The Membership Problem for Subsemigroups of $\mathrm{GL}_2(\mathbb{Z})$ is **NP**-complete

Paul C. Bell[a], Mika Hirvensalo[b], Igor Potapov[c]

[a] *School of Computer Science and Mathematics, Keele University, Keele, ST5 5BG, Staffordshire, UK*
[b] *Department of Mathematics and Statistics, University of Turku, Turku, FIN-20014, Finland*
[c] *Department of Computer Science, University of Liverpool, Liverpool, L69 3BX, Merseyside, UK*

Abstract

We show that the problem of determining if the identity matrix belongs to a finitely generated semigroup of 2×2 matrices from the modular group $\mathrm{PSL}_2(\mathbb{Z})$, the Special Linear group $\mathrm{SL}_2(\mathbb{Z})$ and the General Linear Group $\mathrm{GL}_2(\mathbb{Z})$ is solvable in **NP**. We extend this to prove that the membership problem is decidable in **NP** for $\mathrm{GL}_2(\mathbb{Z})$ and for any arbitrary regular expression over matrices from $\mathrm{SL}_2(\mathbb{Z})$. We then derive that the problems of whether a given finite set of matrices from $\mathrm{SL}_2(\mathbb{Z})$ or $\mathrm{PSL}_2(\mathbb{Z})$ generates a group or a free semigroup are both decidable in **NP**. The previous algorithm for these problems, shown in 2005 by Choffrut and Karhumäki, was in **EXPSPACE**. Our algorithm is based on new techniques allowing us to operate on compressed word representations of matrices without explicit expansions. When combined with the known **NP**-hard lower bound, this proves that the identity (and thus membership) problem over $\mathrm{GL}_2(\mathbb{Z})$ is **NP**-complete, and the group problem and the non-freeness problem in $\mathrm{SL}_2(\mathbb{Z})$ are **NP**-complete. Thus the paper answer the long standing open question on the complexity of the membership problem in semigroups generated by matrices from $\mathrm{GL}_2(\mathbb{Z})$. We develop novel techniques that can be used for solving numerical matrix problems in symbolic form, which are applicable for solving compressed word problems for groups and semigroups, bridging the gap between combinatorial group theory, computational problems on matrices and complexity theory. [1]

[1] A preliminary version of this paper appeared in [4].

Keywords: General linear group, special linear group, nonnumerical algorithms, NP-completeness, matrix semigroups, compressed data structures, computational group theory

1. Introduction

A large number of naturally defined matrix problems are still unanswered despite the long history of matrix theory. Originally, the notion of a matrix naturally arose from abbreviated notations for a set of linear equations [15]. Nowadays, matrix problems emerge in a much larger context, as they appear in the analysis of various population models [14], verification of digital processes [22], in the context of control theory questions [9], etc. In the theory of computation, the analysis of many automata models and abstractions often relies upon determining properties of the matrices which define them. For example, the computation of Weighted Finite Automata (WFA) can be defined as a product of integer matrices [26, 2], the dynamics of Probabilistic Finite Automata (PFA) can be represented by stochastic matrices [28] or in the case of Quantum Finite Automata (QFA) [8] by unitary matrices [2]. Computational problems on matrix semigroups have been associated with several long standing open problems in algebraic number theory and transcendence theory [24], Nash equilibria [23], program verification [41] as well as in a large number of engineering fields [27].

The "embarrassing lack of techniques" for solving computational problems for even simple linear systems has been recently discussed and highlighted by several leading scientists, including Terence Tao [50] and Richard Lipton [31]. Many computational problems on matrix semigroups are either open or else *undecidable*, meaning there is no hope to find a tractable solution to the problem at hand unless one makes several substantial restrictions on the semigroup, such as its *dimension*, *domain* over which it is defined (i.e. $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{C}$, $\mathbb{H}$ or $\mathbb{A}$), *size* of the generating set as well as additional constraints on allowed reachability paths or other simplifications (such as allowing approximate solutions, considering reachability paths over bounded languages or a more specific sub-classes of matrices).

[2]On the other hand, recent developments in automata theory and combinatorics on words have been successfully used in solving algebraic problems in matrix semigroups [18, 42, 25, 8, 5, 47].

Many simply formulated and elementary problems for matrices are inherently difficult to solve even in dimension two, and most of these problems become undecidable in general, starting from dimension three or four. One such hard question is the *Membership Problem*: *Given a finite set of $m \times m$ matrices $F = \{M_1, M_2, \ldots, M_n\}$ and a matrix M, determine if there exist an integer $k \geq 1$ and $i_1, i_2, \ldots, i_k \in \{1, \ldots, n\}$ such that $M_{i_1} M_{i_2} \cdots M_{i_k} = M$, i.e., determine whether matrix M belongs to the semigroup generated by F.* A seminal result in this area by M. Paterson showed that for semigroups generated by a finite number of three-dimensional integer matrices, determining if the zero matrix belongs to the semigroup (the *Mortality Problem*) is undecidable [43]. The decidability of the mortality problem for integer matrices of dimension two still remains an open problem, although the problem is known to be at least **NP**-hard [3].

In this paper, we develop various novel techniques that allow us to replace complex numerical problems on matrix products by combinatorial and computational problems on words. More specifically, by operating with compressed symbolic representations of matrices and matrix products, we dramatically reduce the computational complexity of the proposed algorithms for several computational problems (such as membership, identity and freeness) from **EXPSPACE** to **NP** in the cases of the Projective Special Linear group $\mathrm{PSL}_2(\mathbb{Z})$, Special Linear group $\mathrm{SL}_2(\mathbb{Z})$ and General Linear Group $\mathrm{GL}_2(\mathbb{Z})$, which play central roles in many branches of mathematics. One importance of the proposed techniques, which operate with compressed representations of matrices in $\mathrm{SL}_2(\mathbb{Z})$, is also in the potential applications for larger classes of 2×2 matrices over integers, rationals or complex numbers due to recently studied decomposition techniques (based on the Smith normal form), which convert the original problems on more general classes into problems over symbolic forms of $\mathrm{SL}_2(\mathbb{Z})$, see [46, 47, 48]. These techniques can also contribute to other established areas, such as computational group theory, including problems for compressed words [32], the analysis of cryptosystems based on matrix groups [20] and several variants of the membership problem in similar settings, see [37, 36, 34, 1].

$\mathrm{SL}_2(\mathbb{Z})$, which is the most basic example of a discrete non-abelian group, consists of all integer 2×2 matrices, with determinant one[3] and $\mathrm{PSL}_2(\mathbb{Z})$ is

[3]The subgroup $\mathrm{SL}_2(\mathbb{Z})$ of the group $\mathrm{SL}_2(\mathbb{R})$ has a role somewhat like that of $\mathbb{Z}$ inside of $\mathbb{R}$.

the quotient of $\mathrm{SL}_2(\mathbb{Z})$ by its center $\{I, -I\}$, where I is the identity matrix. In other words, $\mathrm{PSL}_2(\mathbb{Z})$ consists of all integer 2×2 matrices, with determinant 1, where pairs of matrices A and $-A$ are considered to be equivalent. Group $\mathrm{SL}_2(\mathbb{Z})$ is important in the context of many fundamental problems, for example from hyperbolic geometry [52, 16, 21], dynamical systems [44], Lorenz/modular knots [35], braid groups [45], particle physics, high energy physics [51], M/string theories [19], ray tracing analysis, music theory [39] and it plays a central role for the development of efficient solutions of 2×2 matrix problems [46].

The structural properties of $\mathrm{GL}_2(\mathbb{Z})$, $\mathrm{SL}_2(\mathbb{Z})$ and $\mathrm{PSL}_2(\mathbb{Z})$ have been studied extensively in various textbooks and research papers. In this work, we reveal new techniques for efficient computations with compressed representations of elements in these groups in order to answer long-standing algorithmic complexity questions. In particular, we show that for any finitely generated semigroup $S \subseteq \mathrm{GL}_2(\mathbb{Z})$ the membership problem for S (whether or not a given matrix belongs to S) is **NP**-complete, and for $S \subseteq \mathrm{SL}_2(\mathbb{Z})$ the group problem (whether S is a group, i.e. S is closed under inverse) and the freeness problem (whether each matrix in S has a unique factorisation) are **NP**-complete, by reducing the previously known **EXPSPACE** upper bound from [17] to **NP**.

In 1994, Cai, Fuchs, Kozen and Liu proved that the the membership problem for finitely generated subgroups and submonoids of the modular group $\mathrm{PSL}_2(\mathbb{Z})$ can be solved in polynomial time *on average* [11][4]. Later, in 2007, Gurevich and Schupp solved the membership problem for the modular group, showing that the problem for the group case is decidable in polynomial time [25]. The algorithm proposed by Gurevich and Schupp operates on a graph representing a syllabic representation of elements of $\mathrm{PSL}_2(\mathbb{Z})$, and works as a graph contraction algorithm. This approach works efficiently since the authors consider a *group*, not a semigroup. Consequently, in their "daisy graph" (defined later, see Figure 1 for example) representing the group generators, all directed edges run in both directions, and hence it is possible to reduce the nondeterminism in this graph by contracting (i.e. joining "equivalent" paths

[4]Note that the subgroup membership problem can be seen as a special case of the submonoid (semigroup) membership problem. The only difference between the subgroup and submonoid membership problems is that in the subgroup membership problems, inverses are allowed. The subgroup membership problems reduce to the submonoid membership problems by simply including the inverses in the generating set of matrices.

and nodes) gradually. This leads to a deterministic **P** algorithm, whereas in the semigroup case, we cannot apply such contractions without breaking the structure of the graph and hence one cannot do better than an **NP** algorithm. While it is known that the membership problem is **NP**-hard for a semigroup of matrices from $\mathrm{SL}_2(\mathbb{Z})$, the exact complexity for the membership problem in this case was still open.

As mentioned, a main result of this paper states that the identity problem for matrix semigroups generated by any finite set of matrices from $\mathrm{GL}_2(\mathbb{Z})$ is **NP**-complete. We may note that the solution to the identity problem is the most essential special case on the way to building an algorithm for the general membership problem for $\mathrm{GL}_2(\mathbb{Z})$. The previous algorithm for this problem, shown in 2005 by Choffrut and Karhumäki [17], was in **EXPSPACE** mainly due to the translation of matrices into exponentially long words over a binary alphabet $\{s, r\}$ and further constructions with a large nondeterministic finite state automaton that is built on these words. However that decision procedure could also be implemented in **EXPTIME**, as the construction of the automaton relies on words which have an exponential length representation of each matrix from the generator and which then requires an exponential number of steps for the construction of additional edges and checking of the membership problem in the resulting regular language. On the other hand, the problem does not allow any obvious **PSPACE** algorithm, let alone an **NP** algorithm, as it was shown in [7] that there are instances of the identity problem over $\mathrm{SL}_2(\mathbb{Z})$ where the number of generator occurrences needed to produce the identity matrix, or the number of potential solutions to the identity problem are exponential in the description size of the semigroup generator (see §4.1.1 and §4.1.2).

It is rather surprising then, in this context, that we can derive an **NP** algorithm solving the membership problem for $\mathrm{GL}_2(\mathbb{Z})$. Our new algorithm is based on a range of new techniques that allow us to operate directly with compressed word representations of matrices without explicit exponential expansions. The membership problem in $\mathrm{GL}_2(\mathbb{Z})$ is susceptible to an exponential blow up in the space and time requirements, unless elaborate techniques are used to avoid them and simpler approaches often have pathological cases which cause recognisers for the problem to lie outside of **NP**. Our techniques use various properties of $\mathrm{PSL}_2(\mathbb{Z})$ and its rich word structure, which is captured by a succinct syllabic representation initially explored by [25]. In our results, we rely on the fact that we can derive a reasonable characterization of complex long paths within our derived compressed graph that we call *Al-*

ternating Forms, which have extensive properties that can be exploited and help us to greatly simplify some parts of the analysis. We utilise various properties of $\mathrm{PSL}_2(\mathbb{Z})$ and these alternating forms, and we combine them with a variety of ideas from algebra, number theory, and graph theory in a novel way to get the desired **NP** algorithm. When combined with the **NP**-hard lower bound shown in [7], this proves that the membership problem (including the identity problem) and group problem in $\mathrm{GL}_2(\mathbb{Z})$ is **NP**-complete. From this fact, we can immediately derive that the fundamental problem of whether a given finite set of matrices from $\mathrm{SL}_2(\mathbb{Z})$ or $\mathrm{PSL}_2(\mathbb{Z})$ generates a group is also decidable in **NP**.

In fact, we prove a stronger statement that it is decidable whether an arbitrary matrix is in S, where S is an arbitrary regular subset of $\mathrm{SL}_2(\mathbb{Z})$ that is, a subset which is defined by a finite automaton. Since $\mathrm{SL}_2(\mathbb{Z})$ is closed under inverses, we show a construction that solves the freeness problem in **NP**. The non-freeness problem was recently proven to be **NP**-hard [30] so the non-freeness problem in $\mathrm{SL}_2(\mathbb{Z})$ is also **NP**-complete.

The decidability status of the identity problem and the group problem in higher dimensions was unknown for several decades and was only shown in 2010 to be undecidable for integer matrices starting from dimension four [6], see also the solution to Problem 10.3 in [10] [5]. This undecidability result was recently improved by reducing the bound on the size of the generator set from 48 to 8 over $\mathrm{SL}_4(\mathbb{Z})$ in [29]. The freeness problem is known to be undecidable for 3×3 matrices over the integers [12]. Although some partial results for the freeness problem in matrices of dimension two are known, a complete picture is far from clear [13]. The decidability of the identity problem in dimension three remains a long standing open problem as well as many other questions on matrices in dimension two over $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{C}$. The case of dimension two is the most intriguing since there is some evidence that if these problems are undecidable, then this cannot be proven by using any previously known constructions. In particular, there is no injective semigroup morphism from pairs of words over any finite alphabet (with at least two elements) into complex 2×2 matrices [12], which means that the coding of independent pairs of words in 2×2 complex matrices is impossible and the exact encoding of the Post Correspondence Problem or a computation of a

[5] A similar problem for mortality, i.e. the membership of the zero matrix, was shown for $\mathbb{Z}^{3\times3}$ in [43], but the problem for the identity matrix in $\mathbb{Z}^{3\times3}$ is still open.

Turing Machine cannot be used directly for proving undecidability in 2×2 matrix semigroups over $\mathbb{Z}$, $\mathbb{Q}$ or $\mathbb{C}$. The only undecidability result in the case of 2×2 matrices that has been shown so far is the membership, freeness and vector reachability problems over quaternions [5] or more precisely in the case of diagonal matrices over quaternions, which are *dual quaternions*. There is a possibility of finding algorithmic solutions to membership problems in $\mathrm{SL}_3(\mathbb{Z})$ as in [29] it was shown that there is no embedding from pairs of words into 3×3 integral matrices with determinant one, i.e., into $\mathrm{SL}_3(\mathbb{Z})$, and no embedding from pairs of words over a group alphabet into 3×3 integral matrices.

Of note is that the membership problem for sub*groups* of $\mathrm{GL}_2(\mathbb{Z})$, which is known to be decidable in polynomial time [33]. As is often the case for membership problems, there is a distinction in the complexity of membership problems for semigroups and groups.

The paper is organized as follows. In Section 2, we provide essential notations and definitions. In Section 3, we investigate the structure of $\mathrm{SL}_2(\mathbb{Z})$ and $\mathrm{PSL}_2(\mathbb{Z})$, and investigate techniques to convert numerical matrix problems into computational problems on symbolic compressed forms. Section 4 describes a known brute-force **EXPSPACE** algorithm for deciding the identity problem over $\mathrm{PSL}_2(\mathbb{Z})$ and $\mathrm{GL}_2(\mathbb{Z})$. Section 5 contains the main result of this paper, introducing new techniques for operating with compressed syllabic forms and we derive an **NP** algorithm for solving the identity problem in $\mathrm{PSL}_2(\mathbb{Z})$ and then show various corollaries such as deciding membership for a general regular expression in $\mathrm{SL}_2(\mathbb{Z})$ is in **NP**, as is the non-freeness problem. Combined with a known **NP**-hardness result for deciding the identity problem in $\mathrm{SL}_2(\mathbb{Z})$, this proves the **NP**-completeness of several matrix problems. Finally in the conclusion (Section 6), we provide an overview of known results and show future directions for research in this area.

2. Preliminaries

2.1. Semigroup basics

By an alphabet, we understand (usually) a finite set Σ, and call its elements letters. Any alphabet can be furnished with algebraic structure, defining a product by letter juxtaposition (concatenation). The semigroup generated by Σ is denoted by Σ^+ or $\langle \Sigma \rangle_{\mathrm{sg}} = \{\sigma_1 \sigma_2 \dots \sigma_n \mid n \geq 1, \sigma_i \in \Sigma\}$. The assumption that there are no nontrivial relations between the letters such as commutation is another way to say that Σ^+ is *freely generated* by Σ.

An element of the semigroup Σ^+ is called a word, and there is a natural extension of Σ^+ into a monoid, just by adding the neutral element called the *empty word*, which is denoted by ε or 1. The monoid generated by Σ is denoted by Σ^*. Given a word $w = \sigma_1\sigma_2\cdots\sigma_k$, we denote by $w_{i,j}$ the word (also called a factor) $\sigma_i\cdots\sigma_j$, with the assumption that $1 \le i \le j \le k$.

If Σ is included in an algebraic structure which also contains the *inverse* of each $\sigma \in \Sigma$ satisfying $\sigma\sigma^{-1} = \sigma^{-1}\sigma = 1$, we may define the *group* generated by Σ as $\langle\Sigma\rangle_{\mathrm{gr}} = \{\sigma_1^{a_1}\sigma_2^{a_2}\ldots\sigma_n^{a_n} \mid n \ge 0, \sigma_i \in \Sigma, a_i \in \{-1,1\}\}$. If there is no danger of confusion, we omit the subscript 'gr' and simply write $\langle\Sigma\rangle$.

2.2. Matrix Groups in $\mathbb{Z}^{2\times2}$

Notation $\mathbb{Z}^{2\times2}$ stands for the set of all 2×2 integer matrices. This set has a natural ring structure with respect to ordinary matrix addition and multiplication. Unfortunately, the algebraic structure of $\mathbb{Z}^{2\times2}$ seems too complicated to imply any straightforward algorithm for membership questions, hence simpler structures are needed.

A subset of $\mathbb{Z}^{2\times2}$,

$$\mathrm{GL}_2(\mathbb{Z}) = \{A \in \mathbb{Z}^{2\times2} \mid \det(A) \in \{-1,1\}\}.$$

also denoted as $\mathrm{GL}(2,\mathbb{Z})$ is called the *General Linear group*, consisting of all 2×2 integer matrices having integer matrix inverses. Group $\mathrm{GL}_2(\mathbb{Z})$ is clearly the largest multiplicative matrix group contained in $\mathbb{Z}^{2\times2}$. However, as it shortly turns out, a smaller subgroup is useful for computational purposes.

One restriction that turns out to be useful is the *Special Linear group* defined as

$$\mathrm{SL}_2(\mathbb{Z}) = \{A \in \mathrm{GL}_2(\mathbb{Z}) \mid \det(A) = 1\},$$

but the quotient group

$$\mathrm{PSL}_2(\mathbb{Z}) = \mathrm{SL}_2(\mathbb{Z})/\{\pm I\}$$

called the *Projective Special Linear group* appears even more useful. In fact, $\mathrm{PSL}_2(\mathbb{Z})$ has a very useful representation as a free product of two cyclic groups of order 2 and 3. Notice that by the very definition, an element of $\mathrm{PSL}_2(\mathbb{Z})$ is a set $a = \{A, -A\}$ of two matrices in $\mathrm{SL}_2(\mathbb{Z})$, but from now on, we may slightly abuse the notations and write $a = \pm A$, or choose either matrix A or $-A$ to represent a. Intuitively, $\mathrm{PSL}_2(\mathbb{Z})$ can be taken as $\mathrm{SL}_2(\mathbb{Z})$ by ignoring the sign.

2.3. Graph Theory

We will study *labelled multigraphs* with the property that all edges between vertices v_1 and v_2 have distinct labels. Therefore, our notion of multigraphs can be formally defined as follows: V is a finite set of *vertices* (also called *nodes*), L is the set of labels (which may be infinite) and $E \subseteq V \times L \times V$ is the set of labelled *edges* (also called *arcs*). Now $(u, l, v) \in E$ means that there is an edge from u to v labelled with l.

A *path* in a graph is understood as a sequence of adjacent edges, and can hence be presented as a sequence

$$\Pi = (v_1, l_1, v_2)(v_2, l_2, v_3) \dots (v_k, l_k, v_{k+1}) \in E^* \tag{1}$$

Using notation $e_i = (v_i, l_i, v_{i+i})$, the above presentation can be written as $\Pi = e_1 e_2 \dots e_k \in E^*$. The *length* of path (1) is k and its *label* is defined as the concatenation $l_1 l_2 \dots l_k \in L^*$. It is important to notice that if the label set contains the empty word ε, then it is treated under concatenation as usual, i.e. $l_1 \varepsilon l_2 = l_1 l_2$. For a path with label l beginning at vertex u and ending at v we may also use the notation $\Pi = (u, l, v)$.

A *subpath* of (1) is defined as $e_i e_{i+1} \dots e_j$, where $1 \leq i \leq j \leq k$. The subpath is *proper* if $i > 1$ or $j < k$.

Definition 1. A dual edge cycle *is a path of the form* $e_1 e_2 E^* e_1 e_2$*, where* e_1*,* $e_2 \in E$*.*

Remark 2. *The notion of dual edge cycle is essentially different from the usual graph-theoretical notion of a cycle, which requires that a node is visited twice.*

Intuitively, a dual edge cycle is a path at least four edges long that returns to the two initial edges at the very end. Unless otherwise stated, the notion of "cycle" in this article refers to Definition 1. The reason for such a definition is that in the later analysis, we want to remove cycles in the graph but simultaneously preserve local properties of the path from which the cycle was removed.

We call a dual edge cycle *reduced*, if none of its proper subpaths is a dual edge cycle.

Definition 3. *The nondeterminstic reduction function* $\mathrm{red} : E^* \to E^*$ *is defined to remove dual edge cycles: If* $\Pi = \Pi_1 \Pi_2 \Pi_3$*, where* $\Pi_2 = e_1 e_2 E^* e_1 e_2$ *is a*

dual edge cycle and $\Pi_1, \Pi_3 \in E^*$ *are arbitrary paths, then* $\mathrm{red}(\Pi) = \Pi_1 e_1 e_2 \Pi_3$. *Note that an arbitrary path* Π *may contain several dual edge cycles which can be reduced by* red. *We may therefore consider* red *as being nondeterministic when applied to a path, and we define* red^* *as the transitive closure of* red, *i.e.,* $\mathrm{red}^*(\Pi)$ *is a set of paths, none of which contains a dual edge cycle. Thus,* $\mathrm{red}^*(\Pi)$ *contains each consecutive pair of edges of the graph at most once. Such a path is called a* reduced path.

Example 4. *Consider set of edges* $\{e_1, e_2, e_3\} \subseteq V \times L \times V$ *and path*

$$\Pi = e_1 e_2 e_3 e_1 e_3 e_2 e_3 e_1 e_2$$

Now, Π *is a dual edge cycle, since* $e_1 e_2$ *is a prefix and suffix.* Π *is not reduced since it is a dual edge cycle, and anyway,* $e_3 e_1 e_3 e_2 e_3 e_1$ *and* $e_2 e_3 e_1 e_3 e_2 e_3$ *are proper subpaths and dual edge cycles.*

Notice that red, *and thus* red^*, *is nondeterministic in this example, since* $\mathrm{red}(\Pi) \in \{e_1 e_2, e_1 e_2 e_3 e_1 e_2\}$ *and* $\mathrm{red}^*(\Pi) = \{e_1 e_2\}$. *The second path was generated by reduction* $\mathrm{red}(\Pi) = \mathrm{red}(e_1 \cdot e_2 e_3 e_1 e_3 e_2 e_2 \cdot e_1 e_2) = e_1 e_2 e_3 e_1 e_2$ *since* $\mathrm{red}(e_2 e_3 e_1 e_3 e_2 e_2) = e_2 e_3$.

3. The Structure of $\mathbf{PSL_2}(\mathbb{Z})$

3.1. Generating $\mathrm{SL}_2(\mathbb{Z})$

The group $\mathrm{SL}_2(\mathbb{Z})$ is very important in number theory, and its structure has been studied extensively in various textbooks (see [49], for instance), but for pointing out the algorithmic complexity issues, we reproduce the structural properties most relevant to our study here.

Two structurally important elements of $\mathrm{SL}_2(\mathbb{Z})$ are

$$S = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \quad \text{and} \quad T = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}.$$

Evidently $S^2 = -I$ (which implies $S^3 = -S$ and $S^4 = I$, so S has order 4), whereas for each $n \in \mathbb{Z}$,

$$T^n = \begin{pmatrix} 1 & n \\ 0 & 1 \end{pmatrix},$$

implying that T has no finite order. Nevertheless, it can be shown that S and T generate $\mathrm{SL}_2(\mathbb{Z})$, and the following lemma provides even a quantitative version of this fact.

Lemma 5. $\mathrm{SL}_2(\mathbb{Z}) = \langle S, T \rangle_{gr}$. *Furthermore, any matrix*

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \mathrm{SL}_2(\mathbb{Z})$$

can be represented as

$$A = S^{\alpha} T^{q_1} S^3 T^{q_2} \cdot \ldots S^3 T^{q_k} S^{\beta} T^{q_{k+1}}, \tag{2}$$

so that $\alpha, \beta \in \{0, 1, 2, 3\}$, $q_i \in \mathbb{Z}$, $k \leq 1 + \log_2 M$, *and* $|q_i| \leq M$, *with* $M = \max\{|a|, |b|, |c|, |d|\}$. *Representation (2) can be found in time polynomial in* $\log_2 M$.

Proof. By a direct computation we see that left multiplication of A by S and T^n can be described as follows:

$$\begin{aligned} S \begin{pmatrix} a & b \\ c & d \end{pmatrix} &= \begin{pmatrix} -c & -d \\ a & b \end{pmatrix}, \\ T^n \begin{pmatrix} a & b \\ c & d \end{pmatrix} &= \begin{pmatrix} a + nc & b + nd \\ c & d \end{pmatrix}. \end{aligned} \tag{3}$$

If $c = 0$, then

$$A = \begin{pmatrix} a & b \\ 0 & d \end{pmatrix},$$

and since $\det(A) = ad = 1$, it follows that $a = d \in \{-1, 1\}$. Therefore

$$A \in \left\{ \begin{pmatrix} 1 & b \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} -1 & b \\ 0 & -1 \end{pmatrix} \right\} = \{T^b, S^2 T^{-b}\}.$$

If $c \neq 0$ but $a = 0$, then according to (3) $SA \in \{T^{-d}, S^2 T^d\}$, implying that $A \in \{S^3 T^{-d}, S T^d\}$ (since $S^4 = I$). In these cases, the claim evidently holds.

Assume then that $ac \neq 0$. If $|A_{11}| < |A_{21}|$, then according to (3), $|(SA)_{11}| > |(SA)_{21}|$. So define

$$\alpha = \begin{cases} 1 & \text{if } |a| < |c| \\ 0 & \text{if } |a| \geq |c|. \end{cases}$$

to see that

$$A_1 = S^{\alpha} A = \begin{pmatrix} a_1 & b_1 \\ c_1 & d_1 \end{pmatrix}$$

enjoys property $|(A_1)_{11}| = |a_1| \geq |c_1| = |(A_1)_{21}|$. Assume then that

$$A_i = \begin{pmatrix} a_i & b_i \\ c_i & d_i \end{pmatrix}$$

with property $|(A_i)_{11}| = |a_i| \geq |c_i| = |(A_i)_{21}|$ has been defined, but $c_i \neq 0$. Then, due to the (extended) division algorithm, we can find an integer q_i so that $a_i = q_i c_i + r_i$, where $|r_i| \leq \frac{1}{2}|c_i|$.

We define now

$$A_{i+1} = ST^{-q_i}A_i = \begin{pmatrix} -c_i & -d_i \\ r_i & b_i - q_i c_i \end{pmatrix}, \tag{4}$$

and denote $a_{i+1} = -c_i$, $b_{i+1} = -d_i$, $c_{i+1} = r_i$, and $d_{i+1} = b_i - q_i c_i$. Then matrix A_{i+1} clearly satisfies $|(A_{i+1})_{11}| = |a_{i+1}| = |c_i| > |r_i| = |c_{i+1}| = |(A_{i+1})_{21}|$.

The sequence A_1, A_2, ... of matrices is defined until the least k for which $c_k = 0$ and hence

$$A_k = \begin{pmatrix} a_k & b_k \\ 0 & d_k \end{pmatrix},$$

and therefore, as we concluded above,

$$A_k \in \{T^{b_k}, S^2T^{-b_k}\}.$$

Define β and q_k so that $A_k = S^\beta T^{q_k}$, where $\beta \in \{0, 2\}$ and $q_k \in \{\pm b_k\}$. Now $A_{i+1} = ST^{-q_i}A_i$ implies $A_i = T^{q_i}S^3A_{i+1}$, so

$$\begin{aligned} A &= S^{-\alpha}A_1 = S^{-\alpha}T^{q_1}S^3A_2 = S^{-\alpha}T^{q_1}S^3T^{q_2}S^3A_3 \\ &= \vdots \\ &= S^{-\alpha}T^{q_1}S^3T^{q_2}S^3\cdots T^{q_{k-1}}S^3A_k \\ &= S^{-\alpha}T^{q_1}S^3T^{q_2}S^3\cdots T^{q_{k-1}}S^{3+\beta}T^{q_k}. \end{aligned}$$

To estimate the magnitude of the numbers k, q_1, q_2, ..., q_k, let M_i be the absolute value of the largest element of A_i and M the largest M_i. Clearly $M = M_1$ and notice also that according to the process defined above, $|c_{i+1}| \leq \frac{1}{2}|c_i|$ for each i. But if $|c_{i+1}| = |r_i| = \frac{1}{2}|c_i| = \frac{1}{2}|a_{i+1}|$ for some step, c_{i+1} divides a_{i+1} implying that $r_{i+1} = 0$ and the process terminates. Hence we have, if the process has not yet terminated,

$$1 \leq |c_i| < \frac{1}{2}|c_{i-1}| < \frac{1}{2^2}|c_{i-2}| < \ldots < \frac{1}{2^{i-1}}|c_1|,$$

which implies $i - 1 < \log_2 |c_1| \leq \log_2 M_1$. By contraposition, $i \geq 1 + \log_2 M_1$ implies $c_i = 0$. Thus, if k is chosen as the least number so that $c_k = 0$, then $k \leq 1 + \log_2 M_1$. For the magnitude of numbers q_i, notice that as in (4) it always hold that $|r_i| \leq \frac{1}{2}|c_i|$, then $M_{i+1} > M_i$ is possible only in the case $M_{i+1} = |d_{i+1}|$. To analyze this, the determinant condition gives $-c_i d_{i+1} + r_i d_i = 1$, and if $i < k$, then $c_i \neq 0$ and therefore

$$d_{i+1} = \frac{1 - r_i d_i}{-c_i}$$

implying

$$M_i < M_{i+1} = |d_{i+1}| \leq \frac{1}{|c_i|} + \frac{|r_i|}{|c_i|}|d_i| \leq 1 + \frac{1}{2}M_i,$$

But the inequality $M_i < 1 + \frac{1}{2}M_i$ thus obtained can be valid only if $M_i \leq 1$. Now $M_i = 0$ can be true only for the zero matrix, whereas $M_i = 1$ results in a small number of cases which can each be checked to satisfy $M_{i+1} \leq M_i$. For the final step where $c_k = 0$ the determinant condition implies $|d_k| = 1$ anyway, so we can conclude that the process described above cannot increase the absolute value of the maximal matrix entry.

For $i < k$ we can write $q_i = \frac{a_i - r_i}{c_i}$, so

$$|q_i| \leq \left|\frac{a_i}{c_i}\right| + \left|\frac{r_i}{c_i}\right| \leq |a_i| + \frac{1}{2} \leq M_i + \frac{1}{2},$$

and since q_i and M_i are both integers, we can conclude that $|q_i| \leq M_i \leq M_1 = M$. As $q_k \in \{\pm b_k, \pm d_k\}$, trivially $|q_k| \leq M_k \leq M_1 = M$.

It is a straightforward task to analyze that the procedure for finding representation (2) is a polynomial-time algorithm, given the bit representation size of A as the input size. □

Remark 6. *Even though all matrices $A \in \mathrm{SL}_2(\mathbb{Z})$ can be represented in terms of S and T, it is worth noticing that the representation is not unique. A direct computation shows that, for example, $TST = ST^{-1}S^3$.*

For a more canonical representation, let

$$R = ST = \begin{pmatrix} 0 & -1 \\ 1 & 1 \end{pmatrix}.$$

Direct computation shows that

$$R^2 = \begin{pmatrix} -1 & -1 \\ 1 & 0 \end{pmatrix} \quad \text{and} \quad R^3 = -I,$$

implying that $R^6 = I$, so R is of order 6. Since now $T = S^{-1}R = S^3R$, it follows that $\mathrm{SL}_2(\mathbb{Z}) = \langle S, R \rangle$, and that a representation of $A \in \mathrm{SL}_2(\mathbb{Z})$ in terms of R and S can be obtained by substituting $T = S^3R = -SR$ in (2). It is noteworthy that when substituting $T = -SR$ in (2), one can use $R^3 = -I$ and $S^2 = -I$ to get a representation

$$A = (-1)^{\gamma} R^{n_0} S R^{n_1} S \cdot \ldots \cdot R^{n_{l-1}} S R^{n_l}, \tag{5}$$

where $\gamma \in \{0, 1\}$, $n_i \in \{0, 1, 2\}$ and $n_i \in \{1, 2\}$ for $0 < i < l$.

Remark 7. *It can be shown that the representation (5) for a given matrix $A \in \mathrm{SL}_2(\mathbb{Z})$ is unique, but it should be noticed that representation (5) can be exponentially long in the representation size of matrix A in bits, as the example*

$$\begin{pmatrix} 1 & m \\ 0 & 1 \end{pmatrix} = T^m = (-SR)^m = (-1)^m \underbrace{SR \ldots SR}_{m \text{ times}} \tag{6}$$

demonstrates. The representation size of the matrix T^m is proportional to $\log_2 m$, but the representation (6) contains $2m$ matrices.

It is structurally simpler to present (5) ignoring the sign. For that purpose, we introduce two structurally important elements of $\mathrm{PSL}_2(\mathbb{Z})$.

Definition 8. *Let $s = S\{\pm I\}$ and $r = R\{\pm I\}$ be the projections of S and R in $\mathrm{PSL}_2(\mathbb{Z})$.*

Remark 9. *Since $S^2 = R^3 = -I$ in $\mathrm{SL}_2(\mathbb{Z})$, it is clear that $s^2 = r^3 = \varepsilon$ in $\mathrm{PSL}_2(\mathbb{Z})$, with ε corresponding to the identity matrix.*

3.2. Generating $\mathrm{PSL}_2(\mathbb{Z})$

Lemma 10. *[49] -* $\mathrm{PSL}_2(\mathbb{Z}) = \mathrm{SL}_2(\mathbb{Z})/\{\pm I\}$ *is a free product of $\langle s \rangle = \{1, s\}$ and $\langle r \rangle = \{1, r, r^2\}$. That is,* $\mathrm{PSL}_2(\mathbb{Z}) = \langle r, s \mid s^2 = r^3 = 1 \rangle$ *and if*

$$r^{n_0} s r^{n_1} s \cdots r^{n_{p-1}} s r^{n_p} = r^{m_0} s r^{m_1} s \cdots r^{m_{q-1}} s r^{m_q}, \tag{7}$$

where $n_i, m_j \in \{0, 1, 2\}$ and n_i, $m_j \in \{1, 2\}$ for $0 < i < p$ and $0 < j < q$, then $p = q$ and $n_i = m_i$ for each i.

For the proof of Lemma 10 see [49].

Definition 11. *We call a representation w of $a \in \mathrm{PSL}_2(\mathbb{Z})$ a* ground level *presentation, or* $\langle r, s\rangle$-presentation *if $w \in \{r, s\}^*$ strictly, (eg. no parentheses and exponents are involved), and* reduced*, if w contains no subwords ss or rrr.*

Remark 12. *By Lemma 10 every element of $\mathrm{PSL}_2(\mathbb{Z})$ admits a unique reduced ground level representation. However, it follows directly from Remark 7 that the unique representation of the projection of T^m in $\mathrm{PSL}_2(\mathbb{Z})$ is*

$$t^m = \underbrace{rs \dots rs}_{m \text{ times}}, \tag{8}$$

which is exponentially long in the representation size of t^m, being $\Theta(\log_2 m)$ since we can use T^m to represent $t^m = \{T^m, -T^m\}$.

Despite Definition 11, we may refer to the ground level representation using exponents and parentheses, e.g., r^2, or even $(sr)^m$, but it should then be clear from the context that we are not referring to the *succinct* representation which we now define.

It is remarkable that for a given matrix A, the representation (7) of $a = \pm A$ always contains so much periodicity, that it is possible to have a polynomially long description. In the continuation, we will call such a description a *succinct* or *compact* representation.

In fact, substituting $T = -SR$ in (2) and taking the projections $S \to s$ and $R \to r$ we learn that

$$a = s^{\alpha}(sr)^{q_1}s(sr)^{q_2} \cdot \ldots s(sr)^{q_k}s^{\beta}(sr)^{q_{k+1}}, \tag{9}$$

where the estimation for the exponents and k are the same as in Lemma 5. We need to remember that in this representation, numbers q_i are not necessarily positive but, if $q_i < 0$, we can simply write $(sr)^{q_i} = (r^2 s)^{-q_i}$ to get a presentation with positive exponents expressed in the following lemma:

Lemma 13. *Any element $a = \{A, -A\}$ of $\mathrm{PSL}_2(\mathbb{Z})$ admits a unique succinct representation of the form*

$$a = r^{\alpha}(sr)^{n_1}(sr^2)^{n_2}(sr)^{n_3}(sr^2)^{n_4} \dots (sr)^{n_{l-1}}(sr^2)^{n_l}s^{\beta}, \tag{10}$$

where $\alpha \in \{0, 1, 2\}$, $\beta \in \{0, 1\}$ and $n_i > 0$ if $1 < i < l$. The representation size can be bounded analogously to Lemma 5.

It is possible to formalize the notion of the succinct representation by extending alphabet from $\{r, s\}$ into a larger one containing parentheses (and), exponent symbol $\uparrow$, and 0 and 1 to present the exponents in binary. When applying this approach to equation (8), we would have a representation

$$t^m = (rs) \uparrow (m_1 \dots m_k), \tag{11}$$

where $m_1 \dots m_k$ is the binary representation of integer m and hence $k = \lfloor \log_2 m \rfloor + 1$. Now the length of the right hand side of (11) as a string over the larger alphabet described above is approximately $1+2+1+1+1+k+1$, which is proportional to $\log_2 m$, the representation size of t^m.

However, to achieve simplification, we will not use such a formalism for the succinct representations. Instead, we choose to use an infinite alphabet consisting of *syllables* defined in the next section.

3.3. Syllabic Presentation of $\mathrm{PSL}_2(\mathbb{Z})$

A more straightforward version of the compact representation (10) can be obtained by using the notion of a *syllable*. In principle, a syllable is just a word over alphabet $\{r, s\}$, but typically a systematic form is desirable.

Definition 14. *Following Gurevich and Schupp [25], we define the following syllables:*

$$R_i \;= \begin{cases} (rs)^{i-1}r & \text{if } i > 0 \\ (r^2s)^{|i|-1}r^2 & \text{if } i < 0 \\ \varepsilon & \text{if } i = 0 \end{cases}$$

We say that syllable R_i is positive if $i > 0$, and negative if $i < 0$. The representation size *of the syllable is a constant (to define the type) plus the subscript a representation size for R_a type syllable.*

In the continuation, we will introduce more syllables but for the moment, these are sufficient. Notice that R_i is the inverse to R_{-i} for any $i \in \mathbb{Z}$ (thus $R_iR_{-i} = \varepsilon$). As $r = R_1$, the following lemma is trivial but its claim is worth emphasizing.

Lemma 15. *All elements of* $\mathrm{PSL}_2(\mathbb{Z})$ *can be represented by using syllables of the set* $\{s, R_a \mid a \in \mathbb{Z}\}$.

The main advantage of syllables of Definition 14 is that they can be used to write the compact representations (10) in a structural way, and also provide a natural way to handle the potential cancellations of elements.

Remark 16. *It can easily be shown that the syllabic representation of* $\mathrm{PSL}_2(\mathbb{Z})$ *elements is not unique. Consider, for instance an element* $a = R_2R_{-5}$*. By the definition,*

$$\begin{aligned} R_2R_{-5} &= (rs)r(r^2s)^4r^2 = (rs)rr^2s(r^2s)^3r^2 \\ &= r(r^2s)^3r^2 = r(r^2s)(r^2s)^2r^2 = s(r^2s)^2r^2 \end{aligned}$$

but also $sR_{-3} = s(r^2s)^2r^2$*.*

The above example serves as a basis of the following definition.

Definition 17. *Words* w_1 *and* w_2 *over the syllabic alphabet* $\{s, R_a \mid a \in \mathbb{Z}\}$ *(or even over an extended alphabet we introduce later) are* equivalent*, if they are representations of the same* $\mathrm{PSL}_2(\mathbb{Z})$ *element. In the continuation, we will denote the syllabic word equivalence by* $w_1 \equiv w_2$*. It should be noted that for equivalent syllabic words* w_1 *and* w_2*, also* $w_1 = w_2$ *holds, if the equality is understood in* $\mathrm{PSL}_2(\mathbb{Z})$*. To keep notations simpler, we accept this ambiguity.*

It is clear that $\equiv$ *is an equivalence relation, and even a congruence, meaning that if* $w_1 \equiv w_2$*, then* $ww_1 \equiv ww_2$*, and* $w_1w \equiv w_2w$ *for any* $w \in \{s, R_a \mid a \in \mathbb{Z}\}^*$*.*

Even though the syllabic representation is not unique, the following result is proven in [25]. The representation size estimate follows directly from Lemma 13.

Lemma 18. *Each element* $a \in \mathrm{PSL}_2(\mathbb{Z})$ *admits a unique representation of the form*

$$a = s^{\alpha}R_{n_1}sR_{n_2}sR_{n_3}s \ldots sR_{n_l}s^{\beta}, \tag{12}$$

where $\alpha,\ \beta \in \{0, 1\}$ *and the representation is alternating, meaning that* $n_in_{i+1} < 0$ *for each* i*. The size of representation (12) is polynomial in the representation size of* a*.*

Because of the uniqueness, we call representation (12) a *canonical syllabic* representation of $\mathrm{PSL}_2(\mathbb{Z})$ elements.

Lemma 19. *The syllables satisfy the following relations*

- $ss \equiv \varepsilon$
- $R_aR_{-a} \equiv \varepsilon$*, and*

- $R_{a+b} \equiv R_a s R_b$, *if* $ab > 0$.
- $R_1 R_1 \equiv R_{-1}$, *and* $R_{-1} R_{-1} \equiv R_1$.

Proof. The proof is straightforward and uses only the definition of syllables R_a, and relations $r^3 = s^2 = \varepsilon$ in $\mathrm{PSL}_2(\mathbb{Z})$. □

Remark 20. *It can be seen that the above relations give rise to other ones. For example, if* $ab < 0$ *and* $|b| < |a|$, *then* $R_a R_b \equiv R_{a+b} s R_{-b} R_b \equiv R_{a+b} s$, *and a symmetric version is obtained when* $|a| < |b|$. *To summarize:*

- $R_a R_b \equiv R_{a+b} s$, *if* $ab < 0$ *and* $|b| < |a|$
- $R_a R_b \equiv s R_{a+b}$, *if* $ab < 0$ *and* $|a| < |b|$.

Remark 21. *The above rules in Lemma 19 and Remark 20 may seem like cancellation rules: Syllables of type* R_a *with different subindex signs cancel against each other very much like the exponents in a product, but the subindex values close to zero introduce anomalities.*

For example, it is easy to see that

$$\begin{aligned} R_1 R_2^t R_1 &\equiv R_{-1} R_{-1} R_2^t R_1 \\ &\equiv R_{-1} s R_1 R_2^{t-1} R_1 \equiv \dots \\ &\equiv (R_{-1}s)(R_{-1}s)\cdots(R_{-1}s) R_1 R_1 \\ &\equiv (R_{-1}s)^t R_{-1} \equiv R_{-(t+1)} \end{aligned}$$

From this, we can easily derive that $R_2^t R_1 \equiv R_{-1} R_{-(t+1)} \equiv R_1 s R_{-t}$ *and* $R_1 R_2^t \equiv R_{-t} s R_1$. *Similarly we can see that* $R_{-1} R_{-2}^t R_{-1} \equiv R_{t+1}$ *and derive analogous consequences.*

We conclude this section by estimating the "reduction power" of the equivalences of Lemma 19 and Remark 20.

Definition 22. *The* ground level length, *also called* rs-length *of a syllable is defined as the number of occurrences of generators* r *and* s *in the syllable. That is,* $|s|_{\langle r,s\rangle} = 1$, *and*

$$|R_a|_{\langle r,s\rangle} = \begin{cases} 2a-1 & \text{if } a > 0 \\ -3a-1 & \text{if } a < 0 \\ 0 & \text{if } a = 0 \end{cases}$$

The ground-level length of a syllabic word $w = w_1 \dots w_n$ *is defined as* $|w|_{\langle r,s\rangle} = |w_1|_{\langle r,s\rangle} + \dots + |w_n|_{\langle r,s\rangle}$.

Definition 23. *A syllabic word w is* reducible*, if there exists an equivalent syllabic word w' so that $|w'|_{\langle r,s\rangle} < |w|_{\langle r,s\rangle}$.*

Lemma 24. *A syllabic word w is reducible if and only if it contains a factor of the form ss, $R_aR_1R_b$, $R_{-a}R_{-b}$, R_aR_{-b}, or $R_{-a}R_b$, where a, $b > 0$.*

Proof. The proof is based on the fact that a syllabic word can always be interpreted as a word over alphabet $\{r, s\}$, and as such, is reducible if and only if it contains a factor s^2 or r^3.

Part "If": Assume first that a syllabic word contains one of the aforementioned factors. Case ss is trivial, that factor can be removed to obtain an equivalent syllabic word w' so that $|w'|_{\langle r,s\rangle} = |w|_{\langle r,s\rangle} - 2$.

In case $a = b = 1$, we have $R_aR_1R_b = rrr = \varepsilon$. If $a > 1$ but $b = 1$, then $R_aR_1R_b = (rs)^{a-1}r\,r\,r = (rs)^{a-2}rs = R_{a-1}s$, and a similar conclusion follows in case $a = 1$, $b > 1$. In the remaining case, both a, $b > 1$, and a direct calculation shows that $R_aR_1S_b = (rs)^{a-1}r\,r(rs)^{b-1}r$, a word which contains first r^3 and then s^2 to be removed: $R_aR_1R_b = (rs)^{a-2}rs\,rr\,rs(rs)^{b-2}r = (rs)^{a-2}r(rs)^{b-2}r = R_{a-1}R_{b-1}$.

If $a, b > 1$, then

$$\begin{aligned} R_{-a}R_{-b} &= (r^2s)^{a-1}r^2(r^2s)^{b-1}r^2 \\ &= (r^2s)^{a-2}r^2sr^2r^2s(r^2s)^{b-2}r^2 \\ &= R_{-(a-1)}sR_1sR_{-(b-1)} \end{aligned}$$

(r^3 was removed). A similar conclusion holds if either $a = 1$ or $b = 1$.

The cases R_aR_{-b} and $R_{-a}R_b$ are obvious and can be treated analogously.

Part "Only if": Assume then that a syllabic word w is reducible. Since all reductions are done by removing s^2 or r^3 from the underlying presentation over alphabet $\{s, r\}$, we can conduct the following analysis:

1) If ss can be removed, then ss must occur as a subword in the original syllabic word, since the syllables R_a begin and end with an r.

2) The case when factor r^3 can removed can occur only when syllables of type R_a are concatenated.

2.1) In case R_aR_b, where $a, b > 1$, no reduction takes place, since $R_aR_b = (rs)^{a-1}r(rs)^{b-1}r$ contains only two consecutive occurrences of r. However, if $b = 1$, Then R_aR_b ends with rr, and if the next syllable also begins with r, a factor r^3 can be removed. On the other hand, if the next syllable is of type

R_{-b} with $b > 0$, there is a factor $R_1 R_{-b}$, which will fall in the subcase 2.3). Hence we can finish with this subcase by concluding that the reduction takes place if $R_a R_1$ is followed by R_b, where $b > 0$.

2.2) In the case $R_{-a} R_{-b}$, where $a, b > 1$, a reduction (r^3 is removed) always occurs:

$$\begin{aligned} R_{-a} R_{-b} &= (r^2 s)^{a-1} r^2 (r^2 s)^{b-1} r^2 \\ &= (r^2 s)^{a-2} r^2 s r^2 r^2 s (r^2 s)^{b-2} r^2 \\ &= R_{-(a-1)} s r s R_{-(b-1)} \\ &= R_{-(a-1)} s R_1 s R_{-(b-1)}. \end{aligned}$$

2.3) In both cases $R_a R_{-b}$ and $R_{-a} R_b$, a reduction clearly takes place: $R_a R_{-b} = (rs)^{a-1} r (r^2 s)^{b-1} r^2 = R_{a-1} R_{-(b-1)}$ if $a, b > 1$, and the reduction can be applied recursively as long as both subindices remain positive. A similar conclusion can be derived for the supplementary case $R_a R_{-b}$. □

Notice that according to Lemma 24, the canonical form of Lemma 18 is not reducible.

Definition 25. *We define the set of syllables* $\Omega = \{\varepsilon, s, s^\alpha R_{\pm 1} s^\beta, s^\alpha R_{\pm 2} s^\beta\}$, *where* $\alpha, \beta \in \{0, 1\}$. *Intuitively, set* Ω *forms a "neighbourhood" of* ε. *This notion is useful since long syllabic words often reduce not to* ε *but to some other element 'close' to* ε, *i.e., an element of* Ω.

Lemma 26. *Assume that a syllabic word* w *is reducible to* $w' \in \Omega$. *Then the reduction can be performed by using the following syllabic rules:*

1. $ss \mapsto \varepsilon$
2. $R_a R_{-a} \mapsto \varepsilon$
3. $R_a R_{-b} \mapsto R_{a-b} s$, *if* $ab > 0$ *and* $|b| < |a|$
4. $R_a R_{-b} \mapsto s R_{a-b}$, *if* $ab > 0$ *and* $|a| < |b|$
5. $R_{-1} R_{-1} \mapsto R_1$
6. $R_1 \mapsto R_{-1} R_{-1}$

Remark 27. *We do not introduce a rule* $R_1 R_1 \to R_{-1}$, *even though the equivalence* $R_1 R_1 \equiv R_{-1}$ *holds. The asymmetry becomes understandable in the proof below. It should be noted that rule* $R_1 \to R_{-1} R_{-1}$ *is not a ground level reduction, but it is used to incorporate the equivalence* $R_a R_1 R_b \equiv R_{a-1} R_{b-1}$.

Proof. It is straightforward to verify that words over alphabet $\{s, r\}$ together with rewriting rules $r^3 \mapsto \varepsilon$ and $s^2 \mapsto \varepsilon$ form a *locally confluent* system, meaning that if $x \mapsto y$ and $x \mapsto z$ by a single application of a reduction rule, then there is a w so that $y \mapsto^* w$ and $z \mapsto^* w$ (using reduction rules repeatedly). It follows from Newman's lemma [38] that the system is confluent. Especially, for any $x \in \{r, s\}^*$ there is a unique minimal element $x' \in \{r, s\}^*$ obtained by using the reduction rules recursively in any order as long as it is possible to apply any rule.

Let us now assume that a syllabic word w is reducible to $w' \in \Omega$. We need to show that a chain of reduction rules $s^2 \mapsto \varepsilon$ and $r^3 \mapsto \varepsilon$ can be replaced by a chain of the rules mentioned in the statement of this lemma.

1) Factor ss can only occur if it is already present in the syllabic word, and removing that factor corresponds exactly to the syllabic reduction rule 1.

2) The second type $r^3 \mapsto \varepsilon$ can be applied only if w contains three consecutive symbols r. The proof of the previous lemma shows that there are three subcases:

2.1) Reduction of form $R_a R_1 R_b \mapsto R_{a-1} R_{b-1}$ (a, $b > 1$) removes one R_1 and reduces the indices of the surrounding syllables, but it may be simulated by rules 6, 3, 4, and 1:

$$R_a R_1 R_b \mapsto R_a R_{-1} R_{-1} R_b \mapsto R_{a-1} ss R_{b-1} \mapsto R_{a-1} R_{b-1}.$$

2.2) In this case, $R_{-a} R_{-b}$ contains a factor r^3 to be removed, and the resulting representation is $R_{-(a-1)} s R_1 s R_{-(b-1)}$ (assuming a, $b > 1$). However, it is straightforward to see that in order to cancel a word containing such a fragment to the identity word, the first or the last syllable must be cancelled to the identity. More precisely, if a syllabic word

$$u R_{-a} R_{-b} v \mapsto u R_{-(a-1)} s R_1 s R_{-(b-1)} v \mapsto^* \omega$$

is reducible to an element of Ω, then necessarily either $u \mapsto^* u_1 s R_{a-1}$ or $v \mapsto^* R_{b-1} s v_1$. In the first case (the second is analogous), we can change the reduction order to have

$$\begin{aligned} u R_{-a} R_{-b} v &\mapsto^* u_1 s R_{a-1} R_{-a} R_{-b} v \\ &\mapsto u_1 ss R_1 R_{-b} v \\ &\mapsto u_1 R_1 R_{-b} v, \end{aligned}$$

which can be further reduced by using case 2.3. Hence, we can conclude that this subcase is actually not needed when reducing syllabic words to the identity. If one of the subindices, say b, is equal to 1, then the corresponding reduction rule is $R_{-a}R_{-1} \mapsto R_{-(a-1)}sR_1$, but as this form is canonical as well, a similar conclusion can be drawn. On the other hand, if $a = b = 1$, then the rule becomes $R_{-1}R_{-1} \mapsto R_1$, which is exactly the rule number 5.

2.3) This case divides into various subcases. If $a, b \neq 0$, we have $R_aR_{-b} = R_{a-1}R_{-(b-1)}$, a reduction which is obtained by applying $r^3 \mapsto \varepsilon$ and $s^2 \mapsto \varepsilon$. As the system is confluent, we can assume that a reduction of this type is applied recursively, consequently arriving either in rule 2, 3, or 4. □

In the algorithm to be presented, we shall need all reduction rules of Lemma 26 at least implicitly, but the following rules will form the backbone of the algorithm presented in Section 5.

Definition 28 (Reduction function ρ)**.** *We call rules 1-4 of Lemma 26* regular *and define function ρ to represent them as follows:*

i) $\rho(ss) = \varepsilon$

ii) $$\rho(R_xR_{-y}) = \begin{cases} R_{x-y}s, & \text{if } |x| > |y|\,, \\ sR_{x-y}, & \text{if } |y| > |x|\,, \\ \varepsilon, & \text{if } |x| = |y|\,, \end{cases}$$
where $\operatorname{sgn}(x) = \operatorname{sgn}(y)$.

Function ρ can be applied iteratively and nondeterministically. We denote by ρ^* the reflexive transitive closure of ρ. Note that ρ is a locally confluent rewriting system and ρ^* is clearly terminating, thus ρ is globally confluent by Newman's lemma [38] (thus the order that rules of ρ are applied is not important).

Reduction rules 5 and 6 are called anomalous.

4. First (Brute Force) Decision Procedure

Lemma 10 states that the elements of $\mathrm{PSL}_2(\mathbb{Z})$ can be presented as words over $\{r, s\}$ satisfying relations $r^3 = s^2 = \varepsilon$. In this section, we use such a presentation to describe the decision procedure for the identity problem via standard automata-theoretical constructions, although the construction of the automata will require exponential time and space.

We have already described the general formulation of the identity problem in the preliminaries, but for the sake of accuracy, we state the computational problem formally here.

Problem 29 (Identity problem over $\mathrm{PSL}_2(\mathbb{Z})$)**.** *Given a finite set* $\{A_1, \ldots, A_n\} \subset \mathrm{SL}_2(\mathbb{Z})$*; let* $a_i = \{A_i, -A_i\}$ *be the projection of* A_i *on* $\mathrm{PSL}_2(\mathbb{Z})$*. The problem is to decide if the semigroup* $\langle a_1, \ldots, a_n \rangle_{sg}$ *contains the identity element.*

4.1. Input Size Measures

In order to estimate the problem's complexity, it is necessary to define a measure of the size of an input. Here we will use the following:

Definition 30. *Given an integer* a*, we denote by* $|a|_{bit}$ *the bit representation size of* a*, that is* $|a|_{bit} = 1 + \lfloor \log_2 |a| \rfloor + 1$*, where the extra bit serves as the sign of the integer, and* $\log_2(0)$ *is taken as* 0*.*

Definition 31. *For any matrix* $A \in \mathbb{Z}^{2\times 2}$*, we denote by* $|A|_{bit}$ *the* representation size *of matrix* A*, which is given by* $|A|_{bit} = \sum_{1 \le i,j \le 2} |a_{ij}|_{bit}$*.*

Remark 32. *Letting* $M = \max_{1 \le i,j \le 2} |a_{ij}|$*, as in Lemma 5, it is obvious that* $|A|_{bit} = \Theta(\log M)$*.*

Definition 33. *For any finite matrix set* $S = \{A_1, \ldots, A_n\}$*, the bit size of* S *is defined as*

$$|S|_{bit} = |A_1|_{bit} + \ldots + |A_n|_{bit}.$$

When estimating the input size, we ignore the separating symbols needed for representing sets and matrices. It is obvious that including those would produce only a linear increase in the representation size.

It is possible to find instances of Problem 29 where the representation of the identity element requires a high number of generator occurrences.

Example 34. *Let* $n > 1$ *and* $S = \{sR_n, R_{-1}s\}$*. Now the description size of set* S *consists of the description of* $b = R_{-1}s$ *(a constant number of bits) and* $a = sR_n$ *requires a number of bits proportional to* $\log_2 n$ *the length of the number. Using Remark 20 and Lemma 19 we see that* $ab = sR_nR_{-1}s = sR_{n-1}ss = sR_{n-1}$*,* $ab^2 = sR_{n-2}$*, and by induction* $ab^n = \varepsilon$*. It is evident that the identity cannot be found in* S^+*with fewer generator occurrences.*

In this example, the smallest identity in A^+ *is obtained by an exponential (in the description size of the set* A*) number of the generator occurrences, but there is anyway a short sequence of elements in* S^+ *witnessing the existence of the identity: By computing* $O(\log_2 n)$ *elements of sequence* b*,* $b^2 = R_{-1}sR_{-1}s = R_{-2}s$*,* $b^4 = (b^2)^2 = R_{-4}s$*,* $b^8 = (b^4)^2 = R_{-8}s$*,* $b^{16} = (b^8)^2 = R_{-16}s$*, ... it is possible to construct* $R_{-n}s$*, and* $sR_nR_{-n}s = \varepsilon$*.*

4.1.1. An example of an exponential length solution

We now describe an example where the shortest identity is exponentially long, but the parse tree only polynomially deep, originally shown in [7].

Let $Q_4 = \{q_i, q_i^{-1} : 0 \leq i \leq 4\}$, $\Sigma_4 = \{i, i^{-1} : 1 \leq i \leq 4\}$ and

$$W = \left\{ \begin{array}{cccc} q_0^{-1}1q_1, & q_2^{-1}2q_0, & q_3^{-1}3q_0, & q_4^{-1}4q_0, \\ q_1^{-1}1^{-1}q_2, & q_2^{-1}2^{-1}q_3, & q_3^{-1}3^{-1}q_4, & q_4^{-1}4^{-1}q_0 \end{array} \right\} \subseteq Q_4\Sigma_4Q_4.$$

It was proven in [7] that $\varepsilon \in W^*$, but the shortest such ε is of the form:

$$\begin{array}{rcll} X_1 & = & q_0^{-1}1q_1 \cdot q_1^{-1}1^{-1}q_2 & \equiv \; q_0^{-1}q_2 \\ X_2 & = & X_1 \cdot q_2^{-1}2q_0 \cdot X_1 \cdot q_2^{-1}2^{-1}q_3 & \equiv \; q_0^{-1}q_3 \\ X_3 & = & X_2 \cdot q_3^{-1}3q_0 \cdot X_2 \cdot q_3^{-1}3^{-1}q_4 & \equiv \; q_0^{-1}q_4 \\ X_4 & = & X_3 \cdot q_4^{-1}4q_0 \cdot X_3 \cdot q_4^{-1}4^{-1}q_0 & \equiv \; \varepsilon \end{array}$$

We see that $X_4 \equiv \varepsilon$ consists of 30 words from W. In fact, set W can be trivially generalised so that it consists of $2k$ elements and the shortest sequence giving ε uses $2^{k+1} - 2$ elements from W.

It is possible to encode each such word of W into $\mathrm{SL}_2(\mathbb{Z})$ such that the bit representation size of each such matrix is proportional to k, which we now describe briefly (for full details, see [7]). Given an arbitrary sized group alphabet Σ_t, and a binary group alphabet $\Sigma_2 = \{a, b, a^{-1}, b^{-1}\}$, there exists an injective homomorphism $\alpha : \Sigma_t^* \to \Sigma_2^*$. Furthermore, there exists an injective homomorphism $f : \Sigma_2^* \to \mathrm{PSL}_2(\mathbb{Z})$ given by:

$$f(a) = \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix}, f(b) = \begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix}, f(a^{-1}) = \begin{pmatrix} 1 & -2 \\ 0 & 1 \end{pmatrix}, f(b^{-1}) = \begin{pmatrix} 1 & -2 \\ 0 & 1 \end{pmatrix}$$

Note that $f(a) = sR_2, f(b) = sR_{-2}$ and thus $f(a^{-1}) = R_{-2}s, f(b^{-1}) = R_2s$, which makes it clear why $\{f(a), f(b)\}$ form a free group. Given W, we may thus apply $f \circ \alpha$ to each word to give a subset $G \subset \mathrm{SL}_2(\mathbb{Z})$ which can be combined to given the 2×2 identity matrix using at least an exponential number of generator matrices (exponential in both $|G|$ and the bit representation size of each matrix in G).

4.1.2. An example daisy graph with exponentially many paths

Another 'extreme' case of a solution to the identity problem is that there many exist exponentially many possible solutions (paths through the daisy graph, defined in Remark 35) which must be checked. We give a brief proof sketch of this here, see [7] for full details.

As shown above in §4.1.1, there exists an injective morphism $f : \Sigma_2^* \to \mathrm{PSL}_2(\mathbb{Z})$ and in fact it is not difficult to extend this morphism to an arbitrary sized domain *group* alphabet $f : \Sigma_k^* \to \mathrm{PSL}_2(\mathbb{Z})$ while retaining injectivity. One may then define 'border symbols' to enforce restrictions on potential solutions to the identity problem.

Consider the subset sum problem: let $S = \{s_1, s_2, \ldots, s_{k-1}\} \subseteq \mathbb{N}$ and $t \in \mathbb{N}$, does there exist some subset $S' \subseteq S$ such that $\sum_{x \in S'} x = t$? The problem is well known to be NP-complete.

Using border symbols $\Sigma_k = \{1, 2, \ldots, k, 1^{-1}, 2^{-1}, \ldots, k^{-1}\}$, we may define the following set of words:

$$W = \left\{ \begin{array}{cccccc} 1W_1 2^{-1}, & 2W_2 3^{-1}, & \cdots & (k-1)W_{k-1}k^{-1}, & kW_t^{-1}1^{-1}, \\ 1 \cdot \varepsilon \cdot 2^{-1}, & 2 \cdot \varepsilon \cdot 3^{-1}, & \cdots & (k-1) \cdot \varepsilon \cdot k^{-1} & \end{array} \right\},$$

where $W_i = a^{s_i}$ and $W_t^{-1} = a^{-t}$. The encoding of [7] is slightly more complicated but the above is illustrative of the main idea. If we may combine these words together to get the identity, then such a product can be shown to be of the form:

$$1X_1 2^{-1} \cdot 2X_2 3^{-1} \cdots (k-1)X_{k-1}k^{-1} \cdot kW_t^{-1}1^{-1},$$

where $X_i \in \{W_i, \varepsilon\}$. Reaching the identity for such words (with minor modifications to the encoding) is thus equivalent to the subset sum problem. Clearly there are exponentially many such paths to check (exponential in k). It is possible using encoding f to show that the encoding is size preserving (a word a^{s_i} is represented by a matrix $f(a^{s_i})$ whose representation size is logarithmic in s_i) and thus the identity problem is NP-hard (see [7] for full details).

4.2. Automaton for Recognizing the Identity

The decision procedure presented in [17] is based on Lemma 10, which states that all elements of $\mathrm{PSL}_2(\mathbb{Z})$ can be faithfully represented as strings over alphabet $\{r, s\}$ with relations $r^3 = s^2 = \varepsilon$. Briefly described, the procedure works as follows: First, a nondeterministic finite automaton over alphabet $\{r, s\}$ recognizing A^+ is constructed, and then ε-transitions are iteratively added to represent the relations $r^3 = s^2 = \varepsilon$ between the nodes (states) as long as possible. More precisely, whenever a path $q_1 \to q_2$ with label r^3 or s^2 is found, an ε-transition $q_1 \xrightarrow{\varepsilon} q_2$ is introduced. The procedure ends eventually, since the number of states is finite, although exponential in

the description size of A. The decision whether $\varepsilon \in A^+$ is then made based on the observation whether there is an ε-transition from the initial state to the final state.

Another route to the decision procedure, when the aforementioned finite automaton is constructed, is to note that the representations of the identity element in $\mathrm{PSL}_2(\mathbb{Z})$ can be described by a simple context-free grammar (the starting and only nonterminal symbol is Δ)

$$\Delta \to 1 \mid s\Delta s\Delta \mid r\Delta r\Delta r\Delta.$$

It is well-known that the intersection of a regular language L_1 (accepted by a finite automaton) and a context-free language L_2 (that consists of the identity element representations) is context-free, and the decision procedure follows from the fact that the emptiness problem for the intersection of context-free languages (i.e. $L = L_1 \cap L_2$) is decidable.

The construction of an automaton recognizing language $\{a_1, a_2, \ldots, a_n\}^+$ is very straightforward: The automaton has two states q_0 and q_1, and for each a_i, there is a transition $q_0 \xrightarrow{a_i} q_1$, as well as a loop $q_1 \xrightarrow{a_i} q_1$. State q_0 is specified as the initial state, and q_1 as the final state (See Figure 1).

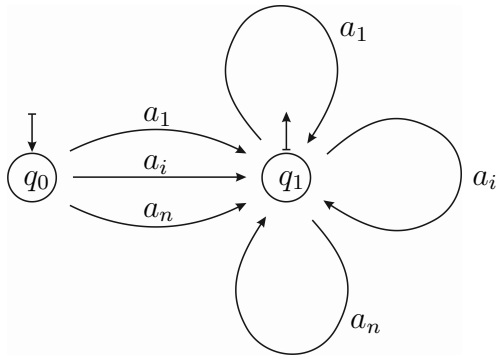

Figure 1: Automaton recognizing $\{a_1, a_2, \ldots, a_n\}^+$. Initial and final states are indicated with short arrows.

Remark 35. *We call the graph of Figure 1* a daisy graph*: Indeed, arrows $q_0 \to q_1$ form the stem, and each arrow $q_1 \to q_1$ forms one petal.*

The automaton of Figure 1 is defined on abstract symbols a_i, and introducing the $\langle r, s\rangle$-representation will result in the automaton being augmented

so that each edge will be replaced with a path as follows: if

$$a_i = t_{i,1} t_{i,2} \dots t_{i,k_i},$$

where each $t_{i,j} \in \{r, s\}$, then each edge $\circ \xrightarrow{a_i} \circ$ of the previous automaton is replaced with a path

$$\circ \xrightarrow{t_{i,1}} \circ \xrightarrow{t_{i,2}} \circ \dots \circ \xrightarrow{t_{i,k_i}} \circ,$$

and all the new nodes are assumed distinct. The replacements result in a larger automaton shown in Figure 2. As described above, the $\langle r, s \rangle$-

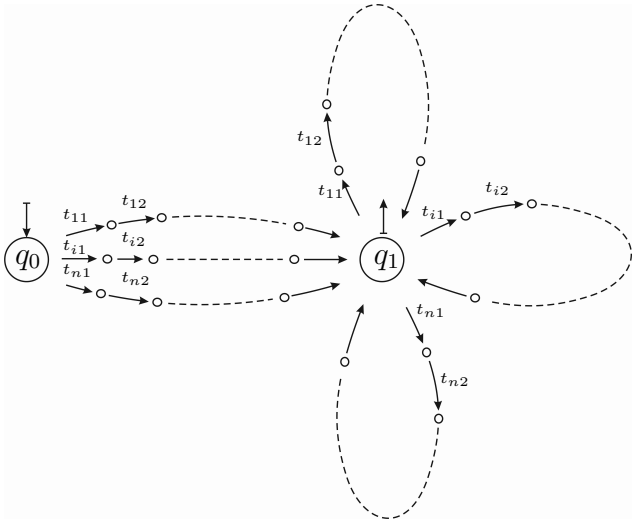

Figure 2: $\langle r, s \rangle$-automaton recognizing $\{a_1, \dots, a_n\}^+$.

automaton of Figure 2 can be used to discover whether the semigroup A^+ contains the identity element.

Now that the lengths of $\langle r, s \rangle$-representations of elements of $\mathrm{PSL}_2(\mathbb{Z})$ can be exponential in the description size of the elements (Remark 12), it follows that the daisy graph of Figure 2 and consequently the described decision procedure requires exponential space in the worst case.

4.3. Syllabic Automaton

An obvious attempt to resolve the identity problem with fewer spatial resources comes from the syllabic representations of the $\mathrm{PSL}_2(\mathbb{Z})$ elements. Using the syllabic representation instead of the ground-level representation, we can redefine the daisy graph of Figure 2 to be only polynomially large in the input size, but the price to pay is that the edge labels then come from an infinite alphabet $\{s, R_a \mid a \in \mathbb{Z}\}$.

The procedure described in Section 4.2 generalizes as well, but instead of introducing ε-transitions only, we introduce new transitions according to Lemma 26: Whenever a path $q_1 \to q_2$ exists bearing a label equal to the left-hand side of one of the syllabic rules of the Lemma, then a new edge $q_1 \to q_2$ with the corresponding right hand side as the label should be introduced.

It is not very difficult to see that such a procedure will also eventually halt, since for the new R_a-labels being introduced, the subscript a has no greater absolute values than those already existing. Finally, the decision can be made by checking whether the procedure has produced an ε-transition from the initial to the final state.

However, the described procedure will produce a multigraph, which may lead to an exponential increase in the amount of space required for the computation.

Example 36. *When applying this procedure to Example 34 we first get a daisy graph with two 'petals': one with label sR_n, and the second one with label $R_{-1}s$. Applying the reduction rules repeatedly will produce new paths $q_1 \to q_1$ with labels sR_{n-1}, sR_{n-2}, sR_{n-3}, etc. Hence the number of new edges eventually added will be exponential in the input description size.*

Remark 37. *It should be mentioned already here that the "daisy" form of the graph is not essential for the decision procedure. On the contrary, it is possible to generalize the procedure to decide if the identity is in $R(a_1, \ldots, a_n)$, where R is any regular expression of a_1, ..., a_n.*

5. Improved Decision Procedure

In the continuation, we will demonstrate how to modify and analyze the syllabic daisy graph in order to achieve a nondeterministic, polynomial time algorithm for resolving Problem 29.

The strategy will avoid exponential growth in the edge set mentioned in Example 36. A cursory description of the algorithm is as follows:

- Given a matrix set $M = \{M_1, \ldots, M_n\} \subseteq \mathrm{SL}_2(\mathbb{Z})$, the procedure starts with constructing a syllabic version of the "daisy graph" $G_M = (Q, E)$ as described in Section 4.3. It follows from Lemma 18 that the size of this graph is polynomial in the input size and the construction can be done in polynomial time. $E_{q_i,q_j} \subseteq E$ stands for labelled edges from node q_i to q_j.

- For a nondeterministically chosen pair of vertices $q_i, q_j \in Q$, it is checked if there is a path $q_i \to q_j$ with label equivalent to a syllabic word in Ω, i.e. one "close" to ε. This may be done via *short*, *medium*, or *long reductions* which we describe later. This steps repeats iteratively whenever such new Ω-edges can be added.

- Finally, it is verified whether there is an ε-edge from the initial state q_0 to the final state q_1. The witness for such an edge gives the positive answer to the identity problem.

The short and medium reductions are straightforward to describe with the already existing notions, but for the long reductions, we need to introduce more terminology. As we shall show, the syllabic words reducing to the identity can be assumed to be of a certain form, which can be locally verified. The form we are aiming at would be much simpler without the reductions shown in Remark 21.

5.1. Syllabic Graph Path Properties

In this section we study various important properties of the syllabic form of the Daisy Graph. Recall from the Definition 25 that Ω-syllables are those "close" to ε.

As shown in Remark 21, there is an option of having an unbounded number of reductions for certain types of paths (where the labels are of the form $R_1 R_2^t R_1 \equiv R_{-(t+1)}$ or $R_{-1} R_{-2}^t R_{-1} \equiv R_{t+1}$), and hence we will also introduce R-minus -type "joker" syllable R_-, and analogous plus -type joker syllable of the form R_+.

5.1.1. Syllables R_- and R_+

Consider a path $\Pi = (q_i, R_2, q_i)(q_i, R_1, q_j)$ in G_M. Note that we have a self loop from q_i to itself, labelled by syllable R_2. This implies that the path $(q_i, R_2, q_i)^t (q_i, R_1, q_j)$ exists for any $t \geq 0$. From Remark 21, $(R_2)^t R_1 \equiv R_{-1} R_{-(t+1)}$, and hence for any $t \in \mathbb{Z}^+$, there is a path from q_i to q_j with label

equivalent to $R_{-1}R_{-(t+1)}$. We thus introduce a syllable R_-, which denotes an R syllable of any negative index.

If there exists a path $\Pi = (q_i, R_{-2}, q_i)(q_i, R_{-1}, q_j)$, then since $R_{-2}^t R_{-1} \equiv R_1 R_{t+1}$, we define a syllable R_+, which denotes an R syllable of any *positive* index.

Definition 38. *Let* $\Gamma_+ = \{R_x, R_+ | x > 2\}$, $\Gamma_- = \{R_x, R_- | x < 2\}$, $\Gamma = \Gamma_+ \cup \Gamma_-$ *and finally* $\Sigma = \Omega \cup \Gamma$ *be the set of all syllables.*

For each syllable in Σ, we now introduce a notion of "weight", which gives a magnitude to each such element.

Definition 39 (Weight)**.** *We define the* weight *of a syllable* $z \in \Sigma$ *as a function* $\mathrm{wgt} : \Sigma \to \mathbb{Z}$*:*

$$\mathrm{wgt}(z) = \begin{cases} x, \text{ if } z = R_x \text{ and } z \in \Gamma; \\ \pm 2, \text{ if } z \in \{s^\alpha R_{\pm 2} s^\beta \,|\, \alpha, \beta \in \{0,1\}\}; \\ \pm 1, \text{ if } z \in \{s^\alpha R_{\pm 1} s^\beta \,|\, \alpha, \beta \in \{0,1\}\}; \\ 0 \text{ if } z \in \{\varepsilon, s\}. \end{cases}$$

We define the absolute weight *of a syllable to be a function* $\mathrm{awgt} : \Sigma \to \mathbb{N} \cup \{0\}$*, given by* $\mathrm{awgt}(z) = |\mathrm{wgt}(z)|$*. Function* wgt *(resp. awgt) can be extended to a word* $w = w_1 w_2 \cdots w_k \in \Sigma^*$ *by defining* $\mathrm{wgt}(w) = \sum_{i=1}^k \mathrm{wgt}(w_i)$ *(resp.* $\mathrm{awgt}(w) = \sum_{i=1}^k \mathrm{awgt}(w_i)$*).*

As described above, syllables R_- *and* R_+ *are essentially 'sets' of syllables, allowing any negative weight for* R_- *and any positive weight for* R_+*. Therefore* $\mathrm{wgt}(R_+)$ *is any positive integer and* $\mathrm{wgt}(R_+)$ *is any negative integer.*

Remark 40. *It is worth noting that equivalent syllabic words may have different (absolute) weights. For example,* $R_{-5}R_{10} \equiv sR_{-5}$*, which shows that the weight may differ, and* $R_1 \equiv R_{-1}R_{-1}$*, which shows that even the absolute weight may differ.*

Therefore, the (absolute) weight is strictly related to a particular syllabic word, not to the $\mathrm{PSL}_2(\mathbb{Z})$ *element it represents.*

The following definition will help to characterize certain syllabic words reducible to the identity and will be essential to the later analysis.

Definition 41 (**Alternating Form** ($\mathcal{AF}$))**.** *Let*

$$\mathcal{AF} = \Sigma^* \setminus \Sigma^* \{R_a s^\alpha s^\alpha R_b, R_a s R_{-b}\} \Sigma^*,$$

where a and b have the same sign, and $\alpha \in \{0,1\}$. In other words, a word $w \in \Sigma^*$ is in alternating form *if it does not contain two consecutive syllables R_a and R_b (possibly with ss in between) with the same sign, or a substring of the form $R_a s R_{-b}$. Given a path $\Pi = (q_i, w, q_j) \in Q \times \Sigma^* \times Q$, we also say $\Pi \in \mathcal{AF}$ if $w \in \mathcal{AF}$ and there is no danger of confusion.*

Definition 42 (Ω**-Minimal Word**). *A syllabic word $w = w_1 w_2 \cdots w_k \in \Sigma^*$ is called an* Ω-minimal word *if and only if $w \equiv w'$, where $w' \in \Omega$ and $w_i w_{i+1} \cdots w_j \equiv w''$ where $w'' \in \Omega$ for $1 \leq i < j \leq k$ implies that $i = 1$, $j = k$ and $w' = w''$. We denote the set of all Ω-minimal words over Σ by Φ.*

For example, $R_{10}R_{-5}sR_{-5} \in \Phi$, since $R_{10}R_{-5}sR_{-5} \equiv R_5 ssR_{-5} \equiv R_5 R_{-5} \equiv \varepsilon$, but no shorter syllabic subword of $R_{10}R_{-5}sR_{-5}$ is reducible to an element of Ω. We later show that Ω-minimal words whose length is greater than 3 are in alternating form which greatly simplifies their analysis.

The length of a path without dual edge cycles is analyzed in the following lemma. Recall that function red : $E^* \to E^*$ is defined in Definition 3 and nondeterministically removes a dual edge cycle from a path, if one is present. Statement i) of Lemma 43 essentially says that if we have a path whose word is in alternating form, then removing a dual edge cycle from that path gives a word still in alternating form. Statement ii) of Lemma 43 says that the length of a reduced path (one containing no dual edge cycle) is no more than the square of the number of edges in the graph.

Lemma 43. *Given a path $\Pi \in Q \times \Sigma^* \times Q$ where $\Pi = (q_i, w, q_j)$ and $w \in \mathcal{AF}$. Then the following two properties hold:*

i) If $\Pi' \in \mathrm{red}(\Pi)$, then $\Pi' = (q_i, w', q_j)$, where $w' \in \mathcal{AF}$;
ii) $|\mathrm{red}^*(w)| \leq |E|^2$.

Proof. To prove i), let $\Pi = \pi_1 \pi_2 \cdots \pi_{|w|} \in \mathcal{AF}$. If $\mathrm{red}(\Pi) = \Pi$, then $\mathrm{red}(\Pi) \in \mathcal{AF}$ as required. Otherwise, $\Pi = \Pi_1 \Pi_2 \Pi_3$, where $\Pi_1, \Pi_3 \in E^*$ and $\Pi_2 = e_1 e_2\ U e_1 e_2 \in E^*$ is a dual edge cycle (for some $e_1, e_2 \in E$) and $\mathrm{red}(\Pi) = \Pi_1 e_1 e_2 \Pi_3$. Notice that checking if an element of Σ^* belongs to $\mathcal{AF}$ is a *local* property of the word; we need only determine if every subword of length two is not of the form $R_a s^\alpha \cdot s^\alpha R_b$, $R_a s \cdot R_{-b}$, $R_a \cdot s R_{-b}$ and every subword of length three is not of the form $R_a \cdot s \cdot R_b$, where $ab > 0$ and $\alpha \in \{0,1\}$.

If $\Pi \in \mathcal{AF}$, then $\Pi_1 e_1 e_2 \in \mathcal{AF}$ and $e_1 e_2 \Pi_3 \in \mathcal{AF}$, which implies that $\mathrm{red}(\Pi) = \Pi_1 e_1 e_2 \Pi_3 \in \mathcal{AF}$, since $e_1 e_2 \in E^2$ and the last syllable of Π_1 agrees with $e_1 e_2$, which in turn agrees with the first syllable of Π_3.

To prove ii) notice that $|\mathrm{red}^*(w)|$ is a reduced path and thus contains each element of E^2 at most once (otherwise we have a dual edge cycle which can be removed). Thus $|\mathrm{red}^*(w)| \leq |E|^2$. □

5.2. Modification Principles of the Daisy Graph

In the analysis below, we shall require that the maximal number of edges in the daisy graph G_M is bounded polynomially in $|M|_{\mathrm{bit}}$. The initial number of labelled edges of the daisy graph G_M is $|E| = \sum_{q_i,q_j \in Q} |E_{q_i,q_j}|$ and this is polynomial in $|M|_{\mathrm{bit}}$ by Lemma 13. The maximal possible number of edges that will be added to G_M by our algorithm will be proven to be polynomial in the initial graph size. Other than the edges that we may add to G_M in the next section, Section 5.2.1, we will only ever add edges with a label from Ω between existing pairs of vertices q_i and q_j in the graph as we see in Section 5.2.2, and therefore the final graph will have a description size polynomial in $|M|_{\mathrm{bit}}$ since $|\Omega|$ is a constant.

5.2.1. Introduction of R_- and R_+ -edges

Consider a path given by $\Pi = (q_i, R_2, q_i)(q_i, R_1, q_j)$ in G_M, which implies that the path: $(q_i, R_2, q_i)^t(q_i, R_1, q_j)$ exists for any $t \geq 0$. Since $(R_2)^t R_1 \equiv R_{-1}R_{-(t+1)}$, we introduce a new vertex q and new edges by defining $E_{q_i,q} = \{R_{-1}\}$ and $E_{q,q_j} = \{R_-\}$, where R_- is the syllable defined previously, which stands for any $R_{-(t+1)}$ where $t \geq 0$.

Similarly, for path $\Pi = (q_i, R_1, q_j)(q_j R_2, q_j)$, we introduce a new vertex q' and new edges $q_i \xrightarrow{R_-} q'$ and $q' \xrightarrow{R_{-1}} q_j$.

Any paths with label $R_{-2}^t R_{-1}$ and $R_{-1}R_{-2}^t$ are treated analogously.

However, we must note that paths where the set of visited vertices are *distinct* with finitely many R_2-labels such as $(q_1, R_2, q_2) \cdots (q_{k-1}, R_2, q_k)(q_k, R_1, q_{k+1})$ in G_M, do not contain a self-loop, and thus arbitrary powers of R_2 are not necessarily possible. In this case, we just add a new vertex q and new edges $q_1 \xrightarrow{R_1} q$ and $q \xrightarrow{R_{-k}} q_{k+1}$. The cases with other path label combinations such as $R_{-1}R_{-2}^t$ are analogous.

In the continuation, we may assume that if we have a subpath of the form $\Pi = (q_i, R_2, q_i)^t(q_i, R_1, q_j)$, then we can alternatively take the (equivalent) path $(q_i, R_{-1}, q)(q, R_{-t}, q_j)$ instead. Similar conclusion holds for subpaths with labels $R_1 R_2^t$, $R_{-2}^t R_{-1}$, and $R_{-1}R_{-2}^t$.

5.2.2. Introduction of Ω-edges

Let $\Pi = (q_i, w, q_j)$ be a path in G_M from vertex q_i to vertex q_j such that $w = w_1 w_2 \cdots w_k \in (\Sigma - \{\varepsilon\})^k$, with $k \geq 2$, $w \equiv w' \in \Omega$, and $w \in \Phi$, i.e. w is Ω-minimal. Throughout this section, we ignore ε transitions, which we assume can be taken at any point without explicitly mentioning them.

We then introduce an edge with label w', i.e. $E_{q_i,q_j} := E_{q_i,q_j} \cup \{(q_i, w', q_j)\}$ (if it does not already exist).

We now describe three ways of showing that there is indeed such a path $q_i \xrightarrow{w'} q_j$.

1. *Short Reductions.* If $|w| \leq 3$, then we call path Π a short reduction. The existence of such a path can be directly checked for any vertex pair (q_i, q_j).
2. *Medium Reductions.* Let $|w| > 3$, such that Π *contains no dual edge cycles*, i.e. no consecutive pair of edges of the graph is used more than once (excluding ε-edges). In this case, we call Π a medium reduction from q_i to q_j.
3. *Long Reductions.* Let $|w| > 3$ such that Π contains at least one dual edge cycle, then we call Π a long reduction from q_i to q_j.

For the study of medium and long reductions of Ω-minimal words over Σ, where $|w| > 3$, the class $\mathcal{AF}$ gives a neat description of such words. We now show that any Ω-minimal word (Definition 42) of length at least four will be in Alternating Form (Definition 41).

Lemma 44. *Let $w \in \Phi$ and $|w| > 3$. Then $w \in \mathcal{AF}$.*

Proof. We proceed by contradiction and show that any word w of length at least 4 which is not in alternating form will not reduce to an element of Ω. To do this, we will use the reduction function ρ from Lemma 28. To simplify the analysis, we will also introduce the rule that $\rho(R_a s R_b) = R_{a+b}$ if $ab > 0$ in this proof. This property can immediately be deduced from the definition of R-syllables in Definition 14 and is simply a rewriting of equivalent ground level representations of a syllable.

Case 1) Assume $w = W_1 R_{x_1} R_{y_1} W_2$ where $W_1, W_2 \in \Sigma^*$ and $x_1 y_1 > 0$ (thus x_1 and y_1 have the same sign). Consider $\rho^*(W_1)$. If it has suffix $R_{-x_1 \pm 1}$ then we have a contradiction since then $R_{-x_1 \pm 1} R_{x_1} \in \Omega$ and thus w is not Ω-minimal and does not belong to Φ (note that $\rho^*(W_1) R_{x_1} R_{y_1} W_2$ is Ω-minimal if and only if $W_1 R_{x_1} R_{y_1} W_2$ is since ρ^* does not alter the suffix, other than

potentially adding symbol 's'). If the suffix is R_{x_2} with $x_1x_2 > 0$, then the syllables do not cancel, similarly if the suffix is $R_{-x_2}s$. If the suffix is $R_{x_2}s$, then since $R_{x_2}sR_{x_1} = R_{x_1+x_2}$, we may recursively consider $w = W_1'R_{x_1+x_2}R_{y_1}W_2$, where $W_1' = W_1R_{-x_2}$. The only other case is that $\rho^*(W_1)$ has suffix R_{-x_2} for $x_1x_2 > 0$.

If $\rho^*(W_1) = XR_{-x_2}$ with $|x_2| > |x_1|$, then we have $\rho^*(W_1R_{x_1}) = XR_{-x_2}R_{x_1} = XR_{-x_2+x_1}s$ and so $\rho^*(W_1R_{x_1})$ ends with a suffix of the form $R_{-x_2'}s$ for some x_2' of the same sign as x_1.

If $\rho^*(W_1) = XR_{-x_2}$ with $|x_2| < |x_1|$, then $\rho^*(W_1R_{x_1}) = XsR_{x_1-x_2}$. X cannot have suffix $R_{-x_3}s$ for $x_3x_2 > 0$, since $\rho(R_{-x_3}sR_{-x_2}) = R_{-(x_2+x_3)}$. If $X = X'R_{x_3}s$, then $\rho^*(W_1R_{x_1}) = X'\rho(R_{x_3}ssR_{x_1-x_2}) = X'R_{x_3}R_{x_1-x_2}$, which does not cancel and ends with a positive R syllable. If $X = X'R_{-x_3}$, then $\rho^*(W_1R_{x_1}) = X'\rho(R_{-x_3}sR_{x_1-x_2}) = X'R_{-x_3}sR_{x_1-x_2}$, again ending with a positive R syllable since there is no cancelation. Thus, $\rho^*(W_1R_{x_1})$ ends with a suffix of the form $R_{x_2'}$ for some x_2' of the same sign as x_1.

The above analysis therefore shows that if there is any left cancelation of syllable R_{x_1}, then the *suffix* of $\rho^*(W_1R_{x_1})$ is $R_{x_2'}$ or $R_{-x_2'}s$ where x_2' has the same sign as x_1. A similar analysis shows that the *prefix* of $\rho^*(R_{y_1}W_2)$ is of the form $R_{y_2'}$ or $R_{-y_2'}s$ where y_2' has the same sign as y_1 (and thus x_1). In fact, we can see that $|x_2'|, |y_2'| > 2$, since otherwise w contains a syllabic reduction to a word of the form $s^\alpha R_{\pm 1}s^\beta \in \Omega$, or $s^\alpha R_{\pm 2}s^\beta \in \Omega$ for $\alpha, \beta \in \{0, 1\}$, which is a contradiction since w is Ω-minimal.

Therefore, we see that $\rho(W_1R_{x_1})\cdot\rho(R_{x_2}W_2)$ has one of the following forms:

$$\begin{aligned} XR_{x_1'} &\cdot R_{y_1'}Y \\ XR_{-x_1'}s &\cdot R_{y_1'}Y \\ XR_{x_1'} &\cdot sR_{-y_1'}Y \\ XR_{-x_1'}s &\cdot sR_{-y_1'}Y \end{aligned}$$

for $X, Y \in \Sigma^*$ and x_1', y_1' of the same sign as x_1, y_1. Since there is no cancelation between the central elements of the first three of these cases, then the word cannot reduce under ρ to a word in Ω. In the final case, $XR_{-x_1'}s \cdot sR_{-y_1'}Y \equiv XR_{-x_1'}R_{-y_1'}X'$ has two central elements $R_{-x_1'}R_{-y_1'}$ with $|x_1'|, |y_1'| > 2$ and thus there is no further cancelation. Thus, a factor $R_{x_1}R_{y_1}$ cannot be present in any Ω-minimal word of length at least four as required.
Case 2) - $w = W_1R_{x_1}sR_{-y_1}W_2$ where $W_1, W_2 \in \Sigma^*$ and $x_1y_1 > 0$. In this case, an identical analysis to that above shows that the suffix of $\rho^*(W_1R_{x_1})$ is of one of the forms $\{XR_{x_1'}, XR_{-x_1'}s\}$ and the prefix of $\rho^*(R_{y_1}W_2)$ is of one of

the forms $\{R_{-y_1'}Y, sR_{y_1'}Y\}$, where $X, Y \in \Sigma^*$ and $|x_1|, |y_1| > 2$. Now we consider what happens when these elements are combined as $\rho^*(W_1 R_{x_1} s R_{-y_1} W_2)$ for these four cases.

In the case $\rho^*(W_1 R_{x_1}) \equiv X R_{x_1'}$ and $\rho^*(R_{y_1} W_2) \equiv R_{-y_1'}Y$, then $X R_{x_1'} \cdot s \cdot R_{-y_1'}Y$ is unchanged by the action of ρ since $R_{x_1'} \cdot s \cdot R_{-y_1'}$ has no cancelation. In the second case $\rho^*(W_1 R_{x_1}) \equiv X R_{x_1'}$ and $\rho^*(R_{y_1} W_2) \equiv s R_{y_1'}Y$, then $\rho(X R_{x_1'} \cdot s \cdot s R_{y_1'}Y) \equiv X R_{x_1'} \cdot R_{y_1'}Y$ with $x_1', y_1' > 2$ has already been considered above.

In case three, $\rho^*(W_1 R_{x_1}) \equiv X R_{-x_1'} s$ and $\rho^*(R_{y_1} W_2) \equiv R_{-y_1'}Y$, then $\rho(X R_{-x_1'} s \cdot s \cdot R_{-y_1'}Y) \equiv X R_{-x_1'} \cdot R_{-y_1'}Y$ with $x_1', y_1' > 2$ has again been considered above. In case four, $\rho^*(W_1 R_{x_1}) \equiv X R_{-x_1'} s$ and $\rho^*(R_{y_1} W_2) \equiv s R_{y_1'}Y$, thus $\rho(X R_{-x_1'} s \cdot s \cdot s R_{y_1'}Y) \equiv X R_{-x_1'} s \cdot R_{y_1'}Y$ which again is unchanged by the action of ρ. □

In fact, we can extend the previous Lemma to show that the weight of a word $w \in \Phi$ must be in the set $\{0, \pm 1, \pm 2\}$ and the value determines which elements in Ω word w may reduce to, as we now see.

Lemma 45. *Given a word $w \in \Phi$ with $|w| > 3$, then $|\mathrm{wgt}(w)| \leq 2$ and $w \equiv w'$, for some $w' \in \Omega$. Furthermore, if*

$$\mathrm{wgt}(w) = \begin{cases} \pm 2 & \Rightarrow \quad w' = s^\alpha R_{\pm 2} s^\beta \\ \pm 1 & \Rightarrow \quad w' = s^\alpha R_{\pm 1} s^\beta \\ 0 & \Rightarrow \quad w' \in \{s, \varepsilon\} \end{cases}$$

where $\alpha, \beta \in \{0, 1\}$.

Proof. Let $w = w_1 w_2 \cdots w_k \in \Phi$. Note that the action of ρ, defined in Definition 28 does not change the weight of word w. Consider thus $\rho^*(w) \equiv w' \in \Omega$. Since the weight of any syllable of Ω is $0, \pm 1, \pm 2$, and by Lemma 26 and Lemma 44, ρ reduces w to w' (since $w \in \Phi$ and thus $w \in \mathcal{AF}$), then the weight of w and w' are the same as required. □

The next technical lemma uses number-theoretic arguments and will be required later in order to bound the number of distinct dual edge cycles required in 'long reductions' to a polynomial value.

Lemma 46. *Let $1 \leq x, c_1, \ldots, c_{k_1}, d_1, \ldots, d_{k_2} < T$ such that there exists a sequence of integers $\alpha_1, \ldots, \alpha_{k_1}, \beta_1, \ldots, \beta_{k_2} > 0$ where:*

$$x + \sum_{j=1}^{k_1} \alpha_j c_j - \sum_{j=1}^{k_2} \beta_j d_j = 0. \tag{13}$$

Then, there exists $\{c'_1, \ldots c'_{k'_1}\} \subseteq \{c_1, \ldots c_{k_1}\}$, $\{d'_1, \ldots d'_{k'_2}\} \subseteq \{d_1, \ldots d_{k_2}\}$, $\alpha'_i, \beta'_i > 0$ *and* $k'_1, k'_2 \in O(\log T)$ *such that*

$$x + \sum_{j=1}^{k'_1} \alpha'_j c'_j - \sum_{j=1}^{k'_2} \beta'_j d'_j = 0. \tag{14}$$

Proof. Let $S = \{c_1, c_2, \ldots, c_k\}$ be a set of positive integers and p_M the largest prime divisor therein. We can then write

$$\begin{aligned} c_1 &= 2^{\alpha_{11}} \cdot 3^{\alpha_{12}} \cdot \ldots \cdot p_M^{\alpha_{1M}} \\ c_2 &= 2^{\alpha_{21}} \cdot 3^{\alpha_{22}} \cdot \ldots \cdot p_M^{\alpha_{2M}} \\ &\vdots \\ c_k &= 2^{\alpha_{k1}} \cdot 3^{\alpha_{k2}} \cdot \ldots \cdot p_M^{\alpha_{kM}} \end{aligned}$$

and if we take the minimal exponent of each column, say $\alpha_j = \min\{\alpha_{1j}, \alpha_{2j}, \ldots, \alpha_{kj}\}$, it is clear that

$$\gcd(c_1, c_2, \ldots, c_k) = 2^{\alpha_1} 3^{\alpha_2} \cdot \ldots \cdot p_M^{\alpha_M}.$$

The same gcd can be obtained by selecting at most M integers from set S: Choose c_{i_1} so that $\alpha_{i_1 1} = \alpha_1$ (the 1st column exponent is minimal), c_{i_2} so that $\alpha_{i_2 1} = \alpha_1$ (the 2nd column exponent is minimal), etc. until c_{i_M}. Some of the numbers c_{i_1}, …, c_{i_M} may be the same, but anyway $|S'| = |\{c_{i_1}, c_{i_2}, \ldots, c_{i_M}\}| \leq M$. To estimate M is straightforward:

$$c_1 = 2^{\alpha_{11}} 3^{\alpha_{12}} \cdot \ldots \cdot p_M^{\alpha_{1M}} \geq 2 \cdot 3 \cdot \ldots \cdot p_M \geq 2^M,$$

hence $M \leq \log_2 c_1$, and a similar estimate holds for any c_i. Hence $M \leq \log_2 T$, where $T = \max\{c_1, c_2, \ldots, c_k\}$. It is clear that for any S'' so that $S' \subset S'' \subset S$, we have $\gcd(S'') = \gcd(S') = \gcd(S)$.

Assume then that a Diophantine equation

$$x + \sum_{j=1}^{k_1} \alpha_j c_j - \sum_{j=1}^{k_2} \beta_j d_j = 0$$

has a solution $(\alpha_1, \ldots, \beta_1, \ldots)$ over the natural numbers (here it is assumed that k_1, $k_2 > 0$, i.e. that both signs really occur). As reasoned above, there is a set $\{c'_1, \ldots, c'_{k'_1}, d'_1, \ldots, d'_{k'_2}\}$ with cardinality at most $\log_2 T + 1$, where $T = \max\{c_1, \ldots, d_1, \ldots\}$ (+1 comes from the requirement that there has to

be at least one number of the opposite sign). Because of the gcd condition, we know that

$$x + \sum_{j=1}^{k_1'} \alpha_j c_j' - \sum_{j=1}^{k_2'} \beta_j d_j' = 0 \tag{15}$$

has a some solution $(\alpha_1, \ldots, \beta_1, \ldots)$ over the integers. To simplify the notations, remove the primes and rewrite (15) as

$$x + \sum_{j=1}^{k_1} \alpha_j c_j - \sum_{j=1}^{k_2} \beta_j d_j = 0. \tag{16}$$

Let then $B = c_1 \ldots c_{k_1} d_1 \ldots d_{k_2}$. Now for any $n \in \mathbb{Z}$,

$$\begin{aligned} & \sum_{j=1}^{k_1} (\alpha_j + nk_2 \frac{B}{c_j}) c_j - \sum_{j=1}^{k_2} (\beta_j + nk_1 \frac{B}{d_j}) d_j \\ = & \sum_{j=1}^{k_1} \alpha_j c_j - \sum_{j=1}^{k_2} \beta_j d_j + nk_1 k_2 B - nk_1 k_2 B, \end{aligned}$$

which shows that for any $n \in \mathbb{Z}$, $\alpha_j \mapsto \alpha_j + nk_2 \frac{B}{c_j}$ (and similarly for β_j) yields another solution to (16). It follows that there is a solution where each α (and β)is positive.

We now estimate the magnitude of the positive integers α and β in the solution. In fact, B could be could even be replaced with $\frac{B}{g^{k_1+k_2}}$, where $g = \gcd(c_1, \ldots, d_1, \ldots)$, but even without such a replacement we have that

$$B = c_1 \ldots c_{k_1} d_1 \ldots d_{k_2} \leq T^{k_1+k_2},$$

hence the bit size of B is at most

$$\log_2 B \leq (k_1 + k_2) \log_2 T \leq (\log_2 T + 1) \log_2 T.$$

□

We also require the following technical lemma. This will allow us to determine that if we have two words $w_1 \in \Phi$ and $w_2 \in \mathcal{AF}$ starting with the same syllable, and ending with the same syllable, then if they have the same weight they will reduce to exactly the same element of Ω.

Lemma 47. *Let $\Sigma' = \Sigma - \{R_-, R_+\}$ and $w_1 = uXv$, where $u, v \in \Sigma'$ and $X \in \Sigma'^*$ such that $|w_1| > 3$, $|\mathrm{wgt}(w_1)| \leq 2$ and $w_1 \in \Phi$. Then $w_1 \equiv w'$ for some unique $w' \in \Omega$ and for any word $w_2 = uYv$ where $Y \in \Sigma'^*$, $Y \in \mathcal{AF}$ and $\mathrm{wgt}(w_2) = \mathrm{wgt}(w_1)$, then $uYv \equiv w'$.*

Proof. Note that if $u = s$ or $v = s$, then $w_1 \notin \Phi$ as is not difficult to see. For example if $u = s$, and $\rho^*(uXv) \in \Omega$, then it implies that $\rho^*(Xv) \in \Omega$ and thus $w_1 \notin \Phi$. We may therefore assume that $u = R_a$ and $v = R_b$ for some $a, b \in \mathbb{Z} - \{0\}$.

If $\mathrm{wgt}(w_1) = 0$, then $w' = \varepsilon$ or $w' = s$ by definition of wgt and Ω. In both cases since $\mathrm{wgt}(w_2) = \mathrm{wgt}(w_1) = 0$, then $w_2 \equiv w'$ since application of the reduction rules of Lemma 26 only remove a multiple of 2 's' syllables from a word as can easily be verified.

Therefore assume that $\mathrm{wgt}(w_1) = t \in \{\pm 1, \pm 2\}$. Thus we have $w_1 \equiv s^{\alpha_1} R_t s^{\beta_1}$ and $w_2 \equiv s^{\alpha_2} R_t s^{\beta_2}$. We prove that $\alpha_1 = \alpha_2$ and $\beta_1 = \beta_2$ which will prove the Lemma.

Clearly $\mathrm{wgt}(X) = \mathrm{wgt}(Y)$ and since $w_1, w_2 \in \mathcal{AF}$, then it follows that $X, Y \in \mathcal{AF}$ because a subword of a word in $\mathcal{AF}$ is also in $\mathcal{AF}$. Assume by contradiction that $\alpha_1 = 1$ and $\alpha_2 = 0$, i.e. that $w_1 = R_a X R_b \equiv s R_t s^{\beta_1}$ and $w_2 = R_a Y R_b \equiv R_t s^{\beta_2}$. Then, $X \equiv R_{-a} s R_t s^{\beta_1} R_{-b}$ and $Y \equiv R_{-a} R_t s^{\beta_2} R_{-b}$. Since $X, Y \in \mathcal{AF}$, then $\mathrm{sgn}(a) = -\mathrm{sgn}(t)$ in order that $R_{-a} s R_t \in \mathcal{AF}$. However, $\mathrm{sgn}(a) = \mathrm{sgn}(t)$ in order that $R_{-a} R_t \in \mathcal{AF}$. Since $t \neq 0$, this give a contradiction. A similar proof shows that if $\alpha_1 = 0$ and $\alpha_2 = 1$, i.e. if $w_1 \equiv R_t s^{\beta_1}$ and $w_2 \equiv s R_t s^{\beta_2}$, then we get a contradiction. Therefore $\alpha_1 = \alpha_2$.

Assume by contradiction that $\beta_1 = 1$ and $\beta_2 = 0$, i.e. that $w_1 = R_a X R_b \equiv s^{\alpha_1} R_t s$ and $w_2 = R_a Y R_b \equiv s^{\alpha_1} R_t$. Then, $X \equiv R_{-a} s^{\alpha_1} R_t s R_{-b}$ and $Y \equiv R_{-a} s^{\alpha_1} R_t R_{-b}$. Since $X, Y \in \mathcal{AF}$, then $\mathrm{sgn}(b) = -\mathrm{sgn}(t)$ in order that $R_t s R_{-b} \in \mathcal{AF}$. However, $\mathrm{sgn}(b) = \mathrm{sgn}(t)$ in order that $R_t R_{-b} \in \mathcal{AF}$. Since $t \neq 0$, this again gives a contradiction. Thus we see that $\alpha_1 = \alpha_2$ and $\beta_1 = \beta_2$ as required. □

Lemma 48. *Let $\Pi = (q_i, w, q_j) \in E^k$ be a path in G_M from a vertex q_i to a vertex q_j such that $w = w_1 w_2 \cdots w_k \in \Phi$ and $k \geq 2$. Then a certificate for the derivation of an edge (q_i, w', q_j), with $w \equiv w' \in \Omega$, can be nondeterministically found in time polynomial in $|M|_{bit}$.*

Proof. We shall deal with three separate cases. In the proof, we again ignore any ε transitions, which we may assume can be taken without explicitly

mentioning them.

1) Short reductions. In this case, $k \leq 3$ and we can verify that $w \equiv w' \in \Omega$ trivially via the reductions shown in Lemma 26. The only remaining cases involve syllables R_- and R_+.

If $w_1w_2 = R_+\lambda_1$, $w_1w_2 = \lambda_2R_-$, $w_1w_2 = R_-\lambda_2$, $w_1w_2 = \lambda_1R_+$, $w_1w_2 = R_-R_+$ or $w_1w_2 = R_+R_-$, where $\lambda_1 \in \Gamma_-$ and $\lambda_2 \in \Gamma_+$: then the following edges all belong to E_{q_i,q_j}: $\{\varepsilon, R_2s, R_1s, sR_1, sR_2\}$.

To see this, let us consider the first rule $w_1w_2 = R_+\lambda_1$, where $\lambda_1 = R_{-x}$ for some $x > 2$ as an example. The other cases follow in a similar analysis. Since syllable R_+ allows us to derive any syllable R_k, where $k \geq 1$, then we can easily verify that the following are all valid labels of edges from q_i to q_j:

$$\begin{array}{lll} R_{x-2}R_{-x} \equiv sR_{-2}; & R_{x-1}R_{-x} \equiv sR_{-1}; & R_xR_{-x} \equiv \varepsilon; \\ R_{x+1}R_{-x} \equiv R_1s; & R_{x+2}R_{-x} \equiv R_2s. & \end{array}$$

Such a path can be found and verified in time polynomial in $|M|_{\text{bit}}$. Thus any short reductions can be found.

2) Medium reductions. In this case, $k > 3$ and Π does not contain a dual edge cycle (as throughout, cycles will mean dual edge cycles unless otherwise stated). We may assume that $w \in \mathcal{AF}$ by Lemma 44. By Lemma 43, we know that $|w| \leq |E|^2$ since $\text{red}(w) = w$. Such a path Π can be guessed in polynomial time and we can verify that $w \equiv w' \in \Omega$ holds by applying the reductions rules of Lemma 26.

3) Long reductions. In this case $k > 3$ and Π contains at least one dual edge cycle. This is the most difficult case and we split the analysis into two subcases. Since $w \in \Phi$, we may assume that $w \in \mathcal{AF}$ by Lemma 44, and that $|\text{wgt}(\Pi)| \leq 2$, with the weight determining which element of Ω we reduce to, up to factors of 's' by Lemma 45. We shall show a way to find an equivalent path $\Pi_2 = (q_i, w_2, q_j)$, such that $w_2 \in \mathcal{AF}$, $\text{wgt}(w_2) = \text{wgt}(w)$ and Π_2 contains no more than a polynomial (in terms of $|M|_{\text{bit}}$) number of reduced dual edge cycles, which will allow us to verify that $w_2 \equiv w \equiv w' \in \Omega$ succinctly.

In this step, we may assume that Π does not contain a subpath of the form $(q_i, R_1, q_j)(q_j, R_2, q_j)$ or $(q_j, R_2, q_j)(q_j, R_1, q_k)$ (or the version with R_{-1} and R_{-2}). This is because an equivalent path exists in the graph using word R_- (R_+ resp.) by Section 5.2.1. In both cases 3a and 3b below, the presence of such a path within Π implies that dual edge cycles of arbitrary positive or negative weight exist, and then in both cases a solution is trivial to find

(since the main difficulty in these cases is finding an equivalent path with low descriptional complexity of a given weight). Therefore in the analysis below we shall exclude syllables R_- and R_+, as well as subwords of the form $R_1R_2^t$, $R_2^tR_1$, $R_{-1}R_{-2}^t$ and $R_{-2}^tR_{-1}$.

3a) Π ***contains both positive and negative weight dual edge cycles***. I.e. $\Pi = X_1C_1Y_1 = X_2C_2Y_2$ such that C_1 and C_2 are dual edge cycles and $\mathrm{wgt}(C_1) \cdot \mathrm{wgt}(C_2) < 0$, with $X_1, X_2, Y_1, Y_2 \in E^*$.

Each *reduced* dual edge cycle C_i present in Π has a weight, which we denote by c_i if the weight is positive and d_i if the weight is negative (we take the absolute value of a negative weight, so all c_i, d_i are positive). Let $x = \mathrm{wgt}(\mathrm{red}^*(\Pi))$ and assume without loss of generality that $x > 0$. Note that x is not unique, since red is nondeterministic. By Lemma 46, if there exists a solution to $x + \sum_{j=1}^{k_1} \alpha_j c_j - \sum_{j=1}^{k_2} \beta_j d_j = 0$, then there also exists a solution when $k_1, k_2 \in O(\log T)$, where T is the sum of absolute values of edge label weights in the daisy graph G_M. This corresponds to choosing a subset of the reduced dual edge cycles of Π.

We now note a technical concern. The proof of Lemma 46 proceeds by removing unnecessary terms from set $\{c_i\}$ and $\{d_i\}$ whilst retaining the gcd. However, we may choose some term c_{i_1}, corresponding to some reduced cycle C_{i_1}, whilst removing some other term c_{i_2}, corresponding to some reduced cycle C_{i_2}. The cycle C_{i_1} may not be directly connected to path $\mathrm{red}^*(\Pi)$ however, and C_{i_2} may need to be present, at least once, in order to allow cycle C_{i_1} to be taken. In this case, we may add c_{i_2} to the set of chosen gcd values however, which potentially increases the size of set $\{c_i\}$ by a factor of two. In this case, the coefficient of c_{i_2} (denoted α_{i_2}) must be nonzero, since C_{i_2} must be chosen at least once, in order to allow C_{i_1} to be traversed. However, if we have a solution to Equation (14) when $\alpha_{i_2} = 0$, then choose any term $\beta_k d_k$ and update $\alpha_{i_2} := d_k$ and $\beta_k := \beta_k + c_{i_2}$ and then a solution still exists and $\alpha_{i_2}, \beta_k > 0$. To see this, note that $0 \cdot c_{i_2} - \beta_k d_k = d_k c_{i_2} - (c_{i_2} + \beta_k) d_k$. A similar analysis holds for the elements of set $\{d_i\}$.

To find a certificate for an Ω-minimal word w along a path from q_i to q_j, which is reducible to $w' \in \Omega$, we can thus:

a) Nondeterministically guess a reduced path Π', in Alternating Form, between nodes q_i and q_j of length $\leq |E|^2$ and of weight x.
b) Nondeterministically guess $O(\log T)$ positive (resp. negative) reduced dual edge cycles that can be 'reinserted' in to Π' and denote their weight by c_i (resp. d_i). The length of each such cycle is bounded by $|E|^2$ by

Lemma 43, since they are reduced. This new path may be denoted $\Pi'' = (q_i, w'', q_j)$. Note that $\Pi \in \mathcal{AF} \Rightarrow \Pi'' \in \mathcal{AF}$ by Lemma 43.

c) Verify that $x + \sum_{j=1}^{k_1} \alpha_j c_j - \sum_{j=1}^{k_2} \beta_j d_j = \mathrm{wgt}(w')$, where $|\mathrm{wgt}(w')| \leq 2$ for some guessed values $\alpha_j, \beta_j \geq 1$.

Note that this procedure is guaranteed to find a syllable in Ω with the same weight as $w' \in \Omega$. Note also in this procedure that since Π'' and Π start and end with the same syllable (since the procedure only removes and reinserts dual edge cycles which leaves the first and last syllables unchanged), and since $\Pi'' \in \mathcal{AF}$, then Lemma 47 implies that $\Pi \equiv \Pi'' \equiv w' \in \Omega$ as required.

3b) Π ***only contains dual edge cycles of the same sign.***

By abuse of notation, let $\Pi'(\tau) \geq 0$ denote the number of occurences of a subpath $\tau \in E^+$ within a path Π'. For example, if $\Pi' = e_1e_2e_3e_1e_2e_4e_3e_1e_2$, then $\Pi'(e_1e_2) = 3$ and $\Pi'(e_4) = 1$.

Our aim is to construct a path Π_z such that $\Pi_z(\tau) = \Pi(\tau)$ for all $\tau \in E^2$, where $\Pi_z \in \mathcal{AF}$. Crucially, Π_z will have a simple description and acts thus as a certificate for path Π from q_i to q_j.

Let $\Pi_1 = \mathrm{red}^*(\Pi)$. Our approach will be to nondeterministically guess a *reduced dual edge cycle* Π_* and 'insert' a power of Π_* into Π_i to give a path Π_{i+1}, starting from $i = 1$. The idea of this procedure is that the description of this new path has a polynomial description in terms of $|M|_{\mathrm{bit}}$. This procedure of inserting powers of a reduced dual edge cycle will generate paths $\Pi_1, \Pi_2, \ldots, \Pi_z$ where we will reach a stopping condition that for all $\tau \in E^2$, then $\Pi_z(\tau) = \Pi(\tau)$. The choice of the dual edge cycles will ensure that $z \leq |E|^2$ and each cycle is taken to a bounded power. We will show that $\Pi_i \in \mathcal{AF} \Rightarrow \Pi_{i+1} \in \mathcal{AF}$ and since $\Pi_1 \in \mathcal{AF}$ by Lemma 43 then by induction this will show that $\Pi_z \in \mathcal{AF}$. The constructed path Π_z will then act as a certificate for path Π.

Now we show how to find Π_* for a given Π_i. Assume that $\Pi_i(\tau) \leq \Pi(\tau)$ for all $\tau \in E^2$. This certainly holds for $i = 1$, since red only removes dual edge cycles. Nondeterministically choose a reduced dual edge cycle $\Pi_* = \pi_1\pi_2 \cdots \pi_m\pi_1\pi_2 \in E^*$, such that $\Pi_i(\pi_1\pi_2) > 0$ and for each $\tau \in E^2$ such that $\Pi_*(\tau) \geq 1$, then $\Pi(\tau) - \Pi_i(\tau) \geq 1$. Note that by the definition of a reduced dual edge cycle, $|\Pi_*| \leq |E|^2 + 2$. For each $\tau \in E^2$, then $\Pi_*(\tau) = 0$ if τ is not a subpath of Π_*, $\Pi_*(\tau) = 2$ if τ is equal to $\pi_1\pi_2$ and $\Pi_*(\tau) = 1$ otherwise. Define $x = \min\{\Pi(\tau) - \Pi_i(\tau); \tau \in E^2 \text{ and } \Pi_*(\tau) \geq 1\}$. Therefore,

$x \geq 1$ by the choice of Π_* and x denotes the minimum difference between the number of times some $\tau \in E^2$ appears in Π and in Π_i.

Recall that we assumed all dual edge cycles have the same sign. Let $b = \text{wgt}(\Pi_1)$. Assume without loss of generality that $b < -2$ and therefore all dual edge cycles of Π have a positive weight (otherwise the weight of Π would certainly be less than -2). Note that b has a description size which is polynomial in $|M|_{\text{bit}}$, since $|\Pi_1| \leq E^2 + 2$ and so $|b|$ is no more than two times the sum of all edge weights in the graph G_M.

Now, since $\Pi_i(\pi_1\pi_2) > 0$, then we can write $\Pi_i = \Pi_i'\pi_1\pi_2\Pi_i'' \in E^*$, where $\Pi_i', \Pi_i'' \in E^*$. We define $\Pi_{i+1} = \Pi_i'(\pi_1\pi_2 \cdots \pi_m)^x\pi_1\pi_2\Pi_i'' \in E^*$ (we intuitively call this 'inserting' Π_*^x into Π_i). Clearly, Π_{i+1} is a path in G_M since $\pi_1\pi_2$ was already a subpath of Π_i and Π_* is a dual edge cycle. Since each cycle has a positive weight (at least 1) then $x \leq |b| + 4$ because otherwise $\text{wgt}(\Pi_{i+1}) > 2$ and any additional (positive) dual edge cycles that are added to Π_{i+1} will only increase the weight, even though $|\text{wgt}(\Pi)| \leq 2$. At this point then, notice that x is bounded polynomially in $|M|_{\text{bit}}$.

Furthermore, invariant $\Pi_{i+1}(\tau) \leq \Pi(\tau)$ still holds for all $\tau \in E^2$ by the choice of x. Crucially, notice that there exists some $\tau \in E^2$ such that $\Pi(\tau) - \Pi_i(\tau) > 0$ and $\Pi(\tau) - \Pi_{i+1}(\tau) = 0$; this is just the τ that defined value x. Each time we repeat this procedure, there exists some new $\tau \in E^2$ such that the number of occurences of τ in Π and Π_{i+1} is equal. Since $\tau \in E^2$, then this procedure can be repeated no more than $|E|^2$ times to generate some path Π_z, after which for every pair $\tau \in E^2$, we have that $\Pi_z(\tau) = \Pi(\tau)$.

By Lemma 43, we know that function red retains Alternating Form for paths (i.e. if path $\Pi' \in \mathcal{AF}$, then $\text{red}(\Pi') \in \mathcal{AF}$). A minor modification of the proof also shows that if $\Pi_i = \Pi_i'\pi_1\pi_2\Pi_i'' \in \mathcal{AF}$, then $\Pi_{i+1} = \Pi_i'\Pi_*^x\Pi_i'' \in \mathcal{AF}$, since inserting a dual edge cycle also retains the required local properties of syllables.

The final part to verify is that this procedure can be carried out iteratively until $\Pi_z(\tau) = \Pi(\tau)$ for all $\tau \in E^2$. The only way that this can fail is if at some point we generate path Π_i and there does not exist a reduced dual edge cycle $\Pi_* \in E^*$ which can be 'inserted' into Π_i, i.e. for some $\tau \in E^2$ which is a subpath of Π_*, then $\Pi(\tau) - \Pi_i(\tau) = 0$, which means that we cannot use τ again while maintaining invariant $\Pi_{i+1}(\tau) \leq \Pi(\tau)$.

Let $\Lambda(\Pi_i) = \{\tau' \mid \tau' \in E^2 \text{ and } \Pi(\tau') - \Pi_i(\tau') \geq 1\}$. Thus, $\Lambda(\Pi_i)$ is just the set of dual edges which are present more in Π than in Π_i.

Assume then by contradiction that $|\Lambda(\Pi_i)| \geq 1$, but there does not exist a reduced dual edge cycle which only uses edges of $\Lambda(\Pi_i)$. In this case, we

cannot insert another cycle into Π_i, even though $\Pi(\tau) - \Pi_i(\tau) \geq 1$ for some $\tau \in \Lambda(\Pi_i)$.

Let $\tau_1 = (q_j, u_1, q) \in E$, $\tau_2 = (q, u_2, q_k) \in E$ and $\tau_c = \tau_1\tau_2 \in \Lambda(\Pi_i) \subseteq E^2$. If there exists some edge $e_l = (q'_j, u'_1, q_j) \in E$ such that $\tau_c(e_l\tau_1) = 0$ and $e_l\tau_1 \in \Lambda(\Pi_i)$, then we 'extend' τ_c to the left to give $\tau_c \mapsto e_l\tau_c$. Note that τ_c is still a valid path. This procedure is performed iteratively. Now, since we only left extend τ_c if it does not cause repetition of some dual edge, then this procedure must eventually halt for some τ_c^* and then $\|\tau_c^*| \leq |\Lambda(\Pi_i)| \leq |E|^2$. Note also that τ_c^* is not a dual edge cycle by our above assumption that no such cycle is possible using only elements of $\Lambda(\Pi_i)$. Now, τ_c^* is a path from some vertex q_1 to q_k that cannot be further left extended by any edges from $\Lambda(\Pi_i)$.

Let $\mathrm{In} : E^* \times Q \to \mathbb{N}$ be a function such that $\mathrm{In}(\Pi', q')$ denotes the number of edges of $\Pi' \in E^*$ going to vertex $q' \in Q$, plus 1 if Π' starts at vertex q'. Similarly, $\mathrm{Out} : E^* \times Q \to \mathbb{N}$ is a function such that $\mathrm{Out}(\Pi', q')$ denotes the number of edges of $\Pi' \in E^*$ leaving vertex $q' \in Q$, plus 1 if Π' ends at vertex q'. For example, given path:

$$\Pi' = (q'_1, w'_1, q'_2)(q'_2, w'_2, q'_3)(q'_3, w'_3, q'_2)(q'_2, w'_5, q'_3),$$

then $\mathrm{In}(\Pi', q'_1) = 1$, $\mathrm{Out}(\Pi', q'_1) = 1$, $\mathrm{In}(\Pi', q'_2) = 2$, $\mathrm{Out}(\Pi', q'_2) = 2$ and $\mathrm{In}(\Pi', q'_3) = 2$, $\mathrm{Out}(\Pi', q'_3) = 2$. These functions can be defined formally for $\Pi' = \pi'_1\pi'_2 \cdots \pi'_{k'} \in E^*$ as follows:

$$\begin{aligned} \mathrm{In}(\Pi', q') &= \sum_{\pi_{i'} = (q'', w', q')} 1 + \sum_{\pi_1 = (q', w', q'')} 1, \\ \mathrm{Out}(\Pi', q') &= \sum_{\pi_{i'} = (q', w', q'')} 1 + \sum_{\pi_{k'} = (q'', w', q')} 1, \end{aligned}$$

where $1 \leq i' \leq k'$, $q'' \in Q$ and $w' \in \Sigma - \{\varepsilon\}$. Note that the second summation of function In/Out adds 1 if and only if Π' begins/ends at vertex q'.

Note that for any path $\Pi' \in E^2$ and vertex $q' \in Q$:

$$\mathrm{In}(\Pi', q') = \mathrm{Out}(\Pi', q'). \tag{17}$$

Consider vertex q_1. Since τ_c^* cannot be further left extended from vertex q_1, then for all $\tau \in E^2$ of the form $(q_{y_1}, w_{y_1}, q_1)(q_1, w_{x_1}, q_{x_1})$, for any $q_{y_1}, q_{x_1} \in Q$ and $w_{x_1}, w_{y_1} \in \Sigma - \{\varepsilon\}$, then $\Pi(\tau) - \Pi_i(\tau) = 0$, and thus $\tau \notin \Lambda(\Pi_i)$. This implies that

$$\mathrm{In}(\Pi, q_1) = \mathrm{In}(\Pi_i, q_1). \tag{18}$$

Since there exists some path $\tau_l = (q_1, w_{x_2}, q_{x_2})(q_{x_2}, w_{y_1}, q_{y_2}) \in E^2$ such that $\tau_l \in \Lambda(\Pi_i)$, then $\Pi(\tau_l) - \Pi_i(\tau_l) > 0$, then it implies that

$$\mathrm{Out}(\Pi, q_1) > \mathrm{Out}(\Pi_i, q_1). \tag{19}$$

Combining Invariant 17, Equality 18 and Inequality 19, we obtain the following contradiction:

$$\begin{aligned} \mathrm{In}(\Pi_i, q_1) &= \mathrm{Out}(\Pi_i, q_1) \\ &< \mathrm{Out}(\Pi, q_1) \\ &= \mathrm{In}(\Pi, q_1) \\ &= \mathrm{In}(\Pi_i, q_1) \end{aligned}$$

To recap then, given $\Pi \in \Phi$ such that $|\Pi| > 2$ and Π contains only dual edge cycles of positive sign, we first define $\Pi_1 = \mathrm{red}^*(\Pi)$, which we showed has a polynomial length (polynomial in terms of $|M|_{\mathrm{bit}}$). We then define some Π_* and some $x > 0$, such that $|\Pi_*|$ and x are polynomial in size and we define Π_{i+1} by 'inserting' Π_*^x into Π_i. We repeat this procedure no more than $|E|^2 + 2$ times, and therefore the procedure is polynomial in $|M|_{\mathrm{bit}}$. Finally this gives us a path Π_z. We showed that $\Pi_i \in \mathcal{AF} \Rightarrow \Pi_{i+1} \in \mathcal{AF}$ and since $\Pi \in \mathcal{AF} \Rightarrow \Pi_1 \in \mathcal{AF}$, by Lemma 43, this implies that $\Pi_z \in \mathcal{AF}$. Since, by definition, $\Pi_z(\tau) = \Pi(\tau)$ for all $\tau \in E^2$, then $\mathrm{wgt}(\Pi_z) = \mathrm{wgt}(\Pi)$. It is clear that the first and last syllables of Π and Π_z are the same, since function red does not alter the first or last two syllables of any word. Therefore by Lemma 47, since $|\Pi| > 3$, $\Pi \in \Phi$, $\mathrm{wgt}(\Pi) = \mathrm{wgt}(\Pi_z)$ and $\Pi_z \in \mathcal{AF}$, then $\Pi_z \equiv \Pi$ as required. □

We conclude the aforementioned procedure in a theorem:

Theorem 49. *The identity problem over* $\mathrm{PSL}_2(\mathbb{Z})$ *is in* **NP**.

Recall from Remark 37 that the described procedure is not limited to the daisy graph and works for any regular expression $R(a_1, \ldots, a_n)$. We define a function ϕ mapping each letter of a regular expression to its corresponding matrix in $\mathrm{PSL}_2(\mathbb{Z})$.

Corollary 50. *The problem of determining whether the identity matrix is in* $\phi(R(a_1, \ldots, a_n)) \subseteq \mathrm{PSL}_2(\mathbb{Z})$ *for an arbitrary regular expression* $R(a_1, \ldots, a_n)$ *is in* **NP**.

Recall also that elements of $\mathrm{PSL}_2(\mathbb{Z})$ are actually matrix pairs: $a = \{A, -A\} \subset \mathrm{SL}_2(\mathbb{Z})$. Let $\langle M' \rangle_{\mathrm{sg}}$ be a semigroup generated by some finite $M' \subseteq \mathrm{SL}_2(\mathbb{Z})$. We may then construct a syllabic automaton for the projection of M' in $\mathrm{PSL}_2(\mathbb{Z})$ only losing the information about the sign. If I belongs to the projection of $\langle M' \rangle_{\mathrm{sg}}$, then either I or $-I$ belongs to $\langle M' \rangle_{\mathrm{sg}}$. But in the latter case, $I = (-I)^2$ also belongs to $\langle M' \rangle_{\mathrm{sg}}$. Hence we obtain the following corollary:

Corollary 51. *The identity problem over* $\mathrm{SL}_2(\mathbb{Z})$ *is in* **NP**.

Theorem 52. *Determining if a matrix* $M \in \mathrm{PSL}_2(\mathbb{Z})$ *is in* $\phi(R(a_1, \ldots, a_n)) \subseteq \mathrm{PSL}_2(\mathbb{Z})$ *for an arbitrary regular expression* $R(a_1, \ldots, a_n)$ *is in* **NP**.

Proof. The decidability of the problem was shown in [17] as it can be reduced to the identity problem for a particular regular expression in $\mathrm{PSL}_2(\mathbb{Z})$, i.e. whether $I \in M^{-1}\phi(R(a_1, \ldots, a_n))$. Let $M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ and since $\det(M) = 1$, it follows that the inverse matrix $M^{-1} = \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$ as well as its syllabic representation will be of the same size as the matrix M. Then the statement of this theorem directly follows from Corollary 50 as deciding whether $I \in M^{-1}\phi(R(a_1, \ldots, a_n))$ is in **NP**. □

We now extend Theorem 52 to cover $\mathrm{SL}_2(\mathbb{Z})$ rather than just $\mathrm{PSL}_2(\mathbb{Z})$. We cannot use the same proof idea as for Corollary 51 since if $-I \in M^{-1}\phi(R(a_1, \ldots, a_n))$, it does not necessarily follow that $I \in M^{-1}\phi(R(a_1, \ldots, a_n))$. We thus modify a proof technique from [17] to show this corollary.

Theorem 53. *Determining if a matrix* M *is in* $\phi(R(a_1, \ldots, a_n)) \subseteq \mathrm{SL}_2(\mathbb{Z})$ *for an arbitrary regular expression* $R(a_1, \ldots, a_n)$ *is in* **NP**.

Proof. We may assume that $M \in \mathrm{SL}_2(\mathbb{Z})$. Let μ and μ^{-1} be new letters (which will be used to represent matrices M and M^{-1} respectively).

Define $\phi : \Sigma^* \to \mathrm{PSL}_2(\mathbb{Z})$ to map alphabet $\Sigma = \{\mu, \mu^{-1}, a_1, a_2, \ldots a_n\}$ to their corresponding matrices in $\mathrm{PSL}_2(\mathbb{Z})$. Let M mod 3 denote matrix M with modulus 3 applied componentwise. Note that I mod $3 = I$ and $-I$ mod $3 = 2I \neq I$. We define a function $\theta : \mathbb{Z}^{2\times 2} \to (\mathbb{Z}/3\mathbb{Z})^{2\times 2}$ which maps each matrix $X \in \mathbb{Z}^{2\times 2}$ to X mod 3. Consider $\phi(\mu^{-1}R(a_1, \ldots, a_n))$. Notice that language

$$L = \{l : l \in \{\mu, \mu^{-1}, a_1, a_2, \ldots, a_n\}^* \text{ and } \theta(\phi(l)) = I\}$$

is regular, thus so is $L' = \mu^{-1} \cdot R(a_1, \ldots, a_n) \cap L$. If $I \in \phi(L')$ it thus implies that $M \in \phi(R(a_1, \ldots, a_n))$. □

We now extend this result to show that the membership problem for $\mathrm{GL}_2(\mathbb{Z})$ is decidable in NP. The difficulty here is that we do not have an embedding from $\mathrm{GL}_2(\mathbb{Z})$ to $\mathrm{PSL}_2(\mathbb{Z})$ and so we need a way to deal with matrices having determinant -1. We use a similar technique to that used in [17] to deal with this problem.

Theorem 54. *The membership problem for* $\mathrm{GL}_2(\mathbb{Z})$ *is in* **NP**.

Proof. Let $X \in \mathrm{GL}_2(\mathbb{Z})$ be the target matrix, and $G = \{X_1, \ldots, X_k\} \subseteq \mathrm{GL}_2(\mathbb{Z})$ the generator set of matrices. Since $\det(X) = \pm 1$, it is clear that $X \in \langle G \rangle$ implies that $I \in X^{-1}\langle G \rangle$. In this proof we consider rational subsets of matrices by allowing regular operations (concatenation, Kleene star etc.) on matrix sets in the natural way.

We may assume that if there exists a product $I = X^{-1}M_1 \cdots M_n$ with $M_j \in G$ for $1 \leq j \leq n$, then the number of matrices in the product with determinant -1 is even. This holds since if $\det(X) = 1$, then $\det(X^{-1}) = 1$ and thus $\det(M_1 \cdots M_n) = 1$. Similarly, if $\det(X) = -1$, then $\det(X^{-1}) = -1$ and thus $\det(M_1 \cdots M_n) = -1$, so again an even number of matrices in the product have negative determinant.

We do not have an embedding from $\mathrm{GL}_2(\mathbb{Z})$ to $\mathrm{PSL}_2(\mathbb{Z})$. Thus we use the following idea from [17].

Assume first that $\det(X) = 1$. Let J denote the subset of integers from $1 \leq i \leq k$ where $\det(X_i) = -1$. Define $X_{i,\ell} = X_i X_\ell X_i^{-1}$ for all $i \in J$ and $\ell \notin J$. Furthermore, denote $Z_{i,k} = X_i X_k$ for $i, k \in J$. Finally, let $A = \{X_i | i \notin J\}$, $B_i = \{X_{i,k} | k \notin J\}$ for $i \in J$, and $Z_i = \{Z_{i,k} | k \in J\}$ for $i \in J$. Note therefore that the determinant of all matrices within A, B_i, and Z_i is 1, and thus $A, B_i, Z_i \subseteq \mathrm{SL}_2(\mathbb{Z})$; thus each such matrix can be embedded into $\mathrm{PSL}_2(\mathbb{Z})$ as previously discussed.

We form the previously described 'petal graph' recognising matrix products of the form $X^{-1}A^*(B_i^* Z_i A^*)^*$ for $i \in J$. Note that the size of this petal graph remains polynomial, giving only a quadratic increase in its overall size.

We now show the correctness of this petal graph, i.e., we show that $I \in X^{-1}\langle G \rangle$ if and only if $I \in X^{-1}A^*(B_i^* Z_i A^*)^*$ which proves that the membership problem is decidable in **NP** by Theorem 53.

Firstly, assume that $I \in X^{-1}\langle G \rangle$. This means that there exists $A_1 \cdots A_n$ with $A_j \in G$ for $1 \leq j \leq n$ with $I = X^{-1}A_1 \cdots A_n$. As before, we may

assume that the number of matrices in the product with determinant -1 is even.

The key argument is to consider the left most two matrices with determinant -1 if they exist, denoted A_i and A_j. We then consider $A_iA_{i+1}\cdots A_{j-1}A_j = X_{i_1}X_{i_2}\cdots X_{i_j}$ with each index $1 \leq i_p \leq k$ and replace it by the product $X_{i_1,i_2}X_{i_1,i_3}\cdots X_{i_1,i_{j-1}}Z_{i_1,i_j} \in B_{i_1}^*Z_{i_1}$, noting that

$$\begin{aligned} X_{i_1,i_2}X_{i_1,i_3}\cdots X_{i_1,i_{j-1}}Z_{i_1,i_j} &= X_{i_1}X_{i_2}X_{i_1}^{-1}\cdot X_{i_1}X_{i_3}X_{i_1}^{-1}\cdots X_{i_1}X_{i_{j-1}}X_{i_1}^{-1}\cdot X_{i_1}X_{i_j} \\ &= X_{i_1}X_{i_2}\cdots X_{i_j} \\ &= A_iA_{i+1}\cdots A_j \end{aligned} \tag{20}$$

Since there are an even number of matrices with determinant -1, this argument can be applied iteratively so that if $I \in X^{-1}\langle G\rangle$, then $I \in X^{-1}A^*(B_i^*Z_iA^*)^*$ as required.

To finish the argument, assume that $I \in X^{-1}A^*(B_i^*Z_iA^*)^*$. We want to prove that this implies $I \in X^{-1}\langle G\rangle$.

Matrices in $B_i^*Z_i$ necessarily have the form of Equation (20). Therefore any product of $X^{-1}A^*(B_i^*Z_iA^*)^*$ can be written in terms of X^{-1} and matrices from G.

We earlier assumed that $\det(X) = 1$. If $\det(X) = -1$, then we can define $X_{0,j} = X^{-1}X_jX$ and $Z_{0,i} = X^{-1}X_i$ for $j \notin J$ and $i \in J$ so that $\det(X_{0,j}) = \det(Z_{0,i}) = 1$. Let $B_0 = \{X_{0,j} | j \notin J\}$ and $Z_0 = \{Z_{0,i} | i \in J\}$ then consider regular subset $B_0^+Z_0A^*(B_i^*Z_iA^*)^*$ for $i, j \in J$. Since now all matrices are over $\mathrm{SL}_2(\mathbb{Z})$, we can apply Theorem 53 to determine if the identity matrix belongs to this rational subset in NP. □

Next we consider the non-freeness problem for regular expressions over $\mathrm{SL}_2(\mathbb{Z})$.

Theorem 55. *The non-freeness problem is* **NP***-complete for finitely generated semigroups in* $\mathrm{SL}_2(\mathbb{Z})$.

Proof. By Corollary 50 the problem of determining whether $I \in \phi(R(a_1, \ldots, a_n))$ is in **NP**, where $R(a_1, \ldots, a_n)$ is an arbitrary regular expression in $\mathrm{PSL}_2(\mathbb{Z})$. We first reduce the non-freeness problem in $\mathrm{PSL}_2(\mathbb{Z})$ into the identity problem.

Let $M = \{m_1, m_2, \ldots, m_n\} \subseteq \mathrm{PSL}_2(\mathbb{Z})$ be a finite set generating a semigroup $\langle M\rangle_{\mathrm{sg}}$. This semigroup is non-free if and only if there exist two different factorizations

$$A \cdot X \cdot B = C \cdot Y \cdot D, \tag{21}$$

where $A, B, C, D \in M$ and $X, Y \in \langle M \rangle_{\text{sg}}$ so that $A \neq C$ and $B \neq D$.

Equation (21) is equivalent to $AXBD^{-1}Y^{-1}C^{-1} = I$, hence the identity element belongs to the language of the regular expression $AM^*BD^{-1}(M^{-1})^*C^{-1}$, where $M^{-1} = \{m^{-1} \mid m \in M\}$. Since there are only $n^2(n-1)^2$ such expressions with $A \neq C$ and $B \neq D$, we can nondeterministically find a witness (if one exists) for the identity for each in polynomial time.

To prove the result for $\mathrm{SL}_2(\mathbb{Z})$, we use the same technique as in Theorem 53, intersecting the regular language corresponding to $AM^*BD^{-1}(M^{-1})^*C^{-1}$ with a regular language L which differentiates between the positive and negative identity matrix so that all potential solutions correpond to the positive identity (I). As before, regular language L is constructed by considering all matrix products over $M \cup M^{-1}$ equivalent to the identity matrix, modulus 3.

The non-freeness problem was shown to be **NP**-hard in [30] (even over $\mathrm{SL}_2(\mathbb{Z})$), and therefore it is **NP**-complete. □

Our final result in this section shows that the problem of determining if a given set of matrices over $\mathrm{SL}_2(\mathbb{Z})$ generates a group is also **NP**-complete by using Theorem 52.

Theorem 56. *Given a finite set of matrices $G \subseteq \mathrm{SL}_2(\mathbb{Z})$, determining if G generates a group is* **NP***-complete.*

Proof. Let $G = \{G_1, G_2, \ldots, G_K\} \subseteq \mathrm{SL}_2(\mathbb{Z})$. We must determine, for each $G_i \in G$, whether $G_i^{-1} \in G^*$.

By Theorem 52, determining if the identity matrix belongs to an arbitrary regular expression is **NP**-complete. We therefore solve the following series of questions sequentially:

1. Does I belong to G_1G^*?
2. Does I belong to G_2G^*?
3. ...
4. Does I belong to G_kG^*?

If I belongs to G_iG^*, then clearly G_i has a multiplicative inverse in G^*. If I does not belong to some regular expression G_iG^*, then G_i does not have a multiplicative inverse in G. Determining if $I \in G_iG^*$ can be done in **NP**, and thus performing this procedure k times means this 'group problem' can be done in **NP**. The hardness result follows since even determining if any element of G has an inverse is known to be **NP**-hard [7]. □

6. Conclusion

The main contribution of this article is a new type of **NP** algorithm applied to low-dimensional matrix problems. In particular, we derive the exact complexity of the identity problem in $\mathrm{GL}_2(\mathbb{Z})$, showing that it is **NP**-complete. Moreover, the **NP** algorithm for checking whether the identity matrix belongs to an arbitrary regular expression is important as many closely related problems for 2×2 matrices can be reduced to it, including the membership and non-freeness problems. In general, many problems for 2×2 matrices are still open. For example, even the decidability of the freeness problem for 2×2 matrices over natural numbers still remains a long-standing open problem [10]. Recently progress was made to show the decidability of the vector reachability problem for $\mathrm{SL}_2(\mathbb{Z})$, see [46] and the decidability of the membership problem for two cases non-singular integer 2×2 matrices see [47] and $\mathrm{GL}_2(\mathbb{Z})$ extended by singular matrices [48]. However, the exact complexity of these problems is not yet known.

The proposed techniques presented in this paper may be helpful for designing more efficient algorithms for similar problems. One of the natural steps would be to extend the **NP** algorithm if possible for the mortality problem for 2×2 matrices whose determinants assume the values 0 or ± 1. This problem was shown to be **NP**-hard in [3] and decidability of this problem was shown in [40] based on the decidability for $\mathrm{SL}_2(\mathbb{Z})$ from [17]. The complexity of matrix problems over rational or complex numbers may be even higher. Very little is still known not only about the complexity, but also about the decidability of these problems.

In the seminal paper of Paterson in 1970 [43], an injective morphism from pairs of words into 3×3 integral matrices was used to prove the undecidability of the mortality problem, and later led to many undecidability results of matrix problems in dimension three. In [29] it was shown that there is no embedding from pairs of words into 3×3 integral matrices with determinant one, i.e., into $\mathrm{SL}_3(\mathbb{Z})$, which provides strong evidence that computational problems in $\mathrm{SL}_3(\mathbb{Z})$ may be decidable, as all known undecidability techniques for low-dimensional matrices are based on encoding of Turing machine computations via Post's Correspondence Problem (PCP), which cannot be applied in $\mathrm{SL}_3(\mathbb{Z})$ following the results of [29]. In the case of the PCP encoding, matrix products extended by right multiplication correspond to a Turing machine simulation, and the only known proof alternatives rely on recursively enumerable sets and Hilbert's Tenth Problem, but provide undecidability for

matrix equations of very high dimensions.

As the decidability status of the *identity problem* in dimension three is still a long standing open problem, it would be plausible to consider the problem in $\mathrm{SL}_3(\mathbb{Z})$, which also has a symbolic representation.

Comparing to the relatively simple representation of $\mathrm{SL}_2(\mathbb{Z}) = \langle S, T \mid S^4 = I_2, (ST)^6 = I_2 \rangle$, where $S = \left(\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right)$ and $T = \left(\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right)$, the case of $\mathrm{SL}_3(\mathbb{Z}) = \langle X, Y, Z \mid X^3 = Y^3 = Z^2 = (XZ)^3 = (YZ)^3 = (X^{-1}ZXY)^2 = (Y^{-1}ZYX)^2 = (XY)^6 = I_3 \rangle$, where

$$X = \left(\begin{smallmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{smallmatrix}\right),\ Y = \left(\begin{smallmatrix} 1 & 0 & 1 \\ 0 & -1 & -1 \\ 0 & 1 & 0 \end{smallmatrix}\right) \text{ and } Z = \left(\begin{smallmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ -1 & -1 & -1 \end{smallmatrix}\right),$$

appears more challenging, containing both non-commutative and partially commutative elements.

Acknowledgements

We sincerely thank the reviewers for their careful checking of this manuscript and helpful suggestions and corrections.

References

[1] L. Babai, R. Beals, and A. Seress. Polynomial-time theory of matrix groups. In *Proceedings of the Forty-first Annual ACM Symposium on Theory of Computing*, STOC '09, pages 55–64, New York, NY, USA, 2009. ACM.

[2] B. Balle and M. Mohri. Learning weighted automata. In *Algebraic Informatics - 6th International Conference, CAI 2015, Stuttgart, Germany, September 1-4, 2015. Proceedings*, pages 1–21, 2015.

[3] P. C. Bell, M. Hirvensalo, and I. Potapov. Mortality for 2×2 matrices is NP-hard. In *Mathematical Foundations of Computer Science 2012: 37th International Symposium, MFCS 2012*, pages 148–159, 2012.

[4] P. C. Bell, M. Hirvensalo, and I. Potapov. The identity problem for matrix semigroups in $\mathrm{SL}_2(\mathbb{Z})$ is NP-complete. In *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'17)*, pages 187–206, 2017.

[5] P. C. Bell and I. Potapov. Reachability problems in quaternion matrix and rotation semigroups. *Information and Computation*, 206(11):1353–1361, 2008.

[6] P. C. Bell and I. 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(6):963–978, 2010.

[7] P. C. Bell and I. Potapov. On the computational complexity of matrix semigroup problems. *Fundamenta Informaticae*, 116:1–13, 2012.

[8] V. Blondel, E. Jeandel, P. Koiran, and N. Portier. Decidable and undecidable problems about quantum automata. *SIAM Journal on Computing*, 34:6:1464–1473, 2005.

[9] V. Blondel and J. Tsitsiklis. The boundedness of all products of a pair of matrices is undecidable. *Systems and Control Letters, Elsevier*, 41:2:135–140, 2000.

[10] V. D. Blondel, J. Cassaigne, and J. Karhumäki. Freeness of multiplicative matrix semigroups. In V. D. Blondel and A. Megretski, editors, *Unsolved Problems in Mathematical Systems and Control Theory*, http://press.princeton.edu/math/blondel/solutions.html, 2004. Princeton University Press.

[11] J.-Y. Cai, W. H. Fuchs, D. Kozen, and Z. Liu. Efficient average-case algorithms for the modular group. In *The 35th Annual Symposium on Foundations of Computer Science (FOCS)*, 1994.

[12] J. Cassaigne, T. Harju, and J. Karhumäki. On the undecidability of freeness of matrix semigroups. *International Journal of Algebra and Computation*, 9(3-4):295–305, 1999.

[13] J. Cassaigne and F. Nicolas. On the decidability of semigroup freeness. *RAIRO - Theoretical Informatics and Applications*, 46(3):355–399, 2012.

[14] H. Caswell. *Matrix population models: Construction, analysis, and interpretation. Second edition.* Sunderland, Massachusetts, USA: Sinauer Associates, 2001.

[15] A. Cayley. A memoir on the theory of matrices. *Philosophical Transactions of the Royal Society of London*, 148:17–37, 1858.

[16] F. Chamizo. Non-euclidean visibility problems. In *Proceedings of the Indian Academy of Sciences - Mathematical Sciences*, volume 116:2, pages 147–160, 2006.

[17] C. Choffrut and J. Karhumäki. Some decision problems on integer matrices. *RAIRO - Theoretical Informatics and Applications*, 39:125–131, 2005.

[18] V. Chonev, J. Ouaknine, and J. Worrell. On the complexity of the orbit problem. *Journal of the ACM*, 63(3):1–18, 2016.

[19] M. G. del Moral, I. Martín, J. M. Peña, and A. Restuccia. $\mathrm{SL}(2,\mathbb{Z})$ symmetries, supermembranes and symplectic torus bundles. *Journal of High Energy Physics 9*, pages 1–12, 2011.

[20] J. Ding, A. Miasnikov, and A. Ushakov. A linear attack on a key exchange protocol using extensions of matrix semigroups. *IACR Cryptology ePrint Archive*, 2015:18, 2015.

[21] J. Elstrodt, F. Grunewald, and J. Mennicke. Arithmetic applications of the hyperbolic lattice point theorem. *Proceedings of the London Mathematical Society*, 57(3):239–283, 1988.

[22] J. Esparza, A. Finkel, and R. Mayr. On the verification of broadcast protocols. In *Proceedings of the 14th Annual IEEE Symposium on Logic in Computer Science*, LICS '99, pages 352–, Washington, DC, USA, 1999. IEEE Computer Society.

[23] K. Etessami and M. Yannakakis. On the complexity of Nash equilibria and other fixed points (extended abstract). In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, 2007.

[24] E. Galby, J. Ouaknine, and J. Worrell. On matrix powering in low dimensions. In *32nd International Symposium on Theoretical Aspects of Computer Science (STACS'15)*, pages 329–340, 2015.

[25] Y. Gurevich and P. Schupp. Membership problem for the modular group. *SIAM Journal on Computing*, 37(2):425–459, 2007.

[26] V. Halava. *Integer Weighted Finite Automata, Matrices, and Formal Power Series over Laurent Polynomials*, pages 81–88. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[27] R. Jungers. *The joint spectral radius: theory and applications*. Lecture Notes in Control and Information Sciences. Springer-Verlag, 2009.

[28] S. K. K, A. Murawski, J. Ouaknine, B. Wachter, and J. Worrell. On the complexity of equivalence and minimisation for Q-weighted automata. *Logical Methods in Computer Science*, Volume 9, Issue 1, Mar. 2013.

[29] S.-K. Ko, R. Niskanen, and I. Potapov. On the identity problem for the special linear group and the Heisenberg group. In *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018*, volume 132:1-132:15, 2018.

[30] S.-K. Ko and I. Potapov. Matrix semigroup freeness problems in $\mathrm{SL}(2,\mathbb{Z})$. In *43rd International Conference on Current Trends in Theory and Practice of Computer Science, (SOFSEM)*, 2017.

[31] R. Lipton. Mathematical embarrassments. Blog entry, Dec. 2009.

[32] M. Lohrey. *The compressed word problem for groups*. Springer Briefs in Mathematics. Springer, 2014.

[33] M. Lohrey. Subgroup Membership in GL(2,Z). In M. Bläser and B. Monmege, editors, *38th International Symposium on Theoretical Aspects of Computer Science (STACS 2021)*, volume 187 of *(LIPIcs)*, pages 51:1–51:17, Dagstuhl, Germany, 2021. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[34] M. Lohrey and G. Zetzsche. Knapsack in graph groups. *Theory of Computing Systems*, 62(1):192–246, Jan 2018.

[35] D. Mackenzie. A new twist in knot theory. *What's Happening in the Mathematical Sciences*, 7, 2009.

[36] A. G. Myasnikov, A. Nikolaev, and A. Ushakov. Knapsack problems in groups. *Mathematics of Computation*, 84(292):987–1016, 2015.

[37] A. G. Myasnikov and A. Weiß. Tc^0 circuits for algorithmic problems in nilpotent groups. In *42nd International Symposium on Mathematical Foundations of Computer Science, MFCS 2017, August 21-25, 2017 - Aalborg, Denmark*, pages 23:1–23:14, 2017.

[38] M. H. A. Newman. On theories with a combinatorial definition of "equivalence". *Ann. Math.*, 43:223–243, 1942.

[39] T. Noll. Musical intervals and special linear transformations. *Journal of Mathematics in Music*, 1(2):121–137, 2007.

[40] C. Nuccio and E. Rodaro. Mortality problem for 2x2 integer matrices. In *Theory and Practice of Computer Science: 34th Conference on Current Trends in Theory and Practice of Computer Science, (SOFSEM)*, pages 400–405, 2008.

[41] J. Ouaknine, J. S. Pinto, and J. Worrell. On termination of integer linear loops. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2015)*, pages 957–969, 2015.

[42] J. Ouaknine, A. Pouly, J. Sousa-Pinto, and J. Worrell. Solvability of matrix-exponential equations. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science (LICS'16)*, pages 798–806, 2016.

[43] M. S. Paterson. Unsolvability in 3×3 matrices. *Studies in Applied Mathematics*, 49(1):105–107, 1970.

[44] L. Polterovich and Z. Rudnick. Stable mixing for cat maps and quasimorphisms of the modular group. *Ergodic Theory Dynam. Systems*, 24(2):609–619, 2004.

[45] I. Potapov. Composition problems for braids. In *proceedings of 33nd International Conference on Foundations of Software Technology and Theoretical Computer Science, LIPIcs. Leibniz Int. Proc. Inform.*, volume 24, pages 175–187, 2013.

[46] I. Potapov and P. Semukhin. Vector reachability problem in $\mathrm{SL}(2,\mathbb{Z})$. In *41st International Symposium on Mathematical Foundations of Computer Science, (MFCS)*, volume 58, pages 1–14, 2016.

[47] I. Potapov and P. Semukhin. Membership problem for 2×2 integer matrices. In *Proceedings of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2017.

[48] I. Potapov and P. Semukhin. Membership problem in $\mathrm{GL}_2(\mathbb{Z})$ extended by singular matrices. In *42nd International Symposium on Mathematical Foundations of Computer Science, MFCS 2017*, volume 44, pages 1–13, 2017.

[49] R. A. Rankin. *Modular forms and functions*. Cambridge University Press, 1977.

[50] T. Tao. Open question: effective Skolem-Mahler-Lech theorem. Blog entry, May 2007.

[51] E. Witten. $\mathrm{SL}(2,\mathbb{Z})$ action on three-dimensional conformal field theories with abelian symmetry. *From fields to strings: circumnavigating theoretical physics*, 2:1173–1200, 2005.

[52] D. Zagier. *Elliptic modular forms and their applications*. The 1-2-3 of Modular Forms : Lectures at a Summer School in Nordfjordeid, Norway. Springer-Verlag, 2008.