The Identity Problem for Matrix Semigroups in $\mathrm{SL}_2(\mathbb{Z})$ is **NP**-complete

Paul C. Bell * Mika Hirvensalo† Igor Potapov‡

Abstract

In this paper, 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 $\mathrm{PSL}_2(\mathbb{Z})$ and thus the Special Linear group $\mathrm{SL}_2(\mathbb{Z})$ is solvable in **NP**. 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 or free semigroup is also decidable in **NP**. The previous algorithm for these problems, shown in 2005 by Choffrut and Karhumäki, 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. Our algorithm is based on various new techniques that allow us to operate with compressed word representations of matrices without explicit expansions. When combined with the known **NP**-hard lower bound, this proves that the membership problem for the identity problem, the group problem and the freeness problem in $\mathrm{SL}_2(\mathbb{Z})$ are **NP**-complete.

1 Introduction

The Projective Special Linear group $\mathrm{PSL}_2(\mathbb{Z})$ and Special Linear group $\mathrm{SL}_2(\mathbb{Z})$ play a central role in many branches of mathematics, see [6]. $\mathrm{SL}_2(\mathbb{Z})$, which is the most basic example of a discrete non-abelian group, consists of all integer 2×2 matrices, with determinant one[1] and $\mathrm{PSL}_2(\mathbb{Z})$ is 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 [25, 9, 12], dynamical systems [19], Lorenz/modular knots [15], braid groups [20], particle physics, high energy physics [24], M/string theories [11], ray tracing analysis, music theory [17] and it plays a central role for the development of efficient solutions of 2×2 matrix problems [21].

The structural properties of $\mathrm{SL}_2(\mathbb{Z})$ and $\mathrm{PSL}_2(\mathbb{Z})$ have been studied extensively in various textbooks and research papers. In this work, we reveal new structural properties and 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 $S \subseteq \mathrm{SL}_2(\mathbb{Z})$ the membership problem for the identity matrix in $\mathrm{SL}_2(\mathbb{Z})$ (whether or not the identity matrix belongs to S), 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 [10] to **NP**.

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 $M_{i_1} \cdot M_{i_2} \cdots M_{i_k} = M$, i.e. determine whether matrix M belongs to the semigroup generated by F.*

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* [6][2]. Later,

*Department of Computer Science, Loughborough University. Email: P.Bell@lboro.ac.uk

†Department of Mathematics and Statistics, University of Turku. Email: mikhirve@utu.fi. Supported by Väisälä Foundation

‡Department of Computer Science, University of Liverpool. Email: potapov@liverpool.ac.uk. This research was supported by EPSRC grant EP/M00077X/1.

[1] 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 $\mathbb{R}$.

[2] 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

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 [13]. 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 is still open.

In this paper, we consider the *Identity Problem*, which is the membership problem for the identity matrix in the semigroup. 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 $\mathrm{SL}_2(\mathbb{Z})$. On the other hand, the identity problem is tightly connected with another two fundamental decision problems on matrices: the *Group Problem*: "is a given matrix semigroup a group?"[3] and the *Freeness Problem*: "does every matrix have a unique factorisation over F, i.e. is F a code?"

One of the main results in this paper states that the identity problem for matrix semigroups generated by any finite set of matrices from $\mathrm{SL}_2(\mathbb{Z})$ is **NP**-complete. The previous algorithm for this problem, shown in 2005 by Choffrut and Karhumäki [10], 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 [4] that there are instances of the identity problem where the number of generator occurrences needed to produce the identity matrix is exponential in the description size of the semigroup generator.

Rather surprisingly, in this context, we show here that the identity problem for $\mathrm{SL}_2(\mathbb{Z})$ can be solved in **NP**. Our new algorithm is based on various new techniques that allow us to operate with compressed word representations of matrices without explicit exponential expansions. The identity problem in $\mathrm{SL}_2(\mathbb{Z})$ 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**. In our results, we rely on the fact that we can find a reasonable characterization of complex long paths within our derived compressed graph called *Alternating Forms*, which have many useful properties that can be exploited and help us to greatly simplify the analysis. When combined with the **NP**-hard lower bound shown in [4], this proves that the membership problem for the identity problem and group problem in $\mathrm{SL}_2(\mathbb{Z})$ 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 the identity 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 [14] so the non-freeness problem in $\mathrm{SL}_2(\mathbb{Z})$ is also **NP**-complete.

Our main results in this paper are therefore to show that the three problems (identity, group and freeness problems) can be solved in **NP** over $\mathrm{SL}_2(\mathbb{Z})$ and they are therefore **NP**-complete following existing hardness results for these problems. The decidability status of the identity problem and the group problem in higher dimensions was unknown for a long time and was only recently shown to be undecidable for integer matrices starting from dimension four [3], see also the solution to Problem 10.3 in [5]. The freeness problem is known to be undecidable for 3×3 matrices over the integers [7]. Although some partial results for the freeness problem in matrices of dimension two are known, a complete picture is far from clear [8]. 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 proved 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 [7], 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 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 [2] or more precisely in the case of

to the submonoid membership problems by simply including the inverses in the generating set of matrices.

[3]The identity and group problems are bilaterally reducible [10].

diagonal matrices over quaternions, which are simply dual quaternions.

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 $\langle\Sigma\rangle_{\mathrm{sg}} = \{\sigma_1\sigma_2\ldots\sigma_n \mid n \geq 1, \sigma_i \in \Sigma\}$. 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 $\sigma_i\cdots\sigma_j$, with the assumption that $1 \leq i \leq j \leq k$.

If Σ is included in an algebraic structure containing also the *inverse* of each $\sigma \in \Sigma$ satisfying $\sigma\sigma^{-1} = \sigma^{-1}\sigma = 1$, we may define the *group* generated by Σ as $\langle\Sigma\rangle_{\mathrm{gr}} = \{\sigma_1^{a_1}\sigma_2^{a_2}\ldots\sigma_n^{a_n} \mid n \geq 0, \sigma_i \in \Sigma, a_i \in \{-1, 1\}\}$. 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 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

$$\Pi = (v_1, l_1, v_2)(v_2, l_2, v_3)\ldots(v_k, l_k, v_{k+1}) \in E^* \tag{2.1}$$

Using notation $e_i = (v_i, l_i, v_{i+i})$, the above presentation can be written as $\Pi = e_1e_2\ldots e_k \in E^*$. The *length* of path (2.1) is k and its *label* is defined as concatenation $l_1l_2\ldots l_k \in L^*$. It is important to notice that if the label set contains the empty word ε, then it is treated in the concatenation as usual, i.e. $l_1\varepsilon l_2 = l_1l_2$. 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 (2.1) is defined as $e_ie_{i+1}\ldots e_j$, where $1 \leq i \leq j \leq k$. The subpath is *proper* if $i > 1$ or $j < k$.

Definition 2.1. A dual edge cycle *is a path of the form* $e_1e_2E^*e_1e_2$, *where* e_1, $e_2 \in E$.

Remark 2.1. *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 2.1. 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 2.2. *The reduction function* $\mathrm{red} : E^* \to E^*$ *is defined to remove dual edge cycles: If* $\Pi = \Pi_1\Pi_2\Pi_3$, *where* $\Pi_2 = e_1e_2E^*e_1e_2$ *is a dual edge cycle and* $\Pi_1, \Pi_3 \in E^*$, *then* $\mathrm{red}(\Pi) = \Pi_1e_1e_2\Pi_3$. *As usual,*

red^* *is defined as the transitive closure of* red. *Thus,* $\text{red}^*(\Pi)$ *contains each consecutive pair of edges of the graph at most once. Such a path is called a* reduced path.

Example. Consider set of edges $\{e_1, e_2, e_3, e_4\} \subseteq V \times L \times V$ and path

$$\Pi = e_1e_2e_3e_1e_3e_2e_3e_1e_2$$

Now, Π is a dual edge cycle, since e_1e_2 is a prefix and suffix. But it is not reduced, since $e_3e_1e_3e_2e_3e_1$ and $e_2e_3e_1e_3e_2e_3$ are proper subpaths and dual edge cycles.

Notice that $\text{red}(\Pi) \in \{e_1e_2, e_1e_2e_3e_1e_2\}$ – recall that red is nondeterministic.

3 The Structure of $\text{PSL}_2(\mathbb{Z})$

3.1 Generating $\text{SL}_2(\mathbb{Z})$. Group $\text{SL}_2(\mathbb{Z})$ is very important in number theory, and its structure has been studied extensively in various textbooks (see [23], 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 $\text{SL}_2(\mathbb{Z})$ 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 $\text{SL}_2(\mathbb{Z})$, and the following lemma provides even a quantitative version of this fact.

LEMMA 3.1. $\text{SL}_2(\mathbb{Z}) = \langle S, T \rangle_{gr}$. *Furthermore, any matrix*

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

can be represented as

$$A = S^{\alpha}T^{q_1}S^3T^{q_2} \cdot \ldots S^3T^{q_k}S^{\beta}T^{q_{k+1}}, \tag{3.2}$$

so that $\alpha, \beta \in \{0, 1, 2, 3\}$, $q_i \in \mathbb{Z}$, $k \leq 1 + \log_2 M$, *and* $|q_i| \leq \frac{3}{2}M^{1+\log_2 \frac{5}{2}}$, *where* $M = \max\{|a|, |b|, |c|, |d|\}$. *Representation (3.2) 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:

$$\begin{aligned} S\begin{pmatrix} a & b \\ c & d \end{pmatrix} &= \begin{pmatrix} -c & -d \\ a & b \end{pmatrix}, \\ T^n\begin{pmatrix} a & b \\ c & d \end{pmatrix} &= \begin{pmatrix} a+nc & b+nd \\ c & d \end{pmatrix}. \end{aligned} \tag{3.3}$$

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

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

If $c \neq 0$ but $a = 0$, then according to (3.3) $SA \in \{T^{-d}, S^2T^d\}$, implying that $A \in \{S^3T^{-d}, ST^d\}$ (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 (3.3), $|(SA)_{11}| > |(SA)_{21}|$. 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 $a_i = q_ic_i + r_i$, where $|r_i| \leq \frac{1}{2}|c_i|$.

We define now

$$A_{i+1} = ST^{-q_i}A_i = \begin{pmatrix} -c_i & -d_i \\ r_i & b_i - q_ic_i \end{pmatrix}, \tag{3.4}$$

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

Sequence $A_1, A_2, \ldots$ 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 (3.4) 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 $-c_id_{i+1} + r_id_i = 1$, and if $i < k$, then $c_i \neq 0$ and therefore

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

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 (3.2) is a polynomial-time algorithm, given the bit representation size of A as the input size. □

REMARK 3.1. *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 (3.2). It is noteworthy that when substituting $T = -SR$ in (3.2), one can use $R^3 = -I$ and $S^2 = -I$ to get a representation

$$A = (-1)^\gamma R^{n_0}SR^{n_1}S \cdot \ldots \cdot R^{n_{l-1}}SR^{n_l}, \tag{3.5}$$

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

REMARK 3.2. *It can be shown that the representation (3.5) for a given matrix $A \in \mathrm{SL}_2(\mathbb{Z})$ is unique, but it should be noticed that representation (3.5) can be exponentially long in the representation size of matrix A in bits, as the example*

$$\begin{pmatrix} 1 & m \\ 0 & 1 \end{pmatrix} = T^m = (-SR)^m = (-1)^m \underbrace{SR\ldots SR}_{m \text{ times}} \tag{3.6}$$

demonstrates. The representation size of the matrix T^m is proportional to $\log_2 m$, but the representation (3.6) contains $2m$ matrices.

It is structurally simpler to present (3.5) ignoring the sign. For that purpose, we introduce two structurally important elements of $\mathrm{PSL}_2(\mathbb{Z})$.

DEFINITION 3.1. *Let $s = S\{\pm I\}$ and $r = R\{\pm I\}$ be the projections of S and R in $\mathrm{PSL}_2(\mathbb{Z})$.*

REMARK 3.3. *Since $S^2 = R^3 = -I$ in $\mathrm{SL}_2(\mathbb{Z})$, it is clear that $s^2 = r^3 = \varepsilon$ in $\mathrm{PSL}_2(\mathbb{Z})$.*

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

LEMMA 3.2. *[23] -* $\mathrm{PSL}_2(\mathbb{Z}) = \mathrm{SL}_2(\mathbb{Z})/\{\pm I\}$ *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*

(3.7)
$$r^{n_0} s r^{n_1} s \cdots r^{n_{p-1}} s r^{n_p} = r^{m_0} s r^{m_1} s \cdots r^{m_{q-1}} s r^{m_q},$$

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 3.2 see [23].

DEFINITION 3.2. *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 3.4. *By Lemma 3.2 every element of* $\mathrm{PSL}_2(\mathbb{Z})$ *admits a unique reduced ground level representation. However, it follows directly from Remark 3.2 that the unique representation of the projection of* T^m *in* $\mathrm{PSL}_2(\mathbb{Z})$ *is*

$$t^m = \underbrace{rs \dots rs}_{m \text{ times}}, \tag{3.8}$$

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

Despite Definition 3.2, 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 (3.7) 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 (3.2) and taking the projections $S \to s$ and $R \to r$ we learn that

$$a = s^{\alpha}(sr)^{q_1} s (sr)^{q_2} \cdot \ldots s (sr)^{q_k} s^{\beta} (sr)^{q_{k+1}}, \tag{3.9}$$

where the estimation for the exponents and k are the same as in Lemma 3.1. 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 3.3. *Any element* $a = \{A, -A\}$ *of* $\mathrm{PSL}_2(\mathbb{Z})$ *admits a unique succinct representation of the form*

(3.10)
$$a = r^{\alpha}(sr)^{n_1}(sr^2)^{n_2}(sr)^{n_3}(sr^2)^{n_4} \dots (sr)^{n_{l-1}}(sr^2)^{n_l} s^{\beta},$$

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 3.1.*

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 (3.8), we would have a representation

$$t^m = (rs) \uparrow (m_1 \dots m_k), \tag{3.11}$$

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 (3.11) 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 (3.10) 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 3.3. *Following Gurevich and Schupp [13] we define the following syllables:*

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

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 $R_i R_{-i} = \varepsilon$). As $r = R_1$, the following lemma is trivial but its claim is worth emphasizing.

LEMMA 3.4. *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 3.3 is that they can be used to write the compact representations (3.10) in a structural way, and also provide a natural way to handle the potential cancellations of elements.

REMARK 3.5. *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 3.4. *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* $w_1w \equiv w_2w$*.*

Even though the syllabic representation is not unique, the following result is proven in [13]. The representation size estimate follows directly from Lemma 3.3.

LEMMA 3.5. *Each element* $a \in \mathrm{PSL}_2(\mathbb{Z})$ *admits a unique representation of the form*

$$a = s^{\alpha}R_{n_1}sR_{n_2}sR_{n_3}s\ldots sR_{n_l}s^{\beta}, \tag{3.12}$$

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 (3.12) is polynomial in the representation size of* a*.*

Because of the uniqueness, we call representation (3.12) a *canonical syllabic* representation of $\mathrm{PSL}_2(\mathbb{Z})$ elements.

LEMMA 3.6. *The syllables satisfy the following relations*

- $ss \equiv \varepsilon$
- $R_aR_{-a} \equiv \varepsilon$*, and*
- $R_{a+b} \equiv R_asR_b$*, if* $ab > 0$*.*
- $R_1R_1 \equiv R_{-1}$*, 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 3.6. *It can be seen that the above relations give rise to other ones. For example, if* $ab < 0$ *and* $|b| < |a|$*, then* $R_aR_b \equiv R_{a+b}sR_{-b}R_b \equiv R_{a+b}s$*, and a symmetric version is obtained when* $|a| < |b|$*. To summarize:*

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

REMARK 3.7. *The above rules in Lemma 3.6 and Remark 3.6 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

$$\begin{aligned} R_1R_2^tR_1 &\equiv R_{-1}R_{-1}R_2^tR_1 \\ &\equiv R_{-1}sR_1R_2^{t-1}R_1 \equiv \ldots \\ &\equiv (R_{-1}s)(R_{-1}s)\cdots(R_{-1}s)R_1R_1 \\ &\equiv (R_{-1}s)^tR_{-1} \equiv R_{-(t+1)} \end{aligned}$$

From this, we can easily derive that $R_2^tR_1 \equiv R_{-1}R_{-(t+1)} \equiv R_1sR_{-t}$ *and* $R_1R_2^t \equiv R_{-t}sR_1$*. Similarly we can see that* $R_{-1}R_{-2}^tR_{-1} \equiv R_{t+1}$ *and derive analogous consequences.*

We conclude this section by estimating the "reduction power" of the equivalences of Lemma 3.6 and Remark 3.6.

DEFINITION 3.5. *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 $w = w_1 \ldots w_n$ *is defined as* $|w|_{\langle r,s\rangle} = |w_1|_{\langle r,s\rangle} + \ldots + |w_n|_{\langle r,s\rangle}$*.*

DEFINITION 3.6. *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 3.7. *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 $R_{-a}R_{-b} = (r^2s)^{a-1}r^2(r^2s)^{b-1}r^2 = (r^2s)^{a-2}r^2sr^2r^2s(r^2s)^{b-2}r^2 = R_{-(a-1)}sR_1sR_{-(b-1)}$ (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_1R_{-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_aR_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:

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

2.3) In both cases R_aR_{-b} and $R_{-a}R_b$, a reduction clearly takes place: $R_aR_{-b} = (rs)^{a-1}r(r^2s)^{b-1}r^2 = R_{a-1}R_{-(b-1)}$ 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_aR_{-b}. □

Notice that according to Lemma 3.7, the canonical form of Lemma 3.5 is not reducible.

Definition 3.7. *We define the set of syllables* $\Omega = \{\varepsilon, s, s^\alpha R_{\pm1}s^\beta, s^\alpha R_{\pm2}s^\beta\}$, *where* α, $\beta \in \{0, 1\}$. *Intuitively, set* Ω *forms a "neighbourhood" of* ε.

Lemma 3.8. *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. $R_aR_{-a} \mapsto \varepsilon$
3. $R_aR_{-b} \mapsto R_{a-b}s$, *if* $ab > 0$ *and* $|b| < |a|$
4. $R_aR_{-b} \mapsto sR_{a-b}$, *if* $ab > 0$ *and* $|a| < |b|$
5. $R_{-1}R_{-1} \mapsto R_1$
6. $R_1 \mapsto R_{-1}R_{-1}$

Remark 3.8. *We do not introduce a rule* $R_1R_1 \to R_{-1}$, *even though the equivalence* $R_1R_1 \equiv R_{-1}$ *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* $R_aR_1R_b \equiv R_{a-1}R_{b-1}$.

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 [16] 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 $R_aR_1R_b \mapsto R_{a-1}R_{b-1}$ (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:

$$R_aR_1R_b \mapsto R_aR_{-1}R_{-1}R_b \mapsto R_{a-1}ssR_{b-1} \mapsto R_{a-1}R_{b-1}.$$

2.2) In this case, $R_{-a}R_{-b}$ contains a factor r^3 to be removed, and the resulting representation is $R_{-(a-1)}sR_1sR_{-(b-1)}$ (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

$$uR_{-a}R_{-b}v \mapsto uR_{-(a-1)}sR_1sR_{-(b-1)}v \mapsto^* \omega$$

is reducible to an element of Ω, then necessarily either $u \mapsto^* u_1sR_{a-1}$ or $v \mapsto^* R_{b-1}sv_1$. In the first case (the second is analogous), we can change the reduction order to have

$$\begin{aligned} uR_{-a}R_{-b}v &\mapsto^* u_1sR_{a-1}R_{-a}R_{-b}v \\ &\mapsto u_1ssR_1R_{-b}v \\ &\mapsto u_1R_1R_{-b}v, \end{aligned}$$

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 3.8 at least implicitly, but the following rules will form the backbone of the algorithm presented in Section 5.

Definition 3.8. (Reduction function ρ) *We call rules 1-4 of Lemma 3.8* regular *and define function ρ to represent them as follows:*

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

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

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 [16] (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 3.2 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 1. (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 4.1. *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 4.2. *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 4.1. *Letting $M = \max_{1\le i,j\le 2} |a_{ij}|$, as in Lemma 3.1, it is obvious that $|A|_{bit} = \Theta(\log M)$.*

Definition 4.3. *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 1 where the representation of the identity element requires a high number of generator occurrences.

Example. 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 3.6 and Lemma 3.6 we see that $ab = sR_nR_{-1}s = sR_{n-1}ss = sR_{n-1}$, $ab^2 = sR_{n-2}$, and by induction $ab^n = 1$. 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 $sR_nR_{-n}s = \varepsilon$.

Here the parsing tree of the identity element is exponentially deep in the semigroup description size. Another example where the shortest identity is exponentially long, but the parsing tree only polynomially deep was given in [4].

4.2 Automaton for Recognizing the Identity. The decision procedure presented in [10] is based on Lemma 3.2, 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 a context-free language $L = L_1 \cap L_2$ 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).

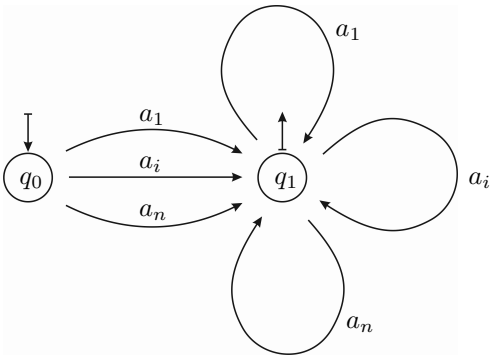

Figure 1: Automaton recognizing $\{a_1, a_2, \ldots, a_n\}^+$. Initial and final states are indicated with short arrows.

REMARK 4.2. *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

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

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

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

and all the new nodes are assumed distinct. The replacements result in a larger automaton shown in Figure 2. As described above, the $\langle r, s\rangle$-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 3.4), 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,

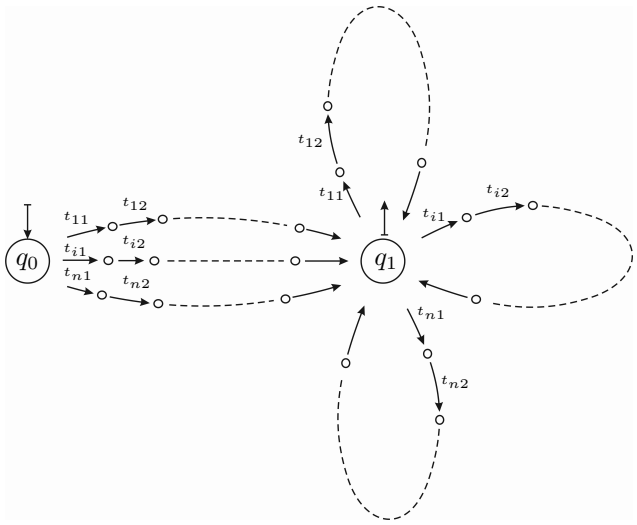

Figure 2: $\langle r, s\rangle$-automaton recognizing $\{a_1, \ldots, a_n\}^+$.

we introduce new transitions according to Lemma 3.8: 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. When applying this procedure to Example 4.1 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 4.3. *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 1.

The strategy will avoid exponential growth in the edge set mentioned in Example 4.3. 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 3.5 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.
- 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 reductions shown in Remark 3.7.

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 3.7 that Ω-syllables are those "close" to ε.

As shown in Remark 3.7, 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 3.7, $(R_2)^t R_1 \equiv R_{-1}R_{-(t+1)}$, 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.

Similarly, 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 5.1. *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 5.2. (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 5.1. *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 absolute weight may differ, and* $R_1 \equiv R_{-1}R_{-1}$*, which shows that even the 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 5.3. ***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 5.4. (**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 $R_{10}R_{-5}sR_{-5}$ has that property. 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.

LEMMA 5.1. *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 $\mathrm{red}(\Pi) = \Pi'$*, 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 $|M|_{\text{bit}}$ by Lemma 3.3. 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 5.1.1, 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 $|M|_{\text{bit}}$ since $|\Omega|$ is a constant.

5.2.1 Introduction of R_- and R_+-edges. Consider a path $\Pi = (q_i, R_2, q_i)(q_i, R_1, q_j)$ in G_M, which implies that the path: $(q_i, R_2, q_i)^t(q_i, R_2, q_j)$ 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 $q \xrightarrow{R_{-1}} q_j$.

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

However, the cases 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, where the states are distinct does 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 $R_1R_2^t$, $R_{-2}^t R_{-1}$, 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. $E_{q_i,q_j} := E_{q_i,q_j} \cup \{w'\}$ (if it does not already exist).

We now describe three ways of showing that there is indeed such a path $q_i \to q_j$.

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 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 as we now show.

LEMMA 5.2. *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 3.8. 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 Lemma. This property can immediately be deduced from the definition of R-syllables in Definition 3.3 and is simply a rewriting of equivalent ground level representations of a syllable.

Case 1) - $w = W_1 R_{x_1} R_{y_1} W_2$ where $W_1, W_2 \in \Sigma^*$ and $x_1, y_1 > 0$. Both W_1 and W_2 cannot equal ε since $k \geq 3$. Let $w' = \rho^*(W_1)R_{x_1}R_{y_1}\rho^*(W_2) = W_1' R_{x_1} R_{y_1} W_2'$. Thus W_1', W_2' do not contain syllabic subwords ss, $R_{z_1} s R_{z_2}$ or $R_{z_1} R_{-z_2}$, where $z_1 z_2 > 0$. Since $w \in \Phi$, then $W_1', W_2' \notin \Omega$.

We now show that $\rho^*(W_1' R_{x_1})$ is of the form $W_1'' R_{x_1'}$ or $W_1'' R_{-x_1'} s$, where $W'' \in \Sigma^*$ and $x_1' > 0$. Consider the suffix of $\rho^*(W_1')$. We have the following cases (where $X, X' \in \Sigma^*$ are arbitrary words and $x_2, x_3 > 0$):

1. If $\rho^*(W_1') = X R_{x_2} s$, then we can apply rule $\rho(R_{x_2} s R_{x_1}) = R_{x_1+x_2}$ and recursively consider the suffix of $\rho(X)$ with $R_{x_1+x_2}$.

2. If $\rho^*(W_1') = X R_{-x_2} s$ or $\rho^*(W_1') = X R_{x_2}$, then there is no cancelation between $\rho^*(W_1')$ and R_{x_1}.

3. If $\rho^*(W_1') = X R_{-x_2}$ with $|x_2| > |x_1|$, then $\rho^*(W_1' R_{x_1}) = X R_{-x_2} R_{x_1} = X R_{-x_2+x_1} s$.

4. If $\rho^*(W_1') = X R_{-x_2}$ with $|x_2| < |x_1|$, then we see that $\rho^*(W_1' R_{x_1}) = X s R_{x_1-x_2}$. X cannot have suffix $R_{-x_3} s$, since $\rho(R_{-x_3} s R_{-x_2}) = R_{-(x_2+x_3)}$. If $X = X' R_{x_3} s$, then $\rho^*(W_1' R_{x_1}) = X' \rho(R_{x_3} s s R_{x_1-x_2}) = X' R_{x_3} R_{x_1-x_2}$, which does not cancel and ends with a positive R syllable. If $X = X' R_{-x_3}$, then $\rho^*(W_1' R_{x_1}) = X' \rho(R_{-x_3} s R_{x_1-x_2}) = X' R_{-x_3} s R_{x_1-x_2}$, again ending with a positive R syllable since there is no cancelation.

5. If $\rho^*(W_1') = XR_{-x_2}$ with $|x_2| = |x_1|$, then this gives a contradiction, since $\rho(R_{-x_2}R_{x_1}) = \varepsilon$ but then w is not an Ω-minimal word.

The above analysis therefore shows that the *suffix* of $\rho^*(W_1'R_{x_1})$ is $R_{x_1'}$ or $sR_{-x_1'}$ for $x_1' > 0$. A similar analysis shows that the *prefix* of $\rho^*(R_{x_2}W_2)$ is of the form $R_{y_1'}$ or $R_{-y_1'}s$ for $y_1' > 0$. In fact, we can see that $x_1', y_1' > 2$, 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: $XR_{x_1'} \cdot R_{y_1'}X'$, $XR_{-x_1'}s \cdot R_{x_2'}X'$, $XR_{x_1'}\cdot sR_{-y_1'}X'$ or $XR_{-x_1'}s\cdot sR_{-y_1'}X' \equiv XR_{-x_1'}R_{-y_1'}X'$. 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. This leaves us with the case that w contains two consecutive negative weight R syllables.

Case 2) - $w = W_1R_{-x_1}R_{-y_1}W_2$ where $W_1, W_2 \in \Sigma^*$ and $x_1, y_1 > 0$. Let $w' = \rho^*(W_1)R_{-x_1}R_{-y_1}\rho^*(W_2) = W_1'R_{-x_1}R_{-y_1}W_2'$. Thus W_1', W_2' do not contain syllabic subwords ss, $R_{z_1}sR_{z_2}$ or $R_{z_1}R_{-z_2}$, where $z_1z_2 > 0$. Since $w \in \Phi$, then $W_1', W_2' \notin \Omega$. We now show that $\rho^*(W_1'R_{-x_1})$ is of the form $W_1''R_{-x_1'}$, $W_1''R_{x_1'}s$ or $W_1''sR_1$, where $W'' \in \Sigma^*$ and $x_1' > 0$.

Consider the suffix of $\rho^*(W_1')$. We have the following cases (where $X, X' \in \Sigma^*$ are arbitrary words and $x_2, x_3 > 0$):

1. If $\rho^*(W_1') = XR_{-x_2}s$, then $\rho^*(W_1'R_{-x_1}) = XR_{-(x_2+x_1)}$ for which there is no cancelation and the suffix is $R_{-(x_2+x_1)}$.

2. If $\rho^*(W_1') = XR_{x_2}s$, then there is no cancelation and $\rho^*(W_1'R_{-x_1}) = XR_{x_2}sR_{-x_1}$.

3. If $\rho^*(W_1') = XR_{x_2}$ with $|x_2| > |x_1|$, then $\rho^*(W_1'R_{-x_1}) = XR_{x_2-x_1}s$, with $x_2 - x_1 > 2$, otherwise $R_{x_2}R_{-x_1} \equiv w' \in \Omega$ which is a contradiction.

4. If $\rho^*(W_1') = XR_{x_2}$ with $|x_2| < |x_1|$, then $\rho^*(W_1'R_{-x_1}) = XsR_{-x_1+x_2}$ with $-x_1 + x_2 < -2$, otherwise $sR_{-x_1+x_2} \in \Omega$ which is a contradiction. X cannot have suffix R_{-x_3} or $R_{x_3}s$ since this suffix would cancel with R_{x_2}. Thus the suffix of X must be either R_{x_3} or $R_{-x_3}s$. If it is R_{x_3}, then $\rho^*(W_1'R_{-x_1}) = X'R_{x_3}sR_{-x_1+x_2}$ which does not cancel any further. If the suffix of X is $R_{-x_3}s$, then $\rho^*(W_1'R_{-x_1}) = X'R_{-x_3}ssR_{-x_1+x_2} = X'R_{-x_3}R_{-x_1+x_2}$ and we again have two consecutive negatively weighted R syllables. Since $-x_1 + x_2 < -2$, then $R_{-x_3}R_{-x_1+x_2}$ has suffix $sR_{-x_1+x_2+1}$, where $-x_1 + x_2 + 1 < -1$.

5. If $\rho^*(W_1') = XR_{x_2}$ with $|x_2| = |x_1|$, then this is a contradiction, since then $\rho^*(R_{x_2}R_{-x_1}) \equiv \varepsilon \in \Omega$, but $w \in \Phi$.

Thus, the *suffix* of $\rho^*(W_1'R_{-x_1})$ is in $\{R_{-x_1'}, R_{x_1'}s; x_1 > 1\}$. A similar analysis shows that the *prefix* of $\rho^*(R_{-y_1}W_2')$ is in $\{R_{-y_1'}, sR_{y_1'}; y_1' > 1\}$. We see that $XR_{-x_1'} \cdot R_{-y_1'}X' \equiv XR_{-(x_1'-1)}sR_1sR_{-(y_1'-1)}X'$ since $x_1', y_1' > 1$, and there is no further reduction. For $XR_{-x_1'} \cdot sR_{y_1'}X'$ there is no further cancelation. Similarly for $XR_{x_1'}s \cdot R_{-y_1'}X'$. Finally, $XR_{x_1'}s \cdot sR_{y_1'}X' \equiv XR_{x_1'}R_{y_1'}X'$ which has already been considered and cannot reduce to an Ω element.

Case 3) - $w = W_1R_{x_1}sR_{-y_1}W_2$ 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 $\{X'R_{x_1'}, X'R_{-x_1'}s\}$ 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 $|x_1|, |y_1| > 1$. Now we consider what happens when these elements are combined as $\rho^*(W_1R_{x_1}sR_{-y_1}W_2)$ for these four cases.

In the case $\rho^*(W_1R_{x_1}) \equiv X'R_{x_1'}$ 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 $\rho^*(W_1R_{x_1}) \equiv X'R_{x_1'}$ and $\rho^*(R_{y_1}W_2) \equiv sR_{y_1'}Y'$, then $\rho(X'R_{x_1'}\cdot s\cdot sR_{y_1'}Y') \equiv X'R_{x_1'}\cdot R_{y_1'}Y'$ with $x_1', y_1' > 1$ has already been considered above.

In case three, $\rho^*(W_1R_{x_1}) \equiv X'R_{-x_1'}s$ 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 $x_1', y_1' > 1$ has already been considered above. In case four, $\rho^*(W_1R_{x_1}) \equiv X'R_{-x_1'}s$ and $\rho^*(R_{y_1}W_2) \equiv sR_{y_1'}Y'$, thus $\rho(X'R_{-x_1'}s \cdot s \cdot sR_{y_1'}Y') \equiv X'R_{-x_1'}s\cdot R_{y_1'}Y'$ 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 5.3. *Given a word $w \in \Phi$, with $|w| > 3$, then $w \equiv w'$, for some $w' \in \Omega$ iff $0 \le |\mathrm{wgt}(w)| \le 2$, and 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 $w = w_1w_2\cdots w_k \in \Phi$. Note that the action of ρ, defined in Definition 3.8 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 3.8 and Lemma 5.2, 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-theoretical 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 5.4. *Let* $1 \leq x, c_1, \ldots, c_{k_1}, d_1, \ldots, d_{k_2} < T$ *such that there exist integers* $\alpha_1, \ldots, \alpha_{k_1}, \beta_1, \ldots, \beta_{k_2} > 0$ *where:*

$$x + \sum_{j=1}^{k_1} \alpha_j c_j - \sum_{j=1}^{k_2} \beta_j d_j = 0. \tag{5.13}$$

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*

$$x + \sum_{j=1}^{k'_1} \alpha'_j c'_j - \sum_{j=1}^{k'_2} \beta'_j d'_j = 0. \tag{5.14}$$

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

$$x + \sum_{j=1}^{k'_1} \alpha_j c'_j - \sum_{j=1}^{k'_2} \beta_j d'_j = 0 \tag{5.15}$$

has a some solution $(\alpha_1, \ldots, \beta_1, \ldots)$ over the integers. To simplify the notations, remove the primes and rewrite (5.15) as

$$x + \sum_{j=1}^{k_1} \alpha_j c_j - \sum_{j=1}^{k_2} \beta_j d_j = 0. \tag{5.16}$$

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

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

which shows that for any $n \in \mathbb{Z}$, $\alpha_j \mapsto \alpha_j + n k_2 \frac{B}{c_j}$ (and similarly for β_j) yields another solution to (5.16). 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 $w_1, w_2 \in \Phi$ 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 5.5. *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* $w \equiv w'$ *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 3.8 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 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 then 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 5.6. *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 3.8. The only remaining cases involve syllables R_- and R_+.

If $w_1 w_2 = R_+ \lambda_1$, $w_1 w_2 = \lambda_2 R_-$, $w_1 w_2 = R_- \lambda_2$, $w_1 w_2 = \lambda_1 R_+$, $w_1 w_2 = R_- R_+$ or $w_1 w_2 = R_+ R_-$, where $\lambda_1 \in \Gamma_-$ and $\lambda_2 \in \Gamma_+$: then the following edges all belong to E_{q_i, q_j}: $\{\varepsilon, R_2 s, R_1 s, s R_1, s R_2\}$.

To see this, let us consider the first rule $w_1 w_2 = R_+ \lambda_1$, 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:

$$R_{x-2} R_{-x} \equiv s R_{-2}; \quad R_{x-1} R_{-x} \equiv s R_{-1}; \quad R_x R_{-x} \equiv \varepsilon;$$
$$R_{x+1} R_{-x} \equiv R_1 s; \quad R_{x+2} R_{-x} \equiv R_2 s.$$

Such a path can be found and verified in time polynomial in $|M|_{\mathrm{bit}}$. 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 5.2. By Lemma 5.1, we know that $|w| \leq |E|^2$ since $\mathrm{red}(w) = w$. 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 3.8.

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 5.2, and that $|\mathrm{wgt}(\Pi)| \leq 2$, with the weight determining which element of Ω we reduce to, up to factors of 's' by Lemma 5.3. We shall show a way to find an equivalent path $\Pi_2 = (q_i, w_2, q_j)$, such that $w_2 \in \mathcal{AF}$, $\mathrm{wgt}(w_2) = \mathrm{wgt}(w)$ and Π_2 contains no more than a polynomial (in terms of $|M|_{\mathrm{bit}}$) 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 $(q_i, R_1, q_j)(q_j, R_2, q_j)$ 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 $R_1 R_2^t$, $R_2^t R_1$, $R_{-1} R_{-2}^t$ and $R_{-2}^t R_{-1}$.

3a) Π contains both positive and negative weight dual edge cycles. I.e. $\Pi = X_1 C_1 Y_1 = X_2 C_2 Y_2$ 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

$x = \text{wgt}(\text{red}^*(\Pi))$ and assume without loss of generality that $x > 0$. Note that x is not unique, since red is nondeterministic. By Lemma 5.4, 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 5.4 proceeds by removing unneccesary 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 $\text{red}^*(\Pi)$ 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 (5.14) 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 5.1, 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 5.1.

c) Verify that $x + \sum_{j=1}^{k_1} \alpha_j c_j - \sum_{j=1}^{k_2} \beta_j d_j = \text{wgt}(w')$, where $|\text{wgt}(w')| \leq 2$ 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 5.5 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 $\Pi_1 = \text{red}^*(\Pi)$. 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 $|M|_{\text{bit}}$. 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 5.1 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 5.1, 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 $\text{In} : E^* \times Q \to \mathbb{N}$ be a function such that $\text{In}(\Pi', q')$ denotes the number of edges of $\Pi' \in E^*$ going to vertex $q' \in Q$, plus 1 if Π' starts at vertex q'. Similarly, $\text{Out} : E^* \times Q \to \mathbb{N}$ is a function such that $\text{Out}(\Pi', q')$ 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 $\text{In}(\Pi', q_1) = 1$, $\text{Out}(\Pi', q_1) = 1$, $\text{In}(\Pi', q_2) = 2$, $\text{Out}(\Pi', q_2) = 2$ and $\text{In}(\Pi', q_3) = 2$, $\text{Out}(\Pi', q_3) = 2$. These functions can be defined formally for $\Pi' = \pi_1'\pi_2' \cdots \pi_{k'}' \in E^*$ as follows:

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

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 $q' \in Q$:

$$\text{In}(\Pi', q') = \text{Out}(\Pi', q'). \tag{5.17}$$

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

$$\text{In}(\Pi, q_1) = \text{In}(\Pi_i, q_1). \tag{5.18}$$

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

$$\text{Out}(\Pi, q_1) > \text{Out}(\Pi_i, q_1). \tag{5.19}$$

Combining Invariant 5.17, Equality 5.18 and Inequality 5.19, we obtain the following contradiction:

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

To recap then, given $\Pi \in \Phi$ such that $|\Pi| > 2$ and Π contains only dual edge cycles of positive sign, we first define $\Pi_1 = \text{red}^*(\Pi)$, which we showed has a polynomial length (polynomial in terms of $|M|_{\text{bit}}$). 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 $|M|_{\text{bit}}$. 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 5.1, this implies that $\Pi_z \in \mathcal{AF}$. Since, by definition, $\Pi_z(\tau) = \Pi(\tau)$ for all $\tau \in E^2$, then $\text{wgt}(\Pi_z) = \text{wgt}(\Pi)$. 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 5.5, since $|\Pi| > 3$, $\Pi \in \Phi$, $\text{wgt}(\Pi) = \text{wgt}(\Pi_z)$ and $\Pi_z \in \mathcal{AF}$, then $\Pi_z \equiv \Pi$ as required. □

We conclude the aforementioned procedure in a theorem:

THEOREM 5.1. *The identity problem over* $\text{PSL}_2(\mathbb{Z})$ *is in* **NP**.

Recall from Remark 4.3 that the described procedure is not limited to the daisy graph and works for any regular expression $R(a_1, \ldots, a_n)$.

COROLLARY 5.1. *The problem of determining whether the identity matrix is in an arbitrary regular expression* $R(a_1, \ldots, a_n) \subseteq \text{PSL}_2(\mathbb{Z})$ *is in* **NP**.

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

COROLLARY 5.2. *The identity problem over* $\text{SL}_2(\mathbb{Z})$ *is in* **NP**.

THEOREM 5.2. *The problem of determining whether a matrix* M *is in an arbitrary regular expression* $R(a_1, \ldots, a_n) \subseteq \text{PSL}_2(\mathbb{Z})$ *is in* **NP**.

Proof. The decidability of the problem was shown in [10] as it can be reduced to the identity problem for a regular expression in $\text{PSL}_2(\mathbb{Z})$, i.e. whether $I \in M^{-1} \cdot R(a_1, \ldots, a_n)$. 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 5.1 as deciding whether $I \in M^{-1} \cdot R(a_1, \ldots, a_n)$ is in NP. □

THEOREM 5.3. *The non-freeness problem for finitely generated semigroups in* $\text{PSL}_2(\mathbb{Z})$ *and* $\text{SL}_2(\mathbb{Z})$ *is* **NP**-*complete.*

Proof. By Corollary 5.1 the problem of determining whether $I \in R(a_1, \ldots, a_n)$ is in **NP**, where $R(a_1, \ldots, a_n)$ is an arbitrary regular expression in $\text{PSL}_2(\mathbb{Z})$. We will reduce the non-freeness problem in $\text{PSL}_2(\mathbb{Z})$ into the identity problem.

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

$$A \cdot X \cdot B = C \cdot Y \cdot D, \tag{5.20}$$

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 (5.20) 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.

The claim for $\text{SL}_2(\mathbb{Z})$ is evident, since if a finitely generated semigroup $\langle M \rangle_{\text{sg}} \subseteq \text{SL}_2(\mathbb{Z})$ is non-free, so clearly is its projection in $\text{PSL}_2(\mathbb{Z})$. Thus the non-freeness problem belongs to class **NP**. The problem was shown to be **NP**-hard in [14], and therefore it is **NP**-complete. □

6 Conclusion

The main contribution of this article is an entirely new type of **NP** algorithm applied to low-dimensional matrix problems. In particular, we derive the exact complexity of the identity problem in $\text{SL}_2(\mathbb{Z})$, 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 [5]. Recently progress was made to show the decidability of the vector reachability problem for $\text{SL}_2(\mathbb{Z})$, see [21] and the decidability of the membership problem for non-singular integer 2×2 matrices see [22]. 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 [1] and decidability of this problem was shown in [18] based on the decidability for $\mathrm{SL}_2(\mathbb{Z})$ from [10]. 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.

References

[1] 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.

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

[3] 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.

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

[5] 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.

[6] 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.

[7] 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.

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

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

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

[11] 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.

[12] J. Elstrodt, F. Grunewald, and J. Mennicke. Arithmetic applications of the hyperbolic lattice point theorem. *Proc. London Math. Soc.*, 57(3):239–283, 1988.

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

[14] 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.

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

[16] M. H. A. Newman. On theories with a combinatorial definition of equivalence. *Annals of Mathematics*, 43:223–243, 1942.

[17] T. Noll. Musical intervals and special linear transformations. *Journal of Mathematics and Music: Mathematical and Computational Approaches to Music Theory, Analysis, Composition and Performance*, 1(2):121–137, 2007.

[18] 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.

[19] L. Polterovich and Z. Rudnick. Stable mixing for cat maps and quasi-morphisms of the modular group. *Ergodic Theory and Dynamical Systems*, 24(2):609–619, 2004.

[20] I. Potapov. Composition problems for braids. In *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.

[21] 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.

[22] 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.

[23] R. A. Rankin. *Modular Forms and Functions*. Cambridge University Press, 1977.

[24] 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.

[25] 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.