductions going from (Δ, ϕ) to (Γ, ψ). Moreover, all the sign-changes and inductions can be performed at the beginning of the sequence.*

Proof. See [ACRK25b]. $\square$

Thus it's enough to deal with the following problem: given two GBS graphs $(\Gamma, \psi), (\Gamma', \psi')$, determine whether there is a sequence of edge sign-changes, inductions, slides, swaps, connections going from one to the other. Note that a sign-change (resp. a slide, an induction, a swap, a connection) induces a natural bijection between the set of vertices of the graph before and after the move. Of course we can ignore the issue of guessing the bijection among the sets of vertices and the sign-changes at the beginning of the sequence, as these choices can be done only in finitely many ways. In what follows, we will also ignore the issue of guessing the inductions at the beginning of the sequence, as this hopefully represents a marginal issue - even though sometimes there are infinitely many possibilities, and thus this issue should be dealt with. In this paper, we will focus on the following question:

Question 3.8. *Given two totally reduced GBS graphs $(\Gamma, \psi), (\Gamma', \psi')$ and a bijection $b : V(\Gamma') \to V(\Gamma)$, is there a sequence of (edge sign-changes,) slides, swaps, connections going from (Γ, ψ) to (Γ', ψ') and inducing the bijection b on the set of vertices?*

In [ACRK25b] we also show that the Question 3.8 can be reduced to the case of one-vertex graphs, and with all edge-labels positive and $\neq 1$. However, the tools we develop in this paper are general and don't make use of these additional assumptions.

3.8 Examples

Theorem 3.7 can already be used to explicitly describe, in some cases, the list of all the possible configurations which can be reached from a given one. We show through an example how this can help in solving the isomorphism problem for GBSs. As usual, in the examples we use $\mathbf{A} = \mathbb{Z}^{\mathcal{P}(\Gamma,\psi)}$, omitting the $\mathbb{Z}/2\mathbb{Z}$ summand.

Example 3.9. Consider the GBS graph (Γ, ψ) with one vertex v and two edges, with labels $8, 48$ and $27, 72$. (see Figure 10). If we apply a connection move, we obtain another GBS graph with one vertex v and two edges, the new labels being $8, 108$ and $27, 162$. It's easy to see that, using slide moves, we can reach all the configurations of the following list L:

1. Two edges with labels $8, 48$ and $27, 72 \cdot 6^k$, for some $k \geq 0$ integer.

2. Two edges with labels $8, 108 \cdot 6^k$ and $27, 162$, for some $k \geq 0$ integer.

But it's also easy to check that if we start at a configuration in L, then we can't apply a swap move, and if we apply a slide or a connection we fall again in a configuration listed in L. Thus L is a complete list of all the configurations which can be reached from (Γ, ψ) using only slides, swaps, connections. By Theorem 3.7 we can algorithmically determine whether a GBS graph (Δ, ϕ) encodes a GBS group isomorphic to the one of (Γ, ψ). In order to do this, first we make (Δ, ϕ) into a totally reduced GBS graph (this can be done algorithmically). Then we apply sign-changes in all the (finitely many) possible ways, and for each of the resulting GBS graphs, we check whether it appears in the list L. Note that induction is not possible on (Γ, ψ), so we don't need to worry about it in this example.

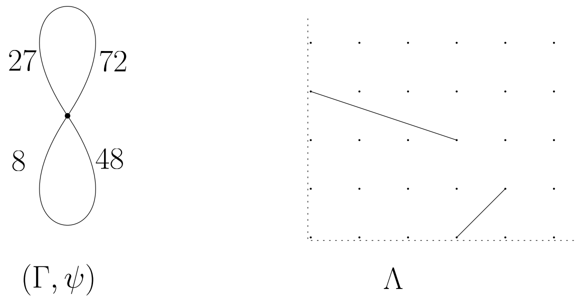

Figure 10: On the left a GBS graph (Γ, ψ) with one vertex and two edges. On the right the corresponding affine representation Λ (on the horizontal axis the number of factors 2, on the vertical axis the number of factors 3).

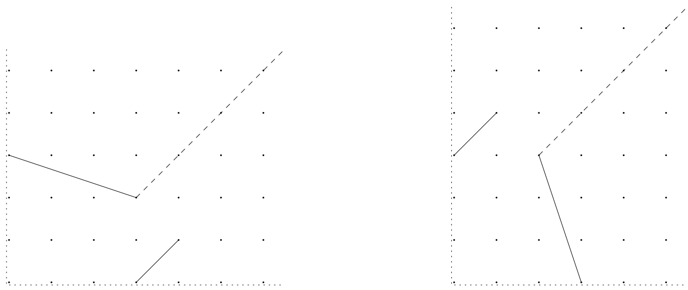

Figure 11: On the left, the configurations that can be reached from (Γ, ψ) by applying only slide moves - the endpoint of one of the edges is allowed to move along the dashed line, while the other three endpoints are required to stay fixed. On the right, the configurations that can be reached from (Γ, ψ) by applying a connection and then only slide moves - again, the endpoint of one of the edges is allowed to move along the dashed line, while the other endpoints stay fixed.

4 Independence of quasi-conjugacy classes

In this section we describe the basic invariants of the moves (which are also essentially isomorphism invariants). These are listed in Definition 4.3, and are the set of primes, the conjugacy and quasi-conjugacy classes, the number of edges in each conjugacy class.

We divide slide moves into two types: *internal* (when they involve two edges in the same quasi-conjugacy class) and *external* (otherwise). We introduce a notion of *external equivalence*, we show that it's an invariant of the moves, and that it can be used to describe external slide moves (without any need to know the exact position of the edges). We show that edges in different quasi-conjugacy classes can only interact with each other using the external equivalence. For the isomorphism problem, this means that distinct quasi-conjugacy classes can be dealt with separately and independently of each other, see Theorem 4.9.

Finally, we introduce the notions of *minimal region.* Every endpoint of every edge must lie above at least one minimal region, making them into excellent "starting points" for slide moves. We show that each of these regions is forced to contain at least one endpoint of some edge, see Lemma 4.17. This decreases a lot the amount of possible configurations that one has to consider.

We show with concrete examples how helpful these notions are when trying to describe the isomorphism problem.

4.1 Basic invariants of the moves

Let (Γ, ψ) be a GBS graph and let Λ be its affine representation. In Section 2.5 we had defined a partition of the set of vertices $V(\Lambda)$, given by the equivalence relation of conjugacy. We had also defined a pre-order $\preceq_c$ on $V(\Lambda)$, by saying that $p \preceq_c q$ if $p \leq q' \sim_c q$ for some other vertex q'. This pre-order $\preceq_c$ induces another partition of $V(\Lambda)$, given by the equivalence relation of quasi-conjugacy. At the same time $\preceq_c$ also defines an order relation on the set of quasi conjugacy classes. These relations can be extended to edges, as follows.

Definition 4.1. *For $e, f \in E(\Lambda)$ we denote:*

1. *$e \sim_c f$ if $\iota(e) \sim_c \iota(f)$, and we say that e, f are **conjugate**.*
2. *$e \preceq_c f$ if $\iota(e) \preceq_c \iota(f)$*
3. *$e \sim_{qc} f$ if $\iota(e) \sim_{qc} \iota(f)$, and we say that e, f, are **quasi-conjugate**.*

We observe that for $e \in E(\Lambda)$ we have $\iota(e) \sim_c \tau(e)$. In particular, in the above Definition 4.1 we can use $\tau(e), \tau(f)$ instead of $\iota(e), \iota(f)$, as we prefer. As for vertices, conjugacy and quasi-conjugacy are equivalence relations on the sets of edges, and $\preceq_c$ induces a partial order on the set of quasi-conjugacy classes of edges. In particular, we obtain a finite poset of quasi-conjugacy classes of edges with the order relation $\preceq_c$.

Definition 4.2. *For $e \in E(\Gamma)$ we say that:*

1. *e belongs to the conjugacy class C if $\iota(e) \in C$. We define the set*

$$\mathrm{E_c}(\Lambda, C) := \{e \in E(\Lambda) : \iota(e) \in C\}$$

of the edges in the conjugacy class C.

2. *e belongs to the quasi-conjugacy class Q if $\iota(e) \in Q$. We define the set*

$$\mathrm{E_{qc}}(\Lambda, Q) := \{e \in E(\Lambda) : \iota(e) \in Q\}$$

of the edges quasi-conjugate to Q.

We are now ready to define the basic invariants of a GBS graph.

Definition 4.3 (Basic invariants)**.** *Let (Γ, ψ) be a GBS graph with affine representation Λ. We define the **basic invariants** of (Γ, ψ) as the following list of data:*

1. *The set of primes $\mathcal{P}(\Gamma, \psi)$.*
2. *The finite list of quasi-conjugacy classes Q containing at least one edge. For each such quasi-conjugacy class we list the finite data $\min_{qc}(Q), \mathrm{supp}_{qc}(Q), \mathrm{lin}_{qc}(Q)$. For each such quasi-conjugacy class, we write the number $|\mathrm{E_{qc}}(\Lambda, Q)|$.*
3. *The finite list of conjugacy classes $C \subseteq Q$ containing at least one edge, for each of them the number $|\mathrm{E_c}(\Lambda, C)|$.*

By direct check with the definitions, we have that the basic invariants are, indeed, invariant by slides, swaps, connections. Sign-changes and induction will preserve the invariants of Items 1 and 2. An edge sign-change will change the numbers $|\mathrm{E_c}(\Lambda, C)|$, as it will move an edge from a conjugacy class C to the conjugacy class $C + \mathbf{e}$ where $\mathbf{e} \in \mathbf{A}$ is the 2-torsion element. A vertex sign-change or an induction will change the structure of the conjugacy classes C and the numbers $|\mathrm{E_c}(\Lambda, C)|$; however, we point out that these changes occur in quite a controlled way, as vertex sign-changes and inductions are just moving the points inside a copy $\mathbf{A}_v^+ \subseteq V(\Lambda)$ by translation. In particular, we have the following:

Corollary 4.4. *Let $(\Gamma, \psi), (\Gamma', \psi')$ be two GBS graph and suppose that there is a sequence of slides, swaps, connections going from one to the other and inducing a bijection $V(\Gamma) = V(\Gamma')$. Then this induces a bijection $V(\Lambda) = V(\Lambda')$, and $(\Gamma, \psi), (\Gamma', \psi')$ have the same basic invariants (Definition 4.3).*

4.2 External equivalence

Let (Γ, ψ) be a GBS graph and let Λ be its affine representation.

Definition 4.5. *Let $p, q \in V(\Lambda)$ with $p \sim_{\mathrm{qc}} q$.*

1. *We say that p, q are **externally equivalent**, denoted $p \sim_{\mathrm{ee}} q$, if there is an affine path, going from p to q, which doesn't use edges quasi-conjugate to p and q.*
2. *We denote $p \preceq_{\mathrm{ee}} q$ if $p \leq q'$ for some $q' \sim_{\mathrm{ee}} q$.*
3. *We say that p, q are **quasi-externally equivalent**, denoted $p \sim_{\mathrm{qee}} q$, if $p \preceq_{\mathrm{ee}} q$ and $q \preceq_{\mathrm{ee}} p$.*

The (quasi-)external equivalence class of p is its (quasi-)conjugacy class in the GBS graph where we remove all edges quasi-conjugate to p. In particular, they can be described in terms of the same data as for (quasi-)conjugacy classes.

Definition 4.6. *Let (Γ, ψ) be a GBS graph and let p be a vertex of its affine representation Λ.*

1. *Define the **external minimal points***

$$\min_{\mathrm{qee}}(p) = \{q \in V(\Lambda) : q \sim_{\mathrm{qee}} p \text{ and } q \preceq_{\mathrm{ee}} q' \text{ for all } q' \sim_{\mathrm{qee}} p\}$$

2. *Define the **external support***

$$\mathrm{supp}_{\mathrm{qee}}(p) = \bigcup \{\mathrm{supp}(\mathbf{u}) : \mathbf{u} \in \mathbf{A}^+ \text{ and } p + \mathbf{u} \sim_{\mathrm{qee}} p\} \subseteq \mathcal{P}(\Gamma, \psi).$$

3. *Define the **external linear algebra***

$$\mathrm{lin}_{\mathrm{qee}}(p) = \{\mathbf{r} \in \mathbf{A} : \mathbf{r} = \mathbf{w} - \mathbf{w}' \text{ for some } \mathbf{w}, \mathbf{w}' \in \mathbf{A}^+ \text{ such that } p + \mathbf{w} \sim_{\mathrm{ee}} p + \mathbf{w}' \sim_{\mathrm{ee}} p\}.$$

Lemmas 2.12 2.14 2.15 2.16 hold for external equivalence in the exact same way as they do for conjugacy. Similarly, it's easy to see that $\min_{\mathrm{qee}}(p), \mathrm{supp}_{\mathrm{qee}}(p), \mathrm{lin}_{\mathrm{qee}}(p)$ are invariant if we change p by quasi-external equivalence. Propositions 2.18 and 2.19 also hold for external equivalence, meaning that $\min_{\mathrm{qee}}(p), \mathrm{supp}_{\mathrm{qee}}(p), \mathrm{lin}_{\mathrm{qee}}(p)$ can be used to describe external equivalence class using a finite set of data. Finally, the algorithms of Propositions 2.27 2.28 and Corollary 2.29 can be used for external equivalence as well, making everything algorithmically computable.

The following Lemma 4.7 shows that the external-equivalence classes are uniquely determined by the basic invariants.

Lemma 4.7. *Let $(\Gamma, \psi), (\Gamma', \psi')$ be GBS graphs with a bijection $V(\Gamma) = V(\Gamma')$. Suppose that they have the same set of primes and the same quasi-conjugacy classes containing at least one edge. Then $(\Gamma, \psi), (\Gamma', \psi')$ have the same external equivalence and quasi-external equivalence relations.*

Proof. Let $p \sim_{\mathrm{qc}} q$ be two vertices in a quasi-conjugacy class Q in the affine representation Λ of (Γ, ψ). Then $p \sim_{\mathrm{ee}} q$ if and only if there is a sequence $p = p_1, p_2, \ldots, p_{\ell-1}, p_\ell = q$ with the following properties:

1. For all $i = 1, \ldots, \ell - 1$ there is a quasi-conjugacy class $Q_i \neq Q$ containing at least one edge, and points $r_i \sim_{\mathrm{c}} s_i \in Q_i$.
2. For all $i = 1, \ldots, \ell - 1$ we have that $p_i = r_i + \mathbf{w}_i$ and $p_{i+1} = s_i + \mathbf{w}_i$ for some $\mathbf{w}_i \in \mathbf{A}^+$.

If $p \sim_{\mathrm{ee}} q$ then the existence of such a sequence follows from the definition. Conversely, if there is such a sequence, then we can easily construct an affine path from p to q which doesn't use any edge in Q. But the existence of such a sequence only depends on the quasi-conjugacy classes containing at least one edge. The conclusion follows. $\square$

4.3 External slide moves

The following Lemma 4.8 introduces a new move, based on external equivalence classes. As we will see, this is the key to separate the main Question 3.8 into several independent problems, each of them internal to a single quasi-conjugacy class.

Lemma 4.8 (External slides)**.** *Let (Γ, ψ) be a GBS graph and let Λ be its affine representation. Suppose that Λ contains an edge p —— q and that $q \sim_{\text{ee}} q'$ for some $p, q, q' \in V(\Lambda)$. Then we can change p —— q into p —— q' by means of a sequence of slide moves.*

Proof. Since $q \sim_{\text{ee}} q'$ there must be an affine path from q to q' that doesn't use any edge quasi-conjugate to q. In particular, the affine paths doesn't use the edge p —— q. Thus we can slide p —— q along the affine path in order to change it into p —— q'. □

Let (Γ, ψ) be a GBS graph and let Λ be its affine representation. Observe that, if we perform a swap or a connection move involving two edges e, f, then $f \sim_{\text{qc}} e$. Instead, when changing an edge e by means of a slide over an edge f, there are two possibilities:

1. $f \sim_{\text{qc}} e$, in which case we say that the slide is **internal**.
2. $f \not\sim_{\text{qc}} e$, in which case we must have $f \preceq_{\text{c}} e$, and the slide move is a particular case of the external slide move of Lemma 4.8.

In particular, this means that we can deal with distinct quasi-conjugacy classes separately and independently from each other.

Theorem 4.9 (Independence of quasi-conjugacy classes)**.** *Let $(\Gamma, \psi), (\Gamma', \psi')$ be two GBS graphs with a bijection $V(\Gamma) = V(\Gamma')$. Then the following are equivalent:*

1. *There is a sequence of slides, swaps, connections going from (Γ, ψ) to (Γ', ψ') inducing the given bijection.*
2. *For every quasi-conjugacy class Q, there is a sequence of swaps, connections, internal slides, external slides (Lemma 4.8) - each of them involving only edges in Q - going from the configuration in (Γ, ψ) to the configuration in (Γ', ψ').*

Proof. Every slide move which involves two edges from different quasi-conjugacy classes can be replaced by an external slide as in Lemma 4.8. But external slides don't depend on the position of the edges in the other quasi-conjugacy classes (but only on the external equivalence relation). Thus, different moves, involving edges in different quasi-conjugacy classes, now commute. Thus we can deal with each quasi-conjugacy class separately and independently. □

In some particular cases, the possible configurations of certain quasi-conjugacy classes are extremely easy to describe.

Corollary 4.10. *Let (Γ, ψ) be a GBS graph with affine representation Λ, and let Q be a quasi-conjugacy class. Suppose that Q contains only one edge e. Then, all the configurations inside Q obtained by sequences of external slides, internal slides, swaps, connections, can also be obtained by performing two external-slides on the two endpoints of e.*

Proof. With only one edge in the quasi-conjugacy class, it's not possible to perform internal slides, swaps, connections. □

Corollary 4.11. *Let (Γ, ψ) be a GBS graph with affine representation Λ, and let Q be a quasi-conjugacy class. Suppose that, for all $p, p' \in Q$, we have that $p \sim_{\text{c}} p'$ if and only if $p \sim_{\text{ee}} p'$. Then, all the configurations inside Q obtained by sequences of external slides, internal slides, swaps, connections, can also be obtained by performing external-slides only, and once on every endpoint of every edge.*

Proof. It's easy to manually check that a single move between internal slide, swap, connection, can be substituted with external slides. The conclusion follows. □

Corollary 4.12. *Let (Γ, ψ) be a GBS graph with affine representation Λ, and let Q be a quasi-conjugacy class. Suppose that $\operatorname{supp}_{\mathrm{qc}}(Q) = \emptyset$. Then only finitely many configurations can be obtained in Q by means of sequences of external slides, internal slides, swaps, connections.*

Proof. If $\operatorname{supp}_{\mathrm{qc}}(Q) = \emptyset$ then Q is finite, and thus finitely many configurations are possible. □

4.4 Mobile edges

Clay and Forester introduced a notion of *mobile edge* [CF08, Definition 3.12], which played an important role in the subsequent literature on GBSs (e.g. [Dud17] [Wan23]). An edge $e \in E(\Lambda)$ is mobile if and only if it belongs to a quasi-conjugacy class Q with $\operatorname{supp}_{\mathrm{qc}}(Q) \neq \emptyset$. As a corollary of the above discussion, we recover the following well-known results:

Corollary 4.13 ([For06, Theorem 8.2])**.** *Let (Γ, ψ) be a GBS graph and suppose that $q(\pi_1(\Gamma)) \cap \mathbf{A}^+ = \{\mathbf{0}\}$. Then only finitely many configurations can be obtained from (Γ, ψ) by means of sequences of sign-changes, inductions, slides, swaps, connections.*

Proof. If $q(\pi_1(\Gamma)) \cap \mathbf{A}^+ = \{\mathbf{0}\}$ then for every quasi-conjugacy class Q we must have $\operatorname{supp}_{\mathrm{qc}}(Q) = \emptyset$, and the result follows from Corollary 4.12. □

Corollary 4.14 ([Dud17, Theorem 2])**.** *There is an algorithm that, given GBS graphs $(\Gamma, \psi), (\Delta, \phi)$ such that (Γ, ψ) has at most one mobile edge, decides whether the corresponding GBS groups are isomorphic or not.*

Proof. Let Q be a quasi-conjugacy class in the affine representation Λ of (Γ, ψ). If Q contains at least one edge, then either $\operatorname{supp}_{\mathrm{qc}}(Q) = \emptyset$ (and we can describe the isomorphism problem using Corollary 4.12) or Q contains only one edge (and we can decide the isomorphism problem using Corollary 4.10). □

4.5 Minimal regions of a quasi-conjugacy class

Let (Γ, ψ) be a GBS graph and let Λ be its affine representation. Let $Q \subseteq V(\Lambda)$ be a quasi-conjugacy class.

Definition 4.15. *A quasi-external equivalence class $R \subseteq Q$ is called* ***region*** *of Q.*

The set of regions of Q is a partition of Q. Moreover, the relation $\preceq_{\mathrm{ee}}$ makes the set of regions of Q into a partially ordered set. A **minimal region** of Q is a region $M \subseteq Q$ which is minimal by $\preceq_{\mathrm{ee}}$. Every minimal region must contain at least one point from $\min_{\mathrm{qc}}(Q)$ - possibly more than one. Each quasi-conjugacy class contains finitely many minimal regions, and always at least one.

Remark 4.16. Not every point in $\min_{\mathrm{qc}}(Q)$ needs to belong to a minimal region.

The reason for which we are interested in the minimal regions of Q is that they often give us important information about the position of the edges in the quasi-conjugacy class (especially when the quasi-conjugacy class contains few edges compared to the number of minimal regions, or when the configuration is particularly "rigid").

Lemma 4.17. *Let Q be a quasi-conjugacy class containing at least one edge. Then every minimal region of Q must contain at least one endpoint of one edge. In particular, if Q contains $d \geq 1$ minimal regions, then Q contains at least $\lceil \frac{d}{2} \rceil$ edges.*

Proof. Edges from other quasi-conjugacy classes don't allow us to change region. Therefore, if we want to move away from a minimal region M, we need to do that using an edge belonging to our quasi-conjugacy class Q. By minimality, this means that there must be an edge with an endpoint in M. □

Remark 4.18. The lower bound $\lceil \frac{d}{2} \rceil$ in the above Lemma 4.17 is optimal.

Definition 4.19. *An edge e is called* ***floating*** *if none of $\iota(e), \tau(e)$ is in a minimal region.*

Note that the number of floating edges isn't invariant when performing moves.

4.6 Examples

As usual, in the examples we use $\mathbf{A} = \mathbb{Z}^{\mathcal{P}(\Gamma,\psi)}$, omitting the $\mathbb{Z}/2\mathbb{Z}$ summand.

Example 4.20. Consider the GBS graph (Γ, ψ) with one vertex v and four edges, as in Figure 12, and call Λ its affine representation. The edges are partitioned into two different quasi-conjugacy classes $B \preceq_c C$, with B containing one edge and C containing three edges (see Figure 13). The quasi-conjugacy class B has exactly one minimal region $N = \{(1, 0)\}$. The quasi-conjugacy class C has exactly two minimal regions, namely $M_1 = \{(k, 1) : k \geq 1\}$ and $M_2 = \{(0, 3)\}$.

Suppose that (Δ, ϕ) is a GBS graph which is obtained from (Γ, ψ) by means of a sequence of slides, swaps, connections. By Corollary 4.4 (Δ, ϕ) must have the same quasi-conjugacy classes and the same regions as (Γ, ψ), and that it must have exactly one edge in B and three in C. By Theorem 4.9 we deduce that (Δ, ϕ) must contain an edge $(1, 0)$ —— $(4, 0)$ since that edge is alone in the quasi-conjugacy class B, and thus by Corollary 4.10 it can only change by external slides, which are trivial in this case. Lemma 4.17 tells us that among the three edges contained in C, at least one must have an endpoint in M_1 and at least one must have an endpoint in M_2.

Suppose that we try and change the GBS graph by means of slides, swaps, connections. The quasi-conjugacy class B contains only one edge $(1, 0)$ —— $(4, 0)$, and thus by Corollary 4.10 that edge will always stay there. The quasi-conjugacy class C is characterized by $\min_{qc}(C) = \{(1, 1), (0, 3)\}$ and $\mathrm{supp}_{qc}(C) = \{2, 3\}$ and $\mathrm{lin}_{qc}(C) = \langle (3, 0), (2, 2)\rangle$. In particular C is partitioned into six conjugacy classes; the three edges belong to three different conjugacy classes.

Suppose that we are given another GBS graph (Γ', ψ') with the same conjugacy classes, and the same number of edges in each conjugacy class. Of course we must have an edge $(1, 0)$ —— $(4, 0)$ in B. Let's call e_1, e_2, e_3 the other three edges, with $e_1 \sim_c (2, 1)$ and $e_2 \sim_c (3, 0)$ and $e_3 \sim_c (2, 2)$. We observe that, by Lemma 4.17, there must always be at least one edge with an endpoint in M_2, and by looking at the conjugacy classes, this edge must be e_2. We deal with cases.

CASE 1: The other endpoint of e_2 falls inside M_1. In this case e_2 must be given by $(0, 3)$ —— $(a, 1)$ for some integer $a \geq 1$, and up to external slide we can assume that e_2 is given by $(0, 3)$ —— $(1, 1)$. If we now keep e_2 fixed, we can move the endpoints of e_1, e_3 at will inside their conjugacy class. In particular, we can arrive at the configuration with e_1 given by $(1, 3)$ —— $(2, 1)$ and e_3 given by $(1, 4)$ —— $(2, 2)$, from which we can reach (Γ, ψ).

CASE 2: e_2 is given by $(0, 3)$ —— (a, b) for some integers $(a, b) \geq (1, 3)$. By Lemma 4.17, we obtain that e_1 must have one endpoint inside M_1. It can't have both endpoints inside M_1, otherwise M_1 would be a quasi-conjugacy class by itself, contradiction; up to external slide, we can assume that e_1 is of the form $(2, 1)$ —— (c, d) for some integers $(c, d) \geq (0, 3)$. Up to external slide, we can obtain $(a, b) \geq (0, 3), (2, 1)$, and up to sliding e_1 along e_2, we can obtain $(c, d) \geq (0, 3), (2, 1)$. By means of slides, we can now make all the edges very long, and then we can use Theorem B.1 to reach (Γ, ψ) - here we are using that the quotient $\mathrm{lin}_{qc}(C)/\mathrm{lin}(C \text{ rel } e_1, e_2, e_3)$ is 2-generated, and thus every triple of generators is Nielsen equivalent to any other.

CASE 3: e_2 is given by $(0, 3)$ —— $(0, a)$ for some integer $a \geq 3$. We can't have $a = 3$, otherwise M_2 would be a quasi-conjugacy class by itself, contradiction; thus $a \geq 4$. One of e_1, e_3 must be of the form $(0, b)$ —— (c, d) for integers $b \geq 3$ and $(c, d) \geq (1, 1)$, otherwise the set $\{(0, k) : k \geq 3\}$ would be a (union of) quasi-conjugacy class(es) by itself, contradiction. But then we can perform a connection move and reduce ourselves to case 2.

To summarize, a GBS graph can be obtained from (Γ, ψ) by means of a sequence of slides, swaps, connections, if and only if it has the same basic invariants as (Γ, ψ). This characterizes the isomorphism problem for (Γ, ψ). As usual, it remains to check the (finitely many) sign-changes. Note that induction doesn't apply to this example.

Example 4.21. We can now describe the isomorphism problem for the GBS graph of Example 2.31. Using the same notation as in Example 2.31, call (Γ, ψ) the GBS graph, with a single vertex v, and Λ its affine representation.

The edges of Λ are partitioned into three conjugacy classes: one in the conjugacy class A of the point $(3, 0, 0, 0)$ in $\mathbf{A}_v^+$, one in the conjugacy class B of the point $(1, 1, 0, 0)$ in $\mathbf{A}_v^+$, and two in the conjugacy class C of the point $(0, 1, 0, 1)$ in $\mathbf{A}_v^+$. For a description of the quasi-conjugacy classes A, B, C see Example 2.31. These three quasi-conjugacy classes form a poset given by $A \preceq_c C$ and $B \preceq_c C$. By Theorem 4.9 we can examine different quasi-conjugacy classes independently.

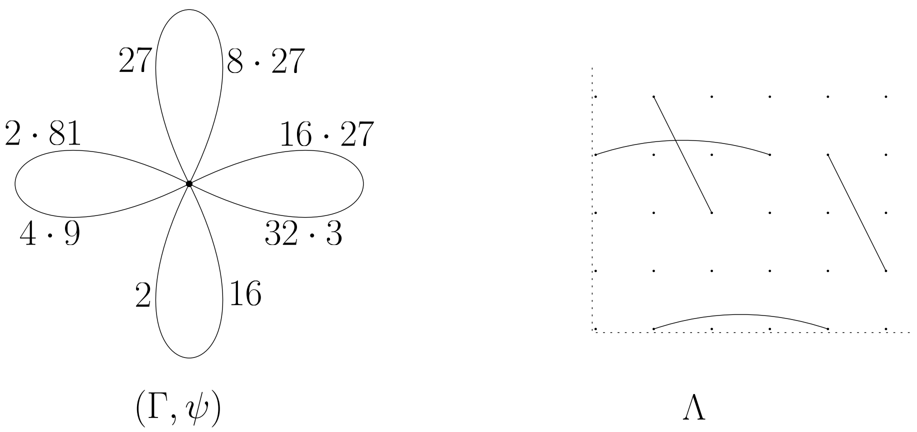

Figure 12: The GBS graph (Γ, ψ) and the affine representation Λ of Example 4.20.

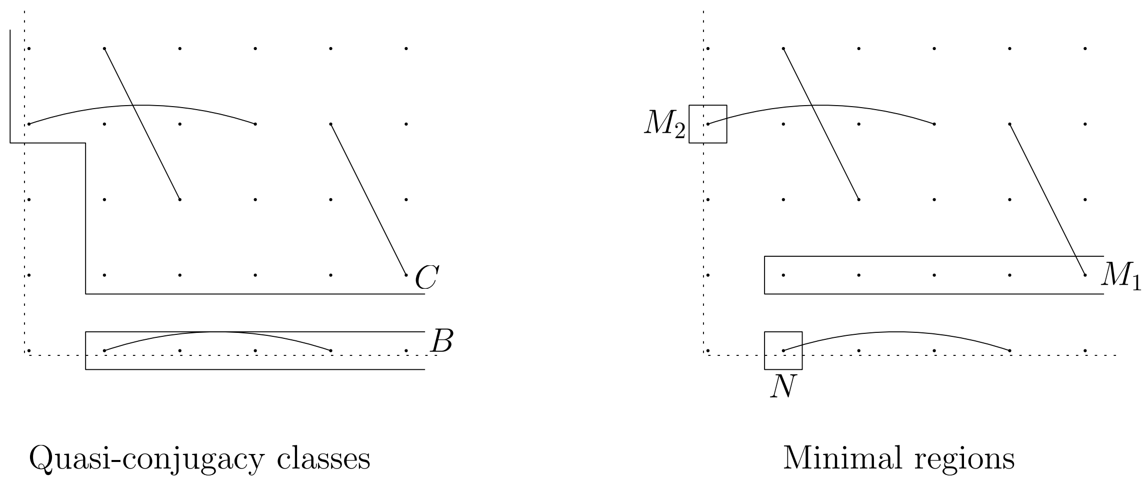

Figure 13: The quasi-conjugacy classes and the minimal regions for the GBS graph of Example 4.20.

The quasi-conjugacy class A only contains the edge $(3,0,0,0)$ —— $(1,0,1,0)$, so by Corollary 4.10 that edge can't be changed by any move. A similar analysis on B tells us that the edge $(1,1,0,0)$ —— $(0,1,1,0)$ will not change either.

The quasi-conjugacy class C has three minimal regions $M_1 = \{(1,0,0,1)\}$ and $M_2 = \{(0,1,0,1)\}$ and $M_3 = \{(a,1,c,0) : a + c \geq 2\}$. By Lemma 4.17, along any sequence of slides, swaps, connections, we will always have (at least) one endpoint in M_1, one endpoint in M_2, one endpoint in M_3. We only need to understand where the fourth endpoint might be. Consider the following list L of configurations:

1. For $a + c, a' + c' \geq 2$ the configurations

$$\begin{cases} (1,0,0,1) \text{ —— } (a,1,c,0) \\ (0,1,0,1) \text{ —— } (a',1,c',0) \end{cases}$$

2. For $a + c \geq 2$ and $b, d \geq 0$ the configurations

$$\begin{cases} (1,0,0,1) \text{ —— } (a,1,c,0) \\ (0,1,0,1) \text{ —— } (1+b,0,d,1) \end{cases}$$

3. For $a + c \geq 2$ and $b, d \geq 0$ the configurations

$$\begin{cases} (1,0,0,1) \text{ —— } (b,1,d,1) \\ (0,1,0,1) \text{ —— } (a,1,c,0) \end{cases}$$

It's easy to see that all the configurations in the list L can be reached from (Γ, ψ) by means of internal slide moves and external slide moves. Moreover, if we are at any configuration in L, then no swap or connection is possible, and by performing an (internal or external) slide we end up at another configuration in L. It follows that L is a complete list of all the configuration that can be obtained from (Γ, ψ) by means of slides, swaps, connections. If we are given a GBS graph (Δ, ϕ), and we want to know whether it encodes the same group as (Γ, ψ), then we just change (Δ, ϕ) into a totally reduced GBS graph (this can be done algorithmically), we perform sign-changes in all the possible (finitely many) ways, and we check whether one of the resulting graphs appears in the list L. Note that, since induction can't be performed on (Γ, ψ), we don't have to worry about it in this case.

Example 4.22. We can now describe the isomorphism problem for the GBS graph of Example 2.30. Using the same notation as in Example 2.30, call (Γ, ψ) the GBS graph, with a single vertex v, and Λ its affine representation. The edges of Λ are partitioned into four quasi-conjugacy classes C_1, C_2, C_3, C_4, see Figure 14, which had been described in Example 2.30. Suppose that we try and perform a sequence of slides, swaps, connections starting from this GBS graph.

The quasi-conjugacy class C_1 contains only the edge $(1,3)$ —— $(3,2)$, and thus by Corollary 4.10 that edge can't be changed. In the quasi-conjugacy classes C_3, C_4, the two notions of conjugacy and external equivalence coincide, and thus by Corollary 4.11 we can change the endpoints of the edges at will using external slides, and that's a complete list of all configurations that can be reached with slides, swaps, connections.

The quasi-conjugacy class C_2 contains two edges, belonging to two distinct conjugacy classes, and one minimal region $M_2 = \{(4,0)\}$; we have $\mathrm{lin}_{\mathrm{qc}}(C_2) = \langle (2,0) \rangle$. For simplicity of notation, call $\mathbf{a} = (4,0)$ and $\mathbf{u} = (2,0)$. Let's call e the edge belonging to the conjugacy class of $(4,0)$: by Lemma 4.17 the edge e must always be given by $(4,0)$ —— $(4+2a, 0)$ for some integer $a \geq 0$. If $a = 0$ then M_2 would be a quasi-conjugacy class by itself, contradiction, so we also always have $a \neq 0$. Let's call f the edge belonging to the other conjugacy class: the edge f must always be given by $(5+2b, 0)$ —— $(5+2c, 0)$ for some integers $b, c \geq 0$. Finally, we must always have that $\langle (2a, 0), (2b-2c, 0) \rangle = \langle (2,0) \rangle$, otherwise the linear algebra of C_2 wouldn't be the correct one. On the other hand, using the results of [ACRK25b], every configuration satisfying the above conditions can actually be reached. To summarize, the configurations for C_2, that we can reach with a sequence of slides, swaps, connections, are exactly the ones such that:

1. e is given by $(4,0)$ —— $(4+2a, 0)$ for some integer $a \geq 1$.
2. f is given by $(5+2b, 0)$ —— $(5+2c, 0)$ for some integers $b, c \geq 0$.
3. $(a, b-c) = 1$.

Finally, we take into account the (finitely many) sign-changes, and we observe that induction moves don't apply to this example. This gives algorithmic solution to the isomorphism problem for an arbitrary GBS graph against this specific one.

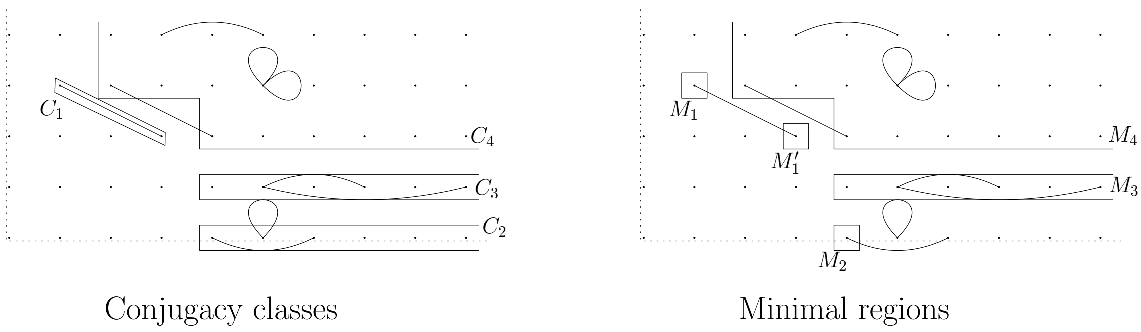

Figure 14: The conjugacy classes and the minimal regions for the GBS graph of Example 4.22.

5 GBSs with one qc-class and full-support gaps

The aim of this section is to classify GBS graphs with one qc-class and full-support gaps.

5.1 One qc-class

Definition 5.1. *Let (Γ, ψ) be a GBS graph with affine representation Λ. We say that (Γ, ψ) has* ***one qc-class*** *if all of its edges belong to the same quasi-conjugacy class.*

Note that the property of having one qc-class is an isomorphism invariant. In this case the external equivalence relation is trivial in that quasi-conjugacy class, and in particular the minimal regions are essentially points (to be formal, couples of points, with only the $\mathbb{Z}/2\mathbb{Z}$ component changing). *Along this section, we will abuse notation and talk about minimal points instead of minimal regions.*

In what follows, we will also assume that the quasi-conjugacy class has non-trivial quasi-conjugacy support: for a quasi-conjugacy class Q with $\mathrm{supp}_{\mathrm{qc}}(Q) = \emptyset$, the isomorphism problem can be solved using Corollary 4.12.

5.2 Graph of families

Let (Γ, ψ) be a totally reduced GBS graph with affine representation Λ. Suppose that (Γ, ψ) has one qc-class Q. Then we will consider the graph of families $\mathcal{F}(Q)$ as defined in Section 2.7. In particular we have the following properties:

1. Each minimal point of Q belongs to exactly one family. Each family contains at least one minimal point. Therefore, the finite set of minimal points of Q is partitioned into $|V(\mathcal{F}(Q))|$ non-empty subsets.

2. According to Lemma 2.23, we have exactly one edge in $\mathcal{F}(Q)$ for every edge in Γ. In other words, we have that $E(\mathcal{F}(Q)) = E(\Gamma) = E(\Lambda)$.

3. We have the modular homomorphism $q_Q : \pi_1(\mathcal{F}(Q)) \to \mathbf{A}$ as defined in Section 2.7.

Definition 5.2. *A family $F \in V(\mathcal{F}(Q))$ is called* ***individual*** *if it contains exactly one minimal point, and* ***non-individual*** *otherwise.*

Remark 5.3. In the particular case when $\mathrm{supp}_{\mathrm{qc}}(Q) = \mathcal{P}(\Gamma, \psi)$, the graph of families is exactly $\mathcal{F}(Q) = \Gamma$. For simplicity, the reader can go through this whole section with this extra hypothesis in mind (this extra assumption won't make any of the arguments easier).

5.3 One endpoint at each minimal point

Let (Γ, ψ) be a totally reduced GBS graph with affine representation Λ. Suppose that (Γ, ψ) has one qc-class Q with $\mathrm{supp}_{\mathrm{qc}}(Q) \neq \emptyset$.

Definition 5.4. *We say that (Γ, ψ) is* ***clean*** *if every minimal point contains exactly one endpoint of exactly one edge, and no edge has both endpoints at minimal points.*

If $\mathrm{supp}_{\mathrm{qc}}(Q) = \emptyset$, then every endpoint of every edge must lie at a minimal point, and thus the isomorphism problem is described through finitely many GBS graphs. If instead $\mathrm{supp}_{\mathrm{qc}}(Q) \neq \emptyset$, then we can arrange for the GBS graph to be clean, as we now explain.

Suppose that $\mathrm{supp}_{\mathrm{qc}}(Q) \neq \emptyset$. Define the graph $\mathcal{L}$ as follows:

- $V(\mathcal{L})$ is the set of minimal points of (Γ, ψ), together with an extra vertex v_{floating}.
- $E(\mathcal{L}) = E(\Lambda)$, with the same reverse map. For $e \in E(\Lambda)$, if $\tau_\Lambda(e)$ is a minimal point then we set $\tau_{\mathcal{L}}(e) = \tau_\Lambda(e)$, otherwise we set $\tau_{\mathcal{L}}(e) = v_{\mathrm{floating}}$.

Note that loops in $\mathcal{L}$ at v_{floating} correspond to floating edges in (Γ, ψ). Since (Γ, ψ) has one qc-class Q, we must have that $\mathcal{L}$ is connected. Note that (Γ, ψ) is clean if and only if in $\mathcal{L}$ every minimal point m has valence one, and the only edge at m connects m to v_{floating}.

We now define a *clean projection* from (Γ, ψ) to a clean GBS graph. Choose a maximal tree of $\mathcal{L}$. Define the GBS graph (Γ', ψ') as follows. If $e \in E(\mathcal{L})$ is an edge which doesn't lie in the maximal tree, then we slide $\tau(e)$ along the maximal tree until it ends up at v_{floating}. We do this for all endpoints of all edges that don't lie in the chosen maximal tree. Now, if $e \in E(\mathcal{L})$ lies in the maximal tree, we consider its two endpoints $\iota(e), \tau(e)$, we take the one that lies nearer to v_{floating}, and we slide it along the maximal tree until it ends up at v_{floating}. We do this for all edges that lie in the chosen maximal tree. The result of this procedure is a clean GBS graph (Γ', ψ').

Lemma 5.5. *The above clean projection has the following properties:*

1. *For a fixed maximal tree, performing the slide moves in a different order will produce the same clean GBS graph (Γ', ψ').*

2. *For different choices of maximal trees, producing clean GBS graphs $(\Gamma', \psi'), (\Gamma'', \psi'')$, there is a sequence of edge sign-changes and slides going from (Γ', ψ') to (Γ'', ψ'') and such that every GBS graph along the sequence is clean.*

Proof. This is an easy check with the definitions. □

Lemma 5.6. *Let $(\Gamma, \psi), (\Delta, \phi)$ be related by a edge sign-change, slide, swap or connection; let $(\Gamma', \psi'), (\Delta', \psi')$ be their respective clean projections. Then there is a sequence of edge sign-changes, slides, swaps, connections passing from (Γ', ψ') to (Δ', ψ') and going only through clean GBS graphs.*

Proof. This is an easy check with the definitions. □

Thanks to the above Lemma 5.5 and Lemma 5.6, for the rest of this section we will be able to assume that our GBS graphs are clean, and that all the sequences of moves go only through clean GBS graphs.

In particular, if (Γ, ψ) is a GBS graph with one qc-class Q, and if $\mathrm{supp}_{\mathrm{qc}}(Q) \neq \emptyset$, then a strong version of Lemma 4.17 holds. To be precise, up to taking the clean projection of (Γ, ψ), for every minimal point m we have exactly one edge with exactly one endpoint at m, and all the other edges are floating edges. In particular, a clean GBS graph (Γ, ψ) has at least as many edges as minimal points; moreover, the inequality is strict if and only if (Γ, ψ) has floating edges. This motivates the following definition.

Definition 5.7. *Let (Γ, ψ) be a GBS graph with one qc-class Q, and with $\mathrm{supp}_{\mathrm{qc}}(Q) \neq \emptyset$. We say that (Γ, ψ) has **floating pieces** if it has strictly more edges than minimal points.*

TO DO maybe it's more convenient to take some time here and explain the structure of the clean projection. Basically, all the rank between the minimal points is converted into loops based at the other endpoint of the "escaping edges". If the maximal tree contains several escaping edges for the same connected component of minimal points, then there will also be an edge connecting the two "escaping points". Maybe a figure might help.

5.4 One endpoint in each individual family

Let (Γ, ψ) be a totally reduced GBS graph with affine representation Λ. Suppose that (Γ, ψ) has one qc-class Q with $\mathrm{supp}_{\mathrm{qc}}(Q) \neq \emptyset$.

Definition 5.8. *If $F \in V(\mathcal{F}(Q))$ is an individual family, then we say that (Γ, ψ) is F**-polished** if F contains exactly one endpoint of one edge.*

Let $F \in V(\mathcal{F}(Q))$ be an individual family, and let m be the unique minimal point in F. A **set of F-polishing data** is given by pairwise distinct edges $e_1, \ldots, e_n, e$ satisfying the following properties:

1. For $j = 1, \ldots, n$ we have that e_j goes from $m + \mathbf{a}_j$ to $m + \mathbf{b}_j$.

2. For $j = 1, \dots, n$ we have $\mathrm{supp}(\mathbf{a}_j) \subseteq (\mathrm{supp}(\mathbf{b}_{j-1}) \cup \ldots \cup \mathrm{supp}(\mathbf{b}_1))$.

3. The edge e goes from $m + \mathbf{c}$ to p for some $p \ngeq m$.

4. We have $\mathrm{supp}(\mathbf{c}) \subseteq (\mathrm{supp}(\mathbf{b}_n) \cup \ldots \cup \mathrm{supp}(\mathbf{b}_1))$.

If p belongs to a family $F' \in V(\mathcal{F}(Q))$, then we say that the set of polishing data is **pointing to** F'. As in [ACRK25b] we can choose natural numbers $k_1, \dots, k_n$ big enough, and define

$$\mathbf{w}_j = (\mathbf{b}_j - \mathbf{a}_j) + k_{j-1}(\mathbf{b}_{j-1} - \mathbf{a}_{j-1}) + k_{j-1}k_{j-2}(\mathbf{b}_{j-2} - \mathbf{a}_{j-2}) + \cdots + k_{j-1}k_{j-2}\ldots k_1(\mathbf{b}_1 - \mathbf{a}_1)$$

for $j = 1, \dots, n$, and $\mathbf{w} = k_n\mathbf{w}_n - \mathbf{c}$. This allows us to define an F**-polishing procedure** that changes the edges

$$\begin{cases} m + \mathbf{a}_1 \text{ —— } m + \mathbf{b}_1 \\ m + \mathbf{a}_2 \text{ —— } m + \mathbf{b}_2 \\ m + \mathbf{a}_3 \text{ —— } m + \mathbf{b}_3 \\ \ldots \\ m + \mathbf{a}_n \text{ —— } m + \mathbf{b}_n \\ m + \mathbf{c} \text{ —— } p \end{cases} \quad \text{into} \quad \begin{cases} m + \mathbf{a}_1 \text{ —— } p + \mathbf{w} + \mathbf{a}_1 \\ p + \mathbf{w} + \mathbf{a}_2 \text{ —— } p + \mathbf{w} + \mathbf{a}_2 + \mathbf{w}_1 \\ p + \mathbf{w} + \mathbf{a}_3 \text{ —— } p + \mathbf{w} + \mathbf{a}_3 + \mathbf{w}_2 \\ \ldots \\ p + \mathbf{w} + \mathbf{a}_n \text{ —— } p + \mathbf{w} + \mathbf{a}_n + \mathbf{w}_{n-1} \\ p \text{ —— } p + \mathbf{w}_n \end{cases}$$

and each other endpoints of the form $m + \mathbf{d}$ into $p + \mathbf{w} + \mathbf{d}$.

In the same way as in [ACRK25b], we can see that for $k_1, \dots, k_n$ big enough the F-polishing procedure can be obtained through a sequence of moves, giving an F-polished GBS graph as a result.

Lemma 5.9. *For different choices of the set of F-polishing data and of big enough constants, producing F-polished GBS graphs $(\Gamma', \psi'), (\Gamma'', \psi'')$, there is a sequence of edge sign-changes, slides, swaps, connections going from (Γ', ψ') to (Γ'', ψ'') and such that every GBS graph along the sequence is F-polished.*

Proof. Analogous to [ACRK25b]. □

Lemma 5.10. *Let $(\Gamma, \psi), (\Delta, \psi)$ be related by an edge sign-change, slide, swap or connection; let $(\Gamma', \psi'), (\Delta', \phi')$ be corresponging F-polished GBS graphs. Then there is a sequence of edge sign-changes, slides, swaps, connections passing from (Γ', ψ') to (Δ', ψ') and going only through F-polished GBS graphs.*

Proof. Analogous to [ACRK25b]. □

Note that, if $F, G \in V(\mathcal{F}(Q))$ are two individual families, and if (Γ, ψ) is G-polished, then there must be a set of F-polishing data pointing to some $F' \neq G$ (unless F, G are the only families). If we use such a set of data to perform the above procedure, the result will be G-polished. Moreover, if we restrict only to set of data pointing to families $F' \neq G$, then Lemma 5.9 and Lemma 5.10 still hold, producing sequences of moves going only through G-polished GBS graphs. In particular, in what follows we can assume that all our GBS graphs are F-polished with respect to all individual families $F \in V(\mathcal{F}(Q))$, and we can consider only sequences of moves going through F-polished GBS graphs.

Remark 5.11. The F-polishing procedure is also compatible with the notion of clean. All of the above discussion, Lemma 5.9 and Lemma 5.10 still hold if we add the additional requirement that all GBS graphs appearing are clean. In particular, we can restrict our attention to GBS graphs which are clean and F-polished for all individual families $F \in V(\mathcal{F}(Q))$, and to sequences of moves going only through GBS graphs of this kind.

5.5 Full-support gaps

Let (Γ, ψ) be a GBS graph with affine representation Λ. Suppose that we are given two vertices $p, q \in V(\Lambda)$ with $q \geq p$: then we can write $q = p + \mathbf{w}$ for a unique $\mathbf{w} \in \mathbf{A}^+$, which we call the *gap* from p to q. We say that the gap from p to q is *full-support* if $\mathrm{supp}(\mathbf{w}) = \mathrm{supp}_{\mathrm{qc}}(p)$ (which in particular implies that $q \sim_{\mathrm{qc}} p$).

Definition 5.12. *Let (Γ, ψ) be a GBS graph and let Q be a quasi-conjugacy class. We say that Q has **full-support gaps** if for every edge e in Q we have that either $\tau(e)$ lies in a minimal region, or $\tau(e)$ lies above a minimal point of Q with full-support gap.*

This means that, whenever an end-point of an edge that falls strictly above a minimal region, the gap can be taken to have the full quasi-conjugacy support. Note that the property of being full-support gap isn't an isomorphism invariant, nor invariant under performing moves.

5.6 GBSs graphs with floating pieces

Let (Γ, ψ) be a totally reduced GBS graph with affine representation Λ. Suppose that (Γ, ψ) has one qc-class Q with $\mathrm{supp}_{\mathrm{qc}}(Q) \neq \emptyset$, and that (Γ, ψ) is clean. Suppose that (Γ, ψ) has floating pieces.

Fix a family $F_0 \in V(\mathcal{F}(Q))$. For every family $F \neq F_0$ we fix a minimal point m_F in that family.

Definition 5.13 (Normal form for GBSs with floating pieces)**.** *We say that (Γ, ψ) is in **normal form** (with respect to the family F_0 and the minimal points $\{m_F\}_{F \neq F_0}$) if the following conditions hold:*

1. *For every family $F \neq F_0$ and for the minimal point m_F, take the unique edge e with $\iota(e) = m_F$. We require that $\tau(e)$ is in F_0, and lies above all minimal points of F_0 with full-support gap.*

2. *For every family F and for every minimal point m (except m_F), take the unique edge e with $\iota(e) = m$. We require that $\tau(e)$ is in F, and lies above all minimal points of F with full-support gap.*

3. *For every floating edge e, we require that $\iota(e), \tau(e)$ are in F_0, and lie above all minimal points of F_0 with full-support gap.*

Proposition 5.14. *Suppose that (Γ, ψ) has full-support gaps. Then, for every family F_0 and minimal points $\{m_F\}_{F \neq F_0}$, we can bring (Γ, ψ) to normal form by means of a sequence of slides, swaps, connections.*

Proof. For every minimal point m, let e_m be the unique edge such that $\iota(e_m) = m$. We say that e_m is *positive* if $\tau(e_m)$ is above m with full-support gap.

In steps 1 and 2 we maximize the number of positive edges, and we observe that the remaining edges at the minimal points for a tree in the graph of families $\mathcal{F}(Q)$. In step 3 we change the tree, and we assume that all non-positive edges at the minimal points point at F_0. In step 4, we make all positive edges very long (taking advantage of floating edges). In step 5, we arrange for the non-positive edges to be at the prescribed minimal points m_F.

STEP 1: If e_m isn't positive, then we choose a minimal point $m' \neq m$ such that $\tau(e_m)$ is above m' with full-support gap. If $e_{m'}$ isn't positive, then we choose m'' such that $\tau(e_{m'})$ is above m'' with full-support gap; and so on. If the sequence doesn't terminate with a positive edge, then we find a cycle, and by sliding one edge of the cycle along the others, we increase the number of positive edges. Note that the hypothesis of having full-support gaps is preserved.

We reiterate this argument until, for every sequence $m, m', m'', \ldots,$ the sequence eventually ends at some $m^{(k)}$ with $e_{m^{(k)}}$ positive. At this point, we can slide $\tau(e_m)$ along $e_{m'}, e_{m''}, \ldots$ in order to get that $\tau(e_m)$ is above $m^{(k)}$ with full-support gap. Similarly for all the other non-positive edges

Thus we can assume that, for all minimal points m, either the edge e_m is positive, or $\tau(e_m)$ lies above some other minimal point m' with full-support gap and with $e_{m'}$ positive.

STEP 2: Now consider the graph of families $\mathcal{F}(Q)$, and remove the floating edges to get a graph $\mathcal{F}'$.

Suppose that $\mathcal{F}'$ contains an embedded (unoriented) closed cycle γ containing a non-positive edge e_m. Thus $\tau(e_m)$ lies above some minimal point m' with full-support gap and with $e_{m'}$ positive. By sliding $\tau(e_m)$ along $e_{m'}$ many times, we can assume that all components of $\tau(e_m)$ in $\operatorname{supp}_{\mathrm{qc}}(Q)$ are very big. We can then slide $\tau(e_m)$ along the other edges of γ until it becomes positive. This increases the number of positive edges.

By reiterating this procedure, we can assume that every embedded closed cycle in $\mathcal{F}'$ is a single loop consisting of a positive edge. This means that $\mathcal{F}'$ is a tree plus some loops at the vertices consisting of positive edges.

STEP 3: We now want all non-positive edges of $\mathcal{F}'$ to point towards a prescribed family F_0. As we said before, non-positive edges of $\mathcal{F}'$ form a tree. Suppose that e_m is a non-positive edge, and $\tau(e_m)$ is more far from F_0 than $\iota(e_m)$ (in the tree). Then we have that $\tau(e_m)$ lies above m' such that $e_{m'}$ is positive, and thus we can perform a connection move. This substitutes $e_m, e_{m'}$ with $f_m, f_{m'}$ respectively, where f_m is positive and $f_{m'}$ is non-positive; what we gain is that now $\tau(f_{m'})$ is more near to F_0 than $\iota(f_{m'})$. If some other non-positive edge $e_{m''}$ has $\tau(e_{m''})$ above m', then we now slide $\tau(e_{m''})$ along $f_{m'}$, in such a way that $\tau(e_{m''})$ ends up above m with positive edge f_m.

Thus we can assume that all non-positive edges point towards F_0. For every non-positive edge e_m, we have that $\tau(e_m)$ is above a positive edge, and thus up to slide we can make all components of $\tau(e_m)$ in $\operatorname{supp}_{\mathrm{qc}}(Q)$ very big, and then slide $\tau(e_m)$ so that it falls in F_0, and above all minimal points of F_0 with full-support gap.

Thus now $\mathcal{F}'$ is made of non-positive edges (one for each family $F \neq F_0$, and all pointing to F_0, and with terminal endpoint with all components in $\operatorname{supp}_{\mathrm{qc}}(Q)$ very big), and of positive edges at every other minimal point.

STEP 4: Up to slides, we can assume that every floating edge has both endpoints in F_0.

Let e_m be a positive edge given by m —— $m + \mathbf{u}$ for $\mathbf{u} \in \mathbf{A}^+$ with $\operatorname{supp}(\mathbf{u}) = \operatorname{supp}_{\mathrm{qc}}(Q)$. Take a floating edge e and, up to slides, assume that $\iota(e)$ lies above m and $\tau(e) = \iota(e) + \mathbf{w}$ for some $\mathbf{w} \in \mathbf{A}^+$ with $\operatorname{supp}(\mathbf{w}) = \operatorname{supp}_{\mathrm{qc}}(Q)$ and with all non-zero components of $\mathbf{w}$ very big. Then we perform a swap move, and we get an edge m —— $m + \mathbf{w}$ and a floating edge. In other words, we can assume that all components in $\operatorname{supp}_{\mathrm{qc}}(Q)$ of all positive edges are very big.

STEP 5: For a family $F \neq F_0$, take a prescribed minimal point m_F. If e_{m_F} is positive, then there is another minimal point m' with $e_{m'}$ non-positive. We have that $\tau(e_{m'})$ lies above some minimal point m'' in F_0, with $e_{m''}$ positive. We now perform a connection to change $e_{m'}, e_{m''}$ into $f_{m'}, f_{m''}$ with $f_{m'}$ positive. Then we perform a connection to change $e_{m_F}, f_{m''}$ into $g_{m_F}, g_{m''}$ with $g_{m''}$ positive. In this way, the edges at m', m'' are now positive, and the edge at m_F is not.

Thus the resulting GBS graph is in normal form, as desired. $\square$

Proposition 5.15. *Suppose that $(\Gamma, \psi), (\Delta, \phi)$ are in normal form. Suppose that they have the same $\operatorname{min}_{\mathrm{qc}}(Q), \operatorname{supp}_{\mathrm{qc}}(Q), \operatorname{lin}_{\mathrm{qc}}(Q)$ and the same number of edges in each conjugacy class (up to edge sign-changes). Then there is a sequence of edge sign-changes, slides, swaps, connections going from one to the other.*

Proof. Let Λ be the affine representation of (Γ, ψ). We have the edges m_F —— c_F for $F \neq F_0$ and $c_F \in F_0$. We can easily assume that all the other edges are given by p_i —— $p_i + \mathbf{x}_i$ with $\mathbf{x}_i \in \mathbf{A}^+$ and $\operatorname{supp}(\mathbf{x}_i) = \operatorname{supp}_{\mathrm{qc}}(Q)$, for $i = 1, \dots, r$.

It's fairly easy to see that, by means of swap moves, and up to translating the floating edges with slide moves, we can permute the vectors $\mathbf{x}_1, \dots, \mathbf{x}_r$ as we want. Once we are able to permute them as we want, we can also perform Nielsen moves among them (since we are able, for example, to change a floating edge by slide over another edge), provided that we keep all components in $\operatorname{supp}_{\mathrm{qc}}(Q)$ of all $\mathbf{x}_i$ big enough, so that the graph remains in normal form.

By Proposition 2.25 we have that $\mathbf{x}_1, \dots \mathbf{x}_r$ generate $\operatorname{lin}_{\mathrm{qc}}(Q)$. But $\operatorname{lin}_{\mathrm{qc}}(Q)$ is isomorphic to either $\mathbb{Z}^k$ or $\mathbb{Z}/2\mathbb{Z} \oplus \mathbb{Z}^k$ for some $k \in \mathbb{N}$. Thus in this case all r-tuples of generators for $\operatorname{lin}_{\mathrm{qc}}(Q)$ are Nielsen equivalent to each other up to permutation. Therefore, using the results of Appendix B, we can perform moves to make $\mathbf{x}_1, \dots, \mathbf{x}_r$ to be the same in (Γ, ψ) and (Δ, ϕ).

Now, the p_i which are minimal points are already the same in (Γ, ψ) and (Δ, ϕ), by assumption (up to performing some edge sign-change). The p_i which are non-minimal can be made to be the same by slides and self-slides (Lemma 3.2) and permutations of $\mathbf{x}_1, \dots, \mathbf{x}_r$, since we are assuming that there are the same number of edges in each conjugacy class in (Γ, ψ) and (Δ, ϕ).

Finally, we can change the endpoints c_F by any combination of $\mathbf{x}_1, \dots, \mathbf{x}_r$ (since we can slide c_F over at least one $\mathbf{x}_i$, and we can permute $\mathbf{x}_1, \dots, \mathbf{x}_r$ as we want in any moment). Since (Γ, ψ) and (Δ, ϕ) have the same conjugacy classes, we can arrange for the c_F to be the same too.

The conclusion follows. □

5.7 Rigid vectors

Let (Γ, ψ) be a totally reduced GBS graph with affine representation Λ. Suppose that (Γ, ψ) has one qc-class Q with $\mathrm{supp}_{\mathrm{qc}}(Q) \neq \emptyset$, and that (Γ, ψ) is clean. Suppose that (Γ, ψ) doesn't have floating pieces.

Definition 5.16. *A* ***rigid cycle*** *is a sequence of $\ell \geq 1$ edges m_1 —— $m_2 + \mathbf{w}_1$, m_2 —— $m_3 + \mathbf{w}_2$, ..., m_ℓ —— $m_1 + \mathbf{w}_\ell$ with the following properties:*

1. *$m_1, \dots, m_\ell$ are pairwise distinct minimal points.*

2. *$\mathbf{w}_1, \dots, \mathbf{w}_\ell \in \mathbf{A}^+$.*

3. *There are no distinct minimal points m, m' such that $m + \mathbf{w}_1 + \cdots + \mathbf{w}_\ell \geq m'$.*

For a rigid cycle we define the associated ***rigid vector*** *as $\mathbf{w} = \mathbf{w}_1 + \cdots + \mathbf{w}_\ell \in \mathbf{A}^+$.*

In the above hypothesis, we must have that $\mathrm{supp}(\mathbf{w}) \neq \emptyset$ since (Γ, ψ) is clean. In a rigid cycle, every endpoint of every edge lies above a unique minimal point. In fact, if $m_{i+1} + \mathbf{w}_i \geq m'$ for some minimal point m', then in particular $m_{i+1} + \mathbf{w} \geq m'$, and by Item 3 of Definition 5.16 this implies that $m' = m_{i+1}$. In particular, an edge can be part of at most one rigid cycle. Note also that $\mathbf{w} \in \mathrm{lin}_{\mathrm{qc}}(Q)$, since we can, for example, slide one edge of the rigid cycle along the others to obtain an edge m —— $m + \mathbf{w}$.

Definition 5.17. *Define the set of* ***rigid vectors*** *of (Γ, ψ) as the multi-set of the vectors $\mathbf{w}$ associated with the rigid cycles of (Γ, ψ).*

Proposition 5.18. *The multi-set of rigid vectors of (Γ, ψ) is invariant under edge sign-changes, slides, swaps, connections.*

Proof. Immediate check with the definitions. □

5.8 GBS graphs with no floating pieces

Let (Γ, ψ) be a totally reduced GBS graph with affine representation Λ. Suppose that (Γ, ψ) has one qc-class Q with $\mathrm{supp}_{\mathrm{qc}}(Q) \neq \emptyset$, and that (Γ, ψ) is clean. Suppose that (Γ, ψ) doesn't have floating pieces.

We denote with $\mathcal{M}$ the finite set of minimal points of Q. This is partitioned into the non-empty sets $\mathcal{M}_F$ for $F \in V(\mathcal{F}(Q))$, according to which family a given minimal point belongs. We give an orientation to each edge of Λ: we say that e is the *oriented edge* of a pair $e, \bar{e} \in E(\Lambda)$ if $\iota(e)$ is a minimal point. Note that exactly one of $e, \bar{e}$ is the oriented edge. Since (Γ, ψ) is clean and doesn't have floating pieces, we have a bijection between the set of oriented edges and $\mathcal{M}$: the oriented edge e corresponds to the minimal point $\iota(e) \in \mathcal{M}$. Recall that $E(\mathcal{F}(Q)) = E(\Lambda)$, and in particular $\mathcal{F}(Q)$ is an oriented graph. We fix a family $F_0 \in V(\mathcal{F}(Q))$, which will play the role of basepoint.

Choose a maximal tree T for $\mathcal{F}(Q)$, and let $t_1, \dots, t_h$ the oriented edges in T. Let $\mathbf{R} = \{\mathbf{r}_1, \dots, \mathbf{r}_k\}$ be the multi-set of rigid vectors as in Definition 5.17. For $i = 1, \dots, k$, choose an oriented edge e_i in the rigid cycle corresponding to $\mathbf{r}_i$, and such that e_i doesn't belong to T. Note that such an e_i exists, since the rigid cycle is a cycle in $\mathcal{F}(Q)$; note that distinct rigid cycles will give distinct edges, as they have no edges in common. Call $f_1, \dots, f_\ell$ the remaining oriented edges. This means that $e_1, \dots, e_k, f_1, \dots, f_\ell, t_1, \dots, t_h$ is a list of all the oriented edges of $\mathcal{F}(Q)$, each appearing exactly once. In particular $\mathcal{M} = \{\iota(e_1), \dots, \iota(e_k), \iota(f_1), \dots, \iota(f_\ell), \iota(t_1), \dots, \iota(t_h)\}$.

For $i = 1, \dots, \ell$ we consider the path σ_i which goes through f_i and then back to the starting point through the maximal tree T. For $i = 1, \dots, \ell$ we define $\mathbf{x}_i = q_Q(\sigma_i) + \langle \mathbf{R} \rangle \in \frac{\mathrm{lin}_{\mathrm{qc}}(Q)}{\langle \mathbf{R} \rangle}$. We consider the tree T with base-point F_0: every edge has an orientation, which is either pointing towards F_0, or away from F_0. We call $\epsilon \in \mathbb{N}$ the number of edges pointing away from F_0.

We define the map $\Theta : \mathcal{M} \to \mathbf{R} \sqcup \frac{\mathrm{lin}_{\mathrm{qc}}(Q)}{\langle \mathbf{R} \rangle} \sqcup (V(\mathcal{F}(Q)) \setminus \{F_0\})$ as follows:

1. $\Theta(\iota(e_i)) = \mathbf{r}_i \in \mathbf{R}$ for $i = 1, \dots, k$.

2. $\Theta(\iota(f_i)) = \mathbf{x}_i \in \frac{\mathrm{lin}_{\mathrm{qc}}(Q)}{\langle \mathbf{R} \rangle}$ for $j = 1, \dots, \ell$.

3. $\Theta(\iota(t_i)) \in V(\mathcal{F}(Q)) \setminus \{F_0\}$ is defined as follows. If t_i is pointing towards F_0, we set $\Theta(\iota(t_i))$ as the family of $\iota(t_i)$. If t_i is pointing away from F_0, we set $\Theta(\iota(t_i))$ as the family of $\tau(t_i)$.

We define the map $(-1)^\epsilon\Theta$ by pre-composing Θ with any permutation of $\mathcal{M}$ of sign $(-1)^\epsilon$.

Definition 5.19. *We define the* ***role-assignment map*** *of* (Γ, ψ) *as the map* $(-1)^\epsilon\Theta$*, considered up to*

1. *Nielsen equivalence of* $\mathbf{x}_1, \dots, \mathbf{x}_\ell$ *in the group* $\frac{\mathrm{lin}_{\mathrm{qc}}(Q)}{\langle \mathbf{R} \rangle}$.

2. *Pre-composing* Θ *with an even permutation of* $\mathcal{M}$.

The above construction can look artificial, but it becomes more natural if one thinks about the normal form introduced below, see Definition 5.23. We encourage the reader to think like this: for an oriented edge e with $\iota(e) = m$ minimal, the map $\Theta(m)$ is telling us what's the role of the edge e in the given configuration. If $\Theta(m) = \mathbf{r} \in \mathbf{R}$, then you should imagine that e is hosting the rigid vector $\mathbf{r}$ (as if e would be m —— $m + \mathbf{r}$). If $\Theta(m) = \mathbf{x} \in \frac{\mathrm{lin}_{\mathrm{qc}}(Q)}{\langle \mathbf{R} \rangle}$, then you should imagine that e is hosting a very long vector $\mathbf{x}$ (as if e would be m —— $m + \mathbf{x}$). If $\Theta(m) = F \in V(\mathcal{F}(Q)) \setminus \{F_0\}$, then you should imagine that e is the edge connecting F to the basepoint F_0 (as if e would be connecting a minimal point in F to a non-minimal point in F_0). And in fact, when (Γ, ψ) is in the normal form of Definition 5.23, this is exactly what the map Θ is encoding.

Lemma 5.20. *A different choice of the edges in the rigid cycles produces the same role-assignment map.*

Proof. Let γ be a rigid cycle with rigid vector $\mathbf{r}$, let T be a maximal tree, and let $s_1, s_2, \dots, s_h$ be the oriented edges of γ that don't belong to T.

For $i = 1, \dots, k$, we call σ_i the path that crosses s_i and then closes through the maximal tree T, and let $\mathbf{x}_i = q_Q(\sigma_i) + \langle \mathbf{R} \rangle \in \mathrm{lin}_{\mathrm{qc}}(Q)/\langle \mathbf{R} \rangle$. The key observation is that $\mathbf{x}_1 + \dots + \mathbf{x}_k = \mathbf{r}$.

It follows that, if we choose s_i to be the edge of the rigid cycle, then we must set $\Theta(\iota(s_i)) = \mathbf{r}$ and $\Theta(\iota(s_j)) = \mathbf{x}_j$ for $j \neq i$. The other values of Θ are independent of i. In particular, a different choice of i will produce the same map Θ up to even permutation and Nielsen equivalence. □

Lemma 5.21. *A different choice of maximal tree T produces the same role-assignment map.*

Proof. Take an embedded cycle γ in $\mathcal{F}(Q)$, and call $\mathbf{x} = q_Q(\gamma) + \langle \mathbf{R} \rangle \in \mathrm{lin}_{\mathrm{qc}}(Q)/\langle \mathbf{R} \rangle$. Suppose that $\gamma = \dots s_1^{\eta_1} \dots s_2^{\eta_2} \dots$ for some oriented edges s_1, s_2 and for some $\eta_1, \eta_2 = \pm 1$. Suppose that some maximal tree T_1 contains all of γ except s_2, and let T_2 be the maximal tree obtained from T_1 by adding s_2 and removing s_1. Let Θ_1, Θ_2 be the maps associated with T_1, T_2 respectively.

For an edge e not appearing in γ, we have two cases. If $e \in T_1$ then $e \in T_2$ and $\Theta_1(\iota(e)) = \Theta_2(\iota(e))$ and the contribution to ϵ doesn't change. If $e \notin T_1$ then $e \notin T_2$ and $\Theta_1(\iota(e)) - \Theta_2(\iota(e))$ is $\mathbf{0}$ or $\pm\mathbf{x}$.

For the edges e appearing in γ, the values of Θ_1, Θ_2 will be $\pm\mathbf{x}$ and $F_1, F_2, \dots, F_r$ (the vertices appearing in γ, except the one nearest to F_0), in some order. When changing from T_1 from T_2, these values get permuted (the permutation consisting of a single cycle), the contributions to ϵ might change, and $\pm\mathbf{x}$ might change sign. It's a routine check that these changes compensate each other, i.e. $\pm\mathbf{x}$ changes sign if and only if the sign of the permutation of the values, plus the sum of the contributions to ϵ, is negative.

Thus we can change from T_1 to T_2 without affecting the role-assignment map. But with changes of this kind, we can go from any tree to any other tree. The conclusion follows. □

Lemma 5.22. *The role-assignment map of (Γ, ψ) is invariant under edge sign-changes, slides, connections.*

Proof. Immediate check with the definitions (using Lemma 5.20 and Lemma 5.21). $\square$

We now define a normal form for GBS graphs with no floating pieces. Fix a non-individual family $F_0 \in V(\mathcal{F}(Q))$. For every family $F \neq F_0$ we fix a minimal point m_F in that family.

Definition 5.23 (Normal form for GBSs with no floating pieces)**.** *Suppose that (Γ, ψ) has full-support gaps. We say that (Γ, ψ) is in **normal form** (with respect to the non-individual family F_0 and the minimal points $\{m_F\}_{F \neq F_0}$) if the following conditions hold:*

1. *For every family $F \neq F_0$ and for the minimal point m_F, take the unique edge e with $\iota(e) = m_F$. We require that $\tau(e)$ is in F_0, and lies above all minimal points of F_0 with full-support gap.*
2. *For every family F and for every minimal point m in F (except possibly m_F), take the unique edge e with $\iota(e) = m$. We require that $\tau(e)$ is in F. If e isn't a rigid cycle, we also require that $\tau(e)$ lies above all minimal points of F with full-support gap.*

Proposition 5.24. *Suppose that (Γ, ψ) has full-support gaps. Then, for every non-individual family F_0 and minimal points $\{m_F\}_{F \neq F_0}$, we can bring (Γ, ψ) to normal form by means of a sequence of slides, swaps, connections.*

Proof. For every minimal point m, let e_m be the unique edge such that $\iota(e_m) = m$. We say that e_m is *positive* if $\tau(e_m)$ is above m with full-support gap.

The first three steps in the proof of Proposition 5.14 apply here. Therefore, we can assume that every edge is positive, except for the edges e_{m_F} for $F \neq F_0$, for which $\tau(e_{m_F})$ lies in F_0 and above all minimal points of F_0 with full-support gap.

Let $F \neq F_0$ be a family containing at least two minimal points m, m_F, and let m_1, m_2 be two minimal points in F_0. Then we have edges $m_1 —— m_1 + \mathbf{x}$ and $m_2 —— m_2 + \mathbf{y}$ and $m —— m + \mathbf{z}$ and $m_F —— p$ with p above both m_1, m_2. Then we can perform connections to change

$$\begin{cases} m_1 —— m_1 + \mathbf{x} \\ m_2 —— m_2 + \mathbf{y} \\ m —— m + \mathbf{z} \\ m_F —— p \end{cases} \text{into} \begin{cases} m_1 —— q \\ m_2 —— m_2 + \mathbf{y} \\ m —— m + \mathbf{z} \\ m_F —— m_F + \mathbf{x} \end{cases} \text{into} \begin{cases} m_1 —— m_1 + \mathbf{z} \\ m_2 —— m_2 + \mathbf{y} \\ m —— p' \\ m_F —— m_F + \mathbf{x} \end{cases}$$

$$\text{into} \begin{cases} m_1 —— m_1 + \mathbf{z} \\ m_2 —— q' \\ m —— m + \mathbf{y} \\ m_F —— m_F + \mathbf{x} \end{cases} \text{into} \begin{cases} m_1 —— m_1 + \mathbf{z} \\ m_2 —— m_2 + \mathbf{x} \\ m —— m + \mathbf{y} \\ m_F —— p'' \end{cases}$$

ultimately getting a cyclic permutation of $\mathbf{x}, \mathbf{y}, \mathbf{z}$ (here p, p', p'' lie in F_0 above m_1, m_2, while q, q' lie in F above m, m_F).

Now suppose that some positive edge e_m given by $m —— m + \mathbf{w}$ isn't a rigid edge, but $m + \mathbf{w}$ doesn't lie above all minimal points in its family. Then we can find two minimal points m', m'' such that $m' + \mathbf{w} \geq m''$. By means of cyclic permutations as explained above, we can bring $\mathbf{w}$ to F_0, and then to m', so that we end up with an edge $m' —— m' + \mathbf{w}$. Using that $m' + \mathbf{w} \geq m''$, we can perform slides and make all components of $\mathbf{w}$ in $\mathrm{supp}_{\mathrm{qc}}(Q)$ very big. Finally, by means of other cyclic permutations, we bring $\mathbf{w}$ back to m.

By reiterating this procedure, we eventually obtain a GBS graph in normal form, as desired. $\square$

Proposition 5.25. *Suppose that $(\Gamma, \psi), (\Delta, \phi)$ are in normal form. Suppose that the following conditions hold:*

1. *$(\Gamma, \psi), (\Delta, \phi)$ have the same $\min_{\mathrm{qc}}(Q), \mathrm{supp}_{\mathrm{qc}}(Q), \mathrm{lin}_{\mathrm{qc}}(Q)$ and the same number of edges in each conjugacy class (up to edge sign-changes).*

2. $(\Gamma, \psi), (\Delta, \phi)$ *have the same multi-set of rigid edges and the same role-assignment map.*

3. *Either* F_0 *contains at least three minimal points, or there is at least another non-individual family besides* F_0.

Then there is a sequence of edge sign-changes, slides, swaps, connections going from one to the other.

Proof. Up to edge sign-changes, we can assume that the minimal points are exactly the same.

As in the proof of Proposition 5.24, we can perform cyclic permutations on the vectors at the edges, by means of connections. In particular, we can assume that all rigid edges are at the same minimal points in (Δ, ϕ) as in (Γ, ψ).

If we have a (possibly rigid) edge m —— $m + \mathbf{x}$ and a non-rigid edge m' —— $m' + \mathbf{y}$, then with a sequence of slides we can change $\mathbf{y}$ into $\mathbf{y} \pm \mathbf{x}$, provided that all components in $\mathrm{supp}_{\mathrm{qc}}(Q)$ of $\mathbf{y}$ remain big enough. Now we deal with two cases: if there is at least one rigid edge, then we want to use Proposition B.4; if there is no rigid edge, then by hypothesis there are at least 3 positive vectors, and we can use Theorem B.1. In both cases, since $(\Gamma, \psi), (\Delta, \phi)$ have the same linear invariant, we can obtain that the values of $\mathbf{x}$ are the same in (Δ, ϕ) as in (Γ, ψ).

Finally, for $F \neq F_0$ and for the edge m_F —— c_F, we can change c_F to any other point in the same conjugacy class, by means of slide moves (and connections, if we would need to add/subtract some vector which lies in F). In particular, we can set also these to be the same in (Δ, ϕ) as in (Γ, ψ). The conclusion follows. □

5.9 The algorithm

Theorem 5.26. *There is an algorithm that, given two GBS graph* $(\Gamma, \psi), (\Delta, \phi)$ *with one qc-class and full-support gaps, decides whether there is a sequence of edge sign-changes, slides, swaps, connections going from one to the other. In case such a sequence exists, the algorithm also computes one such sequence.*

Proof. First of all, we check that the conjugacy class Q has the same $\min_{\mathrm{qc}}(Q), \mathrm{supp}_{\mathrm{qc}}(Q), \mathrm{lin}_{\mathrm{qc}}(Q)$ and the same number of edges in each conjugacy class (up to edge sign-changes), otherwise by Corollary 4.4 we can't change (Γ, ψ) into (Δ, ϕ).

If (Γ, ψ) has floating pieces, then (Δ, ϕ) must have floating pieces too. We make both of them clean and we compute the linear invariant: this must be the same, otherwise there is no sequence of moves going from one to the other. If they have the same linear invariant, then we bring both of them in normal form as in Proposition 5.14, and then we can go from one to the other by Proposition 5.15.

If (Γ, ψ) has no floating pieces, but it has a family with at least three minimal points, or at least two families with at least two minimal points each. Then (Δ, ϕ) must have the same property (as it has the same families and minimal points). We make both of them clean and we compute the multi-set of rigid edges and the linear invariant: these must be the same, otherwise there is no sequence of moves going from one to the other. If they have the same multi-set of rigid edges and the same linear invariant, then we bring both of them in normal form as in Proposition 5.24, and then we can go from one to the other by Proposition 5.25.

Finally, suppose that (Γ, ψ) has no floating pieces, and all families of (Γ, ψ) are individual, except at most one with at most two minimal points; we call F_0 such family. Note that (Δ, ϕ) must have the same property (as it has the same families and minimal points). In this case, we make both (Γ, ψ) and (Δ, ϕ) clean and F-polished for each family $F \neq F_0$, as in Section 5.4. From now on, we only deal with sequences of moves involving GBS graphs which are F-polished for each $F \neq F_0$.

For every family $F \neq F_0$, we take the unique minimal point m_F in that family, and the unique edge m_F —— c_F with $c_F \in F_0$, and we observe that c_F can be changed to any other point in the same conjugacy class, by means of slide moves only; in particular, in any moment, we can arrange them to be the same in (Γ, ψ) as in (Δ, ϕ). Let $(\Gamma', \psi'), (\Delta', \phi')$ be the GBS graphs obtained from $(\Gamma, \psi), (\Delta, \phi)$ respectively, by removing all edges with endpoints outside F_0. Then there is a sequence of moves going from (Γ, ψ) to (Δ, ϕ) if and only if there is a sequence of moves going from (Γ', ψ') to (Δ', ϕ').

But $(\Gamma',\psi'),(\Delta',\phi')$ are GBS graphs with one vertex and at most two edges each. The conclusion thus follows from [?] TO DO fix reference. □

Corollary 5.27. *There is an algorithm that, given two GBS graph $(\Gamma,\psi),(\Delta,\phi)$ with one qc-class and full-support gaps, decides whether the corresponding GBS groups are isomorphic or not. In case they are, the algorithm also computes a sequence of sign-changes, inductions, slides, swaps, connections going from (Γ,ψ) to (Δ,ϕ).*

Proof. By Theorem 5.26, and since vertex sign-changes can be guessed in finitely many ways, we only have to deal with induction.

But if (Γ,ψ) allows for induction, is totally reduced, and has one qc-class, then it must have just one vertex, and by means of a sequence of moves we can make it into a controlled GBS graph. In this case, the thesis follows from [ACRK25b]. □

5.10 Examples

As usual, in the examples we use $\mathbf{A}=\mathbb{Z}^{\mathcal{P}(\Gamma,\psi)}$, omitting the $\mathbb{Z}/2\mathbb{Z}$ summand.

Example 5.28. Consider a GBS graph (Γ,ψ) with affine representation Λ as in Figure 15. This means that Γ has three vertices v_1,v_2,v_3 and set of primes $\mathcal{P}(\Gamma,\psi)$ containing two prime numbers. The affine representation Λ consists of three copies $\mathbf{A}^+_{v_1},\mathbf{A}^+_{v_2},\mathbf{A}^+_{v_3}$ of $\mathbb{Z}^2$, and of edges

$$\begin{cases}(v_1,(11,0)) \text{ —— } (v_1,(21,8))\\ (v_1,(0,11)) \text{ —— } (v_1,(14,9))\\ (v_2,(10,1)) \text{ —— } (v_3,(4,10))\\ (v_2,(0,12)) \text{ —— } (v_1,(15,15))\\ (v_3,(2,6)) \text{ —— } (v_2,(15,4))\end{cases}$$

First, we compute the basic invariants of (Γ,ψ). It's easy to check that all the edges belong to a common quasi-conjugacy class Q, with five minimal points $\mathbf{m}_1=(v_1,(11,0))$ and $\mathbf{m}_2=(v_1,(0,11))$ and $\mathbf{m}_3=(v_2,(10,1))$ and $\mathbf{m}_4=(v_2,(0,12))$ and $\mathbf{m}_5=(v_3,(2,6))$. A computation shows that $\mathrm{supp}_{\mathrm{qc}}(Q)=\mathcal{P}(\Gamma,\psi)$ and $\mathrm{lin}_{\mathrm{qc}}(Q)=\langle(1,0),(0,1)\rangle$, and in particular Q is also a conjugacy class. With this notation, Λ can be rewritten as

$$\begin{cases}\mathbf{m}_1 \text{ —— } \mathbf{m}_1+(10,8)\\ \mathbf{m}_2 \text{ —— } \mathbf{m}_2+(14,9)\\ \mathbf{m}_3 \text{ —— } \mathbf{m}_5+(2,5)\\ \mathbf{m}_4 \text{ —— } \mathbf{m}_2+(15,4)\\ \mathbf{m}_5 \text{ —— } \mathbf{m}_3+(4,3)\end{cases}$$

The graph of families $\mathcal{F}(Q)$ coincides with Γ in this case. We note that (Γ,ψ) is already clean, and has no floating pieces, so we can assign an orientation to each edge, in such a way that edges are always oriented going out of minimal points. We can see that (Γ,ψ) has a unique rigid cycle, given by the edges $\mathbf{m}_5$ —— $\mathbf{m}_3+(4,3)$ and $\mathbf{m}_3$ —— $\mathbf{m}_5+(2,5)$, with corresponding rigid vector $\mathbf{r}=(7,7)$.

We now bring (Γ,ψ) to normal form. First, we maximize the number of positive edges, by performing a slide and a connection, to get the configuration

$$\begin{cases}\mathbf{m}_1 \text{ —— } \mathbf{m}_1+(10,8)\\ \mathbf{m}_2 \text{ —— } \mathbf{m}_4+(13,14)\\ \mathbf{m}_3 \text{ —— } \mathbf{m}_3+(7,7)\\ \mathbf{m}_4 \text{ —— } \mathbf{m}_4+(14,9)\\ \mathbf{m}_5 \text{ —— } \mathbf{m}_3+(4,3)\end{cases}$$

as in Figure 16. This is already near to being a normal form, as all vectors are positive, except the two going from $\mathbf{A}^+_{v_1},\mathbf{A}^+_{v_3}$ to $\mathbf{A}^+_{v_2}$. However, the two positive vectors at $\mathbf{m}_1,\mathbf{m}_4$, and the endpoints of the edge at $\mathbf{m}_5$, have an endpoint which is not high enough. Thus we have to perform further manipulations to reach, for example, the configuration of Figure 17, given by

$$\begin{cases} \mathbf{m}_1 \text{ —— } \mathbf{m}_1 + (7,7) \\ \mathbf{m}_2 \text{ —— } \mathbf{m}_4 + (13,14) \\ \mathbf{m}_3 \text{ —— } \mathbf{m}_3 + (12,13) \\ \mathbf{m}_4 \text{ —— } \mathbf{m}_4 + (17,15) \\ \mathbf{m}_5 \text{ —— } \mathbf{m}_4 + (20,5) \end{cases}$$

where the rigid vector has been moved at $\mathbf{m}_1$, and all the other endpoints are high enough above the minimal points.

For the normal form of Figure 17, the linear invariant $(-1)^\epsilon\Theta : \mathcal{M} \to \{\mathbf{r}\} \sqcup \frac{\mathrm{lin}_{\mathrm{qc}}(Q)}{\langle \mathbf{r} \rangle} \sqcup \{v_1, v_3\}$ is given by $\Theta(\mathbf{m}_1) = \mathbf{r}$ and $\Theta(\mathbf{m}_2) = v_1$ and $\Theta(\mathbf{m}_3) = (12, 13)$ and $\Theta(\mathbf{m}_4) = (17, 15)$ and $\Theta(\mathbf{m}_5) = v_3$. This means that the minimal points $\mathbf{m}_2, \mathbf{m}_5$ are hosting the edges connecting $\mathbf{A}^+_{v_1}, \mathbf{A}^+_{v_3}$ to $\mathbf{A}^+_{v_2}$ respectively, the minimal point $\mathbf{m}_1$ is hosting the rigid vector $\mathbf{r}$, and the minimal points $\mathbf{m}_3, \mathbf{m}_4$ are hosting the vectors $(12, 13), (17, 15)$ of the abelian group $\frac{\mathrm{lin}_{\mathrm{qc}}(Q)}{\langle \mathbf{r} \rangle} \cong \mathbb{Z} \oplus \mathbb{Z}/7\mathbb{Z}$. Note that the couple $(12, 13), (17, 15)$ is Nielsen equivalent to the couple $(1, 0), (0, 1)$ in the abelian group $\frac{\mathrm{lin}_{\mathrm{qc}}(Q)}{\langle \mathbf{r} \rangle}$.

In $\mathbb{Z}\oplus\mathbb{Z}/7\mathbb{Z}$ there are exactly 6 Nielsen equivalence classes of couples of generators, two couples being Nielsen equivalent if and only if the matrix of change of basis has determinant $\equiv 1$ modulo 7 (see Proposition A.5). For example, if in Figure 17 we substitute the edge $\mathbf{m}_4$ —— $\mathbf{m}_4 + (17, 15)$ with $\mathbf{m}_4$ —— $\mathbf{m}_4 + (17, 14)$ while leaving the others unchanged, then we obtain a non-isomorphic GBS group. In fact, $(12, 13), (17, 14)$ is Nielsen equivalent to $(3, 0), (0, 1)$, and the matrix of change of basis from $(1, 0), (0, 1)$ to $(3, 0), (0, 1)$ has determinant $\equiv 3 \not\equiv 1$ modulo 7.

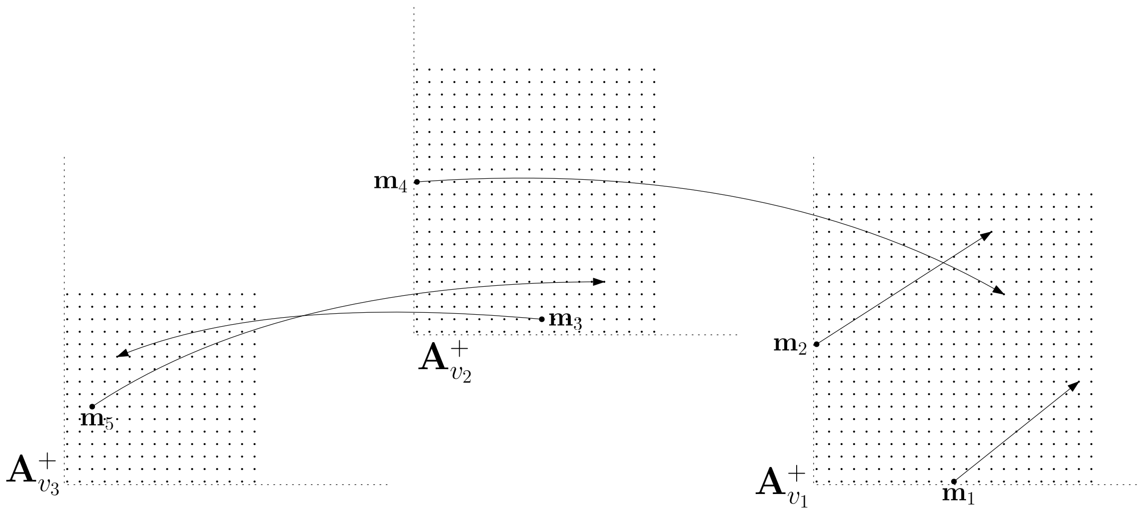

Figure 15: The affine representation for the GBS graph of Example 5.28. The edges have been assigned orientations going out of the minimal points.

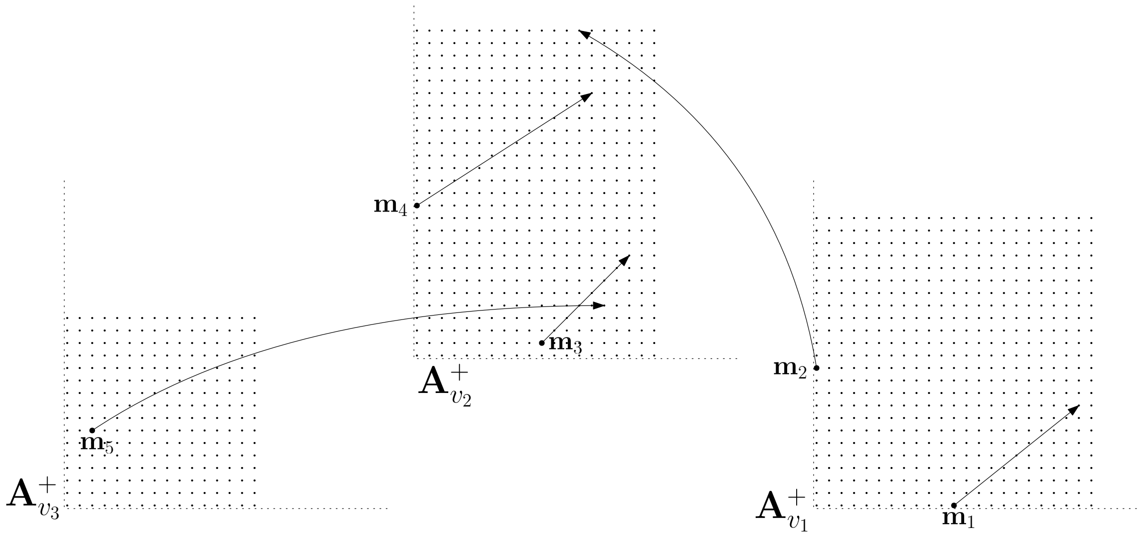

Figure 16: A first manipulation maximizes the number of positive vectors, leaving only two edges (at $\mathbf{m}_5, \mathbf{m}_2$) going from a vertex to another.

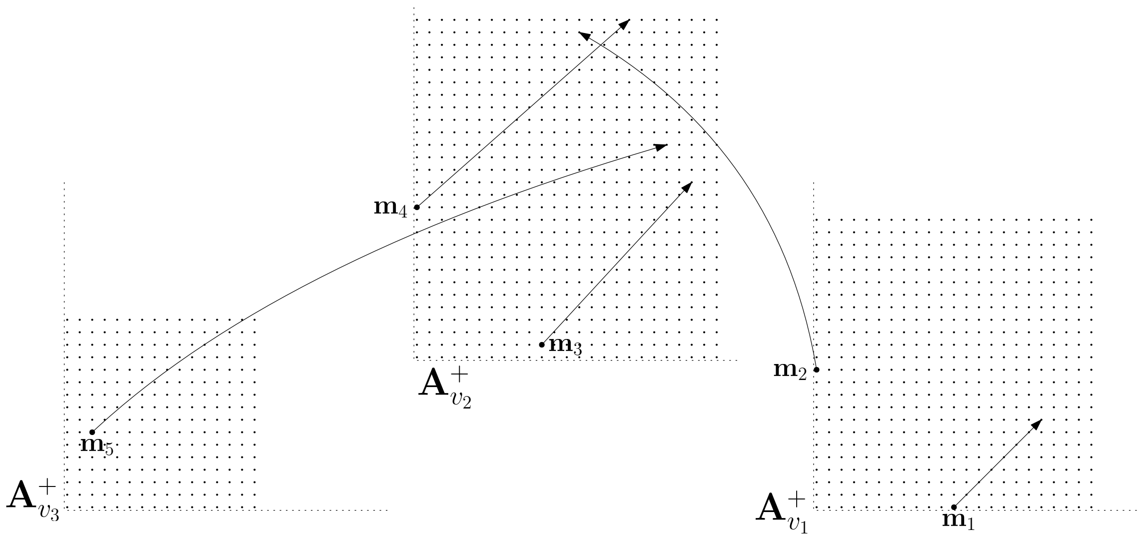

Figure 17: A normal form for the GBS graph of Example 5.28.

6 Further examples

References

[ACRK25a] D. Ascari, M. Casals-Ruiz, and I. Kazachkov. Automorphisms of graphs of groups, 2025. preprint.

[ACRK25b] D. Ascari, M. Casals-Ruiz, and I. Kazachkov. On the isomorphism problem for cyclic JSJ decompositions: vertex elimination, 2025. preprint.

[ACRK25c] D. Ascari, M. Casals-Ruiz, and I. Kazachkov. On the isomorphism problem for generalized Baumslag-Solitar groups: angles and flexible configurations, 2025. preprint.

[Bar18] B. Barrett. Computing JSJ decompositions of hyperbolic groups. *Journal of Topology*, 11(2):527–558, 2018.

[Bau77] A. Baudisch. Kommutationsgleichungen in semifreien Gruppen. *Acta Math. Acad. Sci. Hungar.*, 29(3-4):235–249, 1977.

[Bee15] B. Beeker. Multiple conjugacy problem in graphs of free abelian groups. *Groups Geom. Dyn.*, 9(1):1–27, 2015.

[BF91] M. Bestvina and M. Feighn. Bounding the complexity of simplicial group actions on trees. *Inventiones mathematicae*, 103(3):449–470, 1991.

[BJ96] Hyman Bass and Renfang Jiang. Automorphism groups of tree actions and of graphs of groups. *J. Pure Appl. Algebra*, 112(2):109–155, 1996.

[Bla65] Norman Blackburn. Conjugacy in nilpotent groups. *Proc. Amer. Math. Soc.*, 16:143–148, 1965.

[Bri63] John L. Britton. The word problem. *Ann. of Math. (2)*, 77:16–32, 1963.

[Bum04] Inna Bumagin. The conjugacy problem for relatively hyperbolic groups. *Algebr. Geom. Topol.*, 4:1013–1040, 2004.

[Cas16] C. H. Cashen. Splitting line patterns in free groups. *Algebraic & Geometric Topology*, 16(2):621–673, 2016.

[CF08] M. Clay and M. Forester. On the isomorphism problem for generalized Baumslag-Solitar groups. *Algebraic & Geometric Topology*, 8:2289–2322, 2008.

[CGW09] John Crisp, Eddy Godelle, and Bert Wiest. The conjugacy problem in subgroups of right-angled artin groups. *Journal of Topology*, 2, 2009.

[Cla07] M. Clay. Deformation Spaces of G–trees and Automorphisms of Baumslag–Solitar groups. *Groups, Geometry, and Dynamics*, 3, 2007.

[Cla14] Matt Clay. When does a right-angled Artin group split over $\mathbb{Z}$? *Internat. J. Algebra Comput.*, 24(6):815–825, 2014.

[CM11] C. H. Cashen and N. Macura. Line patterns in free groups. *Geometry & Topology*, 15(3):1419–1475, 2011.

[CRKZ21] M. Casals-Ruiz, I. Kazachkov, and A. Zakharov. Commensurability of Baumslag-Solitar groups. *Indiana Univ. Math. J.*, 70:2527–2555, 2021.

[CV86] M. Culler and K. Vogtmann. Moduli of graphs and automorphisms of free groups. *Inventiones mathematicae*, 84(1):91–119, 1986.

[Day14] Matthew B. Day. Full-featured peak reduction in right-angled Artin groups. *Algebr. Geom. Topol.*, 14(3):1677–1743, 2014.

[DF05] G. Diao and M. Feighn. The Grushko decomposition of a finite graph of finite rank free groups: an algorithm. *Geometry & Topology*, 9(4):1835–1880, 2005.

[DG08a] F. Dahmani and D. Groves. The isomorphism problem for toral relatively hyperbolic groups. *Publications Mathématiques de l'IHÉS*, 107:211–290, 2008.

[DG08b] François Dahmani and Daniel Groves. Detecting free splittings in relatively hyperbolic groups. *Trans. Amer. Math. Soc.*, 360(12):6303–6318, 2008.

[DG11] F. Dahmani and V. Guirardel. The isomorphism problem for all hyperbolic groups. *Geometric & Functional Analysis*, 21(2):223–300, apr 2011.

[DS99] M. J. Dunwoody and M. E. Sageev. JSJ-splittings for finitely presented groups over slender groups. *Inventiones mathematicae*, 135(1):25–44, 1999.

[DT19] François Dahmani and Nicholas Touikan. Deciding isomorphy using Dehn fillings, the splitting case. *Invent. Math.*, 215(1):81–169, 2019.

[Dud17] F. Dudkin. The isomorphism problem for generalized Baumslag–Solitar groups with one mobile edge. *Algebra and Logic*, 56:1–13, 2017.

[Dun85] M. J. Dunwoody. The accessibility of finitely presented groups. *Inventiones mathematicae*, 81(3):449–457, 1985.

[Dun93] M. J. Dunwoody. *An inaccessible group*, volume 1 of *London Mathematical Society Lecture Note Series*, pages 75–78. Cambridge University Press, Cambridge, 1993.

[Eme58] V. A. Emelichev. Commutative semigroups with one defining relation. *Shuya Gosudarstvennyi Pedagogicheskii Institut Uchenye Zapiski*, 5:227–242, 1958.

[ES69] Samuel Eilenberg and M. P. Schützenberger. Rational sets in commutative monoids. *J. Algebra*, 13:173–191, 1969.

[FM98] B. Farb and L. Mosher. A rigidity theorem for the solvable Baumslag-Solitar groups. *Inventiones mathematicae*, 131(2):419–451, 1998.

[For02] M. Forester. Deformation and rigidity of simplicial group actions on trees. *Geometry & Topology*, 6(1):219 – 267, 2002.

[For03] M. Forester. On uniqueness of JSJ decompositions of finitely generated groups. *Commentarii Mathematici Helvetici*, 78(4):740–751, 2003.

[For06] M. Forester. Splittings of generalized Baumslag-Solitar groups. *Geometriae Dedicata*, 121(1):43–59, 2006.

[FP06] K. Fujiwara and P. Papasoglu. JSJ-decompositions of finitely presented groups and complexes of groups. *Geometric & Functional Analysis*, 16:70–125, 02 2006.

[GL11] V. Guirardel and G. Levitt. Trees of cylinders and canonical splittings. *Geometry & Topology*, 15(2):977–1012, 2011.

[GL17] V. Guirardel and G. Levitt. *JSJ decompositions of groups*, volume 395 of *Astérisque*. Société Mathématique de France, 2017.

[Gru40] I. A. Grushko. On the bases of a free product of groups. *Matematicheskii Sbornik*, 8:169–182, 1940.

[GS66] Seymour Ginsburg and Edwin H. Spanier. Semigroups, Presburger formulas, and languages. *Pacific J. Math.*, 16:285–296, 1966.

[GW09] Daniel Groves and Henry Wilton. Enumerating limit groups. *Groups Geom. Dyn.*, 3(3):389–399, 2009.

[Hat] A. Hatcher. Notes on basic 3-manifold topology. https://pi.math.cornell.edu/ hatcher/3M/3Mfds.pdf.

[IJCR91] Oscar H. Ibarra, Tao Jiang, Jik H. Chang, and Bala Ravikumar. Some classes of languages in NC^1. *Inform. and Comput.*, 90(1):86–106, 1991.

[KM79] M. I. Kargapolov and Ju.Ĩ. Merzljakov. *Fundamentals of the theory of groups*, volume 62 of *Graduate Texts in Mathematics*. Springer-Verlag, New York-Berlin, russian edition, 1979.

[KM05] O. Kharlampovich and A. Myasnikov. Effective JSJ decompositions, 2005.

[Kro90] P. H. Kropholler. Baumslag-solitar groups and some other groups of cohomological dimension two. *Commentarii mathematici Helvetici*, 65(4):547–558, 1990.

[Lev05] G. Levitt. On the automorphism group of generalized Baumslag-Solitar groups. *Geometry & Topology*, 11:473 – 515, 2005.

[Lev07] G. Levitt. On the automorphism group of generalized baumslag–solitar groups. *Geometry & Topology*, 11(1):473–515, 2007.

[Lev08] G. Levitt. Unsolvability of the isomorphism problem for [free abelian]-by-free groups. *arXiv: Group Theory*, 2008.

[Loc92] Jody Meyer Lockhart. The conjugacy problem for graph products with infinite cyclic edge groups. *Proceedings of the American Mathematical Society*, 114(3):603–606, 1992.

[LS77] Roger C. Lyndon and Paul E. Schupp. *Combinatorial group theory*, volume Band 89 of *Ergebnisse der Mathematik und ihrer Grenzgebiete [Results in Mathematics and Related Areas]*. Springer-Verlag, Berlin-New York, 1977.

[Mal58] A. I. Mal'cev. Defining relations in categories. *Ivanovo Gosudarstvennyi Pedagogicheskii Institut Uchenye Zapiski*, 18:49–60, 1958.

[McC91] J. McCool. A class of one-relator groups with centre. *Bulletin of the Australian Mathematical Society*, 44(2):245–252, 1991.

[MMNV22] Jeremy Macdonald, Alexei Myasnikov, Andrey Nikolaev, and Svetla Vassileva. Logspace and compressed-word computations in nilpotent groups. *Trans. Amer. Math. Soc.*, 375(8):5425–5459, 2022.

[MS84] John W. Morgan and Peter B. Shalen. Valuations, trees, and degenerations of hyperbolic structures. I. *Ann. of Math. (2)*, 120(3):401–476, 1984.

[Osi06] Denis V. Osin. Relatively hyperbolic groups: intrinsic geometry, algebraic properties, and algorithmic problems. *Mem. Amer. Math. Soc.*, 179(843):vi+100, 2006.

[Pap05] P. Papasoglu. Quasi-isometry invariance of group splittings. *Annals of Mathematics*, 161(2):759–830, 2005.

[RS97] E. Rips and Z. Sela. Cyclic splittings of finitely presented groups and the canonical JSJ decomposition. *Annals of Mathematics*, 146(1):53–109, 1997.

[Sel95] Z. Sela. The isomorphism problem for hyperbolic groups I. *Annals of Mathematics*, 141(2):217–283, 1995.

[Sel01] Z. Sela. Diophantine geometry over groups I: Makanin-Razborov diagrams. *Publications Mathématiques de l'Institut des Hautes Études Scientifiques*, 93(1):31–106, 2001.

[Ser77] J.P. Serre. *Arbres, Amalgames, Sl2*. Asterisque, tome 46. Soc. Math. De France, 1977.

[Sta68] J. R. Stallings. On torsion-free groups with infinitely many ends. *Annals of Mathematics*, 88(2):312–334, 1968.

[Tai68] M. A. Taiclin. Algorithmic problems for commutative semigroups. *Dokl. Akad. Nauk SSSR*, 178:786–789, 1968.

[Tou18] Nicholas W. M. Touikan. Detecting geometric splittings in finitely presented groups. *Trans. Amer. Math. Soc.*, 370(8):5635–5704, 2018.

[Wan23] D. Wang. The isomorphism problem for small-rose generalized Baumslag-Solitar groups, 2023.

[Wei16] Armin Weiss. A logspace solution to the word and conjugacy problem of generalized baumslag-solitar groups. In Delaram Kahrobaei, Bren Cavallo, and David Garber, editors, *Algebra and computer science*. American Mathematical Society, 2016.

[Whi36] J. H. C. Whitehead. On equivalent sets of elements in a free group. *Ann. of Math. (2)*, 37(4):782–800, 1936.

[Why01] K. Whyte. The large scale geometry of the higher Baumslag-Solitar groups. *Geometric & Functional Analysis*, 11(6):1327–1343, 2001.

[Zim90] B. P. Zimmermann. Finite abelian group actions on surfaces. *Yokohama Mathematical Journal*, 38, 1990.

A Nielsen equivalence in finitely generated abelian groups

We consider elements of $\mathbb{Z}^n$ as column vectors. We denote with $\mathcal{M}_{n\times k}(\mathbb{Z})$ the set of matrices with n rows and k columns and integer coefficients. For $M \in \mathcal{M}_{n\times k}(\mathbb{Z})$ we denote with M_i^j the integer number in the i-th row and j-th column of M, for $i = 1, ..., n$ and $j = 1, ..., k$; we denote with $\mathrm{span}(M)$ the subgroup of $\mathbb{Z}^n$ generated by the columns of M. A **column operation** on a matrix consists of choosing a column and adding or subtracting it from another column. A **(column) reordering operation** consists of interchanging two columns of the matrix. Notice that column and column reordering operations don't change $\mathrm{span}(M)$. A **row operation** consists of choosing a row and adding or subtracting it from another row. Row operations correspond to changing basis for the free abelian group $\mathbb{Z}^n$.

Smith normal form

Definition A.1. *A matrix $S \in \mathcal{M}_{n\times k}(\mathbb{Z})$ is in* ***Smith normal form*** *if it satisfies the following conditions:*

1. *$S_i^j = 0$ whenever $i \neq j$.*
2. *$S_i^i \geq 0$ for $i = 1, ..., m$ where $m = \min\{n, k\}$.*
3. *We have $S_{i+1}^{i+1} \mid S_i^i$ for all $i = 1, ..., m-1$.*

Every matrix is equivalent, up to row and column operations and reordering operations, to a unique matrix in Smith normal form. An interesting feature is that the integers S_i^i can be computed directly from the initial matrix, as follows. For $M \in \mathcal{M}_{n\times k}(\mathbb{Z})$, define $D_\ell(M) \geq 0$ as the greatest common divisor of all the determinants of the $\ell \times \ell$ minors of M, for $1 \leq \ell \leq m$ where $m = \min\{n, k\}$. Notice that, if M has rank r, then $D_\ell(M) = 0$ if and only if $\ell \geq r + 1$.

Lemma A.2. *Let $M \in \mathcal{M}_{n\times k}(\mathbb{Z})$ be a matrix of rank r. Let S be the unique matrix in Smith normal form obtained from M by means of row and column operations. Then we have the following:*

1. *$D_1(M) \mid D_2(M) \mid ... \mid D_m(M)$ where $m = \min\{n, k\}$.*
2. *$S_i^i = D_{m+1-i}(M) = 0$ for $i = 1, ..., m-r$.*
3. *$S_i^i = D_{m+1-i}(M)/D_{m-i}(M)$ for $i = m-r+1, ..., m-1$.*
4. *$S_m^m = D_1(M)$.*

Proof. It's immediate to notice that $D_\ell(M)$ is invariant under row and column operations, as well as column swap operation. But for a matrix in Smith normal the thesis holds, so it holds for all matrices. $\square$

Finitely generated abelian groups

Every finitely generated abelian group is isomorphic to

$$\mathbb{Z}/d_1\mathbb{Z} \oplus ... \oplus \mathbb{Z}/d_n\mathbb{Z}$$

for unique integers numbers $n, d_1, ..., d_n \geq 0$ satisfying $1 \neq d_n \mid d_{n-1} \mid ... \mid d_1$. Notice that some d_i can be equal to 0, producing $\mathbb{Z}$ summands.

Lemma A.3. *Let $M \in \mathcal{M}_{n\times k}(\mathbb{Z})$. Let $S \in \mathcal{M}_{n\times k}(\mathbb{Z})$ be the Smith normal form of M. Then $\mathbb{Z}^n/span(M)$ is isomorphic to $\mathbb{Z}/d_1\mathbb{Z} \oplus ... \oplus \mathbb{Z}/d_{n'}\mathbb{Z}$ where*

1. *$n' \leq \min\{n, k\}$ is the maximum integer such that $S_{n'}^{n'} \neq 1$.*
2. *$d_i = S_i^i$ for $i = 1, ..., n'$.*

Proof. Column and reordering operations don't change $\mathrm{span}(M)$, and in particular they don't change the quotient $\mathbb{Z}^n/\mathrm{span}(M)$. Row operations correspond to changing basis for $\mathbb{Z}^n$, so they don't change the isomorphism type of $\mathbf{Z}^n/\mathrm{span}(M)$. Thus we have that $\mathbb{Z}^n/\mathrm{span}(M)$ is isomorphic to $\mathbb{Z}^n/\mathrm{span}(S)$. The conclusion follows. $\square$

Nielsen equivalence in finitely generated abelian groups

Let A be a finitely generated abelian group. Let $(w_1, ..., w_k)$ be an ordered k-tuple of elements of A: a **Nielsen move** on $(w_1, ..., w_k)$ is any operation that consists of the substitution w_i with $w_i + w_j$ or with $w_i - w_j$, for some $j \neq i$. Two ordered k-tuples are **Nielsen equivalent** if there is a sequence of Nielsen moves going from one to the other. Our aim in this section is to classify the k-tuples of generators for A up to Nielsen equivalence (we ignore k-tuples that don't generate the whole group A, as they would require us to change the ambient group we are working in).

Fix an isomorphism

$$A = \mathbb{Z}/d_1\mathbb{Z} \oplus \mathbb{Z}/d_2\mathbb{Z} \oplus ... \oplus \mathbb{Z}/d_n\mathbb{Z}$$

for integers $n, d_1, ..., d_n \geq 0$ with $1 \neq d_n \bigm| d_{n-1} \bigm| ... \bigm| d_1$. Then we can represent elements of A as column vectors with n entries, where the i-th entry is an element of $\mathbb{Z}/d_i\mathbb{Z}$ for $i = 1, ..., n$. We represent an ordered k-tuple $(w_1, ..., w_k)$ as a matrix $M = M(w_1, ..., w_k)$ with n rows and k columns. Nielsen moves correspond to column operations on the matrix M.

Lemma A.4. *The group $A = \mathbb{Z}/d_1\mathbb{Z} \oplus ... \oplus \mathbb{Z}/d_n\mathbb{Z}$ can't be generated by less than n elements.*

Proof. Take a prime $p \bigm| d_n$. Consider the surjective projection map from $\mathbb{Z}/d_1\mathbb{Z} \oplus ... \oplus \mathbb{Z}/d_n\mathbb{Z}$ to $(\mathbb{Z}/p\mathbb{Z})^n$. But $(\mathbb{Z}/p\mathbb{Z})^n$ can't be generated by less than n elements. □

Proposition A.5. *For every $k \geq n+1$, every two ordered k-tuples of generators for A are Nielsen equivalent. Two ordered n-tuples of generators for A are Nielsen equivalent if and only if the corresponding matrices have the same determinant modulo d_n.*

Proof. Let $(w_1, ..., w_k)$ and $(u_1, ..., u_k)$ be k-tuples of generators for A and let $M, N \in \mathcal{M}_{n\times k}(\mathbb{Z})$ be the matrices whose columns represent $w_1, ..., w_k$ and $u_1, ...u_k$ respectively. If $k = n$ and $w_1, ..., w_k$ and $u_1, ..., u_k$ are Nielsen equivalent, then there must be a sequence of columns operations going from M to N, and in particular the two matrices must have the same determinant.

We look at the first row of M, and using column operations we perform the Euclidean algorithm; we obtain a new matrix $M_1^1 \neq 0$ and $M_1^j = 0$ for $j = 2, ..., k$. But since $w_1, ..., w_k$ generate the whole group, we must have that M_1^1 is coprime with d_1. We add d_1 to M_1^2 (we can, since elements of the first row are elements of $\mathbb{Z}/d_1\mathbb{Z}$), and then we perform the Euclidean algorithm again, obtaining a new matrix with $M_1^1 = 1$ and $M_1^j = 0$ for $j = 2, ..., k$.

We reiterate the same reasoning by induction. If $k \geq n+1$ we obtain a matrix with $M_i^i = 1$ for $i = 1, ..., n$ and $M_i^j = 0$ for $j \neq i$. Performing the same procedureon N we obtain the same matrix, and in particular there is a sequence of column operations going from M to N, and thus $w_1, ..., w_k$ is Nielsen equivalent to $u_1, ..., u_k$. If $k = n$, we obtain a matrix with $M_i^i = 1$ for $i = 1, ..., n-1$ and M_n^n invertible modulo d_n and $M_i^j = 0$ for $j \neq n$. Notice that the determinant of M modulo d_n is an invariant under column operations, and for the matrix reduced in this form it's equal to M_n^n. If the determinant of M was equal to the one of N, then performing the same procedure on N we obtain the same matrix, and in particular there is a sequence of column operations going from M to N, and thus $(w_1, ..., w_k)$ is Nielsen equivalent to $(u_1, ..., u_k)$. □

We will also need the following two lemmas.

Lemma A.6. *Let $a \in A$ be an element represented by a vector $(a_1, ..., a_n)$, and suppose that* $\gcd(a_1, ..., a_{n-1}, a_n) = 1$. *Let $(a, w_2, ..., w_k)$, $(a, u_2, ..., u_k)$ be Nielsen equivalent k-tuples of generators for A with $k \geq n+1$. Then $([w_2], ..., [w_k])$, $([u_2], ..., [u_k])$ are Nielsen equivalent $(k-1)$-tuples of generators for $A/\langle a\rangle$.*

Proof. Let $R \in \mathcal{M}_{n\times n}(\mathbb{Z})$ be the matrix given by $R_i^i = d_i$ for $i = 1, ..., n$ and $R_i^j = 0$ for $j \neq i$. We have that $A = \mathbb{Z}^n/\mathrm{span}(R)$. Let R' be the matrix obtained from R by adding an extra column, equal to $(a_1, ..., a_n)$. Then $A/\langle a\rangle = \mathbb{Z}^n/\mathrm{span}(R')$. Since $\gcd(a_1, ..., a_{n-1}) = 1$, we see that $D_1(R') = 1$ and $D_2(R') = d_n \neq 1$. In particular, by Lemmas A.2, A.3 and A.4 we have that the minimum number of generators for $A/\langle a\rangle$ is $n-1$. But then by Proposition A.5 the $(k-1)$-tuples of generators $([w_2], ..., [w_k])$, $([u_2], ..., [u_k])$ for $A/\langle a\rangle$ are Nielsen equivalent, since $k - 1 \geq n$. □

Lemma A.7. *Let $a \in A$ be an element represented by a vector $(a_1, ..., a_n)$, and suppose that $\gcd(a_1, ..., a_{n-1}) = 1$. Let $(a, w_2, ..., w_n)$, $(a, u_2, ..., u_n)$ be Nielsen equivalent n-tuples of generators for A. Then $([w_2], ..., [w_n])$, $([u_2], ..., [u_n])$ are Nielsen equivalent $(n-1)$-tuples of generators for $A/\langle a \rangle$.*

Proof. Let $R \in \mathcal{M}_{n \times n}(\mathbb{Z})$ be the matrix given by $R_i^i = d_i$ for $i = 1, ..., n$ and $R_i^j = 0$ for $j \neq i$. We have that $A = \mathbb{Z}^n/\text{span}(R)$. Let also $M, N \in \mathcal{M}_{n \times k}(\mathbb{Z})$ be matrices whose columns represent $a, w_2, ..., w_n$ and $a, u_2, ..., u_n$ respectively. By Proposition A.5 M and N must have the same determinant modulo d_n.

Let R' be the matrix obtained from R by adding an extra column, equal to the first column of M and of N. We have that $A/\langle a \rangle = \mathbb{Z}^n/\text{span}(R')$ and $D_1(R') = 1$ and $D_2(R') = d_n$ (since $\gcd(a_1, ..., a_{n-1}) = 1$). Thus, by Proposition A.5, we only need to check that the matrices representing $([w_2], ..., [w_n])$ and $([u_2], ..., [u_n])$ have the same determinant modulo d_n.

We perform row operations at the same time on R, M, N, running the Euclidean algorithm on the first column of M (which is also the first column on N). We obtain matrices $\overline{R}, \overline{M}, \overline{N}$ such that the first column of $\overline{M}$ (which is also the first column of $\overline{N}$) is equal to $(1, 0, ..., 0, 0)$. Notice that the determinant of $\overline{M}$ is the same as the one of $\overline{N}$ modulo d_n, since row operations don't change the determinant. Since row operations correspond to changing basis of $\mathbb{Z}^n$, we have an isomorphism $A \cong \mathbb{Z}^n/\text{span}(\overline{R})$, with the two matrices $\overline{M}$ and $\overline{N}$ representing the two k-tuples $a, w_2, ..., w_n$ and $a, u_2, ..., u_n$. Let $\overline{M}''$ be the minor of $\overline{M}$ obtained by cancelling the first row and the first column of $\overline{M}$, and define $\overline{N}''$ analogously; let also $\overline{R}''$ be the matrix obtained from $\overline{R}$ by removing the first row. The isomorphism $A \cong \mathbb{Z}^n/\text{span}(\overline{R})$ induces an isomorphism $A/\langle a \rangle \cong \mathbb{Z}^{n-1}/\text{span}(\overline{R}'')$ and the matrices $\overline{M}''$ and $\overline{N}''$ represent the $(n-1)$-tuples $([w_2], ..., [w_n])$ and $([u_2], ..., [u_n])$. But since the first column of $\overline{M}$ and of $\overline{N}$ is $(1, 0, ..., 0)$ and $\overline{M}, \overline{N}$ have the same determinant modulo d_n, it follows that $\overline{M}''$ and $\overline{N}''$ have the same determinant modulo d_n. The conclusion follows. $\square$

B Nielsen equivalence for big vectors

Fix a free abelian group $\mathbb{Z}^n$. Let $(w_1, ..., w_k)$ be an ordered k-tuple of vectors of $\mathbb{Z}^n$. For an integer $M \geq 1$ we say that an ordered k-tuple $(w_1, ..., w_k)$ is M**-big** if each component of each of $w_1, ..., w_k$ is $\geq M$. Our aim is to classify the M-big ordered k-tuples of vectors up to Nielsen moves. This means that, given two M-big ordered k-tuples, we want to know whether there is a sequence of Nielsen moves that goes from one to the other, and such that all the k-tuples that we write along the sequence are M-big.

We are going to need also a relative version of the same problem, that we now introduce. Let $v_1, ..., v_h$ be vectors of $\mathbb{Z}^n$ and let $(w_1, ..., w_k)$ be an ordered k-tuple of vectors of $\mathbb{Z}^n$. A **Nielsen move relative to** $v_1, ..., v_h$, also called **relative Nielsen move** when there is no ambiguity, is any operation that consists of substituting w_i with $w_i + v_j$ or $w_i - v_j$ for some $i \in \{1, ..., k\}$ and $j \in \{1, ..., h\}$. Fixed vectors $v_1, ..., v_h$, we want to classify the M-big ordered k-tuples up to Nielsen moves and relative Nielsen moves. This means that, given two M-big ordered k-tuples, we want to know whether there is a sequence of Nielsen moves and relative Nielsen moves that goes from one to the other, and such that all the k-tuples that we write along the sequence are M-big.

Given vectors $v_1, ..., v_h \in \mathbb{Z}^n$ and an ordered k-tuple $(w_1, ..., w_k)$, we define the finitely generated abelian group

$$A(w_1, ..., w_k; v_1, ..., v_h) = \langle v_1, ..., v_h, w_1, ..., w_k \rangle / \langle v_1, ..., v_h \rangle$$

It is immediate to see that A is invariant if we change the k-tuple by Nielsen moves and relative Nielsen moves. Moreover, we can look at the ordered k-tuple of generators $([w_1], ..., [w_k])$ for the abelian group A, and we notice that Nielsen moves on $(w_1, ..., w_k)$ correspond to Nielsen moves on $([w_1], ..., [w_k])$, while relative Nielsen moves on $(w_1, ..., w_k)$ correspond to the identity on $([w_1], ..., [w_k])$.

The problem of classification of M-big ordered k-tuples of vectors up to Nielsen moves and relative Nielsen moves, and the problem of classification of ordered k-tuples of generators for A up to Nielsen equivalence, are strictly related. In fact, we will prove that they are equivalent for $k \geq 3$, regardless of M. The main result of Appendix B is the following theorem:

Theorem B.1. *Let $v_1, ..., v_h \in \mathbb{Z}^n$ and let $M \geq 1$. Let $k \geq 3$ and let $(w_1, ..., w_k), (w'_1, ..., w'_k)$ be M-big ordered k-tuples. Then the following are equivalent:*

1. *There is a sequence of Nielsen moves and Nielsen moves relative to $v_1, ..., v_h$ going from $(w_1, ..., w_k)$ to $(w'_1, ..., w'_k)$ such that all the k-tuples that we write along the sequence are M-big.*
2. *We have that $A(w_1, ..., w_k; v_1, ..., v_h) = A(w'_1, ..., w'_k; v_1, ..., v_h) = A$ and the two ordered k-tuples of generators $([w_1], ..., [w_k]), ([w'_1], ..., [w'_k])$ are Nielsen equivalent in A.*

Preliminary results

We provide a couple of preliminary lemmas, that will be useful in what follows. Fix $n \geq 1$ and vectors $v_1, ..., v_h$ in $\mathbb{Z}^n$; fix also an integer $M \geq 1$.

Lemma B.2. *Let $k \geq 1$ and let $(w_1, ..., w_k)$ be an M-big ordered k-tuple. Then for every even permutation σ of $\{1, ..., k\}$ there is a sequence of Nielsen moves going from $(w_1, ..., w_k)$ to $(w_{\sigma(1)}, ..., w_{\sigma(k)})$ such that all the k-tuples that we write along the sequence are M-big.*

Proof. Consider the sequence of Nielsen moves

$$(w_1, w_2, w_3) \to (w_1, w_2 + w_3, w_3) \to (w_1, w_2 + w_3, w_3 + w_1) \to (w_1 + w_2 + w_3, w_2 + w_3, w_3 + w_1) \to$$
$$\to (w_2, w_2 + w_3, w_3 + w_1) \to (w_2, w_3, w_3 + w_1) \to (w_2, w_3, w_1)$$

and notice that in the same way we can obtain any permutation which is a 3-cycle. But 3-cycles generate all even permutations; the conclusion follows. □

Lemma B.3. *Let $k \geq 2$ and let $(w_1, w_2, ..., w_k), (w'_1, w_2, ..., w_k)$ be M-big ordered couples. Suppose that $w'_1 = w_1 + \lambda_2 w_2 + ... + \lambda_k w_k + \mu_1 v_1 + ... + \mu_h v_h$ for some $\lambda_2, ..., \lambda_k, \mu_1, ..., \mu_h \in \mathbb{Z}$. Then there is a sequence of Nielsen moves and relative Nielsen moves going from $(w_1, w_2, ..., w_k)$ to $(w'_1, w_2, ..., w_k)$ such that all the k-tuples that we write along the sequence are M-big.*

Proof. We choose an arbitrary positive integer C which is much bigger than all the components of $w_2, ..., w_k, v_1, ..., v_h$ and than all of $\lambda_2, ..., \lambda_k, \mu_1, ..., \mu_h$. We proceed as follows

$$(w_1, w_2, ..., w_k) \xrightarrow{*} (w_1 + Cw_2, w_2, ..., w_k) \xrightarrow{*} (w'_1 + Cw_2, w_2, ..., w_k) \xrightarrow{*} (w'_1, w_2, ..., w_k)$$

where $\xrightarrow{*}$ denotes a finite sequence of Nielsen moves and relative Nielsen moves. It's evident that the first and the third $\xrightarrow{*}$ can be performed in such a way that every k-tuple that we write along the sequence is M-big. All the k-tuples that we write during the second $\xrightarrow{*}$ are M-big, provided that C had been chosen big enough. □

Proposition B.4. *Let $k \geq 1$ and let $(w_1, ..., w_k), (w'_1, ..., w'_k)$ be M-big ordered k-tuples such that $A(w_1, ..., w_k; v_1, ..., v_h) = A(w'_1, ..., w'_k; v_1, ..., v_h) = A$. Suppose there is a vector in $\langle v_1, ..., v_h \rangle$ such that all of its components are strictly positive. Then the following are equivalent:*

1. *There is a sequence of Nielsen moves and relative Nielsen moves going from $(w_1, ..., w_k)$ to $(w'_1, ..., w'_k)$ such that all the k-tuples that we write along the sequence are M-big.*
2. *The ordered k-uples of generators $([w_1], ..., [w_k]), ([w'_1], ..., [w'_k])$ are Nielsen equivalent in A.*

Proof. $1 \Rightarrow 2$. Trivial.

$2 \Rightarrow 1$. Let v be a vector in $\langle v_1, ..., v_h \rangle$ with all components strictly positive. For every element $[w] \in A$ we can construct a lifting $w \in \langle w_1, ..., w_k, v_1, ..., v_h \rangle$ such that all the components of w are $\geq M$; in fact, we can just pick any lifting and add v repeatedly until the components become all $\geq M$.

Suppose there is a sequence of Nielsen moves going from $([w_1], ..., [w_k])$ to $([w'_1], ..., [w'_k])$ in A. Let $([w_1^r], ..., [w_k^r])$ be such a sequence, for $r = 1, ..., R$, so that $[w_i^1] = [w_i]$ and $[w_i^R] = [w'_i]$ for $i = 1, ..., k$. Let w_i^r be an M-big lifting of $[w_i^r]$ for $i = 1, ..., k$ and $r = 1, ..., R$, and we can take $w_i^1 = w_i$ and $w_i^R = w'_i$ for $i = 1, ..., k$. By Lemma B.3, it's possible to pass from $(w_1^r, ..., w_k^r)$ to $(w_1^{r+1}, ..., w_k^{r+1})$ with a sequence of Nielsen moves and relative Nielsen moves. The conclusion follows. □

Nielsen equivalence for three or more big vectors

Fix $n \geq 1$ and vectors $v_1, ..., v_h$ in $\mathbb{Z}^n$; fix also an integer $M \geq 1$. This section is dedicated to the proof of Theorem B.1. The idea is to try and apply Proposition B.4. When we don't have a positive vector in $\langle v_1, ..., v_h \rangle$, we essentially use one of the w_is instead. This means that, given two M-big ordered k-tuples $(w_1, ..., w_k), (w'_1, ..., w'_k)$, we want to manipulate them until they end up with a common vector. The following Proposition B.5 explains how to manipulate one of the two k-tuples in order to get a vector "almost" in common with the other, up to an integer multiple; the next Propositions B.7 and B.8 are aimed at removing the integer multiple, and to make the vector "really" in common to the k-tuples.

Proposition B.5. *Let $k \geq 3$ and let $(w_1, ..., w_k), (w'_1, ..., w'_k)$ be M-big ordered k-tuples such that $A(w_1, ..., w_k; v_1, ..., v_h) = A(w'_1, ..., w'_k; v_1, ..., v_h) = A$. Then there is an M-big ordered k-tuple $(w''_1, ..., w''_k)$ such that:*

1. *There is a sequence of Nielsen moves and relative Nielsen moves going from $(w_1, ..., w_k)$ to $(w''_1, ..., w''_k)$ such that all the k-tuples that we write along the sequence are M-big.*

2. *We have $w'_1 = d(w''_1 - w''_2) + \mu_1 v_1 + ... + \mu_h v_h$ for some $d, \mu_1, ..., \mu_h \in \mathbb{Z}$.*

Proof. Let $w'_1 = \lambda_1 w_1 + ... + \lambda_k w_k + \mu_1 v_1 + ... + \mu_h v_h$ for $\lambda_1, ..., \lambda_k, \mu_1, ..., \mu_h \in \mathbb{Z}$. We notice that for $i \neq j$ we have the identity

$$\lambda_i w_i + \lambda_j w_j = (\lambda_i - \lambda_j) w_i + \lambda_j (w_j + w_i)$$

For an ordered k-tuple of integers $(\lambda_1, ..., \lambda_k)$ we define a *subtraction* to be any operation that consists of substituting λ_i with $\lambda_i - \lambda_j$ for some $i \neq j$. We can perform any sequence of subtractions to the k-tuple of integers $(\lambda_1, ..., \lambda_k)$, as these are achieved by means of Nielsen moves on the k-tuple of vectors $(w_1, ..., w_k)$ that keep it M-big.

We now work on the ordered k-tuple of integers $(\lambda_1, ..., \lambda_k)$. As $k \geq 3$, it's easy to see that by means of subtractions we can assume that $\lambda_k < 0$. We now subtract λ_k from each of $\lambda_1, ..., \lambda_{k-1}$ enough times, in order to get $\lambda_1, ..., \lambda_{k-1} > 0$. By means of Lemma B.6, we can subtract λ_k from λ_1 the right amount of times, in order to get $\gcd(\lambda_1, ..., \lambda_{k-1}) = \gcd(\lambda_1, ..., \lambda_k)$. We now use subtractions on the non-negative integers $\lambda_1, ..., \lambda_{k-1}$ in order to perform the Euclidean algorithm (and here we need $k \geq 3$): we reduce to a situation of the type $d, 0, ..., 0, -cd$ for some integers $c, d \geq 1$. From this we obtain $d, -d, 0, ..., 0, -cd$ and finally $d, -d, 0, ..., 0$. The conclusion follows. ☐

Lemma B.6. *Let $a, b, c \in \mathbb{Z}$ with $b \neq 0$. Then there is $\lambda \in \mathbb{Z}$ such that*

$$\gcd(a, b, c) = \gcd(a + \lambda c, b)$$

Moreover, the same holds with λ' instead of λ, where λ' is any element of the coset $\lambda + b\mathbb{Z}$.

Proof. Let $p \mid b$ be a prime. If $v_p(c) > v_p(a)$ then we have $v_p(a + \lambda c) = v_p(a) = \min\{v_p(a), v_p(c)\}$. If $v_p(c) < v_p(a)$ then we impose the condition $\lambda \not\equiv 0 \bmod p$; this implies that $v_p(a + \lambda c) = v_p(c) = \min\{v_p(a), v_p(c)\}$. If $v_p(c) = v_p(a)$ then we write $a = p^{v_p(a)} a'$ and $c = p^{v_p(c)} c'$ with a', c' coprime with p, and we impose the condition $\lambda \not\equiv -a'/c' \bmod p$; this implies that $v_p(a + \lambda c) = v_p(a) = v_p(c)$.

By the Chinese reminder theorem, we can find λ satisfying the above conditions for all primes $p \mid b$. Now for every prime p that divides b we have $v_p(\gcd(a + \lambda c, b)) = \min\{v_p(a + \lambda c), v_p(b)\} = \min\{v_p(a), v_p(b), v_p(c)\} = v_p(\gcd(a, b, c))$, and for every prime p that doesn't divide b we have $v_p(\gcd(a + \lambda c, b)) = 0 = v_p(\gcd(a, b, c))$. It follows that $\gcd(a + \lambda c, b) = \gcd(a, b, c)$, as desired. ☐

Proposition B.7. *Let $k \geq 2$ and let $(w_1, ..., w_k)$ be an M-big ordered k-tuple. Suppose $A = A(w_1, ..., w_k; v_1, ..., v_h) \cong \mathbb{Z} \oplus \mathbb{Z} \oplus A'$ for some finitely generated abelian group A'. Then there is an M-big ordered k-tuple $(w'_1, ..., w'_k)$ such that:*

1. *There is a sequence of Nielsen moves and relative Nielsen moves going from $(w_1, ..., w_k)$ to $(w'_1, ..., w'_k)$ such that each k-tuple that we write along the sequence is M-big.*

2. *The first two components of $[w'_1] \in A \cong \mathbb{Z} \oplus \mathbb{Z} \oplus A'$ are coprime, and $[w'_1]$ is not a proper power.*

Proof. Let $[w_i]$ be equal to the vector $(\alpha_i, \beta_i, \gamma_i)$ through the isomorphism $A \cong \mathbb{Z} \oplus \mathbb{Z} \oplus A'$, for $i = 1, ..., k$ and $\alpha_i, \beta_i \in \mathbb{Z}$ and $\gamma_i \in A'$. Since $[w_1], ..., [w_k]$ generate A, we must have $\alpha_i\beta_j - \alpha_j\beta_i \neq 0$ for some $i, j \in \{1, ..., k\}$. By Lemma B.2 we can reorder the vectors with an arbitrary even permutation. Thus we can assume

$$\alpha_1\beta_2 - \alpha_2\beta_1 \neq 0.$$

If $\alpha_2 = 0$ then $\alpha_1 \neq 0$ (since $\alpha_1\beta_2 - \alpha_2\beta_1 \neq 0$) and we substitute w_2 with $w_2 + w_1$. Thus we can assume

$$\alpha_2 \neq 0.$$

By Lemma B.6 we find $\lambda \geq 0$ such that $\gcd(\alpha_1 + \lambda\alpha_3, \alpha_2) = \gcd(\alpha_1, \alpha_2, \alpha_3)$. We substitute w_1 with $w_1 + \lambda w_3$ and thus we can assume $\gcd(\alpha_1, \alpha_2) = \gcd(\alpha_1, \alpha_2, \alpha_3)$. Since λ can be chosen modulo α_2, we can also take λ in such a way that $\alpha_1\beta_2 - \alpha_2\beta_1$ remains $\neq 0$. Reiterating the same reasoning, we obtain $\gcd(\alpha_1, \alpha_2) = \gcd(\alpha_1, ..., \alpha_k)$, which is 1 since $[w_1], ..., [w_k]$ generate A. Thus we can assume

$$\gcd(\alpha_1, \alpha_2) = 1.$$

By Lemma B.6 we find $C \geq 0$ such that $\gcd(\alpha_1 + C\alpha_2, \alpha_1\beta_2 - \alpha_2\beta_1) = \gcd(\alpha_1, \alpha_2, \alpha_1\beta_2 - \alpha_2\beta_1) = 1$. We now have that

$$\gcd(\alpha_1 + C\alpha_2, \alpha_1\beta_2 - \alpha_2\beta_1 - \beta_2(\alpha_1 + C\alpha_2)) = 1$$

$$\gcd(\alpha_1 + C\alpha_2, -\alpha_2(\beta_1 + C\beta_2)) = 1$$

$$\gcd(\alpha_1 + C\alpha_2, \beta_1 + C\beta_2) = 1$$

We substitute w_1 with $w_1 + Cw_2$ and thus obtain that $(\alpha_1, \beta_1) = 1$. This also implies that the element $[w_1]$ can't be a proper power in A. $\square$

Proposition B.8. *Let $k \geq 3$ and let $(w_1, ..., w_k)$ be an M-big ordered k-tuple. Suppose that $A = A(w_1, ..., w_k; v_1, ..., v_h) \cong \mathbb{Z} \oplus \mathbb{Z}/d_2\mathbb{Z} \oplus A'$ for some $d_2 \geq 2$ and A' finitely generated abelian group. Then there is an M-big ordered k-tuple $(w'_1, ..., w'_k)$ such that:*

1. *There is a sequence of Nielsen moves and relative Nielsen moves going from $(w_1, ..., w_k)$ to $(w'_1, ..., w'_k)$ such that each k-tuple that we write along the sequence is M-big.*
2. *The first two components of $[w'_1] \in A \cong \mathbb{Z} \oplus \mathbb{Z}/d_2\mathbb{Z} \oplus A'$ are coprime, and $[w'_1]$ is not a proper power.*

Proof. Let $[w_i]$ be equal to the vector $(\alpha_i, \beta_i, \gamma_i)$ through the isomorphism $A \cong \mathbb{Z} \oplus \mathbb{Z}/d_2\mathbb{Z} \oplus A'$, for $i = 1, ..., k$ and $\alpha_i \in \mathbb{Z}$ and $\beta_i \in \mathbb{Z}/d_2\mathbb{Z}$ and $\gamma_i \in A'$. Let also p be a prime that divides d_2.

Since $[w_1], ..., [w_k]$ generate A, we must have $\alpha_i\beta_j - \alpha_j\beta_i \not\equiv 0 \mod p$ for some $i, j \in \{1, ..., k\}$. By Lemma B.2 we can reorder the vectors with an arbitrary even permutation. Thus we can assume

$$\alpha_1\beta_2 - \alpha_2\beta_1 \not\equiv 0 \mod p.$$

If $\alpha_2 \equiv 0 \mod p$ then $\alpha_1 \not\equiv 0 \mod p$ (since $\alpha_1\beta_2 - \alpha_2\beta_1 \not\equiv 0 \mod p$) and we substitute w_2 with $w_2 + w_1$. Thus we can assume

$$\alpha_2 \not\equiv 0 \mod p.$$

By Lemma B.6 we find $\lambda \geq 0$ such that $\gcd(\alpha_1 + \lambda\alpha_3, \alpha_2) = \gcd(\alpha_1, \alpha_2, \alpha_3)$. We substitute w_1 with $w_1 + \lambda w_3$ and thus we can assume $\gcd(\alpha_1, \alpha_2) = \gcd(\alpha_1, \alpha_2, \alpha_3)$. Since λ can be chosen modulo α_2, which is coprime with p, we can also take λ to be a multiple of p, so that $\alpha_1\beta_2 - \alpha_2\beta_1 \mod p$ remains the same. Reiterating the same reasoning, we obtain $\gcd(\alpha_1, \alpha_2) = \gcd(\alpha_1, ..., \alpha_k)$, which is 1 since $[w_1], ..., [w_k]$ generate A. Thus we can assume

$$\gcd(\alpha_1, \alpha_2) = 1.$$

By means of the Chinese reminder theorem we find $C \geq 0$ such that $\alpha_3 + C\alpha_1 \equiv 1 \mod \alpha_2$ and $(\alpha_3\beta_2 - \alpha_2\beta_3) + C(\alpha_1\beta_2 - \alpha_2\beta_1) \not\equiv 0 \mod p$; here we are using the facts that $\gcd(\alpha_2, p) = 1$ and $\gcd(\alpha_1, \alpha_2) = 1$ and $\gcd(\alpha_1\beta_2 - \alpha_2\beta_1, p) = 1$. Since $\gcd(\alpha_2, p) = 1$ we can find integers $e \geq 0$ as big as we want and such that $p^e \equiv 1 \mod \alpha_2$. Using the fact that $\alpha_3 + C\alpha_1 \equiv 1 \mod \alpha_2$ we are thus able to find integers $D \geq 0$ and $e \geq 0$ such that $\alpha_3 + C\alpha_1 + D\alpha_2 = p^e$. We now have that

$$\gcd(p, (\alpha_3\beta_2 - \alpha_2\beta_3) + C(\alpha_1\beta_2 - \alpha_2\beta_1)) = 1$$

$$\gcd(p, (\alpha_3\beta_2 - \alpha_2\beta_3) + C(\alpha_1\beta_2 - \alpha_2\beta_1) - \beta_2(\alpha_3 + C\alpha_1 + D\alpha_2)) = 1$$

$$\gcd(p, -\alpha_2(\beta_3 + C\beta_1 + D\beta_2)) = 1$$

$$\gcd(p, \beta_3 + C\beta_1 + D\beta_2) = 1$$

We substitute w_3 with $w_3 + Cw_1 + Dw_2$ and thus we can assume that $\alpha_3 = p^e$ and $(p, \beta_3) = 1$. In particular $(\alpha_3, \beta_3) = 1$. The element $[w_3]$ can be a proper power in A: if $[w_3] = d[w]$, then $\alpha_3 = p^e$ implies that $d \mid p^e$, and $(p, \beta_3) = 1$ implies that $(p, d) = 1$; it follows that $d = \pm 1$. To conclude, we permute (w_1, w_2, w_3) using Lemma B.2 in order to bring w_3 in first position. □

Proposition B.9. *Let $k \geq 3$ and let $(w_1, ..., w_k), (w'_1, ..., w'_k)$ be M-big ordered k-tuples such that $A(w_1, ..., w_k; v_1, ..., v_h) = A(w'_1, ..., w'_k; v_1, ..., v_h) = A \cong \mathbb{Z}$. Then there is a sequence of Nielsen moves and relative Nielsen moves from $(w_1, ..., w_k)$ to $(w'_1, ..., w'_k)$ such that every k-tuple that we write along the sequence is M-big.*

Proof. Let $[w_i], [w'_i]$ be equal to α_i, α'_i through the isomorphism $A \cong \mathbb{Z}$, for $i = 1, ..., k$ and $\alpha_i, \alpha'_i \in \mathbb{Z}$. If among α_i, α'_i there is at least one integer > 0 and at least one integer < 0, then $\langle v_1, ..., v_h \rangle$ has to contain a vector whose components are all strictly positive, and thus we are done by Proposition B.4. Thus we assume $\alpha_i, \alpha'_i > 0$ for all $i = 1, ..., k$.

By Lemma B.6 we find $\lambda \geq 0$ such that $\gcd(\alpha_1 + \lambda\alpha_3, \alpha_2) = \gcd(\alpha_1, \alpha_2, \alpha_3)$. We substitute w_1 with $w_1 + \lambda w_3$ and thus we can assume $\gcd(\alpha_1, \alpha_2) = \gcd(\alpha_1, \alpha_2, \alpha_3)$. By reiterating the same reasoning, we can assume $\gcd(\alpha_1, \alpha_2) = \gcd(\alpha_1, ..., \alpha_k) = 1$. Now, by substituting w_3 with $w_3 + \lambda_1 w_1 + \lambda_2 w_2$ for $\lambda_1, \lambda_2 \geq 0$, we can set α_3 to be equal to any natural number big enough. Similarly, we can assume $\gcd(\alpha'_1, \alpha'_2) = 1$, and we are able to set α'_3 to be any natural number big enough. We choose to set $\alpha_3 = \alpha'_3$ and such that $\gcd(\alpha_1, \alpha_3) = \gcd(\alpha'_1, \alpha_3) = 1$. Now, by substituting w_2 with $w_2 + \mu_1 w_1 + \mu_3 w_3$ for $\mu_1, \mu_3 \geq 0$, we can set α_2 to be any natural number big enough. Similarly, we are able to set α'_2 to be any natural number big enough. We choose to set $\alpha_2 = \alpha'_2$ and such that $\gcd(\alpha_2, \alpha_3) = 1$. Reasoning in the same way, and keeping fixed w_2, w_3, w'_2, w'_3, we can set $\alpha_1 = \alpha'_1$ and $\alpha_i = \alpha'_i$ for $i \geq 4$.

To conclude, for $i = 1, ..., k$, we observe that $\alpha_i = \alpha'_i$ implies that $w'_i = w_i + \mu_1 v_1 + ... + \mu_h v_h$ for some $\mu_1, ..., \mu_h \in \mathbb{Z}$ and thus by Lemma B.3 we can obtain $w_i = w'_i$ by means of Nielsen moves and relative Nielsen moves. The thesis follows. □

Proof of Theorem B.1. $1 \Rightarrow 2$. Trivial.

$2 \Rightarrow 1$. If A is a finite abelian group, then we must have $d[w_1] = 0$ in A for some $d \geq 1$, and this implies that $\langle v_1, ..., v_h \rangle$ contains the vector dw_1, which has all the components strictly positive; the conclusion follows from Proposition B.4. If A is isomorphic to $\mathbb{Z}$, then we are done by Proposition B.9.

Otherwise we write $A \cong \mathbb{Z} \oplus \mathbb{Z}/d_2\mathbb{Z} \oplus ... \oplus \mathbb{Z}/d_m\mathbb{Z}$ for integers $m \geq 2$ and $d_2, ..., d_m \geq 0$ with $1 \neq d_m \mid d_{m-1} \mid ... \mid d_2$. By either Proposition B.7 or Proposition B.8, we are able to change $(w'_1, ..., w'_k)$ by means of Nielsen moves and relative Nielsen moves in such a way that $[w'_1]$ isn't a proper power in A, and that $[w'_1]$ can be represented by a vector whose first two components are coprime. We now apply Proposition B.5 and we apply a sequence of Nielsen moves and relative Nielsen moves to $(w_1, ..., w_k)$, in order to obtain that $w'_1 = w_1 - w_2 + \mu_1 v_1 + ... + \mu_h v_h$. By means of Proposition B.3 we can easily change w_1 in such a way that $w_1 = w'_1$.

If $m \geq 3$ then we can apply Lemma A.7, since $[w'_1]$ can be represented by a vector whose first two components are coprime, and we obtain that $([w_2], ..., [w_k]), ([w'_2], ..., [w'_k])$ are Nielsen equivalent $(k-1)$-tuples of generators for $A/\langle [w_1] \rangle$. If $m = 2$ then $[w'_1]$ can be represented by a vector whose components are coprime, and thus we can apply Lemma A.6 to obtain that $([w_2], ..., [w_k]), ([w'_2], ..., [w'_k])$ are Nielsen equivalent $(k-1)$-tuples of generators for $A/\langle [w_1] \rangle$. In both cases, the conclusion follows from Proposition B.4. □