On the average-case bit complexity of the Word Problem for groups of matrices over $\mathbb{Z}$

Frédérique Bassino, bassino@lipn.fr
Université Sorbonne Paris Nord, LIPN, CNRS UMR 7030, F-93430 Villetaneuse, France

Cyril Nicaud, cyril.nicaud@u-pem.fr
LIGM, Univ Gustave Eiffel, CNRS, ESIEE Paris, F-77454, Marne-la-Vallée, France

Pascal Weil, pascal.weil@labri.fr
CNRS, ReLaX, IRL 2000, Siruseri, India
CNRS, Univ. Sorbonne Paris Nord, LIPN, UMR 7030, F-93430 Villetaneuse, France

June 4, 2025

Abstract

We show that the Word Problem in finitely generated subgroups of $\mathsf{GL}_d(\mathbb{Z})$ can be solved in linear average-case complexity. This is done under the bit-complexity model, which accounts for the fact that large integers are handled, and under the assumption that the input words are chosen uniformly at random among the words of a given length.

1 Introduction

Let G be a group and let Σ be a finite non-empty subset of G. Let also $\tilde{\Sigma} = \Sigma \cup \Sigma^{-1}$. The Word Problem for G relative to Σ is the following: given a word w over alphabet $\tilde{\Sigma}$, decide whether the value of w in G is trivial. This problem was introduced by Dehn in the early 20th century [Deh11], and is considered one of the fundamental problems in algorithmic and combinatorial group theory. It is known that the Word Problem is not decidable in general (that is: there exist finitely generated, and even finitely presented groups with undecidable Word Problem), see Novikov [Nov55] and Boone [Boo59]. However, it is known to be decidable in many important classes of groups, for instance in automatic groups [ECH+92] (including finite, free, hyperbolic or braid groups), finitely presented residually finite groups (Simmons [Sim73]), 1-relator groups (see [MKS66, Theorem 4.14] and also Lyndon and Schupp [LS01]), etc.

Here we consider this problem in the very natural context of subgroups of $\mathsf{GL}_d(\mathbb{Z})$, the group of invertible matrices with integer coefficients. We set the following notation: Σ is a finite non-empty subset of $\mathsf{GL}_d(\mathbb{Z})$, $\tilde{\Sigma} = \Sigma \cup \Sigma^{-1}$, H is the subgroup of $\mathsf{GL}_d(\mathbb{Z})$ generated by Σ and $\mathsf{M}\colon \tilde{\Sigma}^* \to \mathsf{GL}_d(\mathbb{Z})$ is the natural morphism, which maps the element $x \in \tilde{\Sigma}$ to the corresponding matrix in $\mathsf{GL}_d(\mathbb{Z})$. That is, if w is a word on alphabet $\tilde{\Sigma}$, then $\mathsf{M}(w)$ is the value of w in $\mathsf{GL}_d(\mathbb{Z})$. The Word Problem in H (relative to Σ), written WP_Σ, is obviously decidable: $\mathsf{M}(w)$ can be computed and compared to the identity matrix Id.

In the following, we consider d and Σ as fixed. The coefficients of the matrices in H can be very large (the coefficients of $\mathsf{M}(w)$ may grow exponentially in the length $|w|$ of w), and we therefore evaluate the complexity of algorithms in the so-called bit-complexity model: integers are identified with their binary expansion and arithmetic operations require more than constant time. In particular, we use the recent result of Harvey and van der Hoeven [HvdH21] which states that multiplying two integers p and q can be done in time $\mathcal{O}(L \log L)$ where $L = \max(\log p, \log q)$ is (roughly) the maximal length of the binary expansions of p and q.

The naive algorithm to solve the Word Problem in H is the following: given $w = a_1 \cdots a_n$, with each $a_i \in \tilde{\Sigma}$, let $w_0 = \mathsf{Id}$ (the identity matrix) and, for each $1 \leq i \leq n$, $w_i = w_{i-1}a_i$. Then $\mathsf{M}(w) = w_n$ and one can decide by inspection whether $\mathsf{M}(w) = \mathsf{Id}$. This algorithm has quadratic worst-case complexity $\mathcal{O}(n^2)$. A direct application of the classical divide-and-conquer strategy lowers this complexity to $\mathcal{O}(n \log^2 n)$, as noted already by Olshanskii and Shpilrain [OS25, Proposition 2] (see Proposition 9 below).

We note that this divide-and-conquer technique yields an $\mathcal{O}(n)$ worst-case complexity when the matrices in Σ are upper-triangular (Proposition 11), a sharpening of Olshanskii's and Shpilrain's result [OS25, Theorem 2]. As noted by these authors, this implies a linear worst-case complexity for the Word Problem in finitely generated torsion-free nilpotent groups.

Our main result (see Theorem 13 below) is the following.

Main Theorem. *WP_Σ has linear time average-case complexity in the bitcost model of computation when inputs follow the uniform distribution on length n words over alphabet $\tilde{\Sigma}$.*

This result was already known in the case of polycyclic groups, which are representable as subgroups of $\mathsf{GL}_d(\mathbb{Z})$ [Weh80]. Indeed, Olshanskii and Shpilrain established a linear average-case complexity of the Word Problem for these groups [OS25, Theorem 3], and in fact, for subgroups $H \leq \mathsf{GL}_d(\mathbb{Z})$ with a non-trivial virtually abelian factor, such as the virtually solvable linear groups [OS25, Remark 4], which goes beyond polycyclic groups.

In contrast, our result holds for all finitely generated subgroups of $\mathsf{GL}_d(\mathbb{Z})$, polycyclic or not, and its proof is seemingly very different from that of Olshanskii and Shpilrain. The two proofs have however an interesting common point, see Remark 1 below.

Several ideas come into play in the proof of our main result. The first is to use modulo computations: if q is an integer, one first solves the Word Problem in the subgroup H_q of $\mathsf{GL}_d(\mathbb{Z}/q\mathbb{Z})$ generated by Σ. That can be done by using the standard (divide-and-conquer) algorithm in $\mathsf{GL}_d(\mathbb{Z}/q\mathbb{Z})$: that is, by computing $\mathsf{M}(w)_q$ (the matrix $\mathsf{M}(w)$ modulo q) and verifying whether it is equal to Id. Here we benefit from the fact that, at each step of the computation, the entries of matrices are in the interval $[0, q-1]$, thus lowering the worst-case complexity. If q is constant, this is faster than computing $\mathsf{M}(w)$ in $\mathsf{GL}_d(\mathbb{Z})$ as the length of w tends to infinity; and if $\mathsf{M}(w)_q$ is not the identity, then neither is $\mathsf{M}(w)$. Thus, if $\mathsf{M}(w)_q \neq \mathsf{Id}$ with high probability, we first compute $\mathsf{M}(w)_q$ and, if it is the identity, we compute $\mathsf{M}(w)$ using the standard $\mathcal{O}(n \log^2 n)$ algorithm.

This however is not sufficient to get our result. Indeed, as we will see, for any value of q, the probability that $\mathsf{M}(w)_q = h$ (for any $h \in \mathsf{GL}_d(\mathbb{Z}/q\mathbb{Z})$) tends to a positive value, namely $\frac{\alpha}{|H_q|}$, where $\alpha = 1$ or $\alpha = 2$ (depending on Σ and q). Thus, with probability tending to $\frac{\alpha}{|H_q|}$, we need to call the divide-and-conquer algorithm, yielding an $\mathcal{O}(n \log^2 n)$ average-case complexity.

The next idea is to choose the modulus q as a function $q(n)$ of the length n of the input word w. The advantage of this algorithm is that, for each input word w, we compute $\mathsf{M}(w)$ modulo a single integer $q(n)$. The challenge is to identify an appropriate function $q(n)$.

More precisely, we analyze the computation of $\mathsf{M}(w)_{q(n)}$ in terms of trajectories in a Markov chain $\mathfrak{M}$, which is a technical variant of the natural Markov chain based on the Cayley graph of the subgroup $H_{q(n)}$ of $\mathsf{GL}_d(\mathbb{Z}/q(n)\mathbb{Z})$ generated by Σ. We want $q(n)$ to be small enough so that $\mathsf{M}(w)_{q(n)}$ is computed rapidly, and we want the Markov chain $\mathfrak{M}$ to have low mixing time (that is, we want it to converge rapidly towards its stationary distribution) and a large dispersion (so that $H_{q(n)}$ should have large cardinality and the probability that $\mathsf{M}(w)_{q(n)} = \mathsf{Id}$ be small). These are seemingly contradictory requirements, and the technical work of the proof consists in identifying a function $q(n)$ with these properties.

Concretely, we choose $q(n)$ to be an increasingly long, but slowly increasing, product of distinct primes, so that we test $\mathsf{M}(w)$ modulo all these primes in a single computation (namely, the modulo $q(n)$ computation), yet $q(n)$ grows sufficiently slowly, see Definition 12.

Remark 1. The proof of [OS25, Theorem 3] also uses computation in a quotient of H, though not a mod q quotient: the projection onto an abelian factor, whose existence is postulated (and which should be known for any implementation of their algorithm).

It is important to note that our algorithm makes no assumption on Σ and on the properties of the subgroup H it generates. The matrices in Σ may be triangular or not; the subgroup H may be finite (and it is well known that the Word Problem is simpler in this case) or infinite; it may be nilpotent, polycyclic or virtually solvable; it may have exponential or polynomial growth (as a consequence of the Tits alternative): we do not need to identify in which situation we are, and we always use the same algorithm.

The paper is organized as follows. In Section 2, we lay out the notation and definitions of the Word Problem and its modulo variant and we give a very quick discussion of models of computation and of the definitions of worst-case and average-case complexity. In Section 2.3, we remind the readers of the precise complexity of computing with integers, and in Section 2.4, we recall the fundamental definitions and results on Markov chains and their convergence to a stationary distribution, inasmuch as they will be needed in this paper.

In Section 3, we describe and analyze the standard divide-and-conquer algorithm to compute $\mathsf{M}(w)$ (Section 3.1), and we briefly discuss its application in the case of triangular matrices (Section 3.2). Our Algorithm $\mathsf{QuickWP}$, which solves the Word Problem in linear average-case complexity, is given in Section 3.3. This includes defining the function $q(n)$ mentioned above. We also reduce the proof of our Main Theorem to a technical statement (Theorem 16) which states that, if H is infinite, then the probability that a word w of length n satisfies $\mathsf{M}(w)_{q(n)} = \mathsf{Id}$ is $\mathcal{O}(\log^{-2} n)$. Finally, the proof of Theorem 16 is given in Section 4.

2 Preliminaries and notation

If A is a matrix in $\mathsf{GL}_d(\mathbb{Z})$, we let $\|A\|_\infty = \max\{|A_{i,j}| \mid 1 \leq i, j \leq d\}$. If $m \geq 2$, A_m denotes the projection of A modulo m, a matrix with entries in $\mathbb{Z}/m\mathbb{Z}$.

If X is a set of matrices in $\mathsf{GL}_d(\mathbb{Z})$, the subgroup they generate is written $\langle X \rangle$. We also denote by X_m the set $\{A_m \mid A \in X\}$. It is interesting to note the following elementary fact.

Fact 2. Let $m \geq 2$. If m is not a prime, then $\mathbb{Z}/m\mathbb{Z}$ is not a field. However, the projection mod m of the group $\mathsf{GL}_d(\mathbb{Z})$ (or of any of its subgroups) is again a group.

2.1 The Word Problem and related algorithmic problems

Let Σ be a fixed, finite, non-empty set of matrices in $\mathsf{GL}_d(\mathbb{Z})$, let $\tilde{\Sigma} = \Sigma \cup \Sigma^{-1}$ and let $\tilde{\Sigma}^*$ be the set of all words on alphabet $\tilde{\Sigma}$ (i.e., finite sequences of elements of $\tilde{\Sigma}$). We denote by $\mathsf{M}\colon \tilde{\Sigma}^* \to \mathsf{GL}_d(\mathbb{Z})$ the natural (monoid) morphism, which maps each element of Σ to itself.

The *Word Problem* WP_Σ, is the following: given a word $w \in \tilde{\Sigma}$, decide whether $\mathsf{M}(w)$ is the identity matrix Id.

We also consider in this paper the closely related *Exact Computation Problem* EC_Σ: on input a word $w \in \tilde{\Sigma}^*$, EC_Σ computes the product $\mathsf{M}(w)$ of that sequence in $\mathsf{GL}_d(\mathbb{Z})$. We will also discuss the same problems *modulo* m, where $m \geq 2$ is an integer. More precisely, Problems $\mathsf{WP}_{\Sigma,m}$ (resp. $\mathsf{EC}_{\Sigma,m}$) takes a word $w \in \tilde{\Sigma}$ as input, and decides whether $\mathsf{M}(w)_m$ is the identity matrix (resp. computes $\mathsf{M}(w)_m$).

Remark 3. If an algorithm $\mathcal{S}$ solves EC_Σ (resp. $\mathsf{EC}_{\Sigma,m}$) on input w — that is, if we have computed $\mathsf{M}(w)$ (resp. $\mathsf{M}(w)_m$), — then a minor tweak solves WP_Σ (resp. $\mathsf{WP}_{\Sigma,m}$): it suffices to examine the d^2 entries of $\mathsf{M}(w)$ (resp. $\mathsf{M}(w)_m$), which is done in constant time (if m is a constant; in time $\mathcal{O}(\log m)$ otherwise).

Convention Throughout the paper, the integer d and the set Σ are fixed. We let $k = |\Sigma|$. We also assume, and this is no loss of generality, that Σ does not contain a matrix A and its inverse. In particular, it does not contain the identity matrix Id, and $|\tilde{\Sigma}| = 2|\Sigma|$.

2.2 About algorithms and complexity

To evaluate the complexity of an algorithm $\mathcal{S}$, we consider the function $S(w)$, where w is an input word, which measures the time (number of elementary operations) needed to run Algorithm $\mathcal{S}$ on input w. Observe that, even though Σ — and thus the coefficients of the matrices in $\tilde{\Sigma}$ — is fixed in our setting, the integers computed by the algorithms can be huge, and one cannot assume that arithmetic operations are performed in constant time. To account for this, we consider the *bit-cost* model of computation, where an integer n is encoded using roughly $\log n$ bits, see Section 2.3.

The worst-case complexity of $\mathcal{S}$ is the function on integers given by

$$S_{\mathsf{wc}}(n) = \max\{S(w) \mid |w| = n\},$$

and its average complexity $S_{\mathsf{ac}}(n)$ is the average value of $S(w)$, when w runs uniformly over inputs of length n. As is traditional, these functions are considered up to asymptotic equivalence, when n tends to infinity.

As mentioned earlier, our average-complexity results assume that input words are taken uniformly at random among words on $\tilde{\Sigma}$ of length n.

2.3 Computing with integers

The *length* (or *bit-size*) $\ell(n)$ of a integer n is the length of its binary expansion, namely $\ell(n) = \lceil \log(|n| + 1) \rceil + 1$ (all logarithms are in base 2), where the additional bit is used to encode the sign. As a result, the integers of length at most $\ell + 1$ have absolute value less than 2^ℓ. We freely use the following facts, all of them elementary — with the exception of Proposition 4 (ii), which is a deep result due to Harvey and van der Hoeven [HvdH21].

Proposition 4. *Let n, n' be integers and let L such that $\ell(n), \ell(n') \leq L$.*

(i) $\ell(nn') \leq \ell(n) + \ell(n') - 1$.

(ii) The product nn' is computed in time $\mathcal{O}(L \log L)$.

(iii) The product nn' is also computed (by the primary school multiplication algorithm) in time $\mathcal{O}(\ell(n)\ell(n'))$, which is interesting if $\ell(n)$ is small with respect to $\ell(n')$.

(iv) The sum of d integers of length at most L, has length at most $1 + \log d + L$ and is computed in time $\mathcal{O}(d \log d + dL)$.

(v) If A, A' are $d \times d$ matrices with entries of length at most L, the entries of the product AA' have length at most $1 + \log d + 2L$, and each is computed in time $\mathcal{O}(d \log d + dL \log L)$.

(vi) If the entries of A (resp. A') are of length at most L (resp. L'), and if L' is much smaller than L, the entries of the product AA' have length at most $1 + \log d + L + L'$, and each is computed in time $\mathcal{O}(d \log d + dLL')$.

Remark 5. Proposition 4 (vi) shows that, if w is a length n word on $\tilde{\Sigma}$, the entries of $\mathsf{M}(w)$ have length at most $(1 + L + \log d)\, n$, where L is the maximum length of the coefficients of the matrices in $\tilde{\Sigma}$: the length of the entries of $\mathsf{M}(w)$ grows at most linearly in n, and their value in $\mathbb{Z}$ at most exponentially.

In the sequel, we will also compute in $\mathbb{Z}/m\mathbb{Z}$ for some integer $m \geq 2$. We record the following result on the complexity of computing in this ring, where every element is represented by a non-negative integer at most equal to m. It follows directly from [vzGG13, Theorem 9.8 and Corollary 9.9], together with Proposition 4 (ii) due to [HvdH21].

Corollary 6. *Let $m \geq 2$. Every arithmetic operation in $\mathbb{Z}/m\mathbb{Z}$ is performed in $\mathcal{O}(\log m \, \log\log m)$.*

2.4 Basic results on probability distributions and Markov chains

For a general discussion of probability distributions and Markov chains, we refer readers to [LP17]. Here we fix some notation and state a few standard results on Markov chains, that will be used in the sequel.

If X is a set, a *probability distribution* (or *probability vector*) on X is a vector $(\mu(x))_{x\in X}$, where each $\mu(x)$ lies in the closed interval $[0,1]$ and $\sum_x \mu(x) = 1$. The *corresponding probability function* is given, for every subset Y of X, by $\mu(Y) = \sum_{y\in Y} \mu(y)$. The *uniform distribution* on X is the vector all of whose entries are $\frac{1}{|X|}$.

If μ and π are two probability distributions on the same finite set X, their *total variation distance* is

$$\|\pi - \mu\|_{\mathrm{Var}} = \sup_{A\subseteq X} |\pi(A) - \mu(A)| = \frac{1}{2}\sum_{x\in X} |\pi(x) - \mu(x)|. \tag{1}$$

The second equality is proven in [LP17, Proposition 4.2].

In this paper, a (finite) *Markov chain* $\mathfrak{M}$ consists in a directed graph with (finite) vertex set S (vertices are also called *states*), edge set a subset of $S \times S$, and, for each edge (called a *transition*) from state q to state q', of a value $M_{q,q'} \in (0,1)$, in such a way that, for each $q \in S$, $(M_{q,q'})_{q'\in S}$ is a probability vector. The matrix $M = (M_{q,q'})_{(q,q')\in S\times S}$ is called the *transition matrix* of $\mathfrak{M}$.

A path $T = (q_0, \ldots, q_n)$ in the underlying graph of $\mathfrak{M}$ is called a *trajectory*. A probability is assigned to T by $\mathfrak{M}$, namely the product $M_{q_0,q_1} M_{q_1,q_2} \ldots M_{q_{n-1},q_n}$. We note that the (q,q')-entry of the n-th power of the transition matrix M is the sum of the probabilities of the length n trajectories from q to q'.

The Markov chain $\mathfrak{M}$ is *symmetric* if its transition matrix is symmetric, that is, if $M_{q,q'} = M_{q',q}$ for all states q, q'. The chain $\mathfrak{M}$ is *irreducible* if its underlying graph is strongly connected, that is: for every $q, q' \in S$, the (q,q')-entry of some positive power of M is non-zero. The chain $\mathfrak{M}$ is said to be *aperiodic* if, for every $q \in S$ and for all n large enough, there exists a length n trajectory from q to q. Finally, the chain $\mathfrak{M}$ is *primitive* if it is both irreducible and aperiodic.

A probability distribution D on X is called *stationary* if $D\,M = D$, that is, if D is a left eigenvector for the eigenvalue 1.

A Markov chain $\mathfrak{M}$ with a stationary distribution π is said to be *reversible with respect to* π (or just *reversible* if the stationary distribution is unique) if $\pi(q)\,M(q,q') = \pi(q')\,M(q',q)$ for all states q, q'.

The following is a classical result on Markov chains.

Theorem 7. *Let $\mathfrak{M}$ be a primitive Markov chain. Then 1 is an eigenvalue of M and the other eigenvalues have modulus less than 1. The eigenspace corresponding to eigenvalue 1 has dimension 1 and $\mathfrak{M}$ has a unique stationary distribution π, which satisfies the following: if μ is any probability distribution on the state set of $\mathfrak{M}$, then $\lim \mu\, M^n = \pi$.*

If $\mathfrak{M}$ is primitive and symmetric, then the uniform distribution is its unique stationary distribution, all the eigenvalues are real, and $\mathfrak{M}$ is reversible.

3 Algorithms for WP_Σ and EC_Σ

With the aim of studying the average-case complexity of WP_Σ, we consider several algorithms solving this problem. We start with standard algorithms, including one with $\mathcal{O}(n\log^2 n)$ worst-case complexity (Section 3.1). In Section 3.3, we introduce a better algorithm with the announced linear average-case complexity for the uniform distribution on words of length n. We then state our main theorem, Theorem 13, and reduce its proof to a technical statement (Theorem 16). The proof of that statement is given in Section 4.

3.1 First algorithms

The naive algorithm to compute $\mathsf{M}(w)$ consists in reading the word w from left to right, one letter at a time, and performing the corresponding $n-1$ matrix multiplications — where $n = |w|$. The

right factor in each of these operations is a matrix in $\tilde{\Sigma}$, a constant set which is independent of n. A direct application of Proposition 4 (vi) then shows that the worst-case bit complexity of this algorithm is $\mathcal{O}(\ell^2 n^2)$, where ℓ is an upper bound of the bit-size of the coefficients of the elements of $\tilde{\Sigma}$. Since ℓ is fixed in our settings, this naive approach runs in $\mathcal{O}(n^2)$ time.

This quadratic upper bound can be significantly improved using a *divide and conquer* strategy.

Algorithm DC_Σ

Input : a sequence w of n elements of $\tilde{\Sigma}$
Output: $\mathsf{M}(w)$
1 **if** $n = 0$ *(resp.* $n = 1$*)* **then return** Id (resp. $\mathsf{M}(w)$)
2 $w_1 \leftarrow$ prefix of w of length $\lfloor n/2 \rfloor$
3 $w_2 \leftarrow$ suffix of w of length $\lceil n/2 \rceil$
4 **return** $\mathsf{DC}_\Sigma(w_1) \times \mathsf{DC}_\Sigma(w_2)$

Before we analyze the complexity of this simple algorithm, let us remind the reader of an instance of the celebrated *Master Theorem* (see, *e.g.*, [CLRS22, Theorem 4.1]), which we will use several times.

Proposition 8. *Let $C(n)$ and $f(n)$ be positive-valued non-decreasing functions on $\mathbb{N}$ satisfying the equation $C(n) = C(\lfloor \frac{n}{2} \rfloor) + C(\lceil \frac{n}{2} \rceil) + f(n)$ for $n \geq 2$.*

- *If $f(n) = \mathcal{O}(n^h)$ for some $0 \leq h < 1$, then $C(n) = \mathcal{O}(n)$.*
- *If $f(n) = \mathcal{O}(n \log^h n)$ for some $h \geq 0$, then $C(n) = \mathcal{O}(n \log^{h+1} n)$.*

Proposition 9. *Algorithm DC_Σ solves Problem EC_Σ, with worst-case complexity $\mathcal{O}(n \log^2 n)$. In addition, Problem WP_Σ can be solved with the same worst-case complexity.*

Moreover, if $H = \langle \Sigma \rangle$ is finite, then the complexity of DC_Σ is linear.

Proof. Since $w = w_1 w_2$, we have $\mathsf{M}(w) = \mathsf{M}(w_1)\,\mathsf{M}(w_2)$, so Algorithm DC_Σ solves Problem EC_Σ.

Proposition 4 (ii) (crucially using [HvdH21]) and Remark 5 show that the complexity $C(n)$ of this algorithm satisfies the equation

$$C(n) = C\left(\lfloor n/2 \rfloor\right) + C\left(\lceil n/2 \rceil\right) + \mathcal{O}(n \log n) \qquad \text{for } n \geq 2. \tag{2}$$

Proposition 8 then yields the fact that $C(n)$ is $\mathcal{O}(n \log^2 n)$. The statement on Problem WP_Σ follows from Remark 3.

If H is finite, then the coefficients of the matrices in H have bounded length and the complexity $C(n)$ now satisfies the equation

$$C(n) = C\left(\lfloor n/2 \rfloor\right) + C\left(\lceil n/2 \rceil\right) + \mathcal{O}(1) \qquad \text{for } n \geq 2. \tag{3}$$

Proposition 8 then yields the fact that $C(n)$ is $\mathcal{O}(n)$. □

The same algorithm can be run on matrices in $\mathbb{Z}/m\mathbb{Z}$, where $m \geq 2$ is any integer. Let $\mathsf{DC}_{\Sigma,m}$ be the algorithm with the same steps as Algorithm DC_Σ, where all arithmetic operations are performed in $\mathbb{Z}/m\mathbb{Z}$ instead of $\mathbb{Z}$. It is immediate that Algorithm $\mathsf{DC}_{\Sigma,m}$ solves Problem $\mathsf{EC}_{\Sigma,m}$, and hence also Problem $\mathsf{WP}_{\Sigma,m}$.

Remark 10. In fact, the same algorithm runs on matrices over any computable ring. Over $\mathbb{Q}$, it also yields an $\mathcal{O}(n \log^2 n)$ worst-case complexity since the addition and multiplication of rationals takes asymptotically the same time as the addition and multiplication of integers.

3.2 The special case of triangular matrices

We note that if A and A' are upper triangular matrices and if $1 \leq i \leq j \leq d$, then the (i,j)-entry of AA' is $\sum_{h=i}^{j} A_{i,h}\, A'_{h,j}$.

Now suppose that the matrices in Σ are upper-triangular, and hence so are their inverses. It is directly verified that, if $w \in \tilde{\Sigma}^*$ has length n and $1 \leq i \leq j \leq d$, then the (i,j)-entry of $\mathsf{M}(w)$ is bounded above by a polynomial in the variable n, with degree $j-i$. In particular, the length of the entries of $\mathsf{M}(w)$ is logarithmic.

Then Proposition 4 shows that if w and w' are words of length at most n, then the product $\mathsf{M}(w)\mathsf{M}(w')$ is computed in polylogarithmic time. Applying the Master Theorem (Proposition 8) directly yields a linear worst-case complexity. Since finitely generated torsion-free nilpotent groups can be represented by subgroups of upper-triangular matrices in $\mathsf{GL}_d(\mathbb{Z})$, we get the following statement, which improves on Olshanskii's and Shpilrain's results [OS25, Theorems 1 and 2].

Proposition 11. *If Σ consists only of upper (resp. lower) triangular matrices, then the worst-case complexity of WP_Σ is $\mathcal{O}(n)$.*

If G is a finitely generated torsion-free nilpotent group, then the Word Problem in G can be solved in linear worst-case complexity.

3.3 A linear average-case algorithm for the Word Problem

The key idea to get an algorithm solving WP_Σ with a better average-case complexity, is to compute $\mathsf{M}(w)_{q(n)}$, where n is the length of w and $q(n)$ is a function such that

(i) $\mathsf{M}(w)_{q(n)}$ is unlikely to be the identity

(ii) $\mathsf{M}(w)_{q(n)}$ can be computed in linear time.

In the rest of the paper, we use the following function q.

Definition 12. Let $q\colon \mathbb{N} \to \mathbb{N}$ be the function given by $q(0) = q(1) = 1$ and, for all $n \geq 2$,

$$q(n) = \prod_{\substack{p \leq \log^5 n \\ p \text{ prime}}} p$$

Algorithm QuickWP is the following.

Algorithm QuickWP

Input : a sequence w of n elements of $\tilde{\Sigma}$
Output: True if $\mathsf{M}(w) = \mathsf{Id}$, and False otherwise

```
Compute q(n)
if DC_{Σ,q(n)}(w) ≠ Id then
    return False
else
    if DC_Σ(w) ≠ Id then
        return False
    else
        return True
```

We can now give a precise version of our main theorem, stated in the introduction.

Theorem 13. *Algorithm QuickWP solves Problem WP_Σ with linear time average-case complexity, when inputs are uniform random words.*

The reader should note that Algorithm QuickWP makes no assumption on the algebraic or combinatorial properties of Σ or the subgroup $H = \langle \Sigma \rangle$. The same algorithm is run, with linear average-case complexity, whether Σ consists of triangular matrices or not, and whether H is finite or infinite. The latter property is decidable (Jacob, [Jac78]) in polynomial time (Babai, Beals and Rockmore [BBR93], see also Detinko and Flannery [DF09]). Similarly, the same algorithm is run, with the same average-case complexity whether H has polynomial or exponential growth, or whether it is nilpotent (see Section 3.2), polycyclic or virtually solvable. In the latter two situations, Olshanskii and Shpilrain recently proved a linear average-case complexity of the Word Problem [OS25, Theorem 3 and Remark 4], using the properties of these subgroups, namely computing in a quotient satisfying a specific algebraic condition.

Towards the proof of Theorem 13, we record the following technical statements.

Proposition 14. *The function q can be computed in polylogarithmic time, and $q(n)$ has polylogarithmic length.*

Proof. A rough upper bound is of the form $q(n) \leq (\log^5 n)^{\log^5 n}$, so that $\log q(n)$ (and hence the length of $q(n)$) is bounded above by $5 \log^6 n$, a polylogarithm.

It follows from algorithms proposed by Mairson [Mai77] and Pritchard [Pri87] that one can list all prime numbers less than or equal to N in (bit-complexity) $\mathcal{O}(N(\log N)(\log\log N))$. In particular, listing the prime numbers at most equal to $\log^5 n$ is done in $\mathcal{O}(\log^5 n(\log\log n)(\log\log\log n))$.

Computing the product of two numbers at most equal to $q(n)$ (and hence with length at most $5\log^6 n$) is done in time $\mathcal{O}(\log^6 n \log\log n)$ by Proposition 4 (ii). It follows that $q(n)$, the product of at most $\log^5 n$ numbers less than or equal to $q(n)$, is computed in time $\mathcal{O}(\log^{11} n \log\log n)$: again a polylogarithm. (Using a divide and conquer method yields a polylogarithm with lesser degree.) □

Lemma 15. *Let P_n be the probability that a word w of length n satisfies $\mathsf{M}(w)_{q(n)} = \mathsf{Id}$. The average-case complexity of QuickWP (when inputs are random words of length n) is $\mathcal{O}(n + P_n\, n \log^2 n)$.*

Moreover if $H = \langle \Sigma \rangle$ is finite, then the average-case complexity of QuickWP is $\mathcal{O}(n)$.

Proof. Once $q(n)$ is computed (in polylogarithmic time, by Proposition 14), the complexity $C(n)$ of the second step of Algorithm QuickWP, that is, of running $\mathsf{DC}_{\Sigma,q(n)}$ on a word w of length n, satisfies

$$C(n) = C\left(\lfloor n/2 \rfloor\right) + C\left(\lceil n/2 \rceil\right) + \mathcal{O}(\log q(n)\, \log\log q(n)) \qquad \text{for } n \geq 2$$

since every arithmetic operation in $\mathbb{Z}/q(n)\mathbb{Z}$ can be performed in time $\mathcal{O}(\log q(n)\, \log\log q(n))$ (see Corollary 6). By Proposition 14 again, we have $\log q(n)\, \log\log q(n) = o(n)$, and Proposition 8 yields the fact that $C(n)$ is linear.

Finally, by Proposition 9, the worst-case complexity of Algorithm DC_Σ on inputs of length n is in $\mathcal{O}(n \log^2 n)$ if H is infinite and is linear if H is finite. The announced result follows. □

In Section 4, we prove the following statement.

Theorem 16. *If $H = \langle \Sigma \rangle$ is infinite, then $\mathsf{M}(w)_{q(n)} = \mathsf{Id}$ with probability $\mathcal{O}(\log^{-2} n)$ (when w is chosen uniformly at random among length n words).*

The proof of Theorem 13 follows directly.

Proof of Theorem 13. It is immediate that Algorithm QuickWP solves WP_Σ. Note that the subgroup H is fixed in our setting (that is: it is not part of the input). If H is finite, the result was established in Lemma 15.

If H is infinite, then Theorem 16 and Lemma 15 show that QuickWP runs with linear average-case complexity. □

We are now left with proving Theorem 16: this is done in Section 4.

4 Proof of Theorem 16

Recall that H denotes the subgroup of $\mathsf{GL}_d(\mathbb{Z})$ generated by Σ, and that H_m denotes the mod m projection of H.

Throughout this section, we let m be an integer greater than $\max_{A\in\tilde{\Sigma}} \|A\|_\infty$. We note that $q(n)$ satisfies this condition for all n large enough.

Remark 17. We have $|\mathsf{GL}_d(\mathbb{Z}/m\mathbb{Z})| \leq m^{d^2}$. We assumed that Σ does not contain mutually inverse matrices, see Section 2.1. Our assumption on m guarantees that the matrices A_m ($A \in \tilde{\Sigma}$) are pairwise distinct and, in particular, none is the identity modulo m. In particular, since H_m is a group, if $M \in H_m$ and $A, B \in \tilde{\Sigma}$ with $A \neq B$, then $M\,A \neq M$ and $MA \neq MB$. An elementary technical consequence is the following: if $M, M' \in H_m$ and $A \in \tilde{\Sigma}$, there exists at most one matrix $B \in \tilde{\Sigma}$ such that $MAB = M'$.

4.1 Uniform random words as trajectories in a Markov chain

The matrices $\mathsf{M}(w)_m$, when w is a random word of length n, are naturally produced by the length n trajectories in the Markov chain $\mathfrak{U}_m$ defined below. Recall that $|\Sigma| = k$.

(i) The state set of $\mathfrak{U}_m$ is the subgroup H_m.

(ii) There is an edge $M \xrightarrow{\frac{1}{2k}} M'$ if and only if there exists a matrix $A \in \tilde{\Sigma}$ such that $M\,A = M'$.

(iii) The initial vector assigns probability 1 to Id and probability 0 to all the other states.

Let P_m be the transition matrix of $\mathfrak{U}_m$. We formulate the following elementary observations.

- In Item (ii), the matrix $A \in \tilde{\Sigma}$ is uniquely determined (if it exists) by the origin M and the end M' of the edge: we will denote this edge by $M \xrightarrow{A:\frac{1}{2k}} M'$ when needed. We call A *the matrix label* of that edge.
- The underlying graph of $\mathfrak{U}_m$ is strongly connected, that is, the Markov chain $\mathfrak{U}_m$ is irreducible.
- The Markov chain $\mathfrak{U}_m$ is symmetric, since $\tilde{\Sigma}$ contains both the matrices in Σ and their inverses.
- In the underlying graph of $\mathfrak{U}_m$, replacing each edge label of the form $A : \frac{1}{2k}$ by its matrix label A, yields the Cayley graph of the subgroup H_m. By Remark 17, no edge of $\mathfrak{U}_m$ is a loop.
- If T_n is a length n trajectory in $\mathfrak{U}_m$ and $\mathsf{lab}(T_n)$ denotes the word obtained from T_n by reading the sequence of matrix labels of the edges traversed by T_n, then $w = \mathsf{lab}(T_n)$ is a uniform random word of $\tilde{\Sigma}^n$, and T_n ends at state $\mathsf{M}(w)_m$.
- Let $\vec{\mathbf{1}}$ be the (column) vector all of whose entries are 1. Then $P_m\,\vec{\mathbf{1}} = \vec{\mathbf{1}}$, so the uniform distribution vector $\frac{1}{|H_m|}\vec{\mathbf{1}}$ is a right eigenvector for the eigenvalue 1. Since P_m is symmetric, it is also a left eigenvector for the eigenvalue 1.

As observed, the Markov chain $\mathfrak{U}_m$ is irreducible, but it may not be aperiodic. However, $\mathfrak{U}_m$ contains cycles of length two (for instance $\mathsf{Id} \xrightarrow{A} A_m \xrightarrow{A^{-1}} \mathsf{Id}$ for any $A \in \Sigma$), and since its period is the gcd of the lengths of its cycle, it is equal to 1 or 2. To deal with both cases at once, we consider the Markov chain $\mathfrak{U}_m^2$, which performs two consecutive steps in $\mathfrak{U}_m$, and whose transition matrix is P_m^2. More precisely, if $M, M' \in H_m$, then

$$P_m^2(M, M') = \sum_{\substack{A,B\in\tilde{\Sigma}\\ M'=MAB}} \frac{1}{4k^2}$$

(where products are taken in H_m). Given our hypothesis on m, for each $A \in \tilde{\Sigma}$, there is at most one matrix $B \in \tilde{\Sigma}$ such that $MAB = M'$, and hence $P_m^2(M, M') \leq \frac{1}{2k}$ (see Remark 17).

In the particular case where $M' = M$, for each $A \in \tilde{\Sigma}$, we have $MAA^{-1} = M$, so $P_m^2(M, M) = \sum_{A \in \tilde{\Sigma}} \frac{1}{4k^2} = \frac{1}{2k}$.

As it contains self-loops, the Markov chain $\mathfrak{U}_m^2$ is aperiodic. However, if $\mathfrak{U}_m$ has period 2, then $\mathfrak{U}_m^2$ is not strongly connected. More precisely, if $\mathfrak{U}_m$ has period 2, then $\mathfrak{U}_m^2$ is the disjoint union of two chains with the same number of states: one for the states at an even distance from Id in $\mathfrak{U}_m$, and one for the states at an odd distance from Id. The state set of the former is $\{M(w)_m : w \in \tilde{\Sigma}^* \text{ and } |w| \text{ even}\} = \langle \tilde{\Sigma}^2 \rangle$, and it has cardinality $\frac{1}{2}|H_m|$.

Let $\tilde{\mathfrak{U}}_m$ be the restriction of $\mathfrak{U}_m$ to the set $\tilde{H}_m$ of states that are accessible from Id in $\mathfrak{U}_m^2$. Again: if $\mathfrak{U}_m$ is aperiodic, then $\tilde{\mathfrak{U}}_m = \mathfrak{U}_m$ and $\tilde{H}_m = H_m$; and if $\mathfrak{U}_m$ has period 2, then $|\tilde{H}_m| = \frac{1}{2}|H_m|$. We let $\tilde{P}_m$ be the restriction of P_m^2 to $\tilde{H}_m$, that is, $\tilde{P}_m(M, M') = P_m^2(M, M')$ for all $M, M' \in \tilde{H}_m$.

To summarize, we have the following statement.

Proposition 18. *$\tilde{\mathfrak{U}}_m$ is a symmetric and primitive Markov chain, whose set of states $\tilde{H}_m$ satisfies $|\tilde{H}_m| \geq \frac{1}{2}|H_m|$. The uniform distribution on $\tilde{H}_m$ is its unique stationary distribution and $\tilde{\mathfrak{U}}_m$ is also reversible. For every $M \in \tilde{H}_m$, we have $\tilde{P}_m(M, M) = \frac{1}{2k}$. Moreover, for all $M, M' \in \tilde{H}_m$ such that $\tilde{P}_m(M, M') > 0$, we have $\frac{1}{4k^2} \leq \tilde{P}_m(M, M') \leq \frac{1}{2k}$.*

4.2 Probability of being the identity in $\tilde{\mathfrak{U}}_m$

By Proposition 18, the distribution of the states reached after a random length n trajectory, starting at any state in $\tilde{H}_m$, converges to the uniform distribution π. We now need to evaluate the rate of this convergence.

Lemma 19. *Let m be an integer greater than $\max_{A \in \tilde{\Sigma}} \|A\|_\infty$. If $n \geq 1$, the distribution vector of the state reached after n random steps in $\tilde{\mathfrak{U}}_m$, starting from Id, namely $\tilde{P}_m^n(\mathsf{Id}, \cdot)$, satisfies*

$$\left\| \tilde{P}_m^n(\mathsf{Id}, \cdot) - \frac{1}{|\tilde{H}_m|} \right\|_{Var} \leq \frac{1}{2} \sqrt{|\tilde{H}_m|} \left(1 - \frac{1}{4k^2 \, |\tilde{H}_m|^2}\right)^n. \tag{4}$$

Proof. The proof is a combination of results in [DS91], on the maximal and minimal eigenvalues of a Markov chain, provided the chain in question is primitive and reversible. The chain $\tilde{\mathfrak{U}}_m$ satisfies these hypotheses, its stationary distribution is the uniform distribution, and its eigenvalues β_i ($i \in [0, |\tilde{H}_m| - 1]$) are real, say $1 = \beta_0 > \beta_1 \geq \beta_2 \geq \ldots > \beta_{|\tilde{H}_m|-1} > -1$, see Theorem 7.

Let $\beta_* = \max\{\beta_1, |\beta_{|\tilde{H}_m|-1}|\}$. Equation (1.9) of [DS91, Prop. 3] states that

$$\left\| \tilde{P}_m^n(\mathsf{Id}, \cdot) - \pi \right\|_{\text{Var}} \leq \frac{1}{2} \sqrt{\frac{1 - \pi(\mathsf{Id})}{\pi(\mathsf{Id})}} \, \beta_*^n.$$

Since $\pi(h) = \frac{1}{|\tilde{H}_m|}$ for every $h \in \tilde{H}_m$, we have $\frac{1-\pi(\mathsf{Id})}{\pi(\mathsf{Id})} = |\tilde{H}_m| - 1$. It follows that

$$\left\| \tilde{P}_m^n(\mathsf{Id}, \cdot) - \frac{1}{|\tilde{H}_m|} \right\|_{\text{Var}} \leq \frac{1}{2} \sqrt{|\tilde{H}_m|} \, \beta_*^n.$$

Thus, we only need to prove that $\beta_* \leq 1 - \frac{1}{4k^2 \, |\tilde{H}_m|^2}$.

We first establish that $\beta_1 \leq 1 - \frac{1}{4k^2 \, |H_m|^2}$ using Poincaré's inequality, as presented in and with the notation of [DS91, Prop. 1].

For each edge e of $\tilde{\mathfrak{U}}_m$, from state x to state y, we let $Q(e) = \tilde{P}_m(x, y)\pi(x) = \tilde{P}_m(y, x)\pi(y)$. Hence for every edge e we have

$$\frac{1}{4k^2 |\tilde{H}_m|} \leq Q(e) = \frac{1}{|\tilde{H}_m|} \tilde{P}_m(x, y) \leq \frac{1}{2k|\tilde{H}_m|}$$

For each ordered pair of distinct states $x, y \in \tilde{H}_m$, we fix a path $\gamma_{x,y}$ from x to y such that a given edge appears at most once in the path, of length at most $|\tilde{H}_m|$. Such a path exists since the chain is irreducible. We define its path length as $|\gamma_{x,y}|_Q = \sum_{e \in \gamma_{x,y}} Q(e)^{-1}$. Then $|\gamma_{x,y}|_Q \leq 4k^2 |\tilde{H}_m|^2$.

Poincaré's inequality ([DS91, Prop. 1]) states that

$$\beta_1 \leq 1 - \frac{1}{\kappa}, \text{ where } \kappa = \max_{e \text{ edge}} \sum_{x,y \text{ such that } e \in \gamma_{x,y}} |\gamma_{x,y}|_Q \, \pi(x)\pi(y).$$

For all x, y, we have $|\gamma_{x,y}|_Q \pi(x)\pi(y) \leq 4k^2$ since $\pi(x) = \pi(y) = \frac{1}{|\tilde{H}_m|}$. Thus $\kappa \leq 4k^2 |\tilde{H}_m|^2$. It follows that $\beta_1 \leq 1 - \frac{1}{4k^2 \, |\tilde{H}_m|^2}$, as announced.

Now we use [DS91, Prop. 2] to prove that $\beta_{|\tilde{H}_m|-1} \geq \frac{1}{k} - 1$. For each state x, we let σ_x be the trajectory consisting of (a single iteration of) the self-loop at x. By [DS91, Prop. 2], we have

$$\beta_{|\tilde{H}_m|-1} \geq -1 + \frac{2}{\iota}, \text{ where } \iota = \max_{e \text{ edge}} \sum_{x \text{ such that } e \in \sigma_x} |\sigma_x|_Q \, \pi(x).$$

Since each σ_x contains only one edge e which satisfies $|\sigma_x|_Q = 2k|\tilde{H}_m|$, we have $\iota = 2k$ since $\pi(x) = \frac{1}{|\tilde{H}_m|}$ and hence $\beta_{|H_m|-1} \geq -1 + \frac{1}{k}$. As a result, $|\beta_{|\tilde{H}_m|-1}| < \beta_1$ and therefore $\beta_* = \beta_1 \leq 1 - \frac{1}{4k \, |\tilde{H}_m|^2}$, thus concluding the proof. □

4.3 Proof of Theorem 16

We use the following linear algebraic results. The first one, Lemma 20, is a slight generalization of [Kur03, Lemma 11] to matrices of dimension greater than 2. Corollary 21 follows from Lemma 20 and the prime number theorem.

Lemma 20. *Let $A \in \mathsf{GL}_d(\mathbb{Z})$ be a matrix of infinite order. The number of primes p such that A_p has order at most $\mathfrak{o}$ in $\mathsf{GL}_d(\mathbb{Z}/p\mathbb{Z})$ is $\mathcal{O}(\mathfrak{o}^2)$.*

Proof. An immediate induction establishes that, for any $\mathfrak{n} \geq 1$, we have $\|A^{\mathfrak{n}}\|_\infty \leq d^{\mathfrak{n}-1} \|A\|_\infty^{\mathfrak{n}}$. It follows that

$$\|A^{\mathfrak{n}} - \mathsf{Id}\|_\infty \leq d^{\mathfrak{n}-1} \|A\|_\infty^{\mathfrak{n}} + 1 \leq (d \, \|A\|_\infty)^{\mathfrak{n}} .$$

Since A has infinite order, $A^{\mathfrak{n}} \neq \mathsf{Id}$ and $\|A^{\mathfrak{n}} - \mathsf{Id}\|_\infty \neq 0$ for every $\mathfrak{n} \geq 1$. Moreover, A_p always has finite order in $\mathsf{GL}_d(\mathbb{Z}/p\mathbb{Z})$. If A_p is of order $\mathfrak{n}$, then p divides $\|A^{\mathfrak{n}} - \mathsf{Id}\|_\infty$, since every coefficient of $A^{\mathfrak{n}} - \mathsf{Id}$ is 0 modulo p.

Observe that if an integer N has x distinct prime factors, then $N \geq 2^x$, and hence $x \leq \log N$. Let $N = \prod_{\mathfrak{n}=1}^{\mathfrak{o}} \|A^{\mathfrak{n}} - \mathsf{Id}\|_\infty$. If p is a prime number such that A_p has order at most $\mathfrak{o}$ in $\mathsf{GL}_d(\mathbb{Z}/p\mathbb{Z})$, then p must divide N. Thus the number of primes p such that A_p has order at most $\mathfrak{o}$ is at most

$$\log N \leq \log \prod_{\mathfrak{n}=1}^{\mathfrak{o}} \|A^{\mathfrak{n}} - \mathsf{Id}\|_\infty \leq \sum_{\mathfrak{n}=1}^{\mathfrak{o}} \mathfrak{n} \log (d \, \|A\|_\infty) ,$$

which is $\mathcal{O}(\mathfrak{o}^2)$ when d and A are fixed. □

Corollary 21. *Let $A \in \mathsf{GL}_d(\mathbb{Z})$ be a matrix of infinite order, and let $q(n)$ be the function defined in Definition 12. For each n large enough, $q(n)$ admits a prime factor p_n such that A_{p_n} has order at least $2 \log^2 n$ in $\mathsf{GL}_d(\mathbb{Z}/p_n\mathbb{Z})$.*

Proof. According to the prime number theorem [Had96, DLVP96], which states that the number of prime numbers less than or equal to N is asymptotically equal to $\frac{N}{\ln N}$, where ln denote the Napierian logarithm, there exists a positive constant C such that, for N large enough, this number is at least $C \frac{N}{\log N}$. Therefore $q(n)$ is a product of at least $C \frac{\log^5 n}{5 \log \log n}$ different prime numbers.

Since this quantity is asymptotically greater than $4 \log^4 n$, Lemma 20 establishes that $q(n)$ has a prime factor p_n such that $|\langle A \rangle_{p_n}| > 2 \log^2 n$. □

Connecting Corollary 21 with Lemma 19, we get the following result.

Corollary 22. *Let $q(n)$ be the map defined in Definition 12, and let $(h_n)_n$ be a sequence such that $h_n \in \tilde{H}_{q(n)}$ for each n. If H is infinite, then the probability that a length n trajectory in $\tilde{\mathfrak{U}}_{q(n)}$ ends in h_n is $\mathcal{O}(\log^{-2} n)$.*

Proof. By a theorem due to Schur [Sch11], a finitely generated subgroup of $\mathsf{GL}_d(\mathbb{C})$ where each element has finite order must be finite. As a result, since H is infinite, it contains an element A with infinite order. Then by Corollary 21, for each n large enough, $q(n)$ admits a prime factor p_n such that A_{p_n} is of order at least $2\log^2 n$ in $\mathsf{GL}_d(\mathbb{Z}/p_n\mathbb{Z})$.

It follows that $|\tilde{H}_{p_n}| \geq \frac{1}{2}|H_{p_n}| \geq \frac{1}{2}|\langle A_{p_n}\rangle| \geq \log^2 n$. Moreover, one trivially has $|\tilde{H}_{p_n}| \leq p_n^{d^2} \leq (\log n)^{5d^2}$, since each prime divisor of $q(n)$ is at most equal to $\log^5 n$.

It follows from these two inequalities that $\log^2 n \leq |\tilde{H}_{p_n}| \leq p_n^{d^2}$, and hence $p_n \geq (\log n)^{2/d^2}$. Thus, for n sufficiently large, p_n is greater than $\max_{B\in\tilde{\Sigma}} \|B\|_\infty$, as required to apply Lemma 19

For each even integer $n = 2\nu$, let $\hat{h}_n$ be the projection of h_n modulo p_n, which is well defined since p_n divides $q(n)$. If $\hat{h}_n \notin \tilde{H}_{p_n}$, then $\tilde{P}^\nu_{p_n}(\mathsf{Id}, \hat{h}_n) = 0$. If $\hat{h}_n \in \tilde{H}_{p_n}$, by Lemma 19, and with the notation of that statement, we have

$$\left|\tilde{P}^\nu_{p_n}(\mathsf{Id}, \hat{h}_n) - \frac{1}{|\tilde{H}_{p_n}|}\right| \leq \left\|\tilde{P}^\nu_{p_n}(\mathsf{Id}, \cdot) - \frac{1}{|\tilde{H}_{p_n}|}\right\|_{\mathrm{Var}} \leq \frac{1}{2}\sqrt{|\tilde{H}_{p_n}|}\left(1 - \frac{1}{4k^2\,|\tilde{H}_{p_n}|^2}\right)^\nu.$$

Therefore, as $\log^2 n \leq |\tilde{H}_{p_n}| \leq (\log n)^{5d_2}$, we have

$$\begin{aligned}\tilde{P}^\nu_{p_n}(\mathsf{Id}, \hat{h}_n) &\leq \frac{1}{|\tilde{H}_{p_n}|} + \frac{1}{2}\sqrt{|\tilde{H}_{p_n}|}\left(1 - \frac{1}{4k^2\,|\tilde{H}_{p_n}|^2}\right)^\nu \\ &\leq \frac{1}{\log^2 n} + \frac{1}{2}\sqrt{(\log n)^{5d^2}}\,\exp\left(-\frac{n}{8k^2(\log n)^{10d^2}}\right).\end{aligned}$$

Therefore, $P^n_{p_n}(\mathsf{Id}, \hat{h}_n) = \tilde{P}^\nu_{p_n}(\mathsf{Id}, \hat{h}_n)$ is $\mathcal{O}(\log^{-2} n)$ if n is even and $\hat{h}_n \in \tilde{H}_{p_n}$. This also holds if $\hat{h}_n \notin \tilde{H}_{p_n}$, as the corresponding probability is equal to zero.

Now suppose that n is odd, $n = 2\nu + 1$. In that case, $P^n_{p_n}(\mathsf{Id}, \hat{h}_n) = \left(\tilde{P}^\nu_{p_n} \times P_{p_n}\right)(\mathsf{Id}, \hat{h}_n)$. This is equal to $\sum_{k_n \in \tilde{H}_{p_n}} \tilde{P}^\nu_{p_n}(\mathsf{Id}, k_n)\, P_{p_n}(k_n, \hat{h}_n)$. Observe that $P_{p_n}(k_n, \hat{h}_n) = 0$ unless $k_n = \hat{h}_n A$ for some $A \in \tilde{\Sigma}$ (which holds for exactly $2k$ values of k_n), in which case $P_{p_n}(k_n, \hat{h}_n) \leq \frac{1}{2k}$. It follows that, in this case as well, $P^n_{p_n}(\mathsf{Id}, \hat{h}_n)$ is $\mathcal{O}(\log^{-2} n)$.

Finally, we note that, if $A \in \mathsf{GL}_d(\mathbb{Z})$ is such that $A_{q(n)} = \mathsf{Id}$, then $A_{p_n} = \mathsf{Id}$ since p_n is a divisor of $q(n)$. Thus, the probability that a trajectory in $\tilde{\mathfrak{U}}_{q(n)}$ ends in h_n is bounded above by the probability that a trajectory following the same steps in $\tilde{\mathfrak{U}}_{p_n}$ ends in $\hat{h}_n$, thus concluding the proof. □

Corollary 22 directly implies the proof of Theorem 16 by taking each h_n to be Id.

Remark 23. It is interesting to note that, in the proof of Corollary 22, we are not concerned with the value of the matrix A or the prime p_n, nor with how hard it would be to compute them. It is enough, for our purpose, to know that they exist.

References

[BBR93] Laszlo Babai, Robert Beals, and Daniel Rockmore. Deciding finiteness of matrix groups in deterministic polynomial time. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation, ISSAC*, pages 117–126. Association for Computing Machinery, 1993.

[Boo59] William W. Boone. The word problem. *Ann. of Math. (2)*, 70:207–265, 1959.

[CLRS22] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to algorithms*. MIT press, 2022.

[Deh11] Max Dehn. Über unendliche diskontinuierliche Gruppen. *Mathematische Annalen*, 71:116–144, 1911.

[DF09] Alla Detinko and David Flannery. On deciding finiteness of matrix groups. *Journal of Symbolic Computation*, 44(8):1037–1043, 2009.

[DLVP96] Charles De La Vallee-Poussin. Recherches analytiques sur la théorie des nombres premiers. *Ann. Soc. Sc. Bruxelles*, 1896.

[DS91] Persi Diaconis and Daniel Stroock. Geometric bounds for eigenvalues of Markov chains. *The Annals of Applied Probability*, 1(1):36–61, 1991.

[ECH+92] David B. A. Epstein, James W. Cannon, Derek F. Holt, Silvio V. F. Levy, Michael S. Paterson, and William P. Thurston. *Word processing in groups*. Jones and Bartlett Publishers, Boston, MA, 1992.

[Had96] Jacques Hadamard. Sur la distribution des zéros de la fonction $\zeta(s)$ et ses conséquences arithmétiques. *Bulletin de la Societé mathematique de France*, 24:199–220, 1896.

[HvdH21] David Harvey and Joris van der Hoeven. Integer multiplication in time $\mathcal{O}(n \log n)$. *Annals of Mathematics*, 193(2):563–617, 2021.

[Jac78] Gérard Jacob. La finitude des représentations linéaires des semi-groupes est décidable. *Journal of Algebra*, 52(2):437–459, 1978.

[Kur03] Pär Kurlberg. On the order of unimodular matrices modulo integers. *Acta Arithmetica*, 110(2):141–151, 2003.

[LP17] David Levin and Yuval Peres. *Markov chains and mixing times, Second edition*. American Mathematical Society Press, 2017.

[LS01] Roger C. Lyndon and Paul E. Schupp. *Combinatorial group theory*. Classics in Mathematics. Springer-Verlag, Berlin, 2001. Reprint of the 1977 edition.

[Mai77] Harry G Mairson. Some new upper bounds on the generation of prime numbers. *Communications of the ACM*, 20(9):664–669, 1977.

[MKS66] Wilhelm Magnus, Abraham Karrass, and Donald Solitar. *Combinatorial group theory: Presentations of groups in terms of generators and relations*. Interscience Publishers [John Wiley & Sons], New York-London-Sydney, 1966.

[Nov55] Petr S. Novikov. *Ob algoritmičeskoĭ nerazrešimosti problemy toždestva slov v teorii grupp (On the algorithmic unsolvability of the word problem in group theory)*. Izdat. Akad. Nauk SSSR, Moscow, 1955. Trudy Mat. Inst. Steklov. no. 44.

[OS25] Alexander Olshanskii and Vladimir Shpilrain. Linear average-case complexity of algorithmic problems in groups. *Journal of Algebra*, 668:390–419, 2025.

[Pri87] Paul Pritchard. Linear prime-number sieves: a family tree. *Sci. Comput. Programming*, 9(1):17–35, 1987.

[Sch11] Issai Schur. Über Gruppen periodischer Substitutionen. *Sitzungsber. Preuss. Akad. Wiss.*, pages 619–627, 1911.

[Sim73] H. Simmons. The word problem for absolute presentations. *J. London Math. Soc. (2)*, 6:275–280, 1973.

[vzGG13] Joachim von zur Gathen and Jürgen Gerhard. *Modern Computer Algebra, Third edition*. Cambridge University Press, 2013.

[Weh80] Bertram A.F. Wehrfritz. On finitely generated soluble linear groups. *Mathematische Zeitschrift*, 170:155–167, 1980.