AIMS Mathematics, 11(4): 9380–9397.
DOI: 10.3934/math.2026389
Received: 08 December 2025
Revised: 18 March 2026
Accepted: 27 March 2026
Published: 07 April 2026

https://www.aimspress.com/journal/Math

Research article

Probabilistic bounds on the number of elements to generate finite nilpotent groups and their applications to quantum algorithms

Ziyuan Dong[1,2], **Xiang Fan**[3,*], **Tengxun Zhong**[3] **and Daowen Qiu**[1,2,*]

[1] Institute of Quantum Computing and Software, School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, 510006, China

[2] The Guangdong Key Laboratory of Information Security Technology, Sun Yat-sen University, Guangzhou, 510006, China

[3] School of Mathematics, Sun Yat-sen University, Guangzhou, 510275, China

* **Correspondence:** Email: fanx8@mail.sysu.edu.cn, issqdw@mail.sysu.edu.cn.

Abstract: This work establishes a new probabilistic bound on the number of elements needed to generate finite nilpotent groups. Let $\varphi_k(G)$ denote the probability that k random elements generate a finite nilpotent group G. For any $0 < \epsilon < 1$, we prove that $\varphi_k(G) \geq 1 - \epsilon$ if $k \geq \text{rank}(G) + \lceil \log_2(2/\epsilon) \rceil$ (a bound based on the group rank) or if $k \geq \text{len}(G) + \lceil \log_2(1/\epsilon) \rceil$ (a bound based on the composition length). Moreover, these bounds are shown to be nearly tight. Both bounds sharpen the previously known requirement of $k \geq \lceil \log_2 |G| + \log_2(1/\epsilon) \rceil + 2$. Our results provide a foundational tool to analyze probabilistic algorithms, thereby enabling a better estimation of the iteration count for the finite abelian hidden subgroup problem (AHSP) standard quantum algorithm and a reduction in the circuit repetitions required by Regev's factoring algorithm.

Keywords: probabilistic group theory; group generation; finite nilpotent group; abelian hidden subgroup problem; quantum algorithms
Mathematics Subject Classification: 20P05, 68Q12

1. Introduction

The probabilistic group theory plays a fundamental role in understanding random generation behavior in finite groups, with significant applications across the computational group theory, cryptography, randomized algorithms, and quantum algorithms [1]. For a finite group G, let $\varphi_k(G)$ denote the probability that k independently and uniformly chosen random elements generate G.

Formally,

$$\varphi_k(G) = \frac{N_k(G)}{|G|^k},$$

where $N_k(G)$ counts the number of k-tuples $(g_1, \ldots, g_k) \in G^k$ that generate G.

For any finite group G, Pak [2] established that $\varphi_k(G) \geq 1-\epsilon$ whenever $k \geq \lceil \log_2 |G| + 2 + \log_2(1/\epsilon) \rceil$ for any $0 < \epsilon < 1$. This was refined by Detomi and Lucchini [3], who proved the existence of a constant c_ϵ such that $\varphi_k(G) \geq 1 - \epsilon$ whenever $k \geq \lfloor \text{rank}(G) + c_\epsilon \log_2 \log_2 |G| \log_2 \log_2 \log_2 |G| \rfloor$, where $\text{rank}(G)$ denotes the minimal number of generators; however, c_ϵ lacks an explicit expression. For more structured classes, stronger results emerge. For solvable groups G, Pak [2] showed $\varphi_k(G) \geq 1/e \approx 0.367879\ldots$ when $k \geq 3.25\,\text{rank}(G) + 10^7$. By further specializing to nilpotent groups G, which form a proper subclass of solvable groups and include all finite abelian groups, the bound improves as follows: $\varphi_k(G) \geq 1/e$, whenever $k \geq \text{rank}(G) + 1$ [2].

Beyond the probability bounds, the expected number $\mathcal{E}(G)$ of random elements required to generate a group has been extensively studied. Pomerance [4] proved that for finite abelian groups (and more generally, nilpotent groups), we have $\mathcal{E}(G) \leq \text{rank}(G) + \sigma$, where $\sigma = 2.118456\ldots$ is a constant given in terms of the Riemann zeta function. Furthermore, for finite nilpotent groups, the generation probability by $\text{rank}(G) + 1$ random elements exceeds $c = 0.435757\ldots$, thus improving the $1/e$ bound in [2]. For general finite groups, Lubotzky [5] established $\mathcal{E}(G) \leq e \cdot \text{rank}(G) + 2e \log_2 \log_2 |G| + 11$, thereby providing the first general upper bound for $\mathcal{E}(G)$. Additionally, Lubotzky showed $\varphi_k(G) \geq 1/e$ whenever $k \geq \text{rank}(G) + 2\log_2 \log_2 |G| + 4.02$, thus confirming Pak's conjecture. More recently, Lucchini [6] derived an exact formula for $\mathcal{E}(G)$ via the Möbius function of the subgroup lattice, thus yielding precise values for nontrivial finite groups.

Despite these advances, the existing probabilistic analyses remain insufficient for algorithmic applications. We bridge this gap in Theorem 12 by proving that for any nilpotent group G and $0 < \epsilon < 1$, $\varphi_k(G) \geq 1 - \epsilon$ whenever

$$k \geq \text{rank}(G) + \lceil \log_2(2/\epsilon) \rceil$$

or

$$k \geq \text{len}(G) + \lceil \log_2(1/\epsilon) \rceil.$$

Both bounds sharpen the previously known requirement $k \geq \lceil \log_2 |G| + \log_2(1/\epsilon) \rceil + 2$. Additionally, in Theorem 13, we show that these bounds are nearly tight. Specifically, via a constructive counterexample, it shows that the composition length bound can be reduced by at most 1, and the group rank bound by at most 2.

Our theorem enables diverse applications, including two major examples in quantum computing. The hidden subgroup problem (HSP) [7] constitutes a central challenge in quantum computation, thereby encompassing most known exponential speedups achieved via the quantum Fourier transform [8]. Each iteration of the standard quantum algorithm for the finite abelian HSP (AHSP) [7,8] yields an element $\mathbf{t_i}$ from $H^\perp$ (see Definition 6), where $H^\perp \leq G$ denotes the dual orthogonal subgroup of the hidden subgroup $H \leq G$. All elements $\mathbf{t}_i \in H^\perp$ are equally likely to be measured [9]. After k iterations, we can only guarantee that $\langle \mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_k \rangle \subseteq H^\perp$, rather than $\langle \mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_k \rangle = H^\perp$. Then, we attempt to recover the hidden subgroup H, by solving the linear congruence system. Consequently, the success probability of the algorithm after k iterations is exactly $\varphi_k(H^\perp)$, that is, the probability that k uniformly random elements generate $H^\perp$.

Applying our probabilistic group theory result, we determine the iteration count to achieve a success probability of $1-\epsilon$ in the standard quantum finite AHSP algorithm to be either $\text{rank}(G)+\lceil\log_2(2/\epsilon)\rceil$ or $\text{len}(G)-\text{len}(H)+\lceil\log_2(1/\epsilon)\rceil$. The former achieves an exponential improvement in ϵ-dependence over the prior bound $\lfloor 4/\epsilon\rfloor\text{rank}(G)$ [10], while the latter improves upon $\lceil\log_2|G|+\log_2(1/\epsilon)+2\rceil$ [11].

Additionally, our result optimizes iteration counts in Regev's recent quantum factoring algorithm [12]. While Corollary 4.2 in Regev's work states that $\text{rank}(G)+4$ uniformly random elements generate G with a probability of at least $1/2$, we demonstrate that only $\text{rank}(G)+2$ elements suffice for the same success probability. This refinement directly reduces the quantum circuit repetition count from $\sqrt{n}+4$ to $\sqrt{n}+2$ in Regev's factoring algorithm while maintaining an identical success probability.

The remainder of this paper is organized as follows: Section 2 presents preparatory lemmas and background knowledge, including properties of $\text{rank}(G)$ and $\text{len}(G)$; Section 3 establishes our main sampling theorem for finite nilpotent groups, thereby analyzing the bound from the dual perspectives of group rank and composition length, and proving that the bounds are nearly tight; Section 4 applies our main theorem to quantum computing, thereby determining the iteration counts for the standard quantum finite AHSP algorithm and reducing quantum circuit repetitions in Regev's factoring algorithm; and Section 5 summarizes our main conclusions.

2. Preliminaries and earlier results

2.1. Earlier results for generating groups

In this subsection, we review prior results on the group generation.

$\text{Rank}(G)$ is defined as the minimal cardinality of a generating set of G. Additionally, the composition length $\text{len}(G)$ is defined as follows.

Definition 1 ([13, Definition 8.1])**.** *A composition series of a group G is a maximal chain of normal subgroups*

$$\{e\}=G_0\trianglelefteq G_1\trianglelefteq\cdots\trianglelefteq G_n=G$$

such that each quotient G_i/G_{i-1} is simple. The integer n is called the composition length of G, which is denoted as $\text{len}(G)$.

Remark 1. For any given finite group G, the composition length $\text{len}(G)$ is an invariant, though the composition series may not be unique.

Definition 2 (Frattini Subgroup [14, Definition 1D.7])**.** *For any group G, the Frattini subgroup $\Phi(G)$ is defined as the intersection of all maximal subgroups:*

$$\Phi(G):=\bigcap_{H\in\mathcal{H}}H\quad \textit{where}\quad \mathcal{H}=\{H\leq G\mid H \textit{ is a maximal subgroup}\}.$$

We conventionally set $\Phi(G)=G$ if G has no maximal subgroups.

Remark 2. $\Phi(G)$ is always a normal subgroup of G.

Definition 2 is equivalent to defining $\Phi(G)$ as the set of all non-generators of G. Formally,

$$\Phi(G):=\{g\in G\mid \forall S\subseteq G,\ \text{if } \langle S,g\rangle=G \text{ then } \langle S\rangle=G\}.$$

This implies that elements in $\Phi(G)$ are redundant to generate G.

Now, we have Lemma 3, which is fundamental to Lemma 4.

Lemma 3 ([15, Problem 8.7]). *Let $\Phi(G)$ be the Frattini subgroup of a finite group G. Then, a set $\{g_1, \ldots, g_k\} \subseteq G$ generates G if and only if $\{\bar{g}_1, \ldots, \bar{g}_k\}$ generates the quotient group $G/\Phi(G)$, where $\bar{g}_i = g_i\Phi(G)$.*

Proof. ($\Longrightarrow$) If $\langle g_1, \ldots, g_k\rangle = G$, then applying the quotient homomorphism gives $\langle \bar{g}_1, \ldots, \bar{g}_k\rangle = G/\Phi(G)$.

($\Longleftarrow$) Suppose that a set $\{g_1, \ldots, g_k\} \subseteq G$ satisfies $\langle \bar{g}_1, \ldots, \bar{g}_k\rangle = G/\Phi(G)$. Let $H = \langle g_1, \ldots, g_k\rangle$.

For any $g \in G$, its coset $\bar{g} = g\Phi(G)$ can be written as $\bar{g} = w(\bar{g}_1, \ldots, \bar{g}_k)$ for some word w. By the homomorphism property, this implies $g\Phi(G) = w(g_1, \ldots, g_k)\Phi(G)$; thus, $g \in H\Phi(G)$.

Thus, $G = H\Phi(G)$. By Remark 2, $\Phi(G)$ consists of non-generators; thus, $H = G$. □

Lemma 4 ([16, Lemma 1]). *Let G be a finite group with the Frattini subgroup $\Phi(G)$. Then,*

$$\varphi_k(G) = \varphi_k(G/\Phi(G)).$$

Proof. Let N be the number of k-tuples that generates $G/\Phi(G)$.

By Lemma 3, a k-tuple $(g_1, \ldots, g_k)$ generates G if and only if its projection $(\bar{g}_1, \ldots, \bar{g}_k)$ generates $G/\Phi(G)$.

For each generating k-tuple $(\bar{g}_1, \ldots, \bar{g}_k)$ of $G/\Phi(G)$, there are exactly $|\Phi(G)|^k$ preimages in G (since we can multiply each g_i by any element of $\Phi(G)$). Therefore, the number of generating k-tuples of G is $N \cdot |\Phi(G)|^k$.

Hence,

$$\varphi_k(G) = \frac{N \cdot |\Phi(G)|^k}{|G|^k} = \frac{N}{|G/\Phi(G)|^k} = \varphi_k(G/\Phi(G)),$$

where we use $|G| = |\Phi(G)| \cdot |G/\Phi(G)|$. □

Lemma 5 (Probabilities of Generating Finite p-Groups [15, Problem 8.26], [2]). *Let G be a finite p-group with* $\mathrm{rank}(G) = r$. *Then, the following holds:*

(i) $G/\Phi(G) \cong (\mathbb{Z}_p)^r$;

(ii) $\varphi_k(G) = \varphi_k((\mathbb{Z}_p)^r) = \begin{cases} \prod_{i=0}^{r-1} \left(1 - p^{i-k}\right), & \text{if } k \geq r, \\ 0, & \text{otherwise.} \end{cases}$

Proof. (i) See [15, Problem 8.26].

(ii) This result appears in the work of Pak [2] without proof; for the sake of completeness, we provide a proof here.

$(\mathbb{Z}_p)^r$ is an r-dimensional vector space over the finite field $\mathbb{Z}_p$. Each uniform sampling generates a row vector in $(\mathbb{Z}_p)^r$, and performing k independent uniform samplings yields a matrix $M \in \mathbb{Z}_p^{k\times r}$.

Our objective is to ensure M has a row rank r. By the fundamental rank equality (row rank = column rank), this implies that M has a full column rank r (i.e., its columns are linearly independent). Therefore, when $k \geq r$,

$$\varphi_k(\mathbb{Z}_p^r) = \frac{(p^k-1)(p^k-p)(p^k-p^2)\cdots(p^k-p^{r-1})}{(p^k)^r} = \prod_{i=0}^{r-1}\left(1-p^{i-k}\right).$$

Otherwise, when $k < r$,

$$\varphi_k(\mathbb{Z}_p^r) = 0.$$

□

Corollary 6. $\varphi_k(\mathbb{Z}_p^r) \geq \varphi_k(\mathbb{Z}_2^r)$.

Proof. The following is clear from Lemma 5(ii):

- When $k \geq r$, $\varphi_k(\mathbb{Z}_p^r) = \prod_{i=0}^{r-1}\left(1-p^{i-k}\right) \geq \prod_{i=0}^{r-1}\left(1-2^{i-k}\right) = \varphi_k(\mathbb{Z}_2^r)$;
- When $k < r$, $\varphi_k(\mathbb{Z}_p^r) = 0 = \varphi_k(\mathbb{Z}_2^r)$.

Thus, $\varphi_k(\mathbb{Z}_p^r) \geq \varphi_k(\mathbb{Z}_2^r)$. □

Definition 3 (Nilpotent Group [14])**.** *Let G be a group. Define $\gamma_1(G) = G$ and $\gamma_{i+1}(G) = [\gamma_i(G), G]$ for $i \geq 1$. We call G nilpotent if there exists a positive integer c such that $\gamma_{c+1}(G) = \{e\}$. The smallest such c is called the nilpotency class of G. In particular, a finite nilpotent group can be decomposed as the direct product of its Sylow subgroups.*

Lemma 7 (Probability Product Formula for Finite Nilpotent Groups [16, Corollary 4])**.** *Let $G \cong G_{p_1} \times G_{p_2} \times \cdots \times G_{p_m}$ be a finite nilpotent group, where G_{p_i} are the Sylow p_i-subgroups for distinct primes $p_1, \ldots, p_m$. Then, for any positive integer k,*

$$\varphi_k(G) = \prod_{i=1}^{m} \varphi_k(G_{p_i}).$$

2.2. *Properties of* rank(G) *and* len(G)

This subsection establishes key properties of rank(G) and len(G) to support our main results in Section 3 and the discussion on the quantum query complexity of the finite AHSP algorithm in Section 4.

Lemma 8 (Additivity of Composition Length for Finite Groups [13, Proposition 6.9])**.** *For any finite group G and normal subgroup $H \trianglelefteq G$,*

$$\text{len}(G) = \text{len}(H) + \text{len}(G/H).$$

Lemma 9 (Subadditivity of Rank for Finite Groups)**.** *For a finite group G and normal subgroup $H \trianglelefteq G$,*

$$\text{rank}(G) \leq \text{rank}(H) + \text{rank}(G/H).$$

Proof. We establish the subadditivity of rank through generating a set construction.

Let $S = \{s_1, \ldots, s_m\}$ be a minimal generating set for H; thus $\text{rank}(H) = m$. Similarly, let $\{\bar{t}_1, \ldots, \bar{t}_n\}$ be a minimal generating set for G/H with $\text{rank}(G/H) = n$. Choose representatives $t_i \in G$ such that $\bar{t}_i = t_i H$, and define $T = \{t_1, \ldots, t_n\}$.

For any $x \in G$, its coset $xH \in G/H$ lies in the subgroup generated by $\{\bar{t}_1, \ldots, \bar{t}_n\}$. Thus, there exists an element g in the subgroup generated by T such that $xH = gH$. This implies $x = gk$ for some $k \in H$.

Since $k \in H$, it lies in the subgroup generated by S. Therefore, x lies in the subgroup generated by $S \cup T$.

This shows that $S \cup T$ generates G, though it may not be minimal. We conclude that

$$\text{rank}(G) \leq |S \cup T| \leq |S| + |T| = \text{rank}(H) + \text{rank}(G/H).$$

□

Definition 4 (Solvable Group [14])**.** *A group G is called solvable if it has a subnormal series*

$$\{e\} = G_0 \trianglelefteq G_1 \trianglelefteq \cdots \trianglelefteq G_n = G$$

such that each quotient G_i/G_{i-1} is abelian. For finite groups, this is equivalent to the condition that all composition factors are cyclic groups of a prime order.

Theorem 10. *Let G be a finite solvable group. Then, the following statements hold:*

(i) *If $|G| = \prod_{j=1}^{k} q_j^{a_j}$ is the prime factorization of $|G|$, then*

$$\text{len}(G) = \sum_{j=1}^{k} a_j.$$

(ii) $\text{len}(G) \geq \text{rank}(G)$. *Moreover, if $G \cong (\mathbb{Z}_p)^k$ for some prime p and integer k, then* $\text{len}(G) = \text{rank}(G)$.

Proof. (i) Since G is finite, it admits a composition series as follows:

$$\{e\} = G_0 \trianglelefteq G_1 \trianglelefteq \cdots \trianglelefteq G_n = G,$$

where each factor G_i/G_{i-1} is simple. By the solvability of G, each factor is abelian and hence cyclic of prime order, say $G_i/G_{i-1} \cong \mathbb{Z}_{p_i}$, for some prime p_i. The length of this series is $\text{len}(G) = n$.

The order of G is the product of the orders of the composition factors:

$$|G| = \prod_{i=1}^{n} |G_i/G_{i-1}| = \prod_{i=1}^{n} p_i.$$

On the other hand, $|G| = \prod_{j=1}^{k} q_j^{a_j}$ is the prime factorization of $|G|$. By the fundamental theorem of arithmetic (uniqueness of prime factorization), the product $\prod_{i=1}^{n} p_i$ must coincide with $\prod_{j=1}^{k} q_j^{a_j}$. Therefore,

the list of primes $p_1, \ldots, p_n$ (counting multiplicities) is exactly the list of primes q_j that each appear a_j times. Hence, $n = \sum_{j=1}^{k} a_j$, thus proving (i).

(ii) Applying the subadditivity of group rank (Lemma 9) iteratively along the composition series yields the following:

$$\begin{aligned} \operatorname{rank}(G) &\leq \operatorname{rank}(G_0) + \operatorname{rank}(G_1/G_0) + \cdots + \operatorname{rank}(G/G_{n-1}) \\ &= 0 + 1 + \cdots + 1 \\ &= n = \operatorname{len}(G), \end{aligned}$$

where each $\operatorname{rank}(G_i/G_{i-1}) = 1$ since $G_i/G_{i-1} \cong \mathbb{Z}_{p_i}$.

If $G \cong (\mathbb{Z}_p)^k$, then clearly $\operatorname{len}(G) = \operatorname{rank}(G) = k$.

□

Theorem 11. *Let G be a finite nilpotent group with a Sylow decomposition as follows:*

$$G \cong G_p \times G_q,$$

where p and q are distinct primes. Then, the rank of G satisfies the following:

$$\operatorname{rank}(G) = \max\{\operatorname{rank}(G_p), \operatorname{rank}(G_q)\}.$$

Proof. Let $r = \operatorname{rank}(G)$, $r_p = \operatorname{rank}(G_p)$, and $r_q = \operatorname{rank}(G_q)$.

We begin by showing that $\max\{r_p, r_q\} \leq r$. Any generating set of G with r elements projects to generating sets of G_p and G_q under the natural projections $\pi_p : G \to G_p$ and $\pi_q : G \to G_q$; thus, $r_p \leq r$ and $r_q \leq r$. Hence, $\max\{r_p, r_q\} \leq r$.

Then, we show that $r \leq \max\{r_p, r_q\}$. Let $m = \max\{r_p, r_q\}$. Choose a minimal generating set $\{g_{p1}, \ldots, g_{pr_p}\}$ of G_p and a minimal generating set $\{g_{q1}, \ldots, g_{qr_q}\}$ of G_q. We extend these generating sets to size m by adding the identity elements e_p of G_p and e_q of G_q if $r_p < m$ or $r_q < m$, respectively.

Now, define elements $x_1, \ldots, x_m \in G$ by the following:

$$x_j = (g_{pj}, g_{qj}), \quad \text{where } g_{pj} = e_p \text{ if } j > r_p, \text{ and } g_{qj} = e_q \text{ if } j > r_q.$$

We claim that $\{x_1, \ldots, x_m\}$ generates G. To prove this, take an arbitrary element $g = (g_p, g_q) \in G$ with $g_p \in G_p$ and $g_q \in G_q$. Since $\{g_{p1}, \ldots, g_{pr_p}\}$ generates G_p and $\{g_{q1}, \ldots, g_{qr_q}\}$ generates G_q, we can write the following:

$$g_p = w_p(g_{p1}, \ldots, g_{pr_p}), \quad g_q = w_q(g_{q1}, \ldots, g_{qr_q}),$$

for some group words w_p and w_q.

Since $|G_p|$ and $|G_q|$ are coprime, by the Chinese Remainder Theorem (see [13, Theorem 2.25]), there exist integers k_p and k_q such that

$$k_p \equiv \begin{cases} 1 \pmod{|G_p|} \\ 0 \pmod{|G_q|} \end{cases}, \quad k_q \equiv \begin{cases} 0 \pmod{|G_p|} \\ 1 \pmod{|G_q|} \end{cases}.$$

Define $h_p, h_q \in G$ by the following:

$$h_p = \left[w_p(x_1, \ldots, x_{r_p})\right]^{k_p}, \quad h_q = \left[w_q(x_1, \ldots, x_{r_q})\right]^{k_q}.$$

Then, the projections satisfy the following:

$$\begin{aligned}
&\pi_p(h_p) = g_p^{k_p} = g_p \quad (\text{since } k_p \equiv 1 \pmod{|G_p|})\\
&\pi_q(h_p) = e_q \quad (\text{since } k_p \equiv 0 \pmod{|G_q|})\\
&\pi_p(h_q) = e_p \quad (\text{since } k_q \equiv 0 \pmod{|G_p|})\\
&\pi_q(h_q) = g_q^{k_q} = g_q \quad (\text{since } k_q \equiv 1 \pmod{|G_q|}).
\end{aligned}$$

Thus, $h_p = (g_p, e_q)$ and $h_q = (e_p, g_q)$; therefore,

$$g = h_p \cdot h_q \in \langle x_1, \ldots, x_m \rangle.$$

Hence, $\{x_1, \ldots, x_m\}$ generates G; thus, $r \leq m = \max\{r_p, r_q\}$.
By combining both inequalities, we conclude that $\text{rank}(G) = \max\{\text{rank}(G_p), \text{rank}(G_q)\}$. □

3. Main results

By synthesizing the results in Section 2, we establish Theorem 12, which is a new result in the probabilistic group theory using dual perspectives: group rank and composition length. Our approach minimally employs inequalities only for estimation, thus yielding a sufficient condition. Moreover, Theorem 13 provides a counterexample which shows that the composition length bound can be reduced by at most 1, and the rank bound by at most 2.

Theorem 12 (Bounds for Generating Finite Nilpotent Groups)**.** *For any finite nilpotent group G and $0 < \epsilon < 1$, let $\varphi_k(G)$ denote the probability that k elements $\{g_1, \ldots, g_k\}$, uniformly and independently sampled with replacement from G, generate the entire group G.*

Then, the following bounds hold:

(i) $\varphi_k(G) \geq 1 - \epsilon$, *if* $k \geq \text{rank}(G) + \left\lceil \log_2 \frac{2}{\epsilon} \right\rceil$.

(ii) $\varphi_k(G) \geq 1 - \epsilon$, *if* $k \geq \text{len}(G) + \left\lceil \log_2 \frac{1}{\epsilon} \right\rceil$.

Proof. Let G be a finite nilpotent group with $\text{rank}(G) = r$. Then, G admits a decomposition into its Sylow subgroups as follows:

$$G \cong \prod_{i=1}^{h} G_{p_i},$$

where $p_1, \ldots, p_h$ are the distinct prime divisors of $|G|$, and each Sylow p_i-subgroup G_{p_i} is a finite p_i-group. Let $\text{rank}(G_{p_i}) = r_i$ for $i = 1, \ldots, h$; then, by the iterated application of Theorem 11, it follows that the rank of G satisfies $\max_{1 \leq i \leq h} r_i = r$.

For a direct product of two groups A and B, since B is a normal subgroup of $A \times B$ and $(A \times B)/B \cong A$, Lemma 8 gives the following:

$$\text{len}(A \times B) = \text{len}(B) + \text{len}((A \times B)/B) = \text{len}(B) + \text{len}(A).$$

Applying this repeatedly yields the following:

$$\text{len}(G) = \sum_{i=1}^{h} \text{len}(G_{p_i}).$$

Then, for all $0 < \epsilon < 1$, we have the following:
(i)

$$\begin{aligned}
\varphi_k(G) &= \prod_{i=1}^{h} \varphi_k(G_{p_i}) \\
&= \prod_{i=1}^{h} \varphi_k\Big((\mathbb{Z}_{p_i})^{r_i}\Big) \qquad \text{(by Lemma 5(ii))} \qquad (3.1)\\
&\geq \prod_{i=1}^{h} \varphi_k\Big((\mathbb{Z}_{p_i})^{r}\Big) \qquad \text{(since } r \geq r_i\text{)} \qquad (3.2)\\
&= \prod_{i=1}^{h} \prod_{j=0}^{r-1} \left(1 - \frac{1}{{p_i}^{k-j}}\right) \qquad \text{(by Lemma 5(ii))} \qquad (3.3)\\
&= \prod_{i=1}^{h} \prod_{j=k-r+1}^{k} \left(1 - \frac{1}{{p_i}^{j}}\right) \qquad \text{(reindexing)}\\
&\geq \prod_{i=1}^{h} \left(1 - \sum_{j=k-r+1}^{k} \frac{1}{{p_i}^{j}}\right) \qquad \text{(Bernoulli's inequality)}\\
&= \prod_{i=1}^{h} \left(1 - \frac{1}{{p_i}^{k-r}} \cdot \frac{1 - p_i^{-r}}{p_i - 1}\right)\\
&\geq \prod_{i=1}^{h} \left(1 - \frac{1}{{p_i}^{k-r}(p_i - 1)}\right) \qquad \text{(since } 1 - p_i^{-r} < 1\text{)}\\
&\geq 1 - \sum_{i=1}^{h} \frac{1}{{p_i}^{k-r}(p_i - 1)}. \qquad \text{(Bernoulli's inequality)} \qquad (3.4)
\end{aligned}$$

However, Eq (3.4) is still difficult to directly handle. Therefore, we further estimate it as follows:

$$\begin{aligned}
1 - \sum_{i=1}^{h} \frac{1}{{p_i}^{k-r}(p_i - 1)} &\geq 1 - \sum_{p \text{ prime}} \frac{1}{p^{k-r}(p-1)}\\
&\geq 1 - \sum_{n=2}^{\infty} \frac{1}{n^{k-r}(n-1)}.
\end{aligned}$$

To ensure the inequality $\varphi_k(G) \geq 1 - \epsilon$, it suffices to bound the following:

$$1 - \sum_{n=2}^{\infty} \frac{1}{n^{k-r}(n-1)} \geq 1 - \epsilon \Longleftrightarrow \frac{1}{2^{k-r}} + \sum_{n=3}^{\infty} \frac{1}{n^{k-r}(n-1)} \leq \epsilon.$$

We establish an upper bound for the series via integral estimates. Observing that

$$\sum_{n=3}^{\infty} \frac{1}{n^{k-r}(n-1)} \leq \int_2^{\infty} \frac{1}{x^{k-r}(x-1)} dx,$$

it suffices to require the following:

$$\frac{1}{2^{k-r}} + \int_2^{\infty} \frac{1}{x^{k-r}(x-1)} dx \leq \epsilon.$$

Proceed step by step as follows:

$$\begin{aligned}
&\frac{1}{2^{k-r}} + \int_2^{\infty} \frac{1}{x^{k-r}(x-1)} dx \\
=&\frac{1}{2^{k-r}} + \int_2^{\infty} \sum_{n=1}^{\infty} \frac{1}{x^{k-r+n}} dx \quad \left(\text{since } \frac{1}{x-1} = \sum_{n=1}^{\infty} \frac{1}{x^n} \text{ for } x > 1\right) \\
=&\frac{1}{2^{k-r}} + \sum_{n=1}^{\infty} \int_2^{\infty} \frac{1}{x^{k-r+n}} dx \quad \text{(uniform convergence permits interchange)} \\
=&\frac{1}{2^{k-r}} + \sum_{n=1}^{\infty} \frac{2^{-(k-r+n-1)}}{k-r+n-1} \\
=&\frac{1}{2^{k-r}} + \sum_{i=k-r}^{\infty} \frac{1}{i \cdot 2^i} \quad \text{(letting } i = k-r+n-1\text{)}.
\end{aligned}$$

Now, observe that

$$\begin{aligned}
\sum_{i=k-r}^{\infty} \frac{1}{i \cdot 2^i} \leq \frac{1}{k-r} \sum_{i=k-r}^{\infty} \frac{1}{2^i} = \frac{1}{k-r} \cdot \frac{1/2^{k-r}}{1-1/2} = \frac{2}{(k-r) \cdot 2^{k-r}} \\
\leq \frac{1}{2^{k-r}}, \quad \text{for } k-r \geq 2.
\end{aligned}$$

Therefore, it is enough to require (with $k - r \geq 2$) the following:

$$\frac{1}{2^{k-r}} + \frac{1}{2^{k-r}} \leq \epsilon \quad \Longleftrightarrow \quad k - r \geq \log_2 \frac{2}{\epsilon}.$$

Hence, a sufficient condition is as follows:

$$k \geq r + \left\lceil \log_2 \frac{2}{\epsilon} \right\rceil, \quad \text{for } \epsilon \in (0, 1).$$

(ii)

$$\begin{aligned}
\varphi_k(G) &= \prod_{i=1}^{h} \varphi_k(G_{p_i}) \\
&= \prod_{i=1}^{h} \varphi_k\left((\mathbb{Z}_{p_i})^{r_i}\right) \qquad \text{(by Lemma 5(ii))}
\end{aligned}$$

$$\geq \prod_{i=1}^{h} \varphi_k\Big((\mathbb{Z}_{p_i})^{\mathrm{len}(G_{p_i})}\Big) \qquad (\mathrm{len}(G_{p_i}) \geq \mathrm{rank}(G_{p_i}) = r_i \text{ by Theorem 10})$$

$$\geq \prod_{i=1}^{h} \varphi_k\Big((\mathbb{Z}_2)^{\mathrm{len}(G_{p_i})}\Big) \qquad (\text{by Corollary 6})$$

$$= \prod_{i=1}^{h} \prod_{l=0}^{\mathrm{len}(G_{p_i})-1} (1 - \frac{1}{2^{k-l}}). \qquad (\text{by Lemma 5(ii)})$$

To continue the estimation, we define $L_i = \sum_{j=1}^{i} \mathrm{len}(G_{p_j})$ for $i = 0, 1, \ldots, h$, with $L_0 = 0$, thus representing the cumulative length after including the first i subgroups. Then, the total length of h subgroups is $L_h = \sum_{j=1}^{h} \mathrm{len}(G_{p_j})$. Then, we have the following:

$$\prod_{i=1}^{h} \prod_{l=0}^{\mathrm{len}(G_{p_i})-1} \left(1 - \frac{1}{2^{k-l}}\right) \geq \prod_{i=1}^{h} \prod_{l=L_{i-1}}^{L_i-1} \left(1 - \frac{1}{2^{k-l}}\right) \tag{3.5}$$

$$= \prod_{l=0}^{L_h-1} \left(1 - \frac{1}{2^{k-l}}\right) \tag{3.6}$$

$$= \prod_{l=0}^{\mathrm{len}(G)-1} \left(1 - \frac{1}{2^{k-l}}\right) \quad (\text{since } \mathrm{len}(G) = \sum_{i=1}^{h} \mathrm{len}(G_{p_i}) = L_h) \tag{3.7}$$

$$= \varphi_k\left((\mathbb{Z}_2)^{\mathrm{len}(G)}\right) \quad (\text{by Lemma 5(ii)}) \tag{3.8}$$

$$\geq 1 - \sum_{l=0}^{\mathrm{len}(G)-1} \frac{1}{2^{k-l}} \quad (\text{by Bernoulli's inequality}). \tag{3.9}$$

To ensure $\varphi_k(G) \geq 1 - \epsilon$, it suffices by Eq (3.9) to satisfy the following:

$$\sum_{l=0}^{\mathrm{len}(G)-1} \frac{1}{2^{k-l}} \leq \epsilon \quad \Longleftrightarrow \quad \frac{2^{\mathrm{len}(G)} - 1}{2^k} \leq \epsilon \quad \Longleftrightarrow \quad 2^k \geq \frac{2^{\mathrm{len}(G)} - 1}{\epsilon}.$$

This condition is satisfied when:

$$2^k \geq \frac{2^{\mathrm{len}(G)}}{\epsilon} \quad \Longleftrightarrow \quad k \geq \mathrm{len}(G) + \log_2 \frac{1}{\epsilon}.$$

Hence, a sufficient condition is as follows:

$$k \geq \mathrm{len}(G) + \left\lceil \log_2 \frac{1}{\epsilon} \right\rceil, \quad \text{for } \epsilon \in (0, 1).$$

□

Theorem 12 provides sufficient conditions, and Theorem 13 establishes their near-tightness in the worst case.

Theorem 13 (Tightness of Theorem 12)**.** *The sufficient conditions in Theorem 12(i) and (ii) are nearly tight in the worst-case sense:*

(i) *For any fixed composition length $n \geq 1$, there exists a finite nilpotent group G with* $\text{len}(G) = n$, *such that if* $k = \text{len}(G) + \left\lceil \log_2 \frac{1}{\epsilon} \right\rceil - 2$, *then we have* $\varphi_k(G) < 1 - \epsilon$.

(ii) *For any fixed rank $r \geq 1$, there exists a finite nilpotent group G with* $\text{rank}(G) = r$, *such that if* $k = \text{rank}(G) + \left\lceil \log_2 \frac{2}{\epsilon} \right\rceil - 3$, *then we have* $\varphi_k(G) < 1 - \epsilon$.

Proof. (i) Consider $G = (\mathbb{Z}_2)^n$. From part(ii) of Lemma 5, we know the following:

$$\varphi_k(G) = \varphi_k\Big((\mathbb{Z}_2)^n\Big) = \prod_{l=0}^{n-1}(1 - \frac{1}{2^{k-l}}), \quad k \geq n.$$

Substituting $k = \text{len}(G) + \left\lceil \log_2 \frac{1}{\epsilon} \right\rceil - 2$ into the above expression yields the following:

$$\begin{aligned}
\varphi_k(G) &= \prod_{l=0}^{n-1}(1 - \frac{1}{2^{k-l}}) \\
&\leq 1 - \frac{1}{2^{k-n+1}} \quad \text{(bounding via the minimum factor at } l = n-1) \\
&= 1 - \frac{1}{2^{\lceil \log_2(1/2\epsilon) \rceil}} \quad \text{(substituting } k = \text{len}(G) + \left\lceil \log_2 \frac{1}{\epsilon} \right\rceil - 2) \\
&< 1 - \frac{1}{2^{\log_2(1/2\epsilon)+1}} \quad \text{(applying } \lceil x \rceil < x + 1 \text{ for } x \in \mathbb{R}) \\
&= 1 - \frac{1}{2^{\log_2(1/\epsilon)}} \\
&= 1 - \epsilon.
\end{aligned}$$

Thus, $k = \text{len}(G) + \left\lceil \log_2 \frac{1}{\epsilon} \right\rceil - 2$ is not enough to ensure that the success probability exceeds $1 - \epsilon$.
(ii) Consider $G = (\mathbb{Z}_2)^r$. From part(ii) of Lemma 5, we know the following:

$$\varphi_k(G) = \varphi_k\Big((\mathbb{Z}_2)^r\Big) = \prod_{l=0}^{r-1}(1 - \frac{1}{2^{k-l}}), \quad k \geq r.$$

Substituting $k = \text{rank}(G) + \left\lceil \log_2 \frac{2}{\epsilon} \right\rceil - 3$ into the above expression yields the following:

$$\begin{aligned}
\varphi_k(G) &= \prod_{l=0}^{r-1}(1 - \frac{1}{2^{k-l}}) \\
&\leq (1 - \frac{1}{2^{k-r+1}}) \quad \text{(bounding via the minimum factor at } l = r-1)
\end{aligned}$$

$$= (1 - \frac{1}{2^{\lceil \log_2(1/2\epsilon) \rceil}}) \quad (\text{substituting } k = \text{rank}(G) + \left\lceil \log_2 \frac{2}{\epsilon} \right\rceil - 3)$$
$$< 1 - \frac{1}{2^{\log_2(1/\epsilon)}} \quad (\text{applying } \lceil x \rceil < x + 1 \text{ for } x \in \mathbb{R})$$
$$= 1 - \epsilon.$$

Thus, $k = \text{rank}(G) + \left\lceil \log_2 \frac{2}{\epsilon} \right\rceil - 3$ is not enough to ensure that the success probability exceeds $1 - \epsilon$. □

Remark 14.

- For a specific finite nilpotent group G, one can theoretically use Eq (3.1) to compute the exact lower bound of k for a given ϵ as follows:

$$\varphi_k(G) = \prod_{p_i \| |G|} \varphi_k\left((\mathbb{Z}_{p_i})^{r_i}\right) = \prod_{p_i \| |G|} \prod_{j=0}^{r_i - 1} \left(1 - \frac{1}{{p_i}^{k-j}}\right) \geq 1 - \epsilon.$$

  However, in practice, closed-form solutions for the exact lower bound of k are generally unavailable. They are only attainable in specific cases, such as for p-groups of rank $r \leq 4$, where the problem of determining k reduces to solving a polynomial equation of degree at most 4 in $x = p^{-k}$, which admits solutions in radicals.

  For all other cases, including higher-rank p-groups ($r \geq 5$) and general non-p-group nilpotent groups, the relevant inequality is transcendental in the variable k, and numerical methods provide a feasible approach.

- Given these computational difficulties, Theorem 12 provides an efficiently computable sufficient condition for k that universally applies to any finite nilpotent group G. The bounds

$$k \geq \text{rank}(G) + \left\lceil \log_2 \frac{2}{\epsilon} \right\rceil \quad \text{and} \quad k \geq \text{len}(G) + \left\lceil \log_2 \frac{1}{\epsilon} \right\rceil$$

  are nearly tight for the worst-case groups, as demonstrated in Theorem 13.

4. Application to analysis of quantum algorithms

4.1. Overview of hidden subgroup problem

This subsection reviews the HSP and AHSP in quantum computing, along with the definition of the orthogonal subgroup $H^{\perp}$.

Definition 5 (HSP [7,8])**.** *Let G be a group and S be a finite set. Given a black-box function $f : G \to S$, suppose there exists an unknown subgroup $H \leq G$ such that for all $x, y \in G$,*

$$f(x) = f(y) \quad \text{if and only if} \quad xH = yH,$$
$$\text{i.e., } x^{-1}y \in H \text{ or } y^{-1}x \in H.$$

The goal of the HSP is to identify the subgroup H (or a generating set for H) by querying f.

Remark 15. xH and yH are both left cosets. The function f is *constant* on left cosets of H and *distinct* across different cosets. When G is a finite abelian group, the problem is specifically referred to as the finite AHSP.

Any finite abelian group G admits a decomposition into cyclic groups. We employ the *elementary divisors* form as follows:

$$G \cong \mathbb{Z}/N_1\mathbb{Z} \oplus \mathbb{Z}/N_2\mathbb{Z} \oplus \cdots \oplus \mathbb{Z}/N_l\mathbb{Z},$$

where $N_i = p_i^{\alpha_i}$ are prime powers. Under this decomposition, elements $x \in G$ are represented as an ordered l-tuple $(x_1, \ldots, x_l)$ with $x_i \in \{0, 1, \ldots, N_i - 1\}$ for each $i \in \{1, 2, \ldots, l\}$. Now, we define the orthogonal subgroup $H^\perp$ within this framework.

Definition 6 ([9]). *Let G be a finite abelian group. For any subgroup $H \leq G$, the subgroup $H^\perp$ is defined as follows:*

$$H^\perp := \left\{ t \in G \;\middle|\; \sum_{i=1}^{l} \frac{t_i s_i}{N_i} \in \mathbb{Z} \text{ for all } s \in H \right\}.$$

Lemma 16 ([11, 17]). *Let G be a finite abelian group and $H \leq G$ be a subgroup. Then,*

(i) $H^\perp \cong G/H$;

(ii) $H^{\perp\perp} := (H^\perp)^\perp = H$.

4.2. Overview of the standard quantum algorithm for finite AHSP

We briefly recall the standard quantum algorithm for the finite AHSP [9]. Let $G \cong \mathbb{Z}_{N_1} \oplus \cdots \oplus \mathbb{Z}_{N_l}$ be a finite abelian group, $H \leq G$ be an unknown subgroup, and $f : G \to S$ be an oracle function that is constant on the cosets of H and takes distinct values on different cosets. The algorithm proceeds as follows:

1. Prepare the uniform superposition $\dfrac{1}{\sqrt{|G|}} \sum\limits_{g \in G} |g\rangle|0\rangle$;
2. Query the oracle U_f to obtain $\dfrac{1}{\sqrt{|G|}} \sum\limits_{g \in G} |g\rangle|f(g)\rangle$;
3. Apply the inverse quantum Fourier transform ($\mathrm{QFT}^\dagger$) on the first register;
4. Measure the first register, thus yielding an element $\mathbf{t} \in H^\perp$.

After the i-th iteration of the standard quantum algorithm for finite AHSP, we obtain an element $\mathbf{t}_i = (t_{i1}, t_{i2}, \ldots, t_{il}) \in H^\perp$, and all elements $\mathbf{t}_i \in H^\perp$ are equally likely to be measured [9]. After k iterations, we can only guarantee that $\langle \mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_k \rangle \subseteq H^\perp$, rather than $\langle \mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_k \rangle = H^\perp$. Then, we attempt to recover the hidden subgroup H by solving the linear congruence system as follows:

$$\mathbf{W}\mathbf{x} \in \mathbb{Z} \quad \text{where } \mathbf{W} = \begin{pmatrix} \mathbf{w}_1 \\ \vdots \\ \mathbf{w}_k \end{pmatrix} = \begin{pmatrix} t_{11}/N_1 & t_{12}/N_2 & \cdots & t_{1l}/N_l \\ \vdots & \vdots & \ddots & \vdots \\ t_{k1}/N_1 & t_{k2}/N_2 & \cdots & t_{kl}/N_l \end{pmatrix}.$$

The solution submodule $A = \{\mathbf{x} \in G \mid \mathbf{W}\mathbf{x} \in \mathbb{Z}\}$ contains $H^{\perp\perp}$. Consequently, the success probability of the algorithm after k iterations is exactly $\varphi_k(H^\perp)$, that is, the probability that k uniformly random elements generate $H^\perp$.

4.3. Application to analysis of quantum algorithm for finite AHSP

Now we present two propositions.

Proposition 17 (Additivity of Composition Length in Abelian Groups)**.** *For a finite abelian group G and subgroup $H \leq G$,*

$$\text{len}(G) = \text{len}(H) + \text{len}(H^{\perp}).$$

Proof. This result is a direct consequence of Lemma 8 in an abelian setting, where all subgroups are normal. Lemma 16 gives $H^{\perp} \cong G/H$, and since isomorphic groups have equal lengths, we obtain the equality. □

Remark 18. Proposition 17 plays a crucial role in our analysis of the standard quantum algorithm for the finite AHSP. The additivity of the composition length allows us to determine $\text{len}(H^{\perp})$ by $\text{len}(G)$ and $\text{len}(H)$. This enables us to directly determine the query complexity from $\text{len}(H^{\perp})$ rather than from $\text{len}(G)$, which can significantly reduce the number of algorithm iterations.

Proposition 19 (Subadditivity of Rank in Abelian Groups)**.** *For a finite abelian group G and subgroup $H \leq G$,*

$$\text{rank}(G) \leq \text{rank}(H) + \text{rank}(H^{\perp}).$$

Proof. This result directly follows directly from Lemma 9 in the abelian setting, where all subgroups are normal. Lemma 16 gives $H^{\perp} \cong G/H$, and since isomorphic groups have equal ranks, we obtain the following:

$$\text{rank}(G) \leq \text{rank}(H) + \text{rank}(H^{\perp}).$$

□

Remark 20. Proposition 19 demonstrates that *a priori* knowledge of $\text{rank}(H)$ does not reduce the minimal number of algorithm iterations. Indeed, the subadditivity of the rank only yields the lower bound

$$\text{rank}(G) \geq \text{rank}(H^{\perp}) \geq \text{rank}(G) - \text{rank}(H),$$

which means that $\text{rank}(H^{\perp})$ could be as large as $\text{rank}(G)$. Since the number of algorithm iterations critically depends on the exact rank of $H^{\perp}$, this inequality cannot reduce the number of iterations.

Theorem 21, which follows from Theorem 12, guarantees that $\langle \mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_k \rangle = H^{\perp}$ with probability that exceeds $1 - \epsilon$ after either $\text{rank}(G) + \lceil \log_2(2/\epsilon) \rceil$ or $\text{len}(G) - \text{len}(H) + \lceil \log_2(1/\epsilon) \rceil$ iterations. The former bound achieves an exponential improvement in the ϵ-dependence over the prior result $\lfloor 4/\epsilon \rfloor \text{rank}(G)$ [10], while the latter improves upon $\lceil \log_2 |G| + \log_2(1/\epsilon) + 2 \rceil$ [11].

If the composition length of the subgroup H is unavailable in advance, then we replace the iteration bound $k \geq \text{len}(G) - \text{len}(H) + \lceil \log_2 \frac{1}{\epsilon} \rceil$ with $k \geq \text{len}(G) + \lceil \log_2 \frac{1}{\epsilon} \rceil$.

Theorem 21. *In the quantum algorithm for the finite AHSP, to achieve a success probability of at least $1 - \epsilon$, it is sufficient to perform*

$$k \geq \text{rank}(G) + \left\lceil \log_2 \frac{2}{\epsilon} \right\rceil \quad \textit{or} \quad k \geq \text{len}(G) - \text{len}(H) + \left\lceil \log_2 \frac{1}{\epsilon} \right\rceil$$

iterations.

Proof. This directly follows from Theorem 12. Note that ensuring $\Pr(A = H) \geq 1 - \epsilon$ is equivalent to ensuring $\Pr(\langle \mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_k \rangle = H^\perp) = \varphi_k(H^\perp) \geq 1 - \epsilon$. By Theorem 12, this probability holds if we choose the following:

$$k \geq \operatorname{rank}(H^\perp) + \left\lceil \log_2 \frac{2}{\epsilon} \right\rceil \quad \text{or} \quad k \geq \operatorname{len}(H^\perp) + \left\lceil \log_2 \frac{1}{\epsilon} \right\rceil.$$

Since $\operatorname{rank}(H^\perp) \leq \operatorname{rank}(G)$ and $\operatorname{len}(H^\perp) = \operatorname{len}(G) - \operatorname{len}(H)$, we obtain the required bounds as follows:

$$k \geq \operatorname{rank}(G) + \left\lceil \log_2 \frac{2}{\epsilon} \right\rceil \quad \text{or} \quad k \geq \operatorname{len}(G) - \operatorname{len}(H) + \left\lceil \log_2 \frac{1}{\epsilon} \right\rceil.$$

□

4.4. Application to analysis of Regev's factoring algorithm

Our result contributes to establishing the requisite number of iterations for Regev's quantum factoring algorithm [12]. Regev's original analysis in [12, Corollary 4.2] shows that for a finite abelian group G, $\operatorname{rank}(G) + 4$ uniformly random elements generate G with a probability of at least $1/2$.

Since finite abelian groups are nilpotent (of class 1), we apply Theorem 12(i) with $\epsilon = 1/2$ to obtain an improved bound.

Theorem 22. *Suppose G is a finite abelian group. Then,* $\operatorname{rank}(G) + 2$ *uniformly random elements of G generate G with probability at least* $1/2$.

As a direct consequence, for an n-bit integers, the quantum circuit repetition count in Regev's factoring algorithm is optimized from $\sqrt{n}+4$ to $\sqrt{n}+2$, while maintaining the same success probability.

5. Conclusions

This paper established a quantitative sampling theorem for finite nilpotent groups. We proved that $k \geq \operatorname{rank}(G) + \lceil \log_2(2/\epsilon) \rceil$ or $k \geq \operatorname{len}(G) + \lceil \log_2(1/\epsilon) \rceil$ random elements suffice to generate G with a probability of at least $1 - \epsilon$, and demonstrated that these bounds are nearly tight. This significantly sharpens the previously known requirement of $k \geq \lceil \log_2 |G| + \log_2(1/\epsilon) \rceil + 2$.

Our theorem provides a foundational tool for the analysis of probabilistic algorithms, thereby leading to concrete improvements in quantum computation as follows:

- **Finite AHSP algorithm:** We determine the iteration count to achieve the success probability $1 - \epsilon$ in the standard quantum algorithm to be either $\operatorname{rank}(G) + \lceil \log_2(2/\epsilon) \rceil$ or $\operatorname{len}(G) - \operatorname{len}(H) + \lceil \log_2(1/\epsilon) \rceil$. The former offers an exponential improvement in ϵ-dependence over the prior bound $\lfloor 4/\epsilon \rfloor \operatorname{rank}(G)$ [10], while the latter improves upon $\lceil \log_2 |G| + \log_2(1/\epsilon) + 2 \rceil$ [11].
- **Regev's factoring algorithm:** Our result optimizes the iteration counts in Regev's recent quantum factoring algorithm [12], thereby directly reducing the quantum circuit repetition count from $\sqrt{n} + 4$ to $\sqrt{n} + 2$ while maintaining an identical success probability.

The probabilistic bounds developed here provide a foundational building block for future quantum and classical randomized algorithms, thus paving the way for applications beyond this work. We will leverage the present results to analyse and design more algorithms in a forthcoming work [18].

Author contributions

Ziyuan Dong: Conceptualization, methodology, writing (original draft), writing (reviewing); Xiang Fan: Methodology, writing (reviewing); Tengxun Zhong: Methodology; Daowen Qiu: Supervision. All authors have read and approved the final version of the manuscript for publication.

Use of Generative-AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

The authors are thankful to the anonymous referees and editor for their significant comments and suggestions that help us improve the quality of the manuscript.

This work is supported in part by the National Natural Science Foundation of China (Nos.12441107, 61876195), Guangdong Basic and Applied Basic Research Foundation (No.2020B1515310017), and Shenzhen Science and Technology Program (Grant No. JCYJ20220818102003006).

Conflict of interest

The authors declare no conflicts of interest.

References

1. A. Detinko, D. Flannery, E. O'Brien, *Probabilistic group theory, combinatorics, and computing: lectures from the Fifth de Brún Workshop*, Springer, 2013.
2. I. Pak, On probability of generating a finite group, *Preprint*, 1999. Available from: `https://www.math.ucla.edu/~pak/papers/sim.pdf`.
3. E. Detomi, A. Lucchini, F. Morini, How many elements are needed to generate a finite group with good probability? *Isr. J. Math.*, **132** (2002), 29–44. https://doi.org/10.1007/BF02784504
4. C. Pomerance, The expected number of random elements to generate a finite abelian group, *Period. Math. Hung.*, **43** (2002),191–198.
5. A. Lubotzky, The expected number of random elements to generate a finite group, *J. Algebra*, **257** (2002), 452–459. https://doi.org/10.1016/S0021-8693(02)00528-8
6. A. Lucchini, The expected number of random elements to generate a finite group, *Monatsh. Math.*, **181** (2016), 123–142. https://doi.org/10.1007/s00605-015-0789-5
7. A. Yu. Kitaev, Quantum measurements and the abelian stabilizer problem, *arXiv preprint*, 1995. Available from: `https://doi.org/10.48550/arXiv.quant-ph/9511026`.
8. M. A. Nielsen, I. L. Chuang, *Quantum computation and quantum information*, Cambridge University Press, 2010.
9. P. Kaye, *Raymond Laflamme, and Michele Mosca*, An introduction to quantum computing, OUP Oxford, 2006.

10. P. Koiran, V. Nesme, N. Portier, The quantum query complexity of the abelian hidden subgroup problem, *Theor. Comput. Sci.*, **380** (2007),115–126. https://doi.org/10.1016/j.tcs.2007.02.057
11. C. Lomont, The hidden subgroup problem - review and open problems, *arXiv preprint*, 2004. https://doi.org/10.48550/arXiv.quant-ph/0411037 .
12. O. Regev, An efficient quantum factoring algorithm, *J. ACM*, **72** (2025), 1–13. https://doi.org/10.1145/3708471
13. T. W. Hungerford, *Algebra*, Graduate texts in mathematics, **73** (1980). https://doi.org/10.1007/978-1-4612-6101-8_11
14. I. M. Isaacs, *Finite group theory*, American Mathematical Soc., **92** (2008).
15. J. D. Dixon, *Problems in group theory*, Courier Corporation, 2007.
16. V. Acciaro, The probability of generating some common families of finite groups, *Utilitas Mathematica*, **49** (1996), 243–254.
17. J. P. Serre, *Linear representations of finite groups*, Springer, **42** (1977).
18. Z. Dong, X. Fan, T. Zhong, D. Qiu, Revisiting finite Abelian hidden subgroup problem and its distributed exact quantum algorithm, *arXiv preprint*, 2025. https://doi.org/10.48550/arXiv.2512.22959

AIMS Press