Semigroup Intersection Problems in the Heisenberg Groups

Ruwen Dong ✉
Department of Computer Science, University of Oxford, UK

Abstract

We consider two algorithmic problems concerning sub-semigroups of Heisenberg groups and, more generally, two-step nilpotent groups. The first problem is *Intersection Emptiness*, which asks whether a finite number of given finitely generated semigroups have empty intersection. This problem was first studied by Markov in the 1940s. We show that Intersection Emptiness is PTIME decidable in the Heisenberg groups $\mathrm{H}_n(\mathbb{K})$ over any algebraic number field $\mathbb{K}$, as well as in direct products of Heisenberg groups. We also extend our decidability result to arbitrary finitely generated 2-step nilpotent groups.

The second problem is *Orbit Intersection*, which asks whether the orbits of two matrices under multiplication by two semigroups intersect with each other. This problem was first studied by Babai et al. (1996), who showed its decidability within commutative matrix groups. We show that Orbit Intersection is decidable within the Heisenberg group $\mathrm{H}_3(\mathbb{Q})$.

2012 ACM Subject Classification Computing methodologies → Symbolic and algebraic manipulation

Keywords and phrases semigroup intersection, orbit intersection, matrix semigroups, Heisenberg group, nilpotent groups

Digital Object Identifier 10.4230/LIPIcs.STACS.2023.25

Funding The author acknowledges support from UKRI Frontier Research Grant EP/X033813/1.

1 Introduction

The computational theory of matrix groups and semigroups is one of the oldest and most well-developed parts of computational algebra. Dating back to the work of Markov [31] in the 1940s, the area plays an essential role in analysing system dynamics, with notable applications in automata theory and program analysis [8, 10, 13, 21]. See [24] for an all-encompassing survey on this topic. While many computational problems are undecidable even for matrix groups of dimension three and four [6, 32, 34], various non-trivial algorithms have been developed for matrix groups satisfying additional constraints, such as commutativity [1], nilpotency [14], solvability [28], and having dimension two [5, 35].

As most algorithmic problems for commutative groups are well-understood due to their relatively simple structure, much effort has focused on problems concerning relaxations of the commutativity requirement, such as nilpotency and solvability. Prominent examples of widely studied groups include the *Heisenberg groups*, as well as the more general *2-step nilpotent groups*. The Heisenberg groups $\mathrm{H}_n(\mathbb{K})$ play an important role in many branches of mathematics, physics and computer science. They first arose in the description of one-dimensional quantum mechanical systems [33, 37], and have now become an important mathematical object connecting domains like representation theory, theta functions, Fourier analysis and quantum algorithms [20, 22, 25, 29, 38]. From a computational point of view, Heisenberg groups are interesting because they are the simplest non-commutative Lie groups. Heisenberg groups are included in the class of 2-step nilpotent groups: these are groups whose quotient by their centre is abelian. Despite being the simplest class of non-commutative groups, 2-step nilpotent groups admit highly non-trivial or even undecidable algorithmic problems, notably due to their ability to encode quadratic equations [27]. For example, decades of research has focused on finding a polynomial-time group isomorphism algorithm for 2-step nilpotent groups, with little success [2, 16].

40th International Symposium on Theoretical Aspects of Computer Science (STACS 2023).
Editors: Petra Berenbrink, Patricia Bouyer, Anuj Dawar, and Mamadou Moustapha Kanté;
Article No. 25; pp. 25:1–25:18
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

For a set $\mathcal{G}$ of matrices in some matrix group G, denote by $\langle \mathcal{G} \rangle$ the *semi*group generated by the set $\mathcal{G}$. In this paper, we consider the following two decision problems for the Heisenberg groups and 2-step nilpotent groups.

i. *(Intersection Emptiness)* Given M finite sets of matrices $\mathcal{G}_1, \dots, \mathcal{G}_M$, decide whether $\langle \mathcal{G}_1 \rangle \cap \cdots \cap \langle \mathcal{G}_M \rangle = \emptyset$.

ii. *(Orbit Intersection)* Given two finite sets of matrices $\mathcal{G}, \mathcal{H}$ and matrices S, T, decide whether $T \cdot \langle \mathcal{G} \rangle \cap S \cdot \langle \mathcal{H} \rangle = \emptyset$.

Intersection Emptiness was one of the first problems studied in algorithmic semigroup theory. In the seminal work of Markov [31], the undecidability of Intersection Emptiness was shown for two sets of 4×4 integer matrices. More recently, by encoding the *Post Correspondence Problem*, Halava and Harju showed its undecidability for two sets of 3×3 upper triangular integer matrices [17]. For 2×2 integer matrices, the problem is only known to be NP-hard [7]. In this paper, we show that Intersection Emptiness is decidable in polynomial time for the Heisenberg groups $\mathrm{H}_n(\mathbb{K})$ over an arbitrary algebraic number field $\mathbb{K}$, as well as for any direct product of such Heisenberg groups. In fact, we will prove the decidability result in the more general case of (finitely generated) 2-step nilpotent groups.

The Orbit Intersection problem was first considered by Babai *et al.* [1], who proved its decidability in *commutative* matrix groups over an algebraic number field. In this paper, we prove the decidability of Orbit Intersection for matrices in the Heisenberg group $\mathrm{H}_3(\mathbb{Q})$.

Let us mention some previous work for semigroup algorithmic problems in the Heisenberg groups and 2-step nilpotent groups. These have seen significant advance in research in recent years. Various results have been shown for the following decision problems.

iii. *(Identity Problem)* Given a finite set of matrices $\mathcal{G}$, decide whether the identity matrix $I \in \langle \mathcal{G} \rangle$.

iv. *(Membership Problem)* Given a finite set of matrices $\mathcal{G}$ and a matrix A, decide whether $A \in \langle \mathcal{G} \rangle$.

v. *(Knapsack Problem)* Given matrices $A_1, A_2, \dots, A_K$ and a matrix A, decide whether there exist $(n_1, n_2 \dots, n_K) \in \mathbb{N}^K$ such that $A = A_1^{n_1} A_2^{n_2} \cdots A_K^{n_K}$.

The Identity Problem in $\mathrm{H}_n(\mathbb{Q})$ was shown to be decidable by Ko, Niskanen and Potapov [26]. Dong [14] then introduced tools from Lie algebra and strengthened this result to PTIME decidability in $\mathrm{H}_n(\mathbb{K})$ for algebraic number fields $\mathbb{K}$. The Membership Problem in $\mathrm{H}_n(\mathbb{Q})$ was shown to be decidable by Colcombet, Ouaknine, Semukhin and Worrell. Their main idea is to use the *Baker-Campbell-Hausdorff (BCH) formula* as well as to incorporate the Membership Problem in a Parikh automaton. It was left as an open problem whether the Membership Problem in $\mathrm{H}_n(\mathbb{K})$ for larger fields $\mathbb{K}$ remains decidable. On the other hand, it is known that there exist 2-step nilpotent groups with undecidable Membership Problem [30]. As for the Knapsack Problem, König, Lohrey and Zetzsche showed its decidability in $\mathrm{H}_n(\mathbb{Z})$ by reducing it to solving a single quadratic equation over the natural numbers [27]. They also constructed a 2-step nilpotent group (namely, a direct product of $\mathrm{H}_3(\mathbb{Z})$) where the Knapsack Problem is undecidable, using an embedding of Hilbert's Tenth Problem.

We point out that by taking $\mathcal{G}_1 = \mathcal{G}$, $\mathcal{G}_2 = \{I\}$, Intersection Emptiness subsumes the Identity Problem. Whereas by taking $T = I, S = A, \mathcal{H} = \{I\}$, the Orbit Intersection problem subsumes the Membership Problem. Hence, the tools developed in this paper provide a more general approach to semigroup problems in 2-step nilpotent groups. Our proofs are based on the logarithm of matrices and the BCH formula, whose usage in studying matrix semigroup problems has been introduced in [12] and [14]. However, our approach goes much deeper in analysing the non-commutative terms of the BCH formula. We show that these terms are connected with a word combinatorics problem concerning subwords of length two, and show a critical result characterizing the behaviour of these terms. This will allow us to reduce equations containing word combinatorial terms to pure linear Diophantine equations.

2 Main results

In this section we state our main results. Denote by $\mathsf{UT}(n, \mathbb{Q})$ the group of $n \times n$ upper triangular rational matrices with ones along the diagonal. Our main result on Intersection Emptiness is the following. For the formal definition of 2-step nilpotency, see Section 3.

▶ **Theorem 1.** *Let G be a 2-step nilpotent subgroup of $\mathsf{UT}(n, \mathbb{Q})$ for some n. Given finite subsets $\mathcal{G}_1, \dots, \mathcal{G}_M$ of G, it is decidable in polynomial time whether $\langle \mathcal{G}_1 \rangle \cap \cdots \cap \langle \mathcal{G}_M \rangle = \emptyset$.*

For $n \geq 3$, the *Heisenberg group* $\mathrm{H}_n(\mathbb{K})$ over a field or commutative ring $\mathbb{K}$ is defined as

$$\mathrm{H}_n(\mathbb{K}) \coloneqq \left\{ \begin{pmatrix} 1 & \boldsymbol{a}^\top & c \\ 0 & I_{n-2} & \boldsymbol{b} \\ 0 & 0 & 1 \end{pmatrix}, \text{ where } \boldsymbol{a}, \boldsymbol{b} \in \mathbb{K}^{n-2},\, c \in \mathbb{K} \right\},$$

where we use the notation I_d for the identity matrix of dimension d. Decidability results for the Heisenberg groups and for 2-step nilpotent groups follow as a corollary of Theorem 1.

▶ **Corollary 2.** *Intersection Emptiness is decidable:*

(i) *in PTIME, for the Heisenberg groups $\mathrm{H}_n(\mathbb{K})$ over any algebraic number field $\mathbb{K}$, and for any direct product of Heisenberg groups.*

(ii) *for finitely generated 2-step nilpotent groups*[1].

Fix a group G. Given an element $T \in G$ and a subset $\mathcal{G}$ of G, denote by $T \cdot \langle \mathcal{G} \rangle$ the orbit of T under right multiplication by the semigroup $\langle \mathcal{G} \rangle$. That is, $T \cdot \langle \mathcal{G} \rangle \coloneqq \{T \cdot s \mid s \in \langle \mathcal{G} \rangle\}$. Our main result concerning Orbit Intersection is the following.

▶ **Theorem 3.** *Given elements $T, S \in \mathrm{H}_3(\mathbb{Q})$ and two finite subsets $\mathcal{G}, \mathcal{H}$ of $\mathrm{H}_3(\mathbb{Q})$, it is decidable whether $T \cdot \langle \mathcal{G} \rangle \cap S \cdot \langle \mathcal{H} \rangle = \emptyset$.*

3 Preliminaries

Convex geometry

Let V be a $\mathbb{Q}$-linear space. A subset $\mathcal{C} \subseteq V$ is called a *cone* if $a \in \mathcal{C}$ implies $a\mathbb{Q}_{\geq 0} \subseteq \mathcal{C}$, and $a, b \in \mathcal{C}$ implies $a + b \in \mathcal{C}$. Given a set of vectors $\mathcal{S} \subseteq V$, denote by $\langle \mathcal{S} \rangle_{\mathbb{Q}_{\geq 0}}$ the *cone generated by* $\mathcal{S}$, that is, the smallest cone of V containing $\mathcal{S}$. The *dimension* of a cone $\mathcal{C}$ is the dimension of the smallest linear space containing $\mathcal{C}$.

The *support* of a vector $\boldsymbol{\ell} = (\ell_1, \dots, \ell_K) \in \mathbb{Z}_{\geq 0}^K$ is defined as the set of indices where the entry of $\boldsymbol{\ell}$ is non-zero:

$$\mathrm{supp}(\boldsymbol{\ell}) \coloneqq \{i \in \{1, \dots, K\} \mid \ell_i > 0\}.$$

The *support* of a subset Λ of $\mathbb{Z}_{\geq 0}^K$ is defined as the union of supports of all vectors in Λ:

$$\mathrm{supp}(\Lambda) \coloneqq \bigcup_{\boldsymbol{\ell} \in \Lambda} \mathrm{supp}(\boldsymbol{\ell}) = \{i \mid \exists (\ell_1, \dots, \ell_K) \in \Lambda, \ell_i > 0\}.$$

In this paper, we will need to compute the support of sets of the form $\Lambda = \mathbb{Z}_{\geq 0}^K \cap V$, where V is a $\mathbb{Q}$-linear subspace of $\mathbb{Q}^K$.

[1] We suppose that the structure of the group is given by a finite presentation or a consistent polycyclic presentation (see [19, Chapter 8]).

▶ **Lemma 4** ([14, Lemma 2.4]). *Given V a $\mathbb{Q}$-linear subspace of $\mathbb{Q}^K$, represented as the solution set of linear homogeneous equations, one can compute the support of $\Lambda = \mathbb{Z}^K_{\geq 0} \cap V$ in polynomial time.*

The group $\mathsf{UT}(n, \mathbb{Q})$ and 2-step nilpotent groups

Denote by $\mathsf{UT}(n, \mathbb{Q})$ the group of $n \times n$ upper triangular rational matrices with ones along the diagonal. Let $\mathbb{K}$ be an algebraic number field. $\mathbb{K}$ can be considered as a linear space over $\mathbb{Q}$ of dimension $d \coloneqq [\mathbb{K} : \mathbb{Q}]$. Let $k_1, \dots, k_d \in \mathbb{K}$ be a $\mathbb{Q}$-basis of this linear space. Throughout this paper, an element k of $\mathbb{K}$ is represented as a tuple $(a_1, \dots, a_d) \in \mathbb{Q}^d$ such that $k = a_1 k_1 + \cdots a_d k_d$. An element k of $\mathbb{K}$ acts on $\mathbb{K}$ by multiplication, and can therefore be considered as an endomorphism of the $\mathbb{Q}$-linear space $\mathbb{K}$. Associate k with the matrix that represents this endomorphism, then we have an (injective) embedding $\iota : \mathbb{K} \hookrightarrow \mathbb{Q}^{d \times d}$. In particular, $\iota(1) = I_d$. This embedding is effectively computable in polynomial time [11].

The embedding ι extends to an embedding $\mathrm{H}_n(\mathbb{K}) \hookrightarrow \mathsf{UT}(nd, \mathbb{Q})$, which we also denote by ι. Note that for any matrix $A \in \mathrm{H}_n(\mathbb{K})$, the total bit size of entries in $\iota(A)$ is at most quadratic in the total bit size of entries in A. Therefore, throughout this paper, we will work with matrices in $\iota(\mathrm{H}_n(\mathbb{K})) \subseteq \mathsf{UT}(nd, \mathbb{Q})$, knowing that any polynomial time algorithm in $\mathsf{UT}(nd, \mathbb{Q})$ will translate to a polynomial time algorithm in $\mathrm{H}_n(\mathbb{K})$.

Let G be an arbitrary group. The *centre* of G is the normal subgroup $Z(G) \trianglelefteq G$ consisting of elements that commute with every element of G (see [15]). We say that G is *2-step nilpotent* if the quotient $G/Z(G)$ is abelian. In particular, the Heisenberg groups $\mathrm{H}_n(\mathbb{K})$, as well as their direct products, are 2-step nilpotent [15, Examples 13.36]. Every finitely generated 2-step nilpotent group can be embedded as a subgroup of the direct product $A \times G_0$, where A is finite and G_0 is a 2-step nilpotent subgroup of $\mathsf{UT}(n, \mathbb{Q})$ for some n [4, Theorem 2.1] [23, Theorem 17.2.5].

Logarithm of matrices and Lie algebra

The *Lie algebra* $\mathfrak{u}(n)$ is defined as the $\mathbb{Q}$-linear space of $n \times n$ upper triangular rational matrices with *zeros* on the diagonal. There exist the *logarithm* map

$$\log : \mathsf{UT}(n, \mathbb{Q}) \to \mathfrak{u}(n), \quad A \mapsto \sum_{k=1}^{n} \frac{(-1)^{k-1}}{k} (A - I)^k$$

and the *exponential* map

$$\exp : \mathfrak{u}(n) \to \mathsf{UT}(n, \mathbb{Q}), \quad X \mapsto \sum_{k=0}^{n} \frac{1}{k!} X^k$$

which are inverse of one another. In particular, $\log I = 0$ and $\exp(0) = I$.

The Lie algebra $\mathfrak{u}(n)$ is equipped with the *Lie bracket* $[\cdot, \cdot] : \mathfrak{u}(n) \times \mathfrak{u}(n) \to \mathfrak{u}(n)$ given by $[X, Y] = XY - YX$. For a subset or subsemigroup $\mathcal{A}$ of $\mathsf{UT}(n, \mathbb{Q})$, we naturally denote by $\log \mathcal{A} \coloneqq \{\log a \mid a \in \mathcal{A}\}$ the set of logarithm of matrices in $\mathcal{A}$.

Parikh Image and length two subwords

Given a finite alphabet $\mathcal{G} = \{A_1, \dots, A_K\}$, the *Parikh Image* of a word $w = M_1 \cdots M_m$ over the alphabet $\mathcal{G}$ is the vector $\mathrm{PI}^{\mathcal{G}}(w) \coloneqq (\mathrm{PI}^{\mathcal{G}}_1(w), \dots, \mathrm{PI}^{\mathcal{G}}_K(w)) \in \mathbb{Z}^K_{\geq 0}$, where $\mathrm{PI}^{\mathcal{G}}_i(w)$ is the number of times A_i appears in w. That is, $\mathrm{PI}^{\mathcal{G}}_i(w) \coloneqq \mathrm{card}(\{j \mid M_j = A_i\})$. When the alphabet $\mathcal{G}$ is clear from the context, we sometimes write $\mathrm{PI}(w), \mathrm{PI}_i(w)$ instead of $\mathrm{PI}^{\mathcal{G}}(w), \mathrm{PI}^{\mathcal{G}}_i(w)$.

For $1 \le i < j \le K$, let w be a word over the alphabet $\mathcal{G}$, denote by $\delta^{\mathcal{G}}_{ij}(w)$ the number of occurrences of the subword $\cdots A_i \cdots A_j \cdots$ minus the number of occurrences of the subword $\cdots A_j \cdots A_i \cdots$ in w. That is, writing $w = M_1 M_2 \cdots M_s$, we have

$$\delta^{\mathcal{G}}_{ij}(w) \coloneqq \delta^{\mathcal{G},+}_{ij}(w) - \delta^{\mathcal{G},-}_{ij}(w),$$

where

$$\delta^{\mathcal{G},+}_{ij}(w) \coloneqq \{(u,v) \mid 1 \le u < v \le s, M_u = A_i, M_v = A_j\},$$
$$\delta^{\mathcal{G},-}_{ij}(w) \coloneqq \{(u,v) \mid 1 \le u < v \le s, M_u = A_j, M_v = A_i\}$$

Again, if the alphabet $\mathcal{G}$ is clear from the context, we write $\delta_{ij}(w)$ instead of $\delta^{\mathcal{G}}_{ij}(w)$. Obviously, we have the parity constraint

$$\delta_{ij}(w) \equiv \delta^{+}_{ij}(w) + \delta^{-}_{ij}(w) = \mathrm{PI}_i(w) \cdot \mathrm{PI}_j(w) \mod 2. \tag{1}$$

The Baker-Campbell-Hausdorff formula

Let G be a 2-step nilpotent subgroup of $\mathsf{UT}(n, \mathbb{Q})$. The *Baker-Campbell-Hausdorff (BCH) formula* [3, 9, 18] states that, given a sequence of matrices $B_1, B_2, \dots, B_s$ in G, we have

$$\log(B_1 B_2 \cdots B_m) = \sum_{i=1}^{m} \log B_i + \frac{1}{2} \sum_{1 \le i < j \le s} [\log B_i, \log B_j]. \tag{2}$$

Fix a finite alphabet $\mathcal{G} = \{A_1, \dots, A_K\}$ in G. For an arbitrary word w with Parikh Image $\boldsymbol{\ell} = (\ell_1, \dots, \ell_K)$, applying Equation (2) to the sequence of matrices in w yields

$$\log w = \sum_{i=1}^{K} \ell_i \log A_i + \frac{1}{2} \sum_{1 \le i < j \le K} \delta_{ij}(w) [\log A_i, \log A_j]. \tag{3}$$

Here, $\log w$ is understood to be the result of multiplying all matrices appearing in w in order, then taking the logarithm. We will adopt this notation throughout this paper.

4 A combinatorial problem for length two subwords

First let us describe the general strategy for solving intersection-type decision problems. Consider a simple example: given two alphabets $\mathcal{G} = \{A_1, \dots, A_K\}$, $\mathcal{H} = \{B_1, \dots, B_M\}$ in a 2-step nilpotent subgroup of $\mathsf{UT}(n, \mathbb{Q})$, we want to decide whether $\langle \mathcal{G} \rangle \cap \langle \mathcal{H} \rangle \neq \emptyset$. This boils down to finding two words v, w respectively in the alphabet $\mathcal{G}$ and $\mathcal{H}$, such that $\log v = \log w$. Denote by $\boldsymbol{x} = (x_1, \dots, x_K)$ the Parikh Image of v, and by $\boldsymbol{y} = (y_1, \dots, y_M)$ the Parikh Image of w, then the BCH formula (3) yields the equivalence between $\log v = \log w$ and

$$\sum_{i=1}^{K} x_i \log A_i + \sum_{i<j} \frac{\delta^{\mathcal{G}}_{ij}(v)}{2} [\log A_i, \log A_j] = \sum_{i=1}^{M} y_i \log B_i + \sum_{i<j} \frac{\delta^{\mathcal{H}}_{ij}(w)}{2} [\log B_i, \log B_j],$$
$$\boldsymbol{x} \in \mathbb{Z}^K_{\ge 0},\ \boldsymbol{y} \in \mathbb{Z}^M_{\ge 0}, \quad \mathrm{PI}^{\mathcal{G}}(v) = \boldsymbol{x},\ \mathrm{PI}^{\mathcal{H}}(w) = \boldsymbol{y}. \tag{4}$$

Hence, deciding whether $\langle \mathcal{G} \rangle \cap \langle \mathcal{H} \rangle \neq \emptyset$ boils down to solving Equation (4) in the numerical variables $\boldsymbol{x}, \boldsymbol{y}$ and the word variables v, w over alphabets $\mathcal{G}, \mathcal{H}$.

Consider a “relaxed” version of this problem. That is, we replace $\delta_{ij}^{\mathcal{G}}(v)$ and $\delta_{ij}^{\mathcal{H}}(w)$ by new variables c_{ij}, d_{ij} over integers, without imposing any constraint. This gives the equation

$$\sum_{i=1}^{K} x_i \log A_i + \sum_{1\leq i<j\leq K} \frac{c_{ij}}{2}[\log A_i, \log A_j] = \sum_{i=1}^{M} y_i \log B_i + \sum_{1\leq i<j\leq M} \frac{d_{ij}}{2}[\log B_i, \log B_j],$$
$$\boldsymbol{x} \in \mathbb{Z}_{\geq 0}^K,\ \boldsymbol{y} \in \mathbb{Z}_{\geq 0}^M, \quad c_{ij}, d_{ij} \in \mathbb{Z} \text{ for all } i, j. \tag{5}$$

Obviously, if Equation (4) has a solution, then the relaxed version (5) will also admit a solution. The converse is not necessarily true. The implicit constraints imposed by the word combinatorial variables $\delta_{ij}^{\mathcal{G}}(v), \delta_{ij}^{\mathcal{H}}(w)$ in Equation (4) are highly non-trivial. (For example, one should at least have $|\delta_{ij}^{\mathcal{G}}(v)| \leq x_i x_j$ for all i, j). However, these constraints are not reflected by the numerical variables c_{ij} and d_{ij} in Equation (5).

The key idea of this paper is the following surprising fact. For the two problems we consider (Semigroup Intersection and Orbit Intersection), it is sufficient to solve the relaxed version of the equation, plus several simple constraints (such as the modulo 2 constraint in Equation (1)). In particular, given a “suitable” solution to the relaxed Equation (5), we can always construct a solution to Equation (4). *A priori*, the values of $\delta_{ij}^{\mathcal{G}}(v)$ cannot reach all integers like the free variables c_{ij}; nevertheless, when $x_1, \ldots, x_K$ tend towards infinity, the vector $\left(\delta_{ij}^{\mathcal{G}}(v)\right)_{1\leq i<j\leq K}$ can in fact reach every value within a ball of radius size $O(|\boldsymbol{x}|^2)$, satisfying modulo 2 constraints. This will suffice to construct a suitable word v, as the quadratic radius will eventually dominate the linear term $\sum_{i=1}^K x_i \log A_i$.

This section aims to formalize this idea. The main result of this section will be Proposition 6. First, we prove a simple case where the alphabet consists of two letters.

▶ **Lemma 5.** *Given an alphabet $\mathcal{G} = \{A_i, A_j\}$ and non-negative integers $s_i, s_j \in \mathbb{Z}_{\geq 0}$, then for every $C \in \mathbb{Z}$ satisfying*

$$|C| \leq s_i s_j \quad \text{and} \quad C \equiv s_i s_j \mod 2, \tag{6}$$

there exists a permutation w of the word $A_i^{s_i} A_j^{s_j}$ such that $\delta_{ij}(w) = C$.

Proof. For an illustration of the proof, see Figure 1. We start with the word $w = A_i^{s_i} A_j^{s_j}$, which satisfies $\delta_{ij}(w) = s_i s_j$. We gradually swap pairs of consecutive letters in w: each time we replace an occurrence of consecutive $A_i A_j$ with $A_j A_i$. An occurrence of $A_i A_j$ can always be found unless we have reached the “final” permutation $A_j^{s_j} A_i^{s_i}$. It is easy to see that each swap reduces the value of $\delta_{ij}(w)$ by 2. Therefore, by swapping consecutive $A_i A_j$ one by one, $\delta_{ij}(w)$ can reach every value between $\delta_{ij}(A_i^{s_i} A_j^{s_j}) = s_i s_j$ and $\delta_{ij}(A_j^{s_j} A_i^{s_i}) = -s_i s_j$ that has the same parity with $s_i s_j$. This proves the lemma. ◀

We then prove the main result of this section, which generalizes Lemma 5 to alphabets of more than two letters.

▶ **Proposition 6.** *Fix a finite alphabet $\mathcal{G}$ of size $K \geq 2$. Then for any tuples $\boldsymbol{\ell} = (\ell_1, \ldots, \ell_K) \in \mathbb{Z}_{\geq 0}^K$ and $\{C_{ij}\}_{1\leq i<j\leq K} \in \mathbb{Z}_{\geq 0}^{K(K-1)/2}$ satisfying*

$$|C_{ij}| \leq \frac{\ell_i \ell_j}{4K^2} - 2K(\ell_i + \ell_j) - 4K^2, \quad \text{for all } 1 \leq i < j \leq K, \tag{7}$$

and

$$C_{ij} \equiv \ell_i \ell_j \mod 2, \quad \text{for all } 1 \leq i < j \leq K, \tag{8}$$

there exists a word w with Parikh Image $\boldsymbol{\ell}$ such that

$$\delta_{ij}(w) = C_{ij}, \quad \text{for all } 1 \leq i < j \leq K. \tag{9}$$

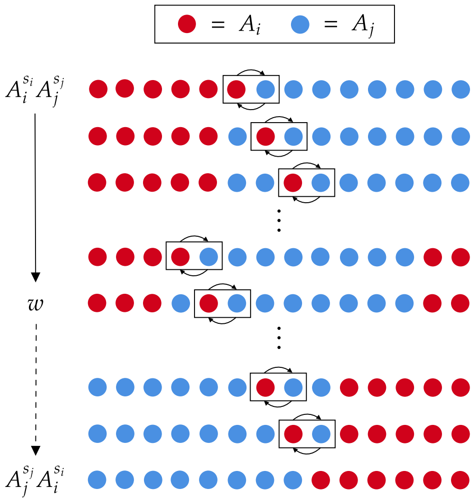

Figure 1 Illustration for the proof of Lemma 5.

Proof. For an illustration of the proof, see Figure 2. For all i, write $\ell_i = 2(K-1)s_i + r_i$ with $0 \leq r_i < 2(K-1)$. Consider the word $W_{init} := W_{res} \cdot W \cdot W_{rev}$, where

$$\begin{aligned}
W_{res} &:= A_1^{r_1} A_2^{r_2} \cdots A_K^{r_K}, \\
W &:= (A_1^{s_1} A_2^{s_2})\,(A_1^{s_1} A_3^{s_3}) \cdots (A_1^{s_1} A_K^{s_K})\,(A_2^{s_2} A_3^{s_3}) \cdots (A_2^{s_2} A_K^{s_K})\,(A_3^{s_3} A_4^{s_4}) \cdots \left(A_{K-1}^{s_{K-1}} A_K^{s_K}\right), \\
W_{rev} &:= \left(A_K^{s_K} A_{K-1}^{s_{K-1}}\right) \left(A_K^{s_K} A_{K-2}^{s_{K-2}}\right) \cdots (A_K^{s_K} A_1^{s_1}) \left(A_{K-1}^{s_{K-1}} A_{K-2}^{s_{K-2}}\right) \left(A_{K-1}^{s_{K-1}} A_{K-3}^{s_{K-3}}\right) \cdots (A_2^{s_2} A_1^{s_1}).
\end{aligned}$$

In particular, W is the concatenation of all words of the form $A_i^{s_i} A_j^{s_j}$ where $i < j$, and W_{rev} is the reverse of W. It is easy to verify that W_{init} contains ℓ_i occurrences of the letter A_i, so its Parikh Image is exactly $\boldsymbol{\ell}$.

We now compute $\delta_{ij}(W_{init})$ for $i < j$. Since $W \cdot W_{rev}$ is a palindrome, we have $\delta_{ij}(W \cdot W_{rev}) = 0$, so

$$\begin{aligned}
\delta_{ij}(W_{init}) &= \delta_{ij}(W_{res}) + \mathrm{PI}_i(W_{res})\,\mathrm{PI}_j(W \cdot W_{rev}) - \mathrm{PI}_j(W_{res})\,\mathrm{PI}_i(W \cdot W_{rev}) \\
&= r_i r_j + r_i \cdot 2(K-1)s_j - r_j \cdot 2(K-1)s_i. \qquad (10)
\end{aligned}$$

In particular, since $0 \leq r_i < 2(K-1)$, we have

$$|\delta_{ij}(W_{init})| \leq 4(K-1)^2 + 2(K-1)^2(s_j + s_i) < 4K^2 + 2(K-1)(\ell_i + \ell_j) \qquad (11)$$

By Condition 7, we have

$$\begin{aligned}
|\delta_{ij}(W_{init}) - C_{ij}| &\leq |\delta_{ij}(W_{init})| + |C_{ij}| \\
&< 4K^2 + 2(K-1)(\ell_i + \ell_j) + \frac{\ell_i \ell_j}{4K^2} - 2K(\ell_i + \ell_j) - 4K^2 \\
&< \left(\frac{\ell_i}{2(K-1)} - 1\right)\left(\frac{\ell_j}{2(K-1)} - 1\right) \\
&< s_i s_j. \qquad (12)
\end{aligned}$$

Since Equation (10) yields $\delta_{ij}(W_{init}) \equiv \ell_i \ell_j \mod 2$, Condition (8) then gives

$$\delta_{ij}(W_{init}) \equiv C_{ij} \mod 2. \qquad (13)$$

We now show how to construct the word w. Starting with the word W_{init}, for every pair $i < j$ perform the following:

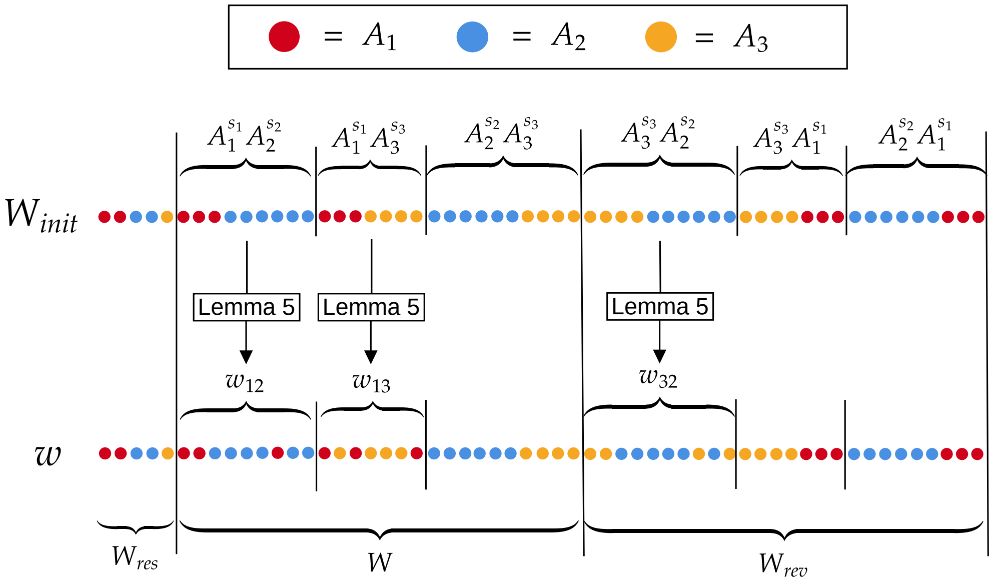

Figure 2 Illustration for the proof of Proposition 6.

1. If $\delta_{ij}(W_{init}) > C_{ij}$. By Lemma 5 there exists a permutation w_{ij} of the word $A_i^{s_i}A_j^{s_j}$ such that $\delta_{ij}(w_{ij}) = s_i s_j + C_{ij} - \delta_{ij}(W_{init})$. Indeed, Equations (12) and (13) guarantee that the conditions (6) in Lemma 5 are satisfied. We then replace the subword $A_i^{s_i}A_j^{s_j}$ in the W-part of W_{init} with the word w_{ij}. The resulting new word W' will satisfy

$$\delta_{ij}(W') = \delta_{ij}(W_{init}) - \delta_{ij}(A_i^{s_i}A_j^{s_j}) + \delta_{ij}(w_{ij}) = C_{ij}.$$

This replacement does not change $\delta_{uv}(W_{init})$ for $(u, v) \neq (i, j)$.

2. If $\delta_{ij}(W_{init}) < C_{ij}$. By Lemma 5 there exists a permutation w_{ji} of the word $A_j^{s_j}A_i^{s_i}$ such that $\delta_{ij}(w_{ji}) = -s_i s_j + C_{ij} - \delta_{ij}(W_{init})$. Again, Equations (12) and (13) guarantee that the conditions (6) in Lemma 5 are satisfied. We then replace the subword $A_j^{s_j}A_i^{s_i}$ in the W_{rev}-part of W_{init} with the word w_{ji}. The resulting new word W' will satisfy

$$\delta_{ij}(W') = \delta_{ij}(W_{init}) - \delta_{ij}(A_j^{s_j}A_i^{s_i}) + \delta_{ij}(w_{ji}) = C_{ij}.$$

This replacement does not change $\delta_{uv}(W_{init})$ for $(u, v) \neq (i, j)$.

3. If $\delta_{ij}(W_{init}) = C_{ij}$, do not perform any change.

Performing all these replacements on W_{init} for all pairs $i < j$ simultaneously, the resulting word w then satisfies $\delta_{ij}(w) = C_{ij}$ for all $1 \leq i < j \leq K$. ◀

5 A polynomial time algorithm for Intersection Emptiness

We prove Theorem 1 in this section. Let G be a 2-step nilpotent subgroup of $\mathsf{UT}(n, \mathbb{Q})$. Let

$$\mathcal{G}_1 = \{A_{11}, A_{12}, \ldots, A_{1K_1}\}, \ldots, \mathcal{G}_M = \{A_{M1}, A_{M2}, \ldots, A_{MK_M}\}$$

be M sets of matrices in G. The following proposition shows that Intersection Emptiness can be reduced to solving linear Diophantine equations with extra constraints on supports. The key to obtaining a PTIME algorithm is the fact that these equations are all *homogeneous*. Hence one can actually solve them over $\mathbb{Q}$, then scale them to obtain integer solutions.

▶ **Proposition 7.** *We have* $\langle \mathcal{G}_1 \rangle \cap \cdots \cap \langle \mathcal{G}_M \rangle \neq \emptyset$ *if and only if there exist non-zero vectors* $\boldsymbol{\ell}_1 \in \mathbb{Z}_{\geq 0}^{K_1} \setminus \{\mathbf{0}\}, \cdots, \boldsymbol{\ell}_M \in \mathbb{Z}_{\geq 0}^{K_m} \setminus \{\mathbf{0}\}$ *as well as rational numbers* c_{mij} *for* $1 \leq m \leq M, i, j \in \operatorname{supp}(\boldsymbol{\ell}_m)$*, such that*

$$\sum_{j=1}^{K_1} \ell_{1j} \log A_{1j} + \sum_{\substack{i<j \\ i,j \in \operatorname{supp}(\boldsymbol{\ell}_1)}} c_{1ij} [\log A_{1i}, \log A_{1j}] =$$

$$\sum_{j=1}^{K_2} \ell_{2j} \log A_{2j} + \sum_{\substack{i<j \\ i,j \in \operatorname{supp}(\boldsymbol{\ell}_2)}} c_{2ij} [\log A_{2i}, \log A_{2j}] = \cdots$$

$$= \sum_{j=1}^{K_M} \ell_{Mj} \log A_{Mj} + \sum_{\substack{i<j \\ i,j \in \operatorname{supp}(\boldsymbol{\ell}_m)}} c_{Mij} [\log A_{Mi}, \log A_{Mj}] \quad (14)$$

Proof. If $\langle \mathcal{G}_1 \rangle \cap \cdots \cap \langle \mathcal{G}_M \rangle \neq \emptyset$, let g be an element in the intersection. There exist non-empty words $w_1, \ldots, w_m$ over the alphabets $\mathcal{G}_1, \ldots, \mathcal{G}_M$ such that $\log g = \log w_1 = \cdots = \log w_m$. By the BCH formula (3),

$$\log g = \sum_{j=1}^{K_i} \mathrm{PI}_i(w_m) \log A_{mj} + \sum_{\substack{i<j \\ i,j \in \operatorname{supp}(\boldsymbol{\ell}_m)}} \frac{\delta_{ij}(w_m)}{2} [\log A_{mi}, \log A_{mj}], \qquad \text{for } m = 1, \ldots, M.$$

This shows that (14) is satisfied by $\boldsymbol{\ell}_m := \mathrm{PI}^{\mathcal{G}_m}(w_m)$ and $c_{mij} := \delta_{ij}(w_m)/2$ for $1 \leq m \leq M, i, j \in \operatorname{supp}(\boldsymbol{\ell}_m)$.

For the other implication, suppose such non-zero vectors $\boldsymbol{\ell}_1, \ldots, \boldsymbol{\ell}_M$ and the rational numbers c_{mij} exist. Then there exists $g \in G$ such that

$$\sum_{j=1}^{K_1} \ell_{mj} \log A_{mj} + \sum_{\substack{i<j \\ i,j \in \operatorname{supp}(\boldsymbol{\ell}_m)}} \frac{2c_{mij}}{2} [\log A_{mi}, \log A_{mj}] = \log g, \qquad \text{for } m = 1, \ldots, M. \quad (15)$$

Note that if $i, j \in \operatorname{supp}(\boldsymbol{\ell}_m)$ then $\ell_{mi}\ell_{mj} \neq 0$.

By homogeneity, for any $N \in \mathbb{Z}_{>0}$, the vectors $N\boldsymbol{\ell}_1, \ldots, N\boldsymbol{\ell}_M$ and Nc_{mij} also satisfy Condition (14). Hence, multiplying all ℓ_{ij}, c_{mij} and $\log g$ by a common denominator, we can suppose all ℓ_{ij} and c_{mij} to be integers. Denote $K = \max_{1 \leq m \leq M} K_m$, then there exists a large enough *even* integer $N \in \mathbb{Z}_{>0}$ such that

$$|N \cdot 2c_{mij}| \leq \frac{N^2 \ell_{mi}\ell_{mj}}{4K^2} - 2NK(\ell_i + \ell_j) - 4K^2 \quad (16)$$

for $1 \leq m \leq M, i, j \in \operatorname{supp}(\boldsymbol{\ell}_m)$. This is because $\ell_{mi}\ell_{mj} > 0$, so the right hand side of (16) is quadratic and dominates the linear term on the left for large enough N. Replace all ℓ_{ij} with $N\ell_{ij}$, all c_{mij} with Nc_{mij}, and $\log g$ with $N \log g$, then the new variables satisfy $2c_{mij} \equiv 0 \equiv \ell_{mi}\ell_{mj} \mod 2$, and

$$|2c_{mij}| \leq \frac{\ell_{mi}\ell_{mj}}{4K^2} - 2K(\ell_i + \ell_j) - 4K^2 \leq \frac{\ell_{mi}\ell_{mj}}{4K_m^2} - 2K_m(\ell_i + \ell_j) - 4K_m^2 \quad (17)$$

for all i, j, m. Equation (15) is still satisfied after the variable replacements. Therefore, by Proposition 6, there exist words $w_1, \ldots, w_M$ over the alphabets $\mathcal{G}_1, \ldots, \mathcal{G}_M$ such that $\mathrm{PI}(w_m) = \boldsymbol{\ell}_m$ and $\delta_{ij}(w_m) = 2c_{mij}$ for all $1 \leq m \leq M, i, j \in \operatorname{supp}(\boldsymbol{\ell}_m)$. These words are non-empty since $\boldsymbol{\ell}_m \neq \mathbf{0}$. Plugging into the BCH formula (3), we have

$$\log w_m = \sum_{j=1}^{K_m} \ell_{mj} \log A_{mj} + \sum_{\substack{i<j \\ i,j \in \mathrm{supp}(\boldsymbol{\ell}_m)}} \frac{2c_{mij}}{2} [\log A_{mi}, \log A_{mj}] = \log g \quad \text{for } m = 1, \ldots, M.$$

This shows that $\log g \in \bigcap_{i=1}^{M} \log\langle \mathcal{G}_i \rangle \neq \emptyset$. ◀

Using Proposition 7, we devise Algorithm 1 that decides Intersection Emptiness.

Algorithm 1 Algorithm for Intersection Emptiness.

Input: M finite sets of matrices $\mathcal{G}_1 = \{A_{11}, A_{12}, \ldots, A_{1K_1}\}, \ldots, \mathcal{G}_M = \{A_{M1}, A_{M2}, \ldots, A_{MK_M}\}$ in the group G.

Output: **True** (intersection is empty) or **False** (intersection is not empty).

Step 1: **Initialization.** Set $S_1 \coloneqq \{1, 2, \ldots, K_1\}, \ldots, S_M \coloneqq \{1, 2, \ldots, K_M\}$.

Step 2: **Main loop.** Repeat the following

a. Represent the $\mathbb{Q}$-linear subspace of $V \coloneqq \mathbb{Q}^{\sum_{m=1}^{M} K_m + \sum_{m=1}^{M} \mathrm{card}(S_m)(\mathrm{card}(S_m)-1)/2}$:

$$\begin{aligned} W \coloneqq \Bigg\{ \big((\ell_{mj})_{1 \le m \le M, 1 \le j \le K_m}, (c_{mij})_{1 \le m \le M, i,j \in S_m} \big) \in V \;\Bigg|\; & \\ \sum_{j=1}^{K_1} \ell_{1j} \log A_{1j} + \sum_{\substack{i<j \\ i,j \in S_1}} c_{1ij} [\log A_{1i}, \log A_{1j}] = \cdots & \\ = \sum_{j=1}^{K_M} \ell_{Mj} \log A_{Mj} + \sum_{\substack{i<j \\ i,j \in S_M}} c_{Mij} [\log A_{Mi}, \log A_{Mj}] \Bigg\} & \end{aligned} \tag{18}$$

as the solution set of homogeneous linear equations.

b. Compute the projection of W onto the coordinates $(\ell_{mj})_{1 \le m \le M, 1 \le j \le K_m}$:

$$\begin{aligned} \pi_{\boldsymbol{\ell}}(W) \coloneqq \Bigg\{ (\ell_{mj})_{1 \le m \le M, 1 \le j \le K_m} \in \mathbb{Q}^{\sum_{m=1}^{M} K_m} \;\Bigg|\; \exists (c_{mij})_{1 \le m \le M, i,j \in S_m}, & \\ \big((\ell_{mj})_{1 \le m \le M, 1 \le j \le K_m}, (c_{mij})_{1 \le m \le M, i,j \in S_m} \big) \in W \Bigg\} & \end{aligned} \tag{19}$$

represented as the solution set of homogeneous linear equations.

c. Define $\Lambda \coloneqq \mathbb{Z}_{\ge 0}^{\sum_{m=1}^{M} K_m} \cap \pi_{\boldsymbol{\ell}}(W)$ and compute $\mathrm{supp}(\Lambda)$ using Lemma 4.

d. If $\mathrm{supp}(\Lambda) \cap S_m = S_m$ for all $1 \le m \le M$, terminate the loop and go to Step 3. Otherwise, let $S_m \coloneqq \mathrm{supp}(\Lambda) \cap S_m$ for every m, and continue with Step 2.

Step 3: **Output.**

a. If $S_m = \emptyset$ for any $1 \le m \le M$, return **True**.

b. Otherwise return **False**.

▶ **Theorem 1.** *Let G be a 2-step nilpotent subgroup of $\mathsf{UT}(n, \mathbb{Q})$ for some n. Given finite subsets $\mathcal{G}_1, \ldots, \mathcal{G}_M$ of G, it is decidable in polynomial time whether $\langle \mathcal{G}_1 \rangle \cap \cdots \cap \langle \mathcal{G}_M \rangle = \emptyset$.*

Proof. Theorem 1 follows from the correctness and polynomial time complexity of Algorithm 1. Their proofs are given in Appendix A, Proposition 10. ◀

6 Decidability of Orbit Intersection in $\mathrm{H}_3(\mathbb{Q})$

We prove Theorem 3 in this section. Let $\mathcal{G}$ and $\mathcal{H}$ be finite sets of matrices in the group $\mathrm{H}_3(\mathbb{Q})$, and T, S be matrices in $\mathrm{H}_3(\mathbb{Q})$. Our goal is to decide whether $T \cdot \langle \mathcal{G} \rangle \cap S \cdot \langle \mathcal{H} \rangle = \emptyset$. Multiplying both $T \cdot \langle \mathcal{G} \rangle$ and $S \cdot \langle \mathcal{H} \rangle$ on the left by T^{-1}, one can without loss of generality suppose $T = I$. That is, it suffices to consider the problem of deciding whether $\langle \mathcal{G} \rangle \cap S \cdot \langle \mathcal{H} \rangle = \emptyset$. Denote by $\varphi \colon \log \mathrm{H}_3(\mathbb{Q}) \to \mathbb{Q}^2$ the projection onto the superdiagonal, and by $\pi \colon \log \mathrm{H}_3(\mathbb{Q}) \to \mathbb{Q}$ the projection onto the upper right entry:

$$\varphi \colon \begin{pmatrix} 0 & a & c \\ 0 & 0 & b \\ 0 & 0 & 0 \end{pmatrix} \mapsto (a, b); \qquad \pi \colon \begin{pmatrix} 0 & a & c \\ 0 & 0 & b \\ 0 & 0 & 0 \end{pmatrix} \mapsto c.$$

One easily verifies that for matrices $X, Y \in \mathrm{H}_3(\mathbb{Q})$, we have $[\log X, \log Y] = 0$ if and only if $\varphi(\log X)$ and $\varphi(\log Y)$ are linearly dependent. Define the cones

$$\mathcal{C}_{\mathcal{G}} \coloneqq \langle \varphi(\log \mathcal{G}) \rangle_{\mathbb{Q}_{\geq 0}}, \quad \mathcal{C}_{\mathcal{H}} \coloneqq \langle \varphi(\log \mathcal{H}) \rangle_{\mathbb{Q}_{\geq 0}}.$$

6.1 Easy case: The cone $\mathcal{C}_{\mathcal{G}} \cap \mathcal{C}_{\mathcal{H}}$ has dimension zero or one

The situation in this case is similar to the one discussed in [12, Section 3, Case I].

▶ **Proposition 8.** *Suppose the cone $\mathcal{C}_{\mathcal{G}} \cap \mathcal{C}_{\mathcal{H}}$ has dimension zero or one. Deciding whether $\langle \mathcal{G} \rangle \cap S \cdot \langle \mathcal{H} \rangle \neq \emptyset$ can be done by solving finitely many linear Diophantine equations.*

6.2 Hard case: The cone $\mathcal{C}_{\mathcal{G}} \cap \mathcal{C}_{\mathcal{H}}$ has dimension two

We have $\langle \mathcal{G} \rangle \cap S \cdot \langle \mathcal{H} \rangle \neq \emptyset$ if and only if there exist words v in the alphabet $\mathcal{G}$ and w in the alphabet $\mathcal{H}$ such that $\log v = \log Sw$. Let $\boldsymbol{x} = (x_1, \ldots, x_K)$ be the Parikh Image of v, and $\boldsymbol{y} = (y_1, \ldots, y_M)$ be the Parikh Image of w. By the BCH formula (2) and (3), $\log v = \log Sw$ is equivalent to

$$\sum_{i=1}^{K} x_i \log A_i + \frac{1}{2} \sum_{1 \leq i < j \leq K} \delta_{ij}^{\mathcal{G}}(v) [\log A_i, \log A_j] = \\ \log S + \sum_{i=1}^{M} y_i (\log B_i + \frac{1}{2}[\log S, \log B_i]) + \frac{1}{2} \sum_{1 \leq i < j \leq M} \delta_{ij}^{\mathcal{H}}(w) [\log B_i, \log B_j] \quad (20)$$

The following proposition shows that it suffices to solve a relaxed version of Equation (20).

▶ **Proposition 9.** *Suppose the cone $\mathcal{C}_{\mathcal{G}} \cap \mathcal{C}_{\mathcal{H}}$ has dimension two. We have $\langle \mathcal{G} \rangle \cap S \cdot \langle \mathcal{H} \rangle \neq \emptyset$ if and only if there exists integers $x_i, 1 \leq i \leq K$ and $y_j, 1 \leq j \leq M$ and $c_{ij}, 1 \leq i < j \leq K$ and $d_{ij}, 1 \leq i < j \leq M$, satisfying*

$$\sum_{i=1}^{K} x_i \varphi(\log A_i) = \varphi(\log S) + \sum_{i=1}^{M} y_i \varphi(\log B_i), \quad (21)$$

$$\sum_{i=1}^{K} x_i \pi(\log A_i) + \frac{1}{2} \sum_{1 \leq i < j \leq K} c_{ij} \pi([\log A_i, \log A_j]) = \\ \pi(\log S) + \sum_{i=1}^{M} y_i \pi(\log B_i + \frac{1}{2}[\log S, \log B_i]) + \frac{1}{2} \sum_{1 \leq i < j \leq M} d_{ij} \pi([\log B_i, \log B_j]) \quad (22)$$

and

$$c_{ij} \equiv x_i x_j \mod 2, \quad 1 \le i < j \le K; \qquad d_{ij} \equiv y_i y_j \mod 2, \quad 1 \le i < j \le M. \tag{23}$$

Proof. If $\langle \mathcal{G} \rangle \cap s \cdot \langle \mathcal{H} \rangle \neq \emptyset$, then let v, w be non-empty words over the respectively alphabets $\mathcal{G}$ and $\mathcal{H}$, such that $\log v = \log Sw$. Let $c_{ij} \coloneqq \delta^{\mathcal{G}}_{ij}(v)$ and $d_{ij} \coloneqq \delta^{\mathcal{H}}_{ij}(w)$ for all i, j. Since Equation (20) is satisfied, projecting it under φ and π gives respectively (21) and (22). The parity condition (23) is obviously due to Equation (1). Hence we have found the integers x_i, y_j, c_{ij}, d_{ij} satisfying Equations (21), (22) and (23).

On the other hand, let x_i, y_j, c_{ij}, d_{ij} be integers that satisfy Equations (21), (22) and (23). Since $\mathcal{C}_{\mathcal{G}}$ and $\mathcal{C}_{\mathcal{H}}$ have dimension two, the commutators $[\log A_i, \log A_j]$ and $[\log B_i, \log B_j]$ are not all zero (since $\varphi(A_i)$ are not all linearly dependant, same for $\varphi(B_i)$). Hence, there exist integers C_{ij}, D_{ij} such that

$$D \coloneqq \sum_{1 \le i < j \le K} C_{ij} \pi([\log A_i, \log A_j]) + \sum_{1 \le i < j \le M} D_{ij} \pi([\log B_i, \log B_j]) \in \mathbb{Q}_{>0}.$$

Denote by E the common denominator of all the entries of the matrices $\log A_i$, $\log B_i$, $\log S$, $\frac{1}{2}[\log S, \log B_i]$, $\frac{1}{2}[\log A_i, \log A_j]$ and $\frac{1}{2}[\log B_i, \log B_j]$. In particular, DE is an integer.

Since the cone $\mathcal{C}_{\mathcal{G}} \cap \mathcal{C}_{\mathcal{H}}$ has dimension two, there exist *strictly positive* integers $X_1, \dots, X_K$ and $Y_1, \dots, Y_M$, such that

$$\sum_{i=1}^{K} X_i \varphi(\log A_i) = \sum_{i=1}^{M} Y_i \varphi(\log B_i). \tag{24}$$

This is because, taking $\boldsymbol{v}$ to be a vector in the *interior* of $\mathcal{C}_{\mathcal{G}} \cap \mathcal{C}_{\mathcal{H}}$ (i.e. $\boldsymbol{v}$ admits an open neighbourhood contained in $\mathcal{C}_{\mathcal{G}} \cap \mathcal{C}_{\mathcal{H}}$), then $\boldsymbol{v}$ is in the interior of both $\mathcal{C}_{\mathcal{G}}$ and $\mathcal{C}_{\mathcal{H}}$. Hence, there exist strictly positive rational numbers $X'_1, \dots, X'_K$ and $Y'_1, \dots, Y'_M$, such that

$$\sum_{i=1}^{K} X'_i \varphi(\log A_i) = \boldsymbol{v} = \sum_{i=1}^{M} Y'_i \varphi(\log B_i).$$

Multiplying $X'_1, \dots, X'_K$ and $Y'_1, \dots, Y'_M$ by their common denominator gives positive integers satisfying Equation (24).

For any $N \in \mathbb{Z}_{>0}$, the integers x_i, y_i, c_{ij}, d_{ij} can be replaced by the integers

$$\begin{aligned}
x'_i &\coloneqq x_i + 2NDEX_i \\
y'_i &\coloneqq y_i + 2NDEY_i \\
c'_{ij} &\coloneqq c_{ij} - 4NEC_{ij} \left(\sum_{k=1}^{K} X_k \pi(\log A_k) - \sum_{k=1}^{M} Y_k \pi(\log B_k + \frac{1}{2}[\log S, \log B_k]) \right) \\
d'_{ij} &\coloneqq d_{ij} + 4NED_{ij} \left(\sum_{k=1}^{K} X_k \pi(\log A_k) - \sum_{k=1}^{M} Y_k \pi(\log B_k + \frac{1}{2}[\log S, \log B_k]) \right)
\end{aligned}$$

for all i, j, while still satisfying Equations (21), (22) and (23). Furthermore, when N is large enough, we have

$$x'_i > 0, y'_j > 0, \quad 1 \le i \le K, 1 \le j \le M, \tag{25}$$

$$|c'_{ij}| \le \frac{x'_i x'_j}{4K^2} - 2K(x'_i + x'_j) - 4K^2, \quad 1 \le i < j \le K, \tag{26}$$

and

$$|d'_{ij}| \leq \frac{y'_i y'_j}{4M^2} - 2M(y'_i + y'_j) - 4M^2, \quad 1 \leq i < j \leq M. \tag{27}$$

This is because the right hand sides of the inequalities (26) and (27) are quadratic in N, whereas the left hand sides grow linearly in N.

Fix an N such that the inequalities (25), (26) and (27) are satisfied. Then, by Proposition 6, there exist non-empty words v, w over the alphabets $\mathcal{G}$ and $\mathcal{H}$, such that

$$\begin{aligned} \mathrm{PI}^{\mathcal{G}}(v) &= (x'_1, \dots, x'_K), \quad \delta^{\mathcal{G}}_{ij}(v) = c'_{ij}, \quad \text{for } 1 \leq i < j \leq K, \\ \mathrm{PI}^{\mathcal{H}}(w) &= (y'_1, \dots, y'_K), \quad \delta^{\mathcal{H}}_{ij}(v) = d'_{ij}, \quad \text{for } 1 \leq i < j \leq M. \end{aligned}$$

(Note that Condition (8) is guaranteed by Equation (23).) For these words v, w, we have

$$\varphi(\log v) = \sum_{i=1}^{K} x'_i \varphi(\log A_i) = \varphi(\log S) + \sum_{i=1}^{M} y'_i \varphi(\log B_i) = \varphi(\log Sw),$$

as well as

$$\pi(\log v) = \sum_{i=1}^{K} x'_i \pi(\log A_i) + \frac{1}{2} \sum_{1 \leq i < j \leq K} c'_{ij} \pi([\log A_i, \log A_j]) =$$

$$\pi(\log S) + \sum_{i=1}^{M} y'_i \pi(\log B_i + \frac{1}{2}[\log S, \log B_i]) + \frac{1}{2} \sum_{1 \leq i < j \leq M} d'_{ij} \pi([\log B_i, \log B_j]) = \pi(\log Sw).$$

This shows $\log v = \log Sw$, hence $\langle \mathcal{G} \rangle \cap s \cdot \langle \mathcal{H} \rangle \neq \emptyset$. ◀

Combining the two cases in Subsections 6.1 and 6.2, we are able to solve the Orbit Intersection problem for $\mathrm{H}_3(\mathbb{Q})$.

▶ **Theorem 3.** *Given elements $T, S \in \mathrm{H}_3(\mathbb{Q})$ and two finite subsets $\mathcal{G}, \mathcal{H}$ of $\mathrm{H}_3(\mathbb{Q})$, it is decidable whether $T \cdot \langle \mathcal{G} \rangle \cap S \cdot \langle \mathcal{H} \rangle = \emptyset$.*

Proof. See Appendix A. ◀

References

1. László Babai, Robert Beals, Jin-yi Cai, Gábor Ivanyos, and Eugene M. Luks. Multiplicative equations over commuting matrices. In *Proceedings of the Seventh Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 498–507, 1996.
2. László Babai, Paolo Codenotti, Joshua A. Grochow, and Youming Qiao. Code equivalence and group isomorphism. In *Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms*, pages 1395–1408. SIAM, 2011.
3. Henry Frederick Baker. Alternants and continuous groups. *Proceedings of the London Mathematical Society*, 2(1):24–47, 1905.
4. Gilbert Baumslag. *Lecture notes on nilpotent groups*. American Mathematical Society, 2007.
5. Paul C Bell, Mika Hirvensalo, and Igor Potapov. The identity problem for matrix semigroups in SL2(Z) is NP-complete. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 187–206. SIAM, 2017.
6. Paul C. Bell and Igor Potapov. On the undecidability of the identity correspondence problem and its applications for word and matrix semigroups. *International Journal of Foundations of Computer Science*, 21(06):963–978, 2010.

7 Paul C. Bell and Igor Potapov. On the computational complexity of matrix semigroup problems. *Fundamenta Informaticae*, 116(1-4):1–13, 2012.

8 Vincent D. Blondel, Emmanuel Jeandel, Pascal Koiran, and Natacha Portier. Decidable and undecidable problems about quantum automata. *SIAM Journal on Computing*, 34(6):1464–1473, 2005.

9 John Edward Campbell. On a law of combination of operators (second paper). *Proceedings of the London Mathematical Society*, 1(1):14–32, 1897.

10 Christian Choffrut and Juhani Karhumäki. Some decision problems on integer matrices. *RAIRO-Theoretical Informatics and Applications-Informatique Théorique et Applications*, 39(1):125–131, 2005.

11 Henri Cohen. *A course in computational algebraic number theory*, volume 8. Springer-Verlag Berlin, 1993.

12 Thomas Colcombet, Joël Ouaknine, Pavel Semukhin, and James Worrell. On reachability problems for low-dimensional matrix semigroups. In Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi, editors, *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece*, volume 132 of *LIPIcs*, pages 44:1–44:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019. `doi:10.4230/LIPIcs.ICALP.2019.44`.

13 Harm Derksen, Emmanuel Jeandel, and Pascal Koiran. Quantum automata and algebraic groups. *Journal of Symbolic Computation*, 39(3-4):357–371, 2005.

14 Ruiwen Dong. On the identity problem and the group problem for nilpotent groups, 2022. Submitted. `doi:10.48550/arXiv.2208.02164`.

15 Cornelia Druţu and Michael Kapovich. *Geometric group theory*, volume 63. American Mathematical Soc., 2018.

16 Max Garzon and Yechezkel Zalcstein. On isomorphism testing of a class of 2-nilpotent groups. *Journal of Computer and System Sciences*, 42(2):237–248, 1991. `doi:10.1016/0022-0000(91)90012-T`.

17 Vesa Halava and Tero Harju. On markov's undecidability theorem for integer matrices. In *Semigroup Forum*, volume 75, pages 173–180. Springer, 2007.

18 Felix Hausdorff. Die symbolische Exponentialformel in der Gruppentheorie. *Berichte über die Verhandlungen der Königlich-Sächsischen Gesellschaft der Wissenschaften zu Leipzig, Mathematisch-Physische Klasse*, 58:19–48, 1906.

19 Derek F. Holt, Bettina Eick, and Eamonn A. O'Brien. *Handbook of Computational Group Theory*. Chapman and Hall/CRC, 2005.

20 Roger Howe. On the role of the heisenberg group in harmonic analysis. *Bulletin (New Series) of the American Mathematical Society*, 3(2):821–843, 1980.

21 Ehud Hrushovski, Joël Ouaknine, Amaury Pouly, and James Worrell. Polynomial invariants for affine programs. In *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 530–539, 2018.

22 Jun-ichi Igusa. *Theta functions*, volume 194. Springer Science & Business Media, 2012.

23 Mikhail Ivanovich Kargapolov and Jurij Ivanovič Merzljakov. *Fundamentals of the Theory of Groups*, volume 62. Springer, 1979.

24 Olga G. Kharlampovich and Mark V. Sapir. Algorithmic problems in varieties. *International Journal of Algebra and Computation*, 5(04n05):379–602, 1995.

25 Aleksandr Aleksandrovich Kirillov. *Lectures on the orbit method*, volume 64. American Mathematical Soc., 2004.

26 Sang-Ki Ko, Reino Niskanen, and Igor Potapov. On the identity problem for the special linear group and the heisenberg group. In Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella, editors, *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic*, volume 107 of *LIPIcs*, pages 132:1–132:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018. `doi:10.4230/LIPIcs.ICALP.2018.132`.

27 Daniel König, Markus Lohrey, and Georg Zetzsche. Knapsack and subset sum problems in nilpotent, polycyclic, and co-context-free groups. *Algebra and Computer Science*, 677:138–153, 2016.
28 V. M. Kopytov. Solvability of the problem of occurrence in finitely generated soluble groups of matrices over the field of algebraic numbers. *Algebra and Logic*, 7(6):388–393, 1968.
29 Hari Krovi and Martin Rötteler. An efficient quantum algorithm for the hidden subgroup problem over Weyl-Heisenberg groups. In *Mathematical Methods in Computer Science*, pages 70–88. Springer, 2008.
30 Engel Lefaucheux. Private Communication, 2022.
31 A. Markov. On certain insoluble problems concerning matrices. *Doklady Akad. Nauk SSSR*, 57(6):539–542, 1947.
32 K. A. Mikhailova. The occurrence problem for direct products of groups. *Matematicheskii Sbornik*, 112(2):241–251, 1966.
33 J. von Neumann. Die Eindeutigkeit der Schrödingerschen Operatoren. *Mathematische Annalen*, 104:570–578, 1931. URL: `http://eudml.org/doc/159483`.
34 Michael S. Paterson. Unsolvability in 3 × 3 matrices. *Studies in Applied Mathematics*, 49(1):105–107, 1970.
35 Igor Potapov and Pavel Semukhin. Decidability of the membership problem for 2× 2 integer matrices. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 170–186. SIAM, 2017.
36 Alexander Schrijver. *Theory of linear and integer programming*. John Wiley & Sons, 1998.
37 Hermann Weyl. *The theory of groups and quantum mechanics*. Courier Corporation, 1950.
38 Jae-Hyun Yang. Harmonic analysis on the quotient spaces of heisenberg groups. *Nagoya mathematical journal*, 123:103–117, 1991.

A Omitted proofs and remarks

▶ **Corollary 2.** *Intersection Emptiness is decidable:*

(i) *in PTIME, for the Heisenberg groups* $\mathrm{H}_n(\mathbb{K})$ *over any algebraic number field* $\mathbb{K}$*, and for any direct product of Heisenberg groups.*

(ii) *for finitely generated 2-step nilpotent groups.*

Proof. (i) By the remark in Section 3, the Heisenberg group $\mathrm{H}_n(\mathbb{K})$ can be embedded as a subgroup of the group $\mathsf{UT}(n', \mathbb{Q})$ for some n', such that the input size only changes at most polynomially. A direct product of Heisenberg groups $\mathrm{H}_{n_1}(\mathbb{K}_1) \times \cdots \times \mathrm{H}_{n_s}(\mathbb{K}_s)$ can hence be embedded as a subgroup of some direct product $\mathsf{UT}(n'_1, \mathbb{Q}) \times \cdots \times \mathsf{UT}(n'_s, \mathbb{Q})$, which is itself a subgroup of $\mathsf{UT}(n'_1 + \cdots + n'_s, \mathbb{Q})$. Again, the input size only changes polynomially during these embeddings. The Heisenberg groups $\mathrm{H}_n(\mathbb{K})$ as well as their direct products are 2-step nilpotent [15, Examples 13.36], and the property of being 2-step nilpotent is preserved under isomorphism. Therefore, Theorem 1 shows that Intersection Emptiness for $\mathrm{H}_n(\mathbb{K})$ as well as for their direct products is decidable in PTIME.

(ii) Given a finite presentation or a consistent polycyclic presentation of G, there exists an embedding $\phi : G \hookrightarrow A \times G_0$ where A is finite and G_0 is a 2-step nilpotent subgroup of some $\mathsf{UT}(n, \mathbb{Q})$. Denote by $\pi_0 : A \times G_0 \to G_0$ the projection onto G_0. The composition $\pi_0 \circ \phi$ can be effectively computed (see proof of [14, Corollary 1.8]).

We claim that $\langle \mathcal{G}_1 \rangle \cap \cdots \cap \langle \mathcal{G}_M \rangle \neq \emptyset$ if and only if $\langle \pi_0(\phi(\mathcal{G}_1)) \rangle \cap \cdots \cap \langle \pi_0(\phi(\mathcal{G}_M)) \rangle \neq \emptyset$. Suppose $g \in \langle \mathcal{G}_1 \rangle \cap \cdots \cap \langle \mathcal{G}_M \rangle$, then obviously $\pi_0(\phi(g)) \in \langle \pi_0(\phi(\mathcal{G}_1)) \rangle \cap \cdots \cap \langle \pi_0(\phi(\mathcal{G}_M)) \rangle$. On the other hand, suppose $h \in \langle \pi_0(\phi(\mathcal{G}_1)) \rangle \cap \cdots \cap \langle \pi_0(\phi(\mathcal{G}_M)) \rangle = \pi_0(\phi(\langle \mathcal{G}_1 \rangle)) \cap \cdots \cap \pi_0(\phi(\langle \mathcal{G}_M \rangle))$, then there exist $a_1, \ldots, a_M \in A$, such that $(a_i, h) \in \langle \mathcal{G}_i \rangle$ for all i. Then $(1, h^{\mathrm{card}(A)}) = (a_i, h)^{\mathrm{card}(A)} \in \langle \mathcal{G}_i \rangle$ for all i, hence $\langle \mathcal{G}_1 \rangle \cap \cdots \cap \langle \mathcal{G}_M \rangle \neq \emptyset$.

Since G_0 is a 2-step nilpotent subgroup of $\mathsf{UT}(n,\mathbb{Q})$, one can decide whether $\langle \pi_0(\phi(\mathcal{G}_1))\rangle \cap \cdots \cap \langle \pi_0(\phi(\mathcal{G}_M))\rangle = \emptyset$ by Theorem 1. Thus, we conclude that it is decidable whether $\langle \mathcal{G}_1\rangle \cap \cdots \cap \langle \mathcal{G}_M\rangle \neq \emptyset$. ◀

We did not attempt to analyse the exact complexity of deciding Intersection Emptiness for arbitrary finitely generated 2-step nilpotent groups. This is because this complexity depends on the computation and representation of the embedding ϕ, as well as the size of the finite group A.

▶ **Theorem 3.** *Given elements $T, S \in \mathrm{H}_3(\mathbb{Q})$ and two finite subsets $\mathcal{G}, \mathcal{H}$ of $\mathrm{H}_3(\mathbb{Q})$, it is decidable whether $T \cdot \langle \mathcal{G}\rangle \cap S \cdot \langle \mathcal{H}\rangle = \emptyset$.*

Proof. As mentioned in the beginning of Section 6, one can without loss of generality suppose $T = I$, and decide whether $\langle \mathcal{G}\rangle \cap S \cdot \langle \mathcal{H}\rangle \neq \emptyset$. Given $\mathcal{G}$ and $\mathcal{H}$, one can effectively compute $\mathcal{C}_{\mathcal{G}} \cap \mathcal{C}_{\mathcal{H}}$ and its dimension using linear programming [36].

If $\mathcal{C}_{\mathcal{G}} \cap \mathcal{C}_{\mathcal{H}}$ has dimension zero or one, then Proposition 8 shows we can decide whether $\langle \mathcal{G}\rangle \cap S \cdot \langle \mathcal{H}\rangle \neq \emptyset$ by solving a finite number of linear Diophantine equations of the form (29).

If $\mathcal{C}_{\mathcal{G}} \cap \mathcal{C}_{\mathcal{H}}$ has dimension two, then Proposition 9 shows we can decide whether $\langle \mathcal{G}\rangle \cap S \cdot \langle \mathcal{H}\rangle \neq \emptyset$ by solving Equations (21), (22) and (23). Equation (23) can be replaced by a boolean combination of conditions of the form "$x_i \equiv 0 \mod 2$", "$x_i \equiv 1 \mod 2$", "$y_i \equiv 0 \mod 2$", …, or "$d_{ij} \equiv 1 \mod 2$". Each of these conditions can be expressed as a linear equation over integers, for example "$x_i \equiv 1 \mod 2$" is equivalent to "$x_i = 2x_i' + 1, x_i' \in \mathbb{Z}$". Therefore, solving Equations (21), (22) and (23) is equivalent to solving a boolean combination of linear equations over integers, which is decidable by integer programming. ◀

▶ **Proposition 8.** *Suppose the cone $\mathcal{C}_{\mathcal{G}} \cap \mathcal{C}_{\mathcal{H}}$ has dimension zero or one. Deciding whether $\langle \mathcal{G}\rangle \cap S \cdot \langle \mathcal{H}\rangle \neq \emptyset$ can be done by solving finitely many linear Diophantine equations.*

Proof. Let $\mathcal{L} \subseteq \mathbb{Q}^2$ be a linear space of dimension one that contains $\mathcal{C}_{\mathcal{G}} \cap \mathcal{C}_{\mathcal{H}}$. Then we decompose $\mathcal{G}$ and $\mathcal{H}$ into disjoint subsets: $\mathcal{G} = \mathcal{G}_0 \cup \mathcal{G}_+$, $\mathcal{H} = \mathcal{H}_0 \cup \mathcal{H}_+$, where

$$\mathcal{G}_0 \coloneqq \{A_i \in \mathcal{G} \mid \varphi(\log A_i) \in \mathcal{L}\}, \quad \mathcal{G}_+ \coloneqq \mathcal{G} \setminus \mathcal{G}_0;$$
$$\mathcal{H}_0 \coloneqq \{B_i \in \mathcal{H} \mid \varphi(\log B_i) \in \mathcal{L}\}, \quad \mathcal{H}_+ \coloneqq \mathcal{H} \setminus \mathcal{H}_0.$$

The key observation is that all matrices in $\mathcal{G}_0$ and in $\mathcal{H}_0$ commute with each other (all $\varphi(\log A_i)$ and $\varphi(\log B_j)$ are linearly dependant, so $[\log A_i, \log A_j] = [\log B_i, \log B_j] = 0$).

Suppose $\langle \mathcal{G}\rangle \cap S \cdot \langle \mathcal{H}\rangle \neq \emptyset$, that is, there exist words v in the alphabet $\mathcal{G}$ and w in the alphabet $\mathcal{H}$ such that $\log v = \log Sw$. We show that the number of occurrences of letters of $\mathcal{G}_+$ in v is bounded; similarly, the number of occurrences of letters of $\mathcal{H}_+$ in w is bounded.

Let $\boldsymbol{n}$ be a non-zero vector orthogonal to $\mathcal{L}$, then $\boldsymbol{x} \mapsto \boldsymbol{n}^\top \boldsymbol{x}$ is the projection parallel to $\mathcal{L}$. Since $\mathcal{C}_{\mathcal{G}} \cap \mathcal{C}_{\mathcal{H}} \subseteq \mathcal{L}$, the values $\boldsymbol{n}^\top \varphi(\log A_i), A_i \in \mathcal{G}$ have signs opposite to that of $\boldsymbol{n}^\top \varphi(\log B_j), B_j \in \mathcal{H}$. Without loss of generality, suppose $\boldsymbol{n}^\top \varphi(\log A_i) \geq 0$ for all $A_i \in \mathcal{G}$ and $\boldsymbol{n}^\top \varphi(\log B_j) \leq 0$ for all $B_j \in \mathcal{H}$. Since $\boldsymbol{n}$ is orthogonal to $\mathcal{L}$, we have furthermore $\boldsymbol{n}^\top \varphi(\log A_i) > 0$ for all $A_i \in \mathcal{G}_+$ and $\boldsymbol{n}^\top \varphi(\log B_j) < 0$ for all $B_j \in \mathcal{H}_+$; as well as $\boldsymbol{n}^\top \varphi(\log X) = 0$ for all $X \in \mathcal{G}_0 \cup \mathcal{H}_0$.

Now, $\log v = \log Sw$ yields $\varphi(\log v) = \varphi(\log S) + \varphi(\log w)$. Projecting onto $\boldsymbol{n}$, this shows

$$\sum_{i, A_i \in \mathcal{G}_+} \mathrm{PI}_i^{\mathcal{G}}(v) \cdot \boldsymbol{n}^\top \varphi(\log A_i) = \boldsymbol{n}^\top \varphi(S) + \sum_{i, B_i \in \mathcal{H}_+} \mathrm{PI}_i^{\mathcal{H}}(w) \cdot \boldsymbol{n}^\top \varphi(\log B_i).$$

This yields

$$\mathrm{PI}_i^{\mathcal{G}}(v) \leq \frac{\boldsymbol{n}^\top \varphi(\log S)}{\boldsymbol{n}^\top \varphi(\log A_i)}, \quad \mathrm{PI}_j^{\mathcal{H}}(v) \leq \frac{\boldsymbol{n}^\top \varphi(\log S)}{\boldsymbol{n}^\top \varphi(\log B_j)}, \tag{28}$$

for all $A_i \in \mathcal{G}_+$ and $B_j \in \mathcal{H}_+$. This gives bounds $\beta_{\mathcal{G}} := \sum_{i, A_i \in \mathcal{G}_+} \frac{\boldsymbol{n}^\top \varphi(\log S)}{\boldsymbol{n}^\top \varphi(\log A_i)}$ and $\beta_{\mathcal{H}} := \sum_{i, B_i \in \mathcal{H}_+} \frac{\boldsymbol{n}^\top \varphi(\log S)}{\boldsymbol{n}^\top \varphi(\log B_i)}$, such that if $\log v = \log Sw$, then the number of letters of $\mathcal{G}_+$ in v is bounded by $\beta_{\mathcal{G}}$; and similarly the number of letters of $\mathcal{H}_+$ in w is bounded by $\beta_{\mathcal{H}}$.

Write $v = v_0 C_1 v_1 C_2 \cdots v_{s-1} C_s v_s$, where $C_1, \dots, C_s$ are matrices in $\mathcal{G}_+$, and $v_0, \dots, v_s$ are words in the alphabet $\mathcal{G}_0$. Similarly, write $w = w_0 D_1 w_1 D_2 \cdots w_{t-1} D_t w_t$, where $D_1, \dots, D_t$ are matrices in $\mathcal{H}_+$, and $w_0, \dots, w_t$ are words in the alphabet $\mathcal{H}_0$. Write $\mathcal{G}_0 = \{A'_1, \dots, A'_{K'}\}$ and $\mathcal{H}_0 = \{B'_1, \dots, B'_{M'}\}$. Define $x_{ij} := \mathrm{PI}_j^{\mathcal{G}_0}(v_i)$ for $0 \leq i \leq s, 1 \leq j \leq K'$, and $y_{ij} := \mathrm{PI}_j^{\mathcal{H}_0}(w_i)$ for $0 \leq i \leq t, 1 \leq j \leq M'$. Then $\log v = \log Sw$ is equivalent to

$$\begin{aligned}
&\sum_{i=1}^{s} \log C_i + \sum_{i=0}^{s}\sum_{j=1}^{K'} x_{ij} \log A'_j + \frac{1}{2} \sum_{0 \leq i < k \leq s} \sum_{j=1}^{K'} x_{ij} [\log A'_j, \log C_k] \\
&+ \frac{1}{2} \sum_{1 \leq k \leq i \leq s} \sum_{j=1}^{K'} x_{ij} [\log C_k, \log A'_j] \\
&= \log S + \sum_{i=1}^{t} (\log D_i + \frac{1}{2}[\log S, \log D_i]) + \frac{1}{2} \sum_{i=0}^{t}\sum_{j=1}^{M'} y_{ij} [\log S, \log B'_j] \\
&\qquad + \frac{1}{2} \sum_{0 \leq i < k \leq t} \sum_{j=1}^{M'} y_{ij} [\log B'_j, \log D_k] + \frac{1}{2} \sum_{1 \leq k \leq i \leq t} \sum_{j=1}^{M'} y_{ij} [\log D_k, \log B'_j]
\end{aligned} \tag{29}$$

All other terms contain $[\log A'_i, \log A'_j]$ or $[\log B'_i, \log B'_j]$ and hence vanish by the commutativity of $\mathcal{G}_0$ and $\mathcal{H}_0$. Note that Equation (29) is a linear Diophantine equation in the variables x_{ij}, y_{ij}. Therefore, $\log v = \log Sw$ has a solution if and only if there exist matrices $C_1, \dots, C_s$ in $\mathcal{G}_+$ and matrices $D_1, \dots, D_t$ in $\mathcal{H}_+$, such that Equation (29) has a solution in non-negative integers, with the additional constraint that, if $s = 0$, then $(x_{01}, \dots, x_{0K'}) \neq \mathbf{0}$; and if $t = 0$, then $(y_{01}, \dots, y_{0M'}) \neq \mathbf{0}$. This additional constraint comes from the condition that v, w are not empty words. Recall the bounds $s \leq \beta_{\mathcal{G}}$ and $t \leq \beta_{\mathcal{H}}$. Hence, deciding whether $\log v = \log Sw$ has a solution amounts to solving finitely many linear Diophantine equations of the form (29). ◀

In theory, it is possible to give a bound on the complexity of the procedure described in Proposition 8. The size of the each bound in Equation (28) is exponential in the bit size of the entries $S, \mathcal{G}, \mathcal{H}$. Hence the procedure consists of solving exponentially many linear Diophantine equations.

▶ **Proposition 10.** *Algorithm 1 is correct and terminates in polynomial time.*

Proof. We prove that Algorithm 1 outputs **False** if and only if $\langle \mathcal{G}_1 \rangle \cap \cdots \cap \langle \mathcal{G}_M \rangle \neq \emptyset$.

After each iteration of Step 2, $\mathrm{card}(S_1) + \cdots + \mathrm{card}(S_M)$ strictly decreases. Therefore, the algorithm terminates after at most $K_1 + \cdots + K_M$ iterations of Step 2.

We now show correctness of the algorithm. We first show that when Algorithm 1 returns **False**, then $\bigcap_{i=1}^{M} \langle \mathcal{G}_i \rangle \neq \emptyset$. Suppose the algorithm terminates with output False, the condition in Step 2(d) shows that $\mathrm{supp}(\Lambda) \cap S_m = S_m$ for all $1 \leq m \leq M$. By the additivity of Λ (that is, $\boldsymbol{a}, \boldsymbol{b} \in \Lambda \implies \boldsymbol{a} + \boldsymbol{b} \in \Lambda$), there exists a vector $\boldsymbol{\ell} = (\boldsymbol{\ell}_1, \dots, \boldsymbol{\ell}_M) \in \Lambda$ such that $\mathrm{supp}(\boldsymbol{\ell}) = \mathrm{supp}(\Lambda)$. This yields $\mathrm{supp}(\boldsymbol{\ell}_m) = \mathrm{supp}(\Lambda) \cap S_m = S_m$ for all m. Since $\mathrm{supp}(\boldsymbol{\ell}_m) = S_m \neq \emptyset$, we have $\boldsymbol{\ell}_m \neq \mathbf{0}$ for all $1 \leq m \leq M$. By the definition (19) of $\pi_{\boldsymbol{\ell}}(W)$, there exist rational numbers $(c_{mij})_{1 \leq m \leq M, i,j \in S_m}$ such that

$$\sum_{j=1}^{K_1} \ell_{1j} \log A_{1j} + \sum_{\substack{i<j \\ i,j \in S_1}} c_{1ij} [\log A_{1i}, \log A_{1j}] = \sum_{j=1}^{K_2} \ell_{2j} \log A_{2j} + \sum_{\substack{i<j \\ i,j \in S_2}} c_{2ij} [\log A_{2i}, \log A_{2j}]$$
$$= \cdots = \sum_{j=1}^{K_M} \ell_{Mj} \log A_{Mj} + \sum_{\substack{i<j \\ i,j \in S_M}} c_{Mij} [\log A_{Mi}, \log A_{Mj}] \quad (30)$$

Since $S_m = \mathrm{supp}(\boldsymbol{\ell}_m)$ for all m, Equation (30) is identical to Equation (14) in Proposition 7. Therefore Proposition 7 shows $\bigcap_{i=1}^{M} \langle \mathcal{G}_i \rangle \neq \emptyset$.

Next, we show that if $\bigcap_{i=1}^{M} \langle \mathcal{G}_i \rangle \neq \emptyset$, then Algorithm 1 returns **False**. Suppose $\bigcap_{i=1}^{M} \langle \mathcal{G}_i \rangle \neq \emptyset$. By Proposition 7, there exist $\boldsymbol{\ell}_1 = (\ell_{1j})_{1 \leq j \leq K_1} \in \mathbb{Z}_{\geq 0}^{K_1} \setminus \{\mathbf{0}\}, \dots, \boldsymbol{\ell}_M = (\ell_{Mj})_{1 \leq j \leq K_M} \in \mathbb{Z}_{\geq 0}^{K_M} \setminus \{\mathbf{0}\}$, and rational numbers $(c_{mij})_{1 \leq m \leq M, i,j \in \mathrm{supp}(\boldsymbol{\ell}_m)}$ that satisfies Equation (14) in Proposition 7. We show that "$\mathrm{supp}(\boldsymbol{\ell}_m) \subseteq S_m$ for all $1 \leq m \leq M$" is an invariant of the algorithm.

At initialization, we obviously have $\mathrm{supp}(\boldsymbol{\ell}_m) \subseteq S_m = \{1, \dots, K_m\}$. Before each iteration of 2(d), suppose we have $\mathrm{supp}(\boldsymbol{\ell}_m) \subseteq S_m$ for all m, then Equation (14) shows that $(\ell_{mj})_{1 \leq m \leq M, 1 \leq j \leq K_m} \in \pi_{\boldsymbol{\ell}}(W)$. Consequently, $\mathrm{supp}(\boldsymbol{\ell}_m) \subseteq \mathrm{supp}(\Lambda)$, meaning $\mathrm{supp}(\boldsymbol{\ell}_m) \subseteq S_m$ still holds after 2(d).

This invariant shows that $\mathrm{supp}(\boldsymbol{\ell}_m) \subseteq S_m$ for all m by the start of Step 3. Since $\boldsymbol{\ell}_m \in \mathbb{Z}_{\geq 0}^{K_m} \setminus \{\mathbf{0}\}$, $\mathrm{supp}(\boldsymbol{\ell}_m)$ is non-empty for every m. We conclude that $S_m \neq \emptyset$ for all m by the start of Step 3. Therefore, Algorithm 1 returns **False**.

Finally, we show that Algorithm 1 terminates in polynomial time. Recall that the algorithm terminates after at most $K_1 + \cdots + K_M$ iterations of Step 2. At each iteration of Step 2(b), the projection can be computed in polynomial time by eliminating the variables $(c_{mij})_{1 \leq m \leq M, i,j \in S_m}$ from the equations defining W. Then, at each iteration of Step 2(c) the support $\mathrm{supp}(\Lambda)$ is computed by Lemma 4. The total input size of the linear programming instances is polynomial with respect to the total bit length of the matrix entries in $\mathcal{G}_1, \dots, \mathcal{G}_M$. Indeed, the total bit length of $\log A_{mi}$ and $[\log A_{mi}, \log A_{mj}]$ is at most of quadratic size in $\mathcal{G}_m$; and the projection performed in Step 2(b) can only alter the total entry bit size at most polynomially. From this, one can express $\pi_{\boldsymbol{\ell}}(W)$ as the solution set of a system of homogeneous linear equations whose total bit length is polynomial in $\mathcal{G}_1, \dots, \mathcal{G}_M$. Hence Lemma 4 computes the support of $\Lambda \coloneqq \mathbb{Z}_{\geq 0}^{\sum_{m=1}^{M} K_m} \cap \pi_{\boldsymbol{\ell}}(W)$ in polynomial time. Therefore, each iteration of Step 2 takes polynomial time, and thus the overall complexity of Algorithm 1 is polynomial with respect to the input $\mathcal{G}_1, \dots, \mathcal{G}_M$. ◀