Qiao YM, Sarma JMN, Tang BS. On isomorphism testing of groups with normal hall subgroups. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 27(4): 687–701 July 2012. DOI 10.1007/s11390-012-1255-7

On Isomorphism Testing of Groups with Normal Hall Subgroups

You-Ming Qiao[1] (乔友明), Jayalal Sarma M.N.[2], and Bang-Sheng Tang[1] (唐邦晟)

[1] *Institute for Theoretical Computer Science, Tsinghua University, Beijing 100084, China*

[2] *Department of Computer Science and Engineering, Indian Institute of Technology Madras, Chennai 600036, India*

E-mail: {jimmyqiao86, jayalal.sarma, bangsheng.tang}@gmail.com

Received March 13, 2011; revised February 29, 2012.

Abstract A normal Hall subgroup N of a group G is a normal subgroup with its order coprime with its index. Schur-Zassenhaus theorem states that every normal Hall subgroup has a complement subgroup, that is a set of coset representatives H which also forms a subgroup of G. In this paper, we present a framework to test isomorphism of groups with at least one normal Hall subgroup, when groups are given as multiplication tables. To establish the framework, we first observe that a proof of Schur-Zassenhaus theorem is constructive, and formulate a necessary and sufficient condition for testing isomorphism in terms of the associated actions of the semidirect products, and isomorphisms of the normal parts and complement parts. We then focus on the case when the normal subgroup is abelian. Utilizing basic facts of representation theory of finite groups and a technique by Le Gall (STACS 2009), we first get an efficient isomorphism testing algorithm when the complement has bounded number of generators. For the case when the complement subgroup is elementary abelian, which does not necessarily have bounded number of generators, we obtain a polynomial time isomorphism testing algorithm by reducing to generalized code isomorphism problem, which asks whether two linear subspaces are the same up to permutation of coordinates. A solution to the latter can be obtained by a mild extension of the singly exponential (in the number of coordinates) time algorithm for code isomorphism problem developed recently by Babai *et al.* (SODA 2011). Enroute to obtaining the above reduction, we study the following computational problem in representation theory of finite groups: given two representations ρ and τ of a group H over $\mathbb{Z}_p^d$, p a prime, determine if there exists an automorphism $\phi : H \to H$, such that the induced representation $\rho_\phi = \rho \circ \phi$ and τ are equivalent, in time $\mathrm{poly}(|H|, p^d)$.

Keywords group isomorphism, normal Hall subgroup, isomorphism testing, problem complexity

1 Introduction

The group isomorphism problem (GPI) is a computational problem intriguing for both complexity theorists as well as computational group theorists. Given two finite groups G and H, the problem asks to test if they are isomorphic, that is the existence of a bijection $\phi : G \to H$ preserving group operations, namely $\forall g, h \in G$, $\phi(g \cdot h) = \phi(g) \cdot \phi(h)$. Naturally, the complexity of the problem depends on how the group is represented: if the groups are given as presentations (generators and relations), then it is undecidable[1-2]. For permutation groups given as generators, the best upper bound known[3] is PSPACE.

The least succinct input format, multiplication table (Cayley table), gives rise to a more interesting scenario from a complexity theoretic perspective. For this case, the problem is known to be easier than the well-known graph isomorphism problem (GRI)[4], thus giving an upper bound of NP $\cap$ coAM. However, unlike many other isomorphism-type problems, a reduction in the reverse direction is not known[4]. A recent work[5] shows that GRI cannot be AC^0 reducible to GPI. Another distinction between GPI and GRI lies in the best known algorithms for them. The best known algorithm for GRI is $2^{\tilde{O}(\sqrt{n})}$[6], where n is the size of the graph. For groups of size n with b generators, in [7] Tarjan is credited for pointing out an $n^{b+O(1)}$ algorithm. Then by the observation that every group has a generating set of size $\lceil \log n \rceil$, we get an $n^{\log n + O(1)}$ algorithm for testing isomorphism of general groups. This is improved by Lipton, Snyder and Zalcstein[8], who gave an algorithm running in $O(\log^2 n)$ space. However, whether a polynomial time algorithm exists is still open.

Regular Paper

The work was supported in part by the National Natural Science Foundation of China under Grant No. 60553001, and the National Basic Research 973 Program of China under Grant Nos. 2007CB807900 and 2007CB807901.

A preliminary version of this work was presented at the 28th International Symposium on Theoretical Aspects of Computer Science (STACS), March 2011.

1.1 Progress on Testing Isomorphism of Restricted Group Classes

There has been some progress on group isomorphism problem for restricted classes of groups. The class of groups with bounded number of generators (say, of size b) can be tested efficiently by the $n^{b+O(1)}$ algorithm. For abelian groups, Savage[9] first gave an $O(n^2)$ algorithm, which was improved to $O(n \log n)$ by Vikas[10] and finally to $O(n)$ by Kavitha[11]. Little is known beyond abelian groups until 2008, when Le Gall[12] showed that isomorphism of groups in the form of semidirect products of an abelian group and a cyclic group, whose orders are coprime, can be tested in almost linear time even in the model of black-box groups. The class of p-groups seems to be the current barrier, though recent works by Wilson[13-14] on the structure of p-groups are noteworthy.

Recently, Kayal and Nezhmetdinov[15] and Wilson[16] address the problem of finding the factors of a group under the direct product operation (Wilson[16] considers a stronger model, that is permutation groups given as generators). They show that given a group, all its direct factors can be computed efficiently. As pointed out in [15], this result can be interpreted in the context of isomorphism testing as follows: by Remak-Krull-Schmidt theorem, two groups are isomorphic if and only if their direct factors are isomorphic up to appropriate correspondence of the factors. Thus, the class of groups that are direct products of groups with known efficient isomorphism testing procedure can be tested efficiently.

This argument suggests the following strategy: suppose for some group class, the groups can be decomposed into smaller subgroups in some canonical way. Then after decomposition, isomorphism testing of the original groups may reduce to testing isomorphism of the building blocks, and then pasting solutions of building blocks back together. In the case of direct product, decomposition is solved in [15] and [16], and "pasting" is trivial due to Remak-Krull-Schmidt theorem. Now it is natural to ask if this strategy can be extended to the case of less stringently defined products. The next natural target is that of semidirect product, which is already considered in [12]. A group G is the semidirect product of a normal subgroup N by a subgroup H if $G = NH$ and $N \cap H = \{\mathrm{id}\}$. Every $h \in H$ can act on N by conjugation, giving rise to a homomorphism from H to $\mathrm{Aut}(N)$, called the action associated with the semidirect product. Unlike direct product, a semidirect product $G = N \rtimes_\tau H$ is canonical only with respect to the associated action. For the special class considered in [12], due to this reason Le Gall needs to solve the problem of testing whether two automorphisms of abelian groups are conjugate or not (when the automorphisms satisfy some property), for which he gives an efficient algorithm.

1.2 Our Result: Framework for Testing Isomorphism of Groups with Normal Hall Subgroups

A Hall divisor m of an integer n is a divisor of n such that $(m, n/m) = 1$. A normal Hall subgroup is a normal subgroup whose order is a Hall divisor of the order of the group. In this paper, we consider the class of groups with at least one normal Hall subgroup, and use $\mathcal{H}$ to denote this group class. It turns out this condition suggests some interesting properties of the group structure. For a given Hall divisor of the size of the group, if the normal Hall subgroup of this size exists then it is a characteristic subgroup. Schur-Zassenhaus theorem states that a normal Hall subgroup always has a complement, that is a set of representatives forming a subgroup. Thus the semidirect product arises naturally for groups in $\mathcal{H}$. Note that $\mathcal{H}$ contains all groups of order $2 \cdot p^k$, p a prime other than 2, and all nilpotent groups that are not p-groups. To see the first point, note that a Sylow p-subgroup is normal as it is of index 2, and the second point follows due to that a nilpotent group is direct product of its Sylow subgroups.

Inspired by [12], we begin with formalizing the strategy for isomorphism testing discussed in Subsection 1.1 for the class $\mathcal{H}$. As a first step, we need to have an efficient decomposition procedure. The observation is that the proof of Schur-Zassenhaus theorem is efficiently constructive, establishing the following theorem about finding a complement of a normal Hall subgroup.

Theorem 1 (Algorithmic Schur-Zassenhaus Theorem). *For a group G of order n, given as multiplication table, all its normal Hall subgroups can be computed in time $O(n^4)$. Given a specific normal Hall subgroup, one of its complements can be computed in time $O(n^4)$.*

In the second step, we need to consider how isomorphism of the original groups connects isomorphisms of the components. One important tool, which has been discovered by Taunt[17] in the context of construction of finite groups, is the formulation of a necessary and sufficient condition of the original groups being isomorphic in terms of actions associated with the semidirect products and isomorphisms of the normal and complement parts. (We state the precise formulation in Theorem 7). From the condition, it is necessary that we are able to test isomorphism of the components, thus, we focus on the case when the factors of semidirect product are efficiently testable. The following notations will help us talk about the group classes of the factors in the semidirect product. Given two groups X

and Y whose orders are coprime, $\mathcal{H}(X,Y)$ is the class of groups with a normal Hall subgroup isomorphic with X, and a complement isomorphic with Y. For two group classes $\mathcal{X}$ and $\mathcal{Y}$, $\mathcal{H}(\mathcal{X},\mathcal{Y})$ is the class of groups with a normal Hall subgroup X from $\mathcal{X}$ and the complement Y from $\mathcal{Y}$. Note that X being a Hall subgroup implies that the orders of X and Y are coprime. That is $\mathcal{H}(\mathcal{X},\mathcal{Y}) = \bigcup_{X\in\mathcal{X},Y\in\mathcal{Y},\gcd(|X|,|Y|)=1} \mathcal{H}(X,Y)$.

We set notations for some group classes with known isomorphism testing/computing procedure. Let $\mathcal{A}$ be the class of abelian groups. As subclasses of $\mathcal{A}$, $\mathcal{A}_p$ is the class of abelian p-groups, and $\mathcal{E}$ is the class of elementary abelian groups. $\prod\mathcal{E}$ is the class of direct products of elementary abelian groups. $\mathcal{B}_b$ is the class of groups with the number of generators bounded by b. Note that $\mathcal{B}_2$ includes all finite simple groups①, symmetric groups and cyclic groups. When the specific number of generators is not required in the context, we drop the subscript and will simply write $\mathcal{B}$. $\mathcal{C} = \mathcal{B}_1$ is the class of cyclic groups. Finally, let $\mathcal{K}$ denote any group from the class of groups for which an efficient isomorphism testing/computing procedure is known. In this article, we mainly consider the case when $\mathcal{K}$ is $\mathcal{A}$ or $\mathcal{B}$, or subclasses of $\mathcal{A}$ or $\mathcal{B}$. To give an example of the use of the notations, the main result of [12] is an efficient isomorphism testing/computing algorithm of $\mathcal{H}(\mathcal{A},\mathcal{C})$, while our main concrete results are efficient algorithms for $\mathcal{H}(\mathcal{A},\mathcal{B})$ (when the complement has bounded number of generators), and $\mathcal{H}(\mathcal{A},\mathcal{E})$ (when the complement is elementary abelian).$\mathcal{H}(\mathcal{A},\mathcal{B})$ includes the class $\mathcal{H}(\mathcal{A},\mathcal{C})$ studied in [12].

1.3 Our Result: Efficient Isomorphism Testing of $\mathcal{H}(\mathcal{A},\mathcal{E})$, $\mathcal{H}(\mathcal{A},\mathcal{B})$

Representation theory of finite groups studies the homomorphisms from abstract groups to general linear groups. Such a homomorphism is called a representation. When the normal subgroup is an elementary abelian group $\mathbb{Z}_p^d$, p a prime, the condition of isomorphism of large groups with respect to the components (cf. Theorem 7) naturally gives rise to the following algorithmic problem in representation theory of finite groups which may be of independent interest. We call it AutoInducedRepEquiv, short for finding the AUTOmorphism INDUCED REPresentation EQUIValence.

Problem 1 (AutoInducedRepEquiv). *Given two representations ρ and τ of a group H over $\mathbb{Z}_p^d$, p a prime, determine if there exists an automorphism $\phi : H \to H$, such that the induced representation $\rho_\phi = \rho\circ\phi$ and τ are equivalent, in time* $\mathrm{poly}(|H|, p^d)$.

The following theorem suggests that AutoInducedRepEquiv cannot be got around in order to solve isomorphism of groups from $\mathcal{H}(\mathcal{E},\mathcal{K})$.

Theorem 2. *For groups from $\mathcal{H}(\mathcal{E},\mathcal{K})$, isomorphism testing is many-one equivalent to* AutoInducedRepEquiv.

Using basic facts from representation theory, it is not hard to solve AutoInducedRepEquiv when the number of generators is bounded, giving an efficient testing algorithm of $\mathcal{H}(\mathcal{E},\mathcal{B})$. The non-trivial case is when the number of generators is not bounded. When the complement is an elementary abelian group, we further reduce AutoInducedRepEquiv to a mild generalization② of the linear code isomorphism problem in singly exponential time, which asks whether two linear subspaces are the same up to permutation of coordinates in time exponential to the number of coordinates.

Theorem 3. *For groups from $\mathcal{H}(\mathcal{E},\mathcal{E})$,* AutoInducedRepEquiv *reduces to generalized code isomorphism problem.*

In a recent work[19] (see also [20]), Babai presents an algorithm solving the code isomorphism problem in singly exponential time in the number of coordinates, which is logarithmic of the size of the group in our case, allowing us to establish the following.

Corollary 1. *There is an $O(n^6)$ algorithm testing isomorphism of groups from $\mathcal{H}(\mathcal{E},\mathcal{E})$.*

It is worth noting that the number of groups in this class is lower bounded by $n^{\Omega(\log n)}$, for certain infinite sequence of group size n. (cf. Appendix A1).

Applying a technique in [12], we extend this further to provide an efficient isomorphism testing of groups from $\mathcal{H}(\mathcal{A},\mathcal{E})$. An $O(n^{b+5})$ algorithm for $\mathcal{H}(\mathcal{A},\mathcal{B}_b)$ can also be derived in this framework, rediscovering what is known in Subsection 8.9, [21] (see Subsection 4.2).

Theorem 4. *For groups of size n from $\mathcal{H}(\mathcal{A},\mathcal{E})$, there is an algorithm in time $O(n^6)$ testing isomorphism.*

The rest of the paper is organized as follows. Section 2 contains the preliminaries. In Section 3 we present the decomposition procedure into normal and complement parts, proving Theorem 1. In Section 4, we first present the condition that shows how testing isomorphism of the original groups relates to that of the small groups. Then we prove Theorem 2, elaborate on the framework, and show that how a technique from [12] allows us to reduce from $\mathcal{H}(\prod\mathcal{E},\mathcal{E})$ to $\mathcal{H}(\mathcal{A},\mathcal{E})$. Finally, in Section 5, we introduce generalized code

①For readers unfamiliar with this fact, cf. the first theorem in [18], and note that a simple abelian group must be a cyclic group with prime order.

②See Section 5 for specific points of generalization.

isomorphism, the reductions (Theorem 3) and show how to test isomorphism of $\mathcal{H}(\mathcal{A}, \mathcal{E})$. Section 6 concludes the paper.

2 Preliminaries

In this section we introduce some preliminary concepts and notations that we will be using. We refer the reader to a standard text book[22] for basic concepts in Group theory.

An abelian group is a group with group operation commutative. Given a prime p, an abelian p-group is an abelian group of order p^k, $k \in \mathbb{Z}^+$, and an elementary abelian p-group is of the form $\mathbb{Z}_p^k$. Every abelian group can be decomposed as direct product of cyclic groups by the fundamental theorem of abelian groups.

Let G be a group and N be a normal subgroup of G (denoted by $N \lhd G$). We say that G is the *semidirect product* of N by H, $H \leqslant G$, written as $G = N \rtimes H$, if $G = NH$ and $N \cap H = \{\mathrm{id}\}$. For a given decomposition of $G = N \rtimes H$, we call N the *normal subgroup* of this decomposition, and H the *complement subgroup*. For a given $N \lhd G$, from the definition of semidirect product it can be seen that $G = N \rtimes H$ if and only if there is a set of coset representatives of G/N closed under group operation. We use C_h^N to denote the automorphism of N induced by h by conjugating action. Formally, $C_h^N : N \to N$ by $n \to hnh^{-1}$. This gives an homomorphism of $\tau : H \to \mathrm{Aut}(N)$, by sending h to C_h^N. When we write $G = N \rtimes_\tau H$, τ is the associated homomorphism from H to $\mathrm{Aut}(N)$ acting by conjugation. Conversely, given two groups N and H, and a homomorphism $\tau : H \to \mathrm{Aut}(N)$ (we will use τ_h to denote the image of h under τ), a group G can be formed as follows: elements in G are from $N \times H$, and we let $(n, h) \cdot (n', h') = (n\tau_h(n'), hh')$. This gives a construction of (outer) semidirect product $G = N \rtimes_\tau H$.③

Theorem 5 (Schur-Zassenhaus Theorem, cf. [22]). *Let G be a finite group of order n, and m is a Hall divisor of n. If there exists $N \lhd G$, $|N| = m$, then we have $H \leqslant G$ such that $G = N \rtimes H$. If H and H' are two complements of N, then H and H' are conjugate.*

Representation Theory of Finite Groups. We list basic notions and facts about representation theory of finite groups, and we refer the reader to a standard text book[23] for further details.

For a finite group G and a vector space V, a *representation* of G over V is a group homomorphism $\phi : G \to \mathrm{GL}(V)$. There is always a trivial representation by mapping every element in G to 1. If the underlying field of V is $\mathbb{F}$, and V is of finite dimension d, a homomorphism $\phi : G \to \mathrm{GL}(d, \mathbb{F})$ is called a representation of G over $\mathbb{F}$ of dimension d. For a given representation $\phi : G \to \mathrm{GL}(d, \mathbb{F})$, a subspace of V, L is an *invariant subspace*, or a *sub-representation* if $\forall g \in G$, $\phi_g(L) = L$. $\mathbf{0}$ and V are called trivial invariant subspaces. A representation without non-trivial invariant subspaces is called an *irreducible representation*. If ϕ and ρ are representations of a group G over spaces V and W (over a field $\mathbb{F}$), then the direct sum $\phi \oplus \rho$ is the representation of G over $V \oplus W$ defined as: $(\phi \oplus \rho)_g(\boldsymbol{u} + \boldsymbol{v}) := \phi_g(\boldsymbol{u}) + \rho_g(\boldsymbol{v})$ for $g \in G$. A representation is completely reducible if it is a direct sum of irreducible representations. Maschke's theorem states that if characteristic of $\mathbb{F}$ is 0 or coprimes with $|G|$, then the representation over $\mathbb{F}$ is completely reducible.

Two representations $\phi : G \to \mathrm{GL}(V)$ and $\psi : G \to \mathrm{GL}(V)$ are equivalent if there exists a general linear map $T : V \to V$ such that $\phi(g) = T\psi(g)T^{-1}$ for every $g \in G$. A fact about completely reducible representations is that two representations are equivalent if and only if irreducible representations (up to equivalence) that appear in their decompositions are the same. Specifically, decomposing a representation gives for every irreducible representation (up to equivalence) its multiplicity in that representation, and two representations are equivalent if and only if for every irreducible representation the multiplicities are the same. For a representation $\phi : G \to \mathrm{GL}(\mathbb{F}, d)$, and $i \in [d]$, let $L_\phi(i)$ be the set of irreducible representations with multiplicity i in the decomposition ϕ, and $L_\phi = (L_\phi(i))_{i \in [d]}$. We say $L_\phi = L_\psi$ if and only if $L_\phi(i) = L_\psi(i)$ for every $i \in [d]$.

We use this straightforward criterion to test whether a representation is irreducible. For $\boldsymbol{v} \in V$, let $g\boldsymbol{v}$ denote the action of the representation of g on $\boldsymbol{v}$.

Proposition 1. *Let $\phi : G \to GL(V)$ be a representation. ϕ is irreducible if and only if $\forall \boldsymbol{v} \in V$, $\boldsymbol{v} \neq \mathbf{0}$, $\langle g\boldsymbol{v} \mid g \in G \rangle = V$.*

Theorem 6 (Maschke's Theorem. Adaptation of Theorem 1 of [23], page 6). *Let $\phi : G \to GL(\mathbb{F}, d)$ be a representation,* $\gcd(|G|, \mathrm{char}(\mathbb{F})) = 1$. *$W \leqslant V$ is a sub-representation of V. Let $p : V \to W$ be a projection of V onto W, and the image of $p' = \frac{1}{|G|} \sum_{g \in G} \phi(g) \circ p \circ \phi(g^{-1})$ be W'. Then W' is a sub-representation and $V = W \oplus W'$.*

Proposition 1 and Theorem 6 suggest the following procedure to decompose a representation into its irreducible components. Let $\phi : G \to \mathrm{GL}(V)$ be a representation. For every $\boldsymbol{v} \in V$, test if $\langle g\boldsymbol{v} \mid g \in G \rangle$ generates V. If so, it is an irreducible representation. Otherwise, for a specific $\boldsymbol{v}$, $\langle g\boldsymbol{v} \mid g \in G \rangle$ is a sub-representation W. Then Theorem 6 helps to identify a sub-representation

③Note that actually $G = N' \rtimes_\tau H'$, where $N' = \{(n, 1) \mid n \in N\}$ and $H' = \{(1, h) \mid h \in H\}$. τ also maps H' to $\mathrm{Aut}(N')$ naturally. As this is a simple embedding, for convenience we write $G = N \rtimes_\tau H$.

W' such that $V = W \oplus W'$. Recursively using the above procedure on W and W' decomposes V into its irreducible components. This gives:

Proposition 2. *Given a representation $\phi : G \to \mathrm{GL}(V)$, where V is a vector space over a finite field, the irreducible components of ϕ can be listed in time $O(dim(V)^2 \cdot |V| \cdot |G|)$.*

Proposition 2 is sufficient for our purpose. But we remark that, in general, the decomposition of modular representation (representations over fields of finite characteristic) can be done much more efficiently (cf. [24] and Chapter 7.4 of [21]). Given two irreducible representations, there is an efficient algorithm to determine whether they are equivalent (cf. [21], Chapter 7.5.3). For factoring polynomials of degree n over $\mathbb{Z}_p$, we use the $O(p^{1/2}(\log p)^2 n^{2+\varepsilon})$ algorithm in [25]. For computing canonical normal form of a linear transformation, Steel's algorithm[26] in time $O(n^4)$ suffices.

Permutation Group Algorithm and Coset Intersection Problem. The most studied model in Computational Group theory[27] is permutation groups given as generators. It subsumes the Cayley table representation since every group can be viewed as a permutation group acting on the group itself. Let S_n be the symmetric group of order n. A coset of a permutation group H, Hx is represented as a set of generators of H and a coset representative. When we apply permutation group algorithms, or when the input or output is clearly a permutation group or a coset of permutation group, we assume they are given as above. In particular, the output of the singly exponential algorithm for code isomorphism problem in [19] (see also [20]) is given as such. Given two cosets Hx and $H'y$, where H and H' are permutation groups given as generators acting on a domain of size n, to compute their intersection (another coset) is called the *coset intersection problem.* The best known algorithm computing coset intersection is an $\exp(n^{1/2+o(1)})$ presented in [28], while in this paper the bound 2^n suffices for us. We refer the reader to [29] for a relatively simple $\exp(O(n))$ algorithm.

3 Decomposition into Normal and Complement Parts

In this section we describe that for a given group, all its normal Hall subgroups and their complements can be listed, proving Theorem 1, by providing the following two propositions.

Proposition 3. *Let G be a group of size n. For a Hall divisor m, if a normal Hall subgroup of order m exists then it can be computed in time $O(n^3)$.*

Proposition 4. *Let G be a group of order n, and N a normal Hall subgroup of order m. Then a complement of N can be found in time $O(n^4)$.*

The two propositions give a natural way of listing the normal Hall subgroups and their complements: for a given Hall divisor m of the group size n, compute the normal Hall subgroup of size m by Proposition 3 if it exists. Then compute its complement by Proposition 4. Going over all Hall divisors lists all normal Hall subgroups and their complements.

3.1 Listing Normal Hall Subgroups: Proof of Proposition 3

Lemma 1. *If a group $G = \{g_1, \ldots, g_n\}$ can be written as $G = N \rtimes H$, where N is a normal Hall subgroup, and $|H| = l$. Then $N = M \doteq \langle g_1^l, g_2^l, \ldots, g_n^l \rangle$.*

Proof.

$\supseteq$: Any g_i can be written as $n_i h_i$ for some $n_i \in N$, $h_i \in H$, so $g_i^l = (n_i h_i)^l = n_i' h_i^l = n_i'$, for some $n_i' \in N$.

$\subseteq$: Since $\gcd(n/l, l) = 1$, $\exists p, q \in \mathbb{Z}$, s.t. $p(n/l) + ql = 1$. For any $x \in N$, $x^l \in M$, thus $x^{ql} \in M$. Note that $\mathrm{id} = x^{n/l} = x^{p(n/l)}$, as $|N| = n/l$. Now we have

$$x = x^{p(n/l)+ql} = x^{p(n/l)} x^{ql} = x^{ql} \in M. \qquad \square$$

For a given group G of size n, Lemma 1 implies that for a given Hall divisor m of n, if a normal Hall subgroup of size m exists then it is unique (thus a characteristic subgroup). For any group G given as multiplication table, and a given subset $S \subseteq G$, $\langle S \rangle$ can be computed in time $O(n^3)$. Indeed, we can do this in stages by simply computing all pairwise product of elements in the closure produced at the previous stage and stop exactly when no new elements are produced. Along with this, Lemma 1 implies Proposition 3.

3.2 Finding the Complement: Proof of Proposition 4

We state the following algorithmic problem called SubgroupComplement.

Problem 2 (SubgroupComplement). *Given a group G and $N \lhd G$, decide if there is a complement $H \leqslant G$ such that $G = N \rtimes H$. If there exists, compute H.*

When N is abelian, this problem can be solved even in the model of permutation groups given as generators[30], but beyond abelian we are not aware of any result. For the case when N is a normal Hall subgroup of G, Schur-Zassenhaus theorem states that a complement always exists, thus solving the decision version of SubgroupComplement. Our observation is that, the proof of Schur-Zassenhaus theorem is efficiently constructive when groups are represented as Cayley tables, thus providing an efficient algorithm solving the search version of SubgroupComplement when the normal subgroup is a Hall subgroup.

3.2.1 When the Normal Subgroup is Abelian

In this subsection we use addition as the operation in the normal subgroup.

The correctness of the algorithm follows from the proof of Schur-Zassenhaus theorem[22], which is adapted slightly and put in Appendix A2.1. Let us analyze the time complexity of Algorithm 1. It is easy to get a set of representatives in $O(n)$ time. To compute f there are $O(l^2)$ computations involved. To get p and q only $O(\max(\log m, \log l))$ is needed, and σ needs $O(l^2)$ computations. Thus the algorithm runs in $O(\max(l^2, n)) = O(n^2)$.

Algorithm 1. Finding the Complement When the Normal Subgroup $N \lhd G$ is Abelian, AbelianComplement(G, N).

Input: Group G of size n given as multiplication table. A normal Hall abelian subgroup $N \lhd G$. $|N| = m$, $[G : N] = l = n/m$, $(m, l) = 1$.

Output: $H \leqslant G$ such that $G = N \rtimes H$.

1: Let $A = \{a_1 = \mathrm{id}, \ldots, a_l\}$ be any representative set of G/N, and let $\phi : G \to A$ be the natural map.
2: Compute $f : A \times A \to N$ by $f(a_i, a_j) = \phi(a_i a_j) \cdot (a_i a_j)^{-1}$.
3: Let p, q be two integers such that $pm + ql = 1$. Compute $\sigma : A \to N$ by $\sigma(a) = q \cdot \sum_{b \in A} f(a, b)$.
4: **return** $H = \{\sigma(a_i) \cdot a_i\}_{a_i \in A}$.

3.2.2 When the Normal Subgroup is not Abelian

In this subsection we show the algorithm that computes the complement when the normal subgroup is not abelian. The idea is to transform the proof of Schur-Zassenhaus theorem into a recursive algorithm. The correctness of the algorithm follows from the proof of Schur-Zassenhaus theorem[22], and we give an analysis of the proof strategy in Appendix A2.2. Before the actual algorithm we need to set up some auxiliary algorithms which basically follow from definitions. We present the algorithms in Appendix A2.3 and only introduce their functionality and running time here. Algorithm A1 computes the normalizer of a given set $S \subseteq G$ within G. Algorithm A2 tests whether a given subgroup N of G is a minimal normal subgroup. Algorithm A3 finds a Sylow p-subgroup given a group G and a prime p dividing $|G|$. For groups of size n, the first subroutine runs in time $O(n^2)$, the second and third ones can be verified to be running in $O(n^4)$ and $O(\log n \cdot n^3)$, respectively.

Now we analyze Algorithm 2. The base case (normal subgroup being abelian) runs in time $O(n^2)$. As explained in Appendix A2.2, if N is not abelian and is a minimal normal subgroup of G, then the p-Sylow subgroup G' is a proper subgroup of G and hence the size of the group reduces by at least one half. If N is not a minimal subgroups, then the algorithm makes two recursive calls, one with respect to G/T and N/T, the other with respect to K and T. By Correspondence theorem, K is a subgroup. So $|G|/|T|$, $|K| \leqslant |G|/2$. Thus the time complexity follows from the expression $T(n) = 2T(n/2) + O(n^4)$, which gives the running time to be $O(n^4)$.

Algorithm 2. Finding the Complement of $N \lhd G$, ComputeComplement(G, N).

Input: Group G of size n given as multiplication table. A normal Hall subgroup $N \lhd G$. $|N| = m$, $[G : N] = n/m$, $(m, n/m) = 1$.

Output: $H \leqslant G$ such that $G = N \rtimes H$.

1: **if** N is abelian **then**
2: **return** AbelianComplement(G, N). {Base case.}
3: **end if**
4: {N is non-abelian}
5: **if** MinimalNormal$(G, N) =$ true **then**
6: For some p that divides $|N|$, $P \leftarrow$ Sylow(N, p).
7: $G' \leftarrow$ Normalizer(G, P) {$G' < G$, see Appendix A2.2}.
8: $N' \leftarrow$ Normalizer(N, P).
9: **return** ComputeComplement(G', N').
10: **else**
11: $T \leftarrow$ MinimalNormal(G, N).
12: $K/T \leftarrow$ ComputeComplement$(G/T, N/T)$.
13: **return** ComputeComplement(K, T).
14: **end if**

4 Condition for Isomorphism Testing

The next theorem shows how isomorphism of big groups reduces to that of components for groups with normal Hall subgroups. This has been discovered by Taunt[17] in the context of construction of finite groups, though he did not apply it to normal Hall subgroups explicitly (because the fact that complement subgroups are conjugate was not proved then). We present the proof here as this theorem is crucial to further development of this work.

Theorem 7 (Theorem 3.3, [17]). *Given $G_1 = N_1 \rtimes_\tau H_1$, $G_2 = N_2 \rtimes_\gamma H_2$, with $|N_1| = |N_2|$, $|H_1| = |H_2|$. N_1 and N_2 are normal Hall. Then $G_1 \cong G_2$ if and only if there exist an isomorphism $\psi : N_1 \to N_2$, and an isomorphism $\phi : H_1 \to H_2$, such that, $\forall h \in H_1$,*

$$\tau(h) = \psi^{-1} \circ \gamma(\phi(h)) \circ \psi. \tag{1}$$

Proof. $\Rightarrow$: Since $G_1 \cong G_2$, let $f : G_1 \to G_2$ be an isomorphism. As f preserves normal groups, and N_2 is the unique normal subgroup of G_2 with order

$|N_2|$, $f(N_1) = N_2$. Thus $f(H_1)$ is another complement of N_2. By Theorem 5, two different complements of a normal Hall subgroup are conjugate, so there exists $a \in G_2$ so that $af(H_1)a^{-1} = H_2$. Denote I_a by the inner automorphism of G_2 induced by a, and $\hat{f} = I_a \circ f$. $\hat{f} : G_1 \to G_2$ is an isomorphism, and $\hat{f}(N_1) = N_2$, $\hat{f}(H_1) = H_2$. Let $\psi = \hat{f}|_{N_1}$, and $\phi = \hat{f}|_{H_1}$. Now we verify the third condition. For any $n \in N_1$, $(\psi^{-1} \circ \gamma(\phi(h)) \circ \psi)(n) = (\hat{f}^{-1} \circ \gamma(\hat{f}(h)) \circ \hat{f})(n) = \hat{f}^{-1}(\hat{f}(h)\hat{f}(n)\hat{f}^{-1}(h)) = hnh^{-1} = \tau(h)(n)$.

$\Leftarrow$: Given $g \in G_1$, since it can be written as nh, for $n \in N_1$ and $h \in H_1$, we define $f : G_1 \to G_2$ by $nh \to \psi(n)\phi(h)$. f is bijection, as $f(nh) = f(n'h') \Rightarrow \psi(n)\phi(h) = \psi(n')\phi(h') \Rightarrow \psi(n) = \psi(n')$ and $\phi(h) = \phi(h')$, which yields $n = n'$ and $h = h'$. f is a homomorphism since

$$\begin{aligned} f(nhn'h') =& \psi(n\tau_h(n'))\phi(hh') \\ =& \psi(n)\psi(\tau_h(n'))\phi(h)\phi(h') \\ =& \psi(n)\phi(h)(\phi(h)^{-1}\psi(\tau_h(n'))\phi(h))\phi(h') \\ =& \psi(n)\phi(h)((\gamma(\phi(h^{-1})) \circ \psi \circ \tau_h)(n'))\phi(h') \\ =& \psi(n)\phi(h)\psi(n')\phi(h') \quad \text{(By (1))} \\ =& f(nh)f(n'h'). \end{aligned}$$ □

Remark 1. *From the algorithmic point of view, a first observation about Theorem* 7 *is that we only need to test* (1) *for a generating set of H_1, rather than all of H_1, as τ, γ, ϕ and ψ are all homomorphisms.*

4.1 Proof of Theorem 2

Theorem 2 states that isomorphism of $\mathcal{H}(\mathcal{E}, \mathcal{K})$ is equivalent to AutoInducedRepEquiv. In this subsection we show the two reductions here.

Isomorphism of Groups in $\mathcal{H}(\mathcal{E}, \mathcal{K})$ to AutoInducedRepEquiv. By listing all normal Hall subgroups and their complements we can find two normal Hall subgroups of the same size from two groups. Thus in order to test isomorphism of the original pair of groups, we first solve the isomorphism problem for normal and complement parts. Given the isomorphisms of the normal and complement parts, the only task left is to test (1), which, by composing the isomorphisms of the normal and complement parts, gives an instance of the problem AutoInducedRepEquiv. This gives the reduction.

AutoInducedRepEquiv *to Isomorphism of Groups in $\mathcal{H}(\mathcal{E}, \mathcal{K})$.* In Section 2 we described the standard construction that, given groups N, H and $\tau : H \to \mathrm{Aut}(N)$, defines a group $G = N \rtimes_\tau H$. Thus, given two representations τ and γ of H over $\mathbb{Z}_p^k$, we can construct $G_1 = \mathbb{Z}_p^k \rtimes_\tau H$ and $G_2 = \mathbb{Z}_p^k \rtimes_\gamma H$, and then call the oracle to test if G_1 and G_2 are isomorphic. By Theorem 7, the two representations are equivalent up to automorphism action if and only if G_1 and G_2 are isomorphic. This gives the reduction.

4.2 Framework for Testing Isomorphism of Groups from $\mathcal{H}(\mathcal{K}, \mathcal{K})$

Suppose we want to test isomorphism of two groups G_1 and G_2 from $\mathcal{H}(\mathcal{K}, \mathcal{K})$. Given Theorem 1, for any group all its normal Hall subgroups can be listed efficiently, so we can first compare the orders of the normal Hall subgroups of G_1 and G_2, and output "not isomorphic" if there are no normal Hall subgroups of the same size. For normal Hall subgroups with the same order, compute their complements using Proposition 4. Suppose we decompose $G_1 = N_1 \rtimes H_1$ and $G_2 = N_2 \rtimes H_2$, with $|N_1| = |N_2|$. As the normal and complement parts are from groups with known isomorphism computing procedure, run the isomorphism tests between N_1, N_2 and H_1, H_2. If they are not isomorphic output "not isomorphic". Now the only task left is to test (1). Recall that $\prod \mathcal{E}$ denotes the class of direct products of elementary abelian groups. The cases $\mathcal{H}(\mathcal{E}, \mathcal{B})$ and $\mathcal{H}(\prod \mathcal{E}, \mathcal{B})$ are immediate: for $\mathcal{H}(\mathcal{E}, \mathcal{B})$, the automorphisms of complements can be efficiently enumerated and they correspond to the ϕ in (1). For a given automorphism of the complement, the problem is to test if two representations are equivalent. It can be solved by decomposing the representations, and then noticing that equivalence of irreducible representations can be determined efficiently. For $\mathcal{H}(\prod \mathcal{E}, \mathcal{B})$, like in $\mathcal{H}(\mathcal{E}, \mathcal{B})$, as the automorphisms of the complement can be enumerated, for a given automorphism, the problem is to test if the representations over the direct factors of the normal subgroup are equivalent. These instances can be solved separately.

We remark that when the complement is in $\mathcal{B}$, to find the complement it is easy to come up with an efficient enumeration procedure (without using algorithmic Schur-Zassenhaus). From the above discussion, the difficult case is when the complement has no generating set of size $O(1)$.

4.3 From $\mathcal{H}(\prod \mathcal{E}, \mathcal{K})$ to $\mathcal{H}(\mathcal{A}, \mathcal{K})$: Le Gall's Technique

Let A be an abelian p-group with minimal generating set of size s, and let ϕ_1, $\phi_2 \in \mathrm{Aut}(A)$ such that $p \nmid |\phi_i|$, $i = 1, 2$. In [12], Le Gall devised a map $\Lambda_p : \mathrm{Aut}(A) \to \mathrm{GL}(\mathbb{F}_p, s)$, such that ϕ_1 and ϕ_2 are conjugate if and only if $\Lambda_p(\phi_1)$ and $\Lambda_p(\phi_2)$ are conjugate. In [31], the same reduction was proved to work for $S_1, S_2 \leqslant \mathrm{Aut}(A)$, as explained formally in the following lemma. We refer it as *Le Gall's technique* in this paper,

and explain it in detail in Appendix A3.

Lemma 2 (Le Gall's Technique). *For a given abelian p-group A with minimal generating set of size s, let $S_1, S_2 \leqslant \mathrm{Aut}(A)$, given by a set of generators. Assume that $p \nmid |S_i|$, for $i = 1, 2$. Then there exists a map $\Lambda_p : \mathrm{Aut}(A) \to \mathrm{GL}(\mathbb{F}_p, s)$, such that S_1 and S_2 are conjugate in $\mathrm{Aut}(A)$ if and only if $\Lambda_p(S_1)$ and $\Lambda_p(S_2)$ are conjugate in $\mathrm{GL}(\mathbb{F}_p, s)$.*

We show that Le Gall's technique allows us to reduce testing isomorphism of $\mathcal{H}(\mathcal{A}, \mathcal{K})$ to that of $\mathcal{H}(\prod \mathcal{E}, \mathcal{K})$. For convenience we first explain how Le Gall's technique allows us to reduce isomorphism of $\mathcal{H}(\mathcal{A}_p, \mathcal{K})$ to $\mathcal{H}(\mathcal{E}, \mathcal{K})$. Let G_1 and G_2 be decomposed as $N_1 \rtimes_\tau H_1$ and $N_2 \rtimes_\gamma H_2$, where N_1 and N_2 are abelian p-groups. Then decompose N_1 and N_2 into the canonical form, and identify H_1 and H_2 as isomorphic. Now by Theorem 7, we need to test if there exist $\psi \in \mathrm{Aut}(N_1)$, and $\phi \in \mathrm{Aut}(H)$, such that $\tau(h)$ and $\gamma(\phi(h))$ are conjugate by the same ψ, for every $h \in H$. This is equivalent to test if there exist ψ and ϕ such that $\tau(H)$ and $\gamma(\phi(H))$ are conjugate by ψ as subgroups in $\mathrm{Aut}(N_1)$. Noting that $p \nmid |H|$, Lemma 2 can be applied, reducing the problem to test whether $\Lambda_p(\tau(H))$ and $\Lambda_p(\gamma(\phi(H)))$ are conjugate. Note that $\Lambda_p \circ \tau$ sends H to $\mathrm{GL}(\mathbb{F}_p, s)$, where s is the size of the minimal generating set of H. This finishes the reduction. To go from $\mathcal{H}(\mathcal{A}, \mathcal{K})$ to $\mathcal{H}(\prod \mathcal{E}, \mathcal{K})$ we need to consider the factors of $\prod \mathcal{E}$ separately and apply the appropriate Λ_p to each factor.

5 Isomorphism of $\mathcal{H}(\mathcal{A}, \mathcal{E})$

The main result of this section is a reduction of the isomorphism testing problem for groups in $\mathcal{H}(\mathcal{A}, \mathcal{E})$ to the problem of generalized code isomorphism problem. We first introduce this problem. Let $\mathbb{F}$ be a field. For $\mathbb{F}^n$, a linear code of dimension d is a d-dimensional linear subspace of $\mathbb{F}^n$. A generating matrix of a code $\boldsymbol{C}$ of dimension d is a d by n matrix with row vectors being a basis of $\boldsymbol{C}$. With abuse of notation we will also use $\boldsymbol{C}$ to denote the generating matrix of the code $\boldsymbol{C}$. Two codes $\boldsymbol{C}$ and $\boldsymbol{D}$ of dimension d over $\mathbb{F}$ are isomorphic if they are equivalent up to permutation of coordinates. Formally, if there exists a d by d non-singular matrix $\boldsymbol{G}$ and an n by n permutation matrix $\boldsymbol{P}$ such that $\boldsymbol{GCP} = \boldsymbol{D}$. It is noted that linear code isomorphism problem is hard for Graph Isomorphism problem[32], and the following singly exponential algorithm presented by Babai in [19] (see also [20]) is the main algorithmic tool we shall use.

Theorem 8 ([19], see also [20]). *Given two linear codes $\boldsymbol{C}$ and $\boldsymbol{D}$, presented as generating matrices, it can be tested if they are isomorphic, and the coset of the isomorphism can be computed in time $(2 + o(1))^n$.*

We generalizes code isomorphism problem slightly to get:

Problem 3 (Generalized Code Isomorphism Problem). *Given two $d' \times n$ matrices $\boldsymbol{C}'$ and $\boldsymbol{D}'$ over the field $\mathbb{F}$, and a permutation group $S \leqslant S_n$, determine the existence of $\boldsymbol{G} \in \mathrm{GL}(\mathbb{F}, d')$ and a permutation matrix $\boldsymbol{P} \in S$, such that $\boldsymbol{GC'P} = \boldsymbol{D'}$. If such $\boldsymbol{G}$ and $\boldsymbol{P}$ exist, then $\boldsymbol{C}'$ and $\boldsymbol{D}'$ are called isomorphic.*

The generalized code isomorphism problem generalizes code isomorphism problem in two ways: first we do not require row vectors of $\boldsymbol{C}'$ and $\boldsymbol{D}'$ to be linearly independent. Secondly the permutation matrix $\boldsymbol{P}$ must come from a certain permutation group S. Its solution in singly exponential time can be viewed as a corollary to Theorem 8, by applying a coset intersection running in singly exponential time[28]. A detailed proof of the following corollary can be found in Appendix A4.

Corollary 2. *Given two $d' \times n$ matrices $\boldsymbol{C}'$ and $\boldsymbol{D}'$, and a permutation group S, whether $\boldsymbol{C}'$ and $\boldsymbol{D}'$ are isomorphic, and if so the coset of permutation matrices can be computed in time $(2 + o(1))^n$.*

5.1 Representation of $\mathbb{Z}_q^\ell$ over $\mathbb{Z}_p$

In this subsection, we recall basic facts concerning representations of $\mathbb{Z}_q^\ell$ over $\mathbb{Z}_p$, p, q two different primes, and we refer the reader to Appendix A5 for more detailed explanations. First suppose the cyclotomic polynomial $\Phi_q(x)$ factors as $g_1 \cdot g_2 \cdot \ldots \cdot g_r$ over $\mathbb{Z}_p$, in which g_i's are monic polynomials with the same degree $d = (q-1)/r$. It is noted that d is the order of p in the multiplicative group $(\mathbb{Z}/q\mathbb{Z})^\times$. Let $\boldsymbol{M} \in \mathrm{GL}(\mathbb{Z}_p, d)$ be the companion matrix of g_1[④]. For $\boldsymbol{v} \in \mathbb{Z}_q^\ell$, $v \neq \boldsymbol{0}$, we define $\boldsymbol{v}^* : \mathbb{Z}_q^\ell \to \mathbb{Z}_q$ by mapping $\boldsymbol{v}^*(\boldsymbol{u}) = (\boldsymbol{v}, \boldsymbol{u})$ (the inner product of $\boldsymbol{v}$ and $\boldsymbol{u}$). Now define $f_{\boldsymbol{v}} : \mathbb{Z}_q^\ell \to \mathrm{GL}(\mathbb{Z}_p, d)$ by sending $\boldsymbol{u} \to \boldsymbol{M}^{\boldsymbol{v}^*(\boldsymbol{u})}$. To unify notation let $f_{\boldsymbol{0}} : \mathbb{Z}_q^\ell \to \mathbb{Z}_p$ be the trivial representation. Then $f_{\boldsymbol{v}}$ gives an irreducible representation of $\mathbb{Z}_q^\ell$ over $\mathbb{Z}_p$, and $\{f_{\boldsymbol{v}} \mid \boldsymbol{v} \in V\}$ is the set of all irreducible representations. However, $f_{\boldsymbol{v}}$ and $f_{\boldsymbol{u}}$ may be equivalent, for $\boldsymbol{u}, \boldsymbol{v} \in V$, as described by the following claims, whose proofs are put in Appendix A5.

Claim 1. *Let $f_{\boldsymbol{v}}$ and $f_{\boldsymbol{u}}$ be two irreducible representations of $\mathbb{Z}_q^\ell$ over $\mathbb{Z}_p$ induced by $\boldsymbol{v}, \boldsymbol{u} \in \mathbb{Z}_q^\ell$, $\boldsymbol{v}, \boldsymbol{u} \neq \boldsymbol{0}$ as above. $f_{\boldsymbol{v}}$ and $f_{\boldsymbol{u}}$ are equivalent if and only if $\boldsymbol{u} = s\boldsymbol{v}$ for $s \in \mathbb{Z}_q$, and M^s and M are conjugate.*

Corollary 3. *Let $S_{p,q}$ be the set of s satisfying the condition in Claim 1, and d be the order of p in the multiplicative group $(\mathbb{Z}/q\mathbb{Z})^\times$. Then $|S_{p,q}| = d$.*

Let $\tau : \mathbb{Z}_q^\ell \to \mathrm{GL}(\mathbb{Z}_p, k)$ be a representation. Due to Maschke's theorem, representations of $\mathbb{Z}_q^\ell$ over $\mathbb{Z}_p$

[④] In fact, any d by d matrix with characteristic polynomial as g_1 would suffice, and it does not matter if we choose, say companion matrix of g_i, for any $i \in [r]$.

are completely reducible. Suppose $\tau = f_{\boldsymbol{v}_1}^{k_1} \oplus \ldots \oplus f_{\boldsymbol{v}_t}^{k_t}$, for $\boldsymbol{v}_i \in V$, $i \in [t]$, $k_1 \geqslant \ldots \geqslant k_t \geqslant 1$. Note that t is bounded by $1 + \lfloor (k-1)/d \rfloor$ or k/d, depending on whether the trivial representation exists or not. For a given multiplicity $w \in [k]$, recall that $L_\tau(w)$ is the set of irreducible representations with multiplicity w appearing in τ, and $L_\tau = (L_\tau(w))_{w\in[k]}$ determines a representation up to equivalence. The problem of working with L_τ is that the irreducible representations are "abstract", while we need to actually know the form of the irreducible representations. The idea is to use vectors to index irreducible representations (which must be induced from vectors as described before), at the cost of losing uniqueness.

Definition 1. *Given a representation $\tau : \mathbb{Z}_q^\ell \to \mathrm{GL}(\mathbb{Z}_p, k)$, and $w \in [k]$, $\mathcal{L}_\tau(w)$ is a set of vectors such that for every irreducible representation $f \in L_\tau(w)$, there is a unique vector $\boldsymbol{v} \in \mathcal{L}_\tau(w)$ such that $f_{\boldsymbol{v}}$ and f are equivalent. $\mathcal{L}_\tau = (\mathcal{L}_\tau(w))_{w\in[k]}$. Such a tuple of sets of vectors is called an indexing tuple of L_τ.*

Remark 2. *By Corollary 3, the number of different indexing tuples of L_τ is bounded by $d^{k/d} \leqslant (e^{1/e})^k < 2^k$. (Note that we do not need to consider $f_{\mathbf{0}}$.)*

For two representations $\tau : \mathbb{Z}_q^\ell \to \mathrm{GL}(\mathbb{Z}_p, k)$ and $\gamma : \mathbb{Z}_q^\ell \to \mathrm{GL}(\mathbb{Z}_p, k)$, τ and γ are equivalent if and only if $L_\tau = L_\gamma$. For two indexing tuples $\mathcal{L}_\tau$ and $\mathcal{L}_\gamma$ of τ and γ, we also use $\mathcal{L}_\tau = \mathcal{L}_\gamma$ to denote the fact that for every $w \in [k]$, $\mathcal{L}_\tau(w) = \mathcal{L}_\gamma(w)$. An immediate consequence is the following claim.

Claim 2. *Let $\tau : \mathbb{Z}_q^\ell \to \mathrm{GL}(\mathbb{Z}_p, k)$ and $\gamma : \mathbb{Z}_q^\ell \to \mathrm{GL}(\mathbb{Z}_p, k)$ be two representations. τ and γ are equivalent if and only if there exist indexing tuples of τ and γ, $\mathcal{L}_\tau$ and $\mathcal{L}_\gamma$, such that $\mathcal{L}_\tau = \mathcal{L}_\gamma$.*

The induced representation of $f_{\boldsymbol{v}}$ by $\phi \in \mathrm{GL}(\mathbb{Z}_q, l)$ has a nice form: $(f_{\boldsymbol{v}} \circ \phi)(\boldsymbol{u}) = f_{\boldsymbol{v}}(\phi(\boldsymbol{u})) = M^{v^*(\phi(\boldsymbol{u}))} = M^{(\phi^{\mathrm{T}}(\boldsymbol{v}))^*(\boldsymbol{u})} = f_{\phi^{\mathrm{T}}(\boldsymbol{v})}(\boldsymbol{u})$. That is $f_{\boldsymbol{v}} \circ \phi = f_{\phi^{\mathrm{T}}(\boldsymbol{v})}$. Note that for any two representations g and h of an arbitrary group G and $\phi' \in \mathrm{Aut}(G)$, $(g \oplus h) \circ \phi' = (g \circ \phi') \oplus (h \circ \phi')$. If follows that $\tau \circ \phi = f_{\phi^{\mathrm{T}}(\boldsymbol{v}_1)}^{k_1} \oplus \ldots \oplus f_{\phi^{\mathrm{T}}(\boldsymbol{v}_t)}^{k_t}$. For $\phi \in \mathrm{GL}(\mathbb{Z}_q, l)$, and $S \subseteq \mathbb{Z}_q^\ell$, S^ϕ is the set obtained by applying ϕ^{T} to every vector in S. Thus $\mathcal{L}_{\tau\circ\phi} = \mathcal{L}_\tau^\phi \doteq (\mathcal{L}_\tau(w)^\phi \mid w \in [k])$.

5.2 Isomorphism of $\mathcal{H}(\mathcal{E}, \mathcal{E})$: Proof of Theorem 3

To test isomorphism of two groups G_1 and G_2 identified as $\mathbb{Z}_p^k \rtimes_\tau \mathbb{Z}_q^\ell$ and $\mathbb{Z}_p^k \rtimes_\gamma \mathbb{Z}_q^\ell$, by Theorem 2 we can view τ and γ as two representations of $\mathbb{Z}_q^\ell$ over $\mathbb{Z}_p$ of dimension k. Then we need to solve AutoInducedRepEquiv problem for τ and γ. This is done, as shown in Theorem 3, by reducing to generalized code isomorphism problem.

Since τ and γ are equivalent if and only if $L_\tau = L_\gamma$, using Proposition 2 we decompose τ and γ as $\tau = f_{\boldsymbol{v}_1}^{k_1} \oplus \ldots \oplus f_{\boldsymbol{v}_t}^{k_t}$ and $\gamma = f_{\boldsymbol{u}_1}^{\ell_1} \oplus \ldots \oplus f_{\boldsymbol{u}_{t'}}^{\ell_{t'}}$ to get two specific indexing sets $\mathcal{L}_\tau$ and $\mathcal{L}_\gamma$. Along with the decomposition, we can calculate the change of basis matrices S and T, such that the images of $S(\tau \circ \phi)S^{-1}$ and $T\gamma T^{-1}$ are sets of block diagonal matrices with blocks representing the irreducible representations. Also note that for a specific irreducible representation, it is easy to identify an indexing vector of it, by examining which vector maps to M, the companion matrix of some predetermined factor of $\Phi_q(x)$ over $\mathbb{Z}_p$.

Given the decomposition, we first need to test if $t = t'$, and $|\mathcal{L}_\tau(w)| = |\mathcal{L}_\gamma(w)|$, $\forall w \in [k]$. If the conditions are not satisfied τ and γ cannot be equivalent under automorphism. For now assume that the conditions are satisfied. By $\mathcal{L}_{\tau\circ\phi} = \mathcal{L}_\tau^\phi$, we know the indexing tuple of $\mathcal{L}_{\tau\circ\phi}$ applies ϕ^{T} to the vectors in $\mathcal{L}_\tau$. From a specific indexing tuple $\mathcal{L}_\tau$, all indexing tuples of L_τ can be enumerated based on Claim 1. From Remark 2, we can afford the enumeration of all indexing tuples. Finally, by Claim 2, the only task left is to determine whether there exists $\phi \in \mathrm{GL}(\mathbb{Z}_p, \ell)$, such that $\mathcal{L}_\tau^\phi$ is a specific indexing tuple of L_γ, in time $\mathrm{poly}(p^k, q^\ell)$, where $p^k \cdot q^\ell$ is the size of the original group.

Proposition 5. *Testing the existence of ϕ so that of $\mathcal{L}_\tau^{\phi^{\mathrm{T}}} = \mathcal{L}_\gamma$ in time $\mathrm{poly}(p^k, q^\ell)$ reduces to generalized code isomorphism problem in singly exponential time.*

Proof. Expand $\mathcal{L}_\tau = (\mathcal{L}_\tau(1), \ldots, \mathcal{L}_\tau(k))$ as $(\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{s_1}\}, \{\boldsymbol{v}_{s_1+1}, \ldots, \boldsymbol{v}_{s_2}\}, \ldots, \{\boldsymbol{v}_{s_{k-1}+1}, \ldots, \boldsymbol{v}_{s_k}\})$ in which $s_1 \leqslant s_2 \leqslant \cdots \leqslant s_k = t$.

Similarly expand $\mathcal{L}_\gamma$ and $\mathcal{L}_\tau^{\phi^{\mathrm{T}}}$ respectively as $(\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{s_1}\}, \{\boldsymbol{u}_{s_1+1}, \ldots, \boldsymbol{u}_{s_2}\}, \ldots, \{\boldsymbol{u}_{s_{k-1}+1}, \ldots, \boldsymbol{u}_{s_k}\})$ and $(\{\phi(\boldsymbol{v}_1), \ldots, \phi(\boldsymbol{v}_{s_1})\}, \{\phi(\boldsymbol{v}_{s_1+1}), \ldots, \phi(\boldsymbol{v}_{s_2})\}, \ldots, \{\phi(\boldsymbol{v}_{s_{k-1}+1}), \ldots, \phi(\boldsymbol{v}_{s_k})\})$.

Thus, testing $\mathcal{L}_\tau^{\phi^{\mathrm{T}}} = \mathcal{L}_\gamma$ can be done by finding $\phi \in \mathrm{GL}(\mathbb{Z}_q, \ell)$ and $\sigma \in S_{s_1} \times S_{s_2-s_1} \times \ldots \times S_{s_k-s_{k-1}}$ such that $\phi(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_t)\sigma = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_t)$. This is just generalized code isomorphism problem with the permutation group $S_{s_1} \times S_{s_2-s_1} \times \ldots \times S_{s_k-s_{k-1}}$, whose generators can be computed as symmetric groups can be generated by two elements. The reduction takes time $\mathrm{poly}(k, \ell)$. □

Thus the solution for generalized code isomorphism in singly exponential time gives the algorithm for AutoInducedRepEquiv for elementary abelian groups, finishing the proof of Theorem 3.

5.3 Isomorphism of $\mathcal{H}(\mathcal{A}, \mathcal{E})$: Proof of Theorem 4

The idea for $\mathcal{H}(\mathcal{E}, \mathcal{E})$ can be extended to $\mathcal{H}(\prod \mathcal{E}, \mathcal{E})$, as follows. Suppose we have G_1 and G_2 identified as $(\prod_{i\in[s]} \mathbb{Z}_{p_i}^{k_i}) \rtimes \mathbb{Z}_q^\ell$, with the associated actions as τ and

γ, respectively. Now we need to test if there exist $\psi \in \prod_{i\in[s]} \mathrm{GL}(\mathbb{Z}_{p_i}, k_i)$ and $\phi \in \mathrm{GL}(\mathbb{Z}_q, l)$ such that $\tau(h) = \psi^{-1}\circ\gamma(\phi(h))\circ\psi$, for every $h \in \mathbb{Z}_q^\ell$. Let $\tau_i : H_1 \to \mathrm{GL}(\mathbb{Z}_{p_i}, k_i)$ be the projection of τ into the i-th component, and similarly we have $\gamma_i : H_2 \to \mathrm{GL}(\mathbb{Z}_{p_i}, k_i)$. This reduces to testing for every $i \in [s]$, if $\tau_i(h)$ and $\gamma_i(\phi(h))$ are conjugate by $\psi_i \in \mathrm{GL}(\mathbb{Z}_{p_i}, k_i)$, for every $h \in \mathbb{Z}_q^\ell$. Viewing τ_i's and γ_i's as representations and going through the decomposition into irreducibles, we get $\mathcal{L}_{\tau_i}$'s and $\mathcal{L}_{\gamma_j}$'s and similarly we need to determine if there exists $\phi \in \mathrm{GL}(\mathbb{Z}_q, l)$ such that $\mathcal{L}_{\tau_i}^{\phi^{\mathrm{T}}} = \mathcal{L}_{\gamma_i}$, for every $i \in [s]$. Now it is enough to group $\mathcal{L}_{\tau_i}$'s and $\mathcal{L}_{\gamma_j}$'s respectively, and view them as a single generalized code isomorphism instance. Finally, Le Gall's technique gives an efficient algorithm for groups from $\mathcal{H}(\mathcal{A}, \mathcal{E})$.

Putting Together: The Algorithm. In this subsection we put together the ideas from the above discussion to derive an algorithm testing isomorphism of groups from $\mathcal{H}(\mathcal{A}, \mathcal{E})$. We first need a subroutine testing if two given groups G_1 and G_2 from $\mathcal{H}(\mathcal{A}, \mathcal{E})$ can be decomposed as $N_1 \rtimes H_1$ and $N_2 \rtimes H_2$, such that $N_1 \cong N_2 \in \mathcal{A}$, and $H_1 \cong H_2 \in \mathcal{E}$ or not. This can be done by listing all normal subgroups and utilizing the algorithm of finding the structure of abelian groups, in time $O(n^6)$. Name the above subroutine as TestStructure, returning (N_1, H_1, N_2, H_2) if such a tuple exists, and false otherwise. We also need Ψ_p as in Lemma 2, and $S_{p,q} \subseteq \mathbb{Z}_q$ as in Claim 1. The algorithm is summarized using pseudocode in Algorithm 3.

Algorithm 3. Test Isomorphism of Groups from $\mathcal{H}(\mathcal{A}, \mathcal{E})$.
Input: G_1, G_2 from $\mathcal{H}(\mathcal{A}, \mathcal{E})$, as multiplication tables.
Output: An isomorphism if $G_1 \cong G_2$. False otherwise.

1: **if** TestStructure$(G_1, G_2) =$ false **then**
2: **return** false.
3: **else**
4: $(N_1, H_1, N_2, H_2) \leftarrow$ TestStructure(G_1, G_2).
5: **end if**
6: Decompose N_1 and N_2 as $A_{p_1} \times \ldots \times A_{p_s}$.
7: Identify H_1 and H_2 as $\mathbb{Z}_q^\ell$.
8: **for all** $i \in [s]$ **do**
9: $k_i \leftarrow$ size of basis of A_{p_i}.
10: Compute $\tau_i : H_1 \to \mathrm{Aut}(A_{p_i})$, $\gamma_i : H_2 \to \mathrm{Aut}(A_{p_i})$.
11: Compute $T_i = \Psi_{p_i} \circ \tau_i$, $\Gamma_i = \Psi_{p_i} \circ \gamma_i$. {$T_i$ and Γ_i are representations.}
12: Decompose T_i and Γ_i to get $\mathcal{L}_{T_i} = (\mathcal{L}_{T_i}(w) \mid w \in [k_i])$ and $\mathcal{L}_{\Gamma_i} = (\mathcal{L}_{\Gamma_i}(w) \mid w \in [k_i])$.
13: $P_i \leftarrow \prod_{w\in[k_i]} S_{|\mathcal{L}_{T_i}(w)|}$, $Q_i \leftarrow \prod_{w\in[k_i]} S_{|\mathcal{L}_{\Gamma_i}(w)|}$. {Permutations within groups of vectors.}
14: **if** $P_i \neq Q_i$ {That is, for some $w \in [k_i]$, $|\mathcal{L}_{T_i}| \neq |\mathcal{L}_{\Gamma_i}|$.} **then**
15: **return** false.
16: **end if**
17: **end if**
18: $\mathcal{L}_T \leftarrow (\mathcal{L}_{T_1}, \ldots, \mathcal{L}_{T_s})$, $\mathcal{L}_\Gamma \leftarrow (\mathcal{L}_{\Gamma_1}, \ldots, \mathcal{L}_{\Gamma_s})$.
19: $P \leftarrow \prod_{i\in[s]} P_i = \prod_{i\in[s]} Q_i$.
20: **for all** possible indexing tuples of $\mathcal{L}_T$ **do**
21: **if** GenCodeIsom$(\mathcal{L}_T, \mathcal{L}_\Gamma, P) \neq \varnothing$ **then**
22: $(\phi, \sigma) \leftarrow$ GenCodeIsom$(\mathcal{L}_T, \mathcal{L}_\Gamma, P)$.
23: **for all** $i \in [s]$ **do**
24: Calculate conjugacies $\psi_i' \in \mathrm{GL}(\mathbb{Z}_{p_i}, k_i)$ in (1) from σ.
25: Calculate the $\psi_i \in \mathrm{Aut}(A_{p_i})$ from ψ_i' by Remark 2.
26: **end for**
27: $\psi \leftarrow \mathrm{diag}_{i\in[s]}(\psi_i)$.
28: **return** the isomorphism of G_1 to G_2 from (ϕ, ψ) by Theorem 7.
29: **end if**
30: **end for**
31: **return** false.

The analysis of Algorithm 3 goes as follows. Let $|G_1| = |G_2| = n$, we first note that the iterations in step 8 and in step 20 are bounded by s and $2^{\sum_{i\in[s]} k_i}$, which are bounded by $\log n$ and n, respectively. Then the following routines are used. In step 6, we need to decompose an abelian group in canonical form. This can be done in time $O(n)$ by [33]. In step 12, to decompose a representation requires $O(n^4)$ by Proposition 2. To identify the indexing vectors for irreducible components we need to factor Φ_q over different fields, and test conjugation of matrices of size at most q. These two operations can be done in time $O(p \cdot q^2)$ and $O(q^4)$, bounded by $O(n^3)$ and $O(n^4)$, respectively. GenCodeIsom takes time $2^{\sum_{i\in[s]} k_i}$ bounded by n. Other steps can be verified to be computed in at most $O(n^2)$ time. To summarize we have an $O(n^6)$ algorithm testing isomorphism of group class $\mathcal{H}(\mathcal{A}, \mathcal{E})$.

6 Conclusions

In this paper we design polynomial-time algorithm for groups in the following form: firstly it has an abelian normal Hall subgroup; secondly, that subgroup has a complement with bounded number of generators, or elementary abelian. These results greatly expand the previous work of Le Gall[12]. Furthermore, we put the study of this group class (groups with abelian normal Hall subgroup) in a representation-theoretic framework, which lies the foundation for further research.

Acknowledgements Part of this work was done while the second author was a postdoctoral fellow at the Institute for Theoretical Computer Science at Tsinghua University, Beijing, China. Part of the work was done while You-Ming Qiao was visiting the University of Chicago, and he would like to thank Laci Babai and Sasha Razborov for hosting him. You-Ming would also like to thank J. L. Alperin and James B. Wilson for several useful discussions. The authors are also grateful

to Laci Babai for sharing the results in [19] (see [20]).

References

[1] Dehn M. Über unendliche diskontinuierliche gruppen. *Mathematische Annalen*, 1911, 71: 116-144.

[2] Adian S. The unsolvability of certain algorithmic problems in the theory of groups. *Trudy Moskov. Math. Obshch*, 1957, 6: 231-298.

[3] Babai L, Szemerédi E. On the complexity of matrix group problems *i*. In *Proc. IEEE Annual Symposium on Foundations of Computer Science* (*FOCS*), West Palm Beach, Florida, USA, October 24-26, 1984, pp.229-240.

[4] Köbler J, Schöning U, Torán J. The Graph Isomorphism Problem: Its Structural Complexity. Boston: Birkhauser, 1993.

[5] Chattopadhyay A, Torán J, Wagner F. Graph isomorphism is not AC^0 reducible to group isomorphism. In *Proc. Annual Conference on Foundations of Software Technology and Theoretical Computer Science* (*FSTTCS 2010*), Chennai, India, Dec. 15-18, 2010, pp.317-326.

[6] Babai L, Luks E M. Canonical labeling of graphs. In *Proc. the 15th Annual ACM Symposium on Theory of Computing* (*STOC*), Boston, Massachusetts, USA, April 25-27, 1983, pp.171-183.

[7] Miller G L. On the $n\log n$ isomorphism technique. In *Proc. the 10th Annual ACM Symposium on Theory of Computing*, San Diego, California, USA, May 1-3, 1978, pp.51-58.

[8] Lipton R J, Snyder L, Zalcstein Y. The complexity of word and isomorphism problems for finite groups. Technical Report, John Hopkins, 1976.

[9] Savage C. An $O(n^2)$ algorithm for abelian group isomorphism. Technical Report, North Carolina State University, 1980.

[10] Vikas N. An $O(n)$ algorithm for abelian p-group isomorphism and an $O(n \log n)$ algorithm for abelian group isomorphism. *Journal of Computer and System Sciences*, 1996, 53(1): 1-9.

[11] Kavitha T. Linear time algorithms for abelian group isomorphism and related problems. *Journal of Computer and System Sciences*, 2007, 73(6): 986-996.

[12] Gall F L. Efficient isomorphism testing for a class of group extensions. In *Proc. the 26th International Symposium on Theoretical Aspects of Computer Science* (*STACS 2009*), Freiburg, Germany, February 26-28, 2009, pp.625-636.

[13] Wilson J B. Decomposing p-groups via Jordan algebras. *Journal of Algebra*, 2009, 322(8): 2642-2679.

[14] Wilson J B. Finding central decompositions of p-groups. *Journal of Group Theory*, 2009, 12(6): 813-830.

[15] Kayal N, Nezhmetdinov T. Factoring groups efficiently. In *Proc. the 36th International Colloquium on Automata, Languages and Programming* (*ICALP 2009*), Rhodes, Greece, July 5-12, 2009, pp.585-596.

[16] Wilson J B. Finding direct product decompositions in polynomial time. 2010. http://arxiv.org/pdf/1005.0548.pdf.

[17] Taunt D R. Remarks on the isomorphism problem in theories of construction of finite groups. *Mathematical Proceedings of the Cambridge Philosophical Society*, 1955, 51(1): 16-24.

[18] Menegazzo F. The number of generators of a finite group. *Irish Math. Soc. Bulletin*, 2003, 50: 117-128.

[19] Babai L. Equivalence of linear codes. Technical Report, University of Chicago, 2010.

[20] Babai L, Codenotti P, Grochow J, Qiao Y. Towards efficient algorithm for semisimple group isomorphism. In *Proc. ACM-SIAM Annual Symposium of Discrete Algorithms* (*SODA*), San Francisco, California, USA, January 23-25, 2011.

[21] Holt D F, Eick B, O'Brien E A. Handbook of Computational Group Theory. London: Chapman and Hall/CRC, 2005.

[22] Rotman J J. An Introduction to the Theory of Groups (4th edition). Springer-Verlag, 1995.

[23] Serre J P. Linear Representations of Finite Groups. New York: Springer-Verlag, 1977.

[24] Rónyai L. Computing the structure of finite algebras. *Journal of Symbolic Computation*, 1990, 9(3): 355-373.

[25] Shoup V. On the deterministic complexity of factoring polynomials over finite fields. *Information Processing Letters*, 1990, 33(5): 261-267.

[26] Steel A. A new algorithm for the computation of canonical forms of matrices over fields. *Journal of Symbolic Computation*, 1997, 24(3-4): 409-432.

[27] Seress A. Permutation Group Algorithms. Cambridge: Cambridge University Press, 2003.

[28] Babai L. Coset intersection in moderately exponential time. *Chicago Journal of Theoretical Computer Science*, to appear.

[29] Luks E M. Hypergraph isomorphism and structural equivalence of Boolean functions. In *Proc. the 31st Annual ACM Symposium on Theory of Computing*, Atlanta, Georgia, USA, May 1-4, 1999, pp.652-658.

[30] Kantor W M, Luks E M, Mark P D. Sylow subgroups in parallel. *Journal of Algorithms*, 1999, 31(1): 132-195.

[31] Babai L, Qiao Y. Polynomial-time isomorphism test for groups with abelian Sylow towers. In *Proc. the 29th International Symposium on Theoretical Aspects of Computer Science*, Pairs, France, Feb. 28-March 3, 2012, pp.453-464.

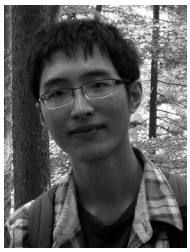
[32] Petrank E, Roth R M. Is code equivalence easy to decide? *IEEE Trans. Information Theory*, 1997, 43(5): 1602-1604.

[33] Buchmann J, Schmidt A. Computing the structure of a finite abelian group. *Mathematics of Computation*, 2005, 74(252): 2017-2026.

[34] Ranum A. The group of classes of congruent matrices with application to the group of isomorphisms of any abelian group. *Transactions of the American Mathematical Society*, 1907, 8(1): 71-91.

You-Ming Qiao is a fourth year Ph.D. candidate at Institute for Interdisciplinary Information Sciences, Tsinghua University, China. Before joining this institute, he received his Bachelor's degree of Engineer from Department of Computer Science and Technology, Tsinghua University in 2008. He is interested in complexity theory and algebraic computation, with an emphasis on the interplay with group theory. His advisor is Prof. Andrew Yao.

Jayalal Sarma M.N. is currently an assistant professor at the Department of Computer Science and Engineering, Indian Institute of Technology Madras, Chennai, India. He received his doctorate degree in 2008 from the Institute of Mathematical Sciences, Chennai, India. Later he spent two years at the Institute of Theoretical Computer Science as a postdoctoral researcher hosted by Prof. Andrew Yao. His research interests are in computational and circuit complexity theory and algebraic and combinatorial problems related to them.



Bang-Sheng Tang received his bachelor's degree from Department of Computer Science and Technology, Tsinghua University in 2008. After that he joined Institute of Theoretical Computer Science (later became one part of Institute for Interdisciplinary Information Sciences) to pursuit his Ph.D. degree. Now he is a Ph.D. candidate in his fourth year, supervised by Prof. Periklis Papakonstantinou and co-supervised by Prof. Andrew Yao. His research interests are complexity theory and algorithm design, and also computational group theory.

Appendix

A1 Lower Bound on the Number of Groups in $\mathcal{H}(\mathcal{E},\mathcal{E})$

In this subsection we give an $n^{\Omega(\log n)}$ lower bound on the number of non-isomorphic groups in the class $\mathcal{H}(\mathcal{E},\mathcal{E})$, for some sequence of n.

Consider $\mathbb{Z}_p^k \rtimes \mathbb{Z}_q^\ell$. Let $k = 2\ell$. The primes p and q are fixed, and chosen so that Φ_q factors into linear terms over $\mathbb{Z}_p$. So k and ℓ are of order $\Theta(\log n)$, where $n = p^k \cdot q^\ell$. The groups of the form $\mathbb{Z}_p^k \rtimes \mathbb{Z}_q^\ell$ are associated with homomorphisms from $\mathbb{Z}_q^\ell$ to $\mathrm{GL}(\mathbb{Z}_p, k)$, and two groups $\mathbb{Z}_p^k \rtimes_\rho \mathbb{Z}_q^\ell$ and $\mathbb{Z}_p^k \rtimes_\gamma \mathbb{Z}_q^\ell$ are isomorphic if and only if ρ and γ are isomorphic up to automorphism, by Theorem 7. Break ρ into the irreducibles. Note that an irreducible representation is indexed by a unique vector in $\mathbb{Z}_q^\ell$ due to the choice of p and q. Furthermore assume the irreducibles of ρ and γ are all of multiplicity 1. Suppose ρ breaks into $f_{\boldsymbol{v}_1} \oplus \ldots \oplus f_{\boldsymbol{v}_k}$, and γ breaks into $f_{\boldsymbol{u}_1} \oplus \ldots \oplus f_{\boldsymbol{u}_k}$, where $\boldsymbol{v}_i$, $\boldsymbol{u}_j \in \mathbb{Z}_q^\ell$. Let $\boldsymbol{V} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k)$, and $\boldsymbol{U} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_k)$. $\boldsymbol{V}$ and $\boldsymbol{U}$ are two $\ell \times k$ matrices over $\mathbb{Z}_q$. Thus $\mathbb{Z}_p^k \rtimes_\rho \mathbb{Z}_q^\ell$ and $\mathbb{Z}_p^k \rtimes_\gamma \mathbb{Z}_q^\ell$ are isomorphic if and only if the row vectors of $\boldsymbol{V}$ and $\boldsymbol{U}$ generate the same subspace of $\mathbb{Z}_q^k$. First note that the number of ℓ-dimension subspaces of $\mathbb{Z}_q^k$ is lower bounded by $q^{\ell^2/4}$, while $\ell = \Theta(\log n)$. By permuting the coordinates, at most $k! = n^{\log\log n}$ different subspaces can be generated. So the number of groups in $\mathbb{Z}_p^k \rtimes \mathbb{Z}_q^\ell$ is lower bounded by $n^{\Omega(\log n)}$, as claimed.

A2 Details for Decomposition Into the Normal Hall Subgroup and Complement Groups

A2.1 Proof of Correctness of Algorithm 1

All we need to show is that H is a subgroup, as clearly it is a set of representatives. That is for every a_i, a_j pair, it is required that

$$(\sigma(a_i)\cdot a_i)(\sigma(a_j)\cdot a_j) = \sigma(\phi(a_ia_j))\cdot\phi(a_ia_j).$$

On the other hand, we have

$$\begin{aligned}&(\sigma(a_i)\cdot a_i)(\sigma(a_j)\cdot a_j)\\ &\quad=\sigma(a_i)C_{a_i}(\sigma(a_j))(a_ia_j/\phi(a_ia_j))\cdot\phi(a_ia_j)\\ &\quad=(\sigma(a_i)+C_{a_i}(\sigma(a_j))+a_ia_j/\phi(a_ia_j))\cdot\phi(a_ia_j),\end{aligned}$$

so we only need to show

$$\sigma(\phi(a_ia_j)) = \sigma(a_i) + C_{a_i}(\sigma(a_j)) - f(a_i,a_j).$$

Given $a,b,c \in A$, from $(ab)c = a(bc)$ the following equation can be derived (note that $\phi(\phi(ab)c) = \phi(a\phi(bc))$):

$$f(a,b) + f(\phi(ab),c) = C_a(f(b,c)) + f(a,\phi(bc)).$$

By summing over $c \in A$, and multiply each side by q we have

$$mqf(a,b) + \sigma(\phi(ab)) = C_a(\sigma(b)) + \sigma(a).$$

But noting that $mqf(a,b) = (1-np)f(a,b) = f(a,b)$ as $nf(a,b) = 0$, we have

$$\sigma(\phi(ab)) = C_a(\sigma(b)) - f(a,b) + \sigma(a).$$

Thus finishing the proof.

A2.2 Analysis of Strategy of Algorithm 2

We present the idea behind Algorithm 2 which is essentially following the proof of Schur-Zassenhaus theorem in [22]. We include it here for completeness.

The strategy is to use the case of N being abelian as the base case, and use the recursive algorithm presented in Algorithm 2. Given N not abelian, the recursion goes into two branches depending on whether N is minimal or not. Recall that $|G| = n$, $|N| = m$ and $l = n/m$. Note that it is enough to find a subgroup of order l.

First we consider the case of N being minimal. Let p be a prime dividing $|N|$ and let P be a p-Sylow subgroup of N. Consider G' and N' which are normalizers of P in G and N respectively. Then the following holds,

$$\begin{aligned}G/N &= NG'/N \text{ (Frattini argument)}\\ &\cong G'/(N\cap G') \text{ (the second Iso. theorem)}\\ &= G'/N' \text{ (definition of } G', N').\end{aligned}$$

Notice that $|N'| > 1$, $|N'|$ divides $|N|$, and hence $(|N'|, |G'/N'|) = 1$, then we can recursively call the algorithm with G' and N'. Now we argue that G' must be a proper subgroup of G. Suppose not, $G' = G$. This implies, $P \lhd G$. But since N is a minimal normal subgroup of G and $P \leqslant N$, this gives $P = N$ as well. Now the center of P, $Z(P)$ must be a non-trivial normal subgroup of G since a p-group always has a non-trivial center. Minimality gives that $P = N = Z(P)$ which contradicts the assumption that N is not abelian.

If N was not a minimal subgroup to begin with, then we can find the minimal normal subgroup $T \lhd N$. Now it follows that $N/T \lhd G/T$ and $(G/T)/(N/T) \cong G/N$. Let $t = |T|$, then N/T has order m/t, and $(G/T)/(N/T)$ has order n/m. As $(m/t, n/m) = 1$, N/T is a Hall subgroup of G/T. Recursive invocation of the algorithm with G/T and N/T returns a K/T which is the complement of N/T satisfying $K/T \cong G/N$. Notice here when doing the recursive invocation, quotient groups are represented as sets of representatives; multiplication tables are defined according to G; and the representatives are consistent in the sense that N/T, K/T are represented by subsets of G/T's representatives.

Given K/T (as a set R of representatives) and T, we calculate K as $R \cdot T = \{r \cdot t | r \in R, t \in T\}$. By normality of T and the definition of R, $\forall r_1, r_2 \in R, t_1, t_2 \in T$, $r_1 t_1 (r_2 t_2)^{-1} = r_1 r_2^{-1} t' = r t'' \in K$, where r is the representative of $r_1 r_2^{-1}$ in R, $t', t'' \in T$, we proved $K \leqslant G$. Since $(|T|, |K/T|) = (t, n/m) = 1$, T is a Hall subgroup of K, another recursive call with K and T will return a complement H of T in K, $H \cong K/T \cong G/N$. Now we have $H \leqslant K \leqslant G$, which is exactly what we need.

A2.3 Auxiliary Algorithms for Algorithm 2

This subsection contains descriptions of auxiliary algorithms of computing normalizer, testing minimality of a normal subgroup and computing a Sylow p-group of a given group.

Algorithm A1. Compute the Normalizer of a Set $S \subseteq G$, NORMALIZER(G, S).
Input: Group G given as multiplication table, $S \subseteq G$.
Output: Normalizer of S.
1: Let $T \leftarrow \varnothing$.
2: **for all** $g \in G$ **do**
3: **if** for **all** $s \in S$, $gsg^{-1} \in S$ **then**
4: $T \leftarrow T \cup \{g\}$.
5: **end if**
6: **end for**
7: **return** T.

Algorithm A2. Test Whether $N \lhd G$ is a Minimal Normal Subgroup, MINIMALNORMAL(G, N).
Input: Group G given as multiplication table, $N \lhd G$.
Output: True if N is minimal. $S \lhd G$, $S \subsetneq N$ otherwise.
1: $S \leftarrow \varnothing$.
2: **for all** $g \in N$ **do**
3: $S = \langle g^G \rangle$.
4: **if** $S \subsetneq N$ **then**
5: **return** S.
6: **end if**
7: **end for**
8: **return** true.

Algorithm A3. Compute a Sylow p-Group of G, sylow(G, p).
Input: Group G given as multiplication table. A prime number p that divides $|G|$.
Output: A p-Sylow subgroup of G.
1: $l \leftarrow$ the highest power of p in $|G|$.
2: Fix some $g \in G$, such that $|g| = p$. {By Macay's Theorem, such g must exist.}
3: $S \leftarrow \langle g \rangle$, $K \leftarrow \varnothing$.
4: **for** $i = 2$ to l **do**
5: $K =$ NORMALIZER(G, S).
6: Compute K/S.
7: Fix some $kS \in K/S$, such that $|k| = p$.
8: $S \leftarrow \langle k, S \rangle$.
9: **end for**
10: **return** S.

A3 On Testing Conjugation of Automorphisms of Abelian Groups

In [12], Le Gall presented a technique that reduces testing conjugation of automorphisms of an abelian group to that of linear maps, when the orders of the automorphisms are coprime with that of the abelian group. We refer it as Le Gall's technique in this paper, and elaborate the main result as follows.

Once a basis is fixed, any element $x \in A$ can be uniquely written as $x_1^{a_1} x_2^{a_2} \cdots x_s^{a_s}$, for $a_i \in [p_i^{k_i}]$, $i \in [s]$. Thus any element can be identified as a vector $(a_1, \ldots, a_s)$, and any homomorphism of an abelian group can be written as a matrix describing images of basis. Note that for this matrix, different entries may be from different fields. For simplicity we first consider the automorphisms of abelian p-groups. Then the following classes of matrices can be defined[12,34].

Definition A1. *Given an abelian p-group A_p, define $M(A_p)$ as the following set of integer matrices.*

$$M(A_p) = \{(u_{ij})_{s\times s} \in \mathbb{Z}^{s\times s} \mid 0 \leqslant u_{ij} < p^{k_i} \wedge p^{k_i - k_{\min(i,j)}} \text{ divides } u_{ij},\ \forall i, j \in [s]\}.$$

Given two matrices $\boldsymbol{U}$ and $\boldsymbol{U}'$ in $M(A_p)$, the multiplication $*$ is defined as: $\boldsymbol{U} * \boldsymbol{U}'$ is the integer matrix $\boldsymbol{W}$ of size $s \times s$ such that $w_{ij} = (\sum_{k=1}^{s} u_{ik} u'_{kj} \ mod\ p^{k_i})$. Define $R(A_p)$ to be the set $R(A_p) = \{\boldsymbol{U} \in M(A_p) \mid det(\boldsymbol{U}) \not\equiv 0 \ mod\ p\}$.

The class of matrices $R(A_p)$ fully characterizes automorphism group of $\mathrm{Aut}(A_p)$ in the following sense.

Theorem A1. $(R(A_p), *) \cong (Aut(A_p), \circ)$[34].

So testing conjugacy of the automorphism groups of abelian p-groups is the same as testing conjugacy of matrices from $R(A_p)$. Recall that, in linear algebra, testing conjugacy of two matrices over a field is easy, because we can calculate the normal forms (rational normal form, Jordan normal form, etc.) and test

whether the two normal forms coincide. If $A_p = \mathbb{Z}_p^s$, and hence A_p is a vector space, we can use the linear algebraic result above. In general A_p may not be elementary abelian. However, we can transform a matrix in $R(A_p)$ to a matrix in $\mathrm{GL}(\mathbb{Z}_p, s)$ in the following way.

For $\boldsymbol{U} \in M(A_p)$, let $D_i(\boldsymbol{U})$ be the i-th diagonal block of $\boldsymbol{U}$, namely, the square block indexed from $k_1+\cdots+k_{i-1}+1$ to $k_1+\cdots+k_{i-1}+k_i$, and let $[D_i(\boldsymbol{U})]$ be the result by reducing each entry of $D_i(\boldsymbol{U})$ mod p. Now we define a mapping $\Lambda_p : M(A_p) \to \mathrm{GL}(\sum k_i, p)$ as

$$\Lambda_p : \boldsymbol{U} \mapsto diag([D_1(\boldsymbol{U})], \ldots, [D_s(\boldsymbol{U})]).$$

When p is clear from the context we drop the subscript from Λ_p to get Λ.

In general, an abelian group is direct product of abelian p_i-groups with different p_i's. We define $R(A)$ as the class of big diagonal matrices where its i-th diagonal block is from $R(A_{p_i})$ corresponding to the component of the p_i-group, constructed as above. We can define Λ by applying Λ_{p_i} for the block corresponding to $R(A_{p_i})$. Then Lemma 2 can be extended to $R(A)$ naturally by considering each component independently.

This finishes the description of the reduction Λ_p used in Subsection 4.3. The most interesting property about Λ_p is Lemma 2. In [12], a weaker version of Lemma 2, namely when the subgroup of $\mathrm{Aut}(A)$ is cyclic, is proved. The form presented in Subsection 4.3 is also proved in [31]. We refer the reader to these references for more details.

A4 Proof of Corollary 2

First of all the ranks of $\boldsymbol{C}'$ and $\boldsymbol{D}'$ must be the same, as multiplying $\boldsymbol{G}$ and P will not change the rank, and two matrices being equal necessarily implies their ranks are the same. Let $\boldsymbol{C}$ and $\boldsymbol{D}$ be the linear subspaces generated by row vectors of $\boldsymbol{C}'$ and $\boldsymbol{D}'$, respectively. We claim that $\boldsymbol{C}'$ and $\boldsymbol{D}'$ are isomorphic if and only if $\boldsymbol{C}$ and $\boldsymbol{D}$ are isomorphic as linear codes.

The "If" Direction. $\boldsymbol{C}$ and $\boldsymbol{D}$ being isomorphic as codes implies that $\boldsymbol{C}$ and $\boldsymbol{D}$ are the same subspaces up to permutation of coordinates. Let P be the permutation, then row vectors of $\boldsymbol{C}'P$ and $\boldsymbol{D}'$ generate the same subspace. Now every row vector of $\boldsymbol{D}'$ can be generated by an appropriate combination of row vectors in $\boldsymbol{C}'P$, and an appropriate general linear map $\boldsymbol{G}$ can be recovered.

The "Only If" Direction. If $\boldsymbol{G}\boldsymbol{C}'P = \boldsymbol{D}'$, then row vectors of $\boldsymbol{G}\boldsymbol{C}'P$ and row vectors of $\boldsymbol{D}'$ generate the same subspace. It follows that row vectors of $\boldsymbol{C}'P$ and row vectors of $\boldsymbol{D}'$ generate the same subspace. This means that under the permutation of coordinates by P, $\boldsymbol{C}$ and $\boldsymbol{D}$ are the same subspace.

So to test isomorphism of $\boldsymbol{C}'$ and $\boldsymbol{D}'$, it is enough to take a set of linearly independent row vectors of $\boldsymbol{C}'$ and $\boldsymbol{D}'$ to form generating matrices of $\boldsymbol{C}$ and $\boldsymbol{D}$, then to apply Theorem 8. Finally, to make sure that the permutation of coordinates comes from the permutation group S, we take a coset intersection which runs in time 2^n to finish the proof. This gives a permutation matrix, and the general linear map can be recovered easily.

A5 Representation of $\mathbb{Z}_q^\ell$ over $\mathbb{Z}_p$

The proofs in this section are only sketched.

We give an alternative construction of representation of $\mathbb{Z}_q^\ell$ over $\mathbb{Z}_p$, as follows. Factor Φ_q as $g_1 \cdot g_2 \cdot \ldots \cdot g_r$, and let $\boldsymbol{M}_i$ be the companion matrix of g_i, for $i \in [r]$. Let $d = (q-1)/r$, then $\boldsymbol{M}_i \in \mathrm{GL}(\mathbb{Z}_p, d)$. Choose an $(\ell-1)$-dimension subspace L of $\mathbb{Z}_q^\ell$, and let $Q = \mathbb{Z}_q^\ell / L$. Fix a generator of Q, then by mapping this generator to any of $\boldsymbol{M}_i$'s, we get an irreducible representation. By going over all $(\ell-1)$-dimension subspaces and all $\boldsymbol{M}_i$'s (fixing the generator for each linear subspace), we get all different irreducible representations, except the trivial representation.

To see that this representation is irreducible, we prove that for a non-trivial representation $\phi : \mathbb{Z}_q^\ell \to \mathrm{GL}(\mathbb{Z}_p, d')$, if $d \nmid d'$, ϕ cannot be irreducible. If this is true, the above construction is irreducible as follows. Fix the linear subspace L and the chosen companion matrix to be $\boldsymbol{M}$. Firstly, if $d = 1$, there is nothing to prove. If $d > 1$, and the above construction contains a sub-representation of dimension $< d$, this sub-representation is reducible itself. Continuing the argument gives rise to a representation of dimension 1, that is a vector $\boldsymbol{v} \in \mathbb{Z}_p^d$ satisfying $\boldsymbol{M}\boldsymbol{v} = \lambda \boldsymbol{v}$, $\lambda \in \mathbb{Z}_p$. This is a contradiction, since the eigenvalues of $\boldsymbol{M}$ are q-th roots of unity over the algebraic closure of $\mathbb{Z}_p$, which do not lie in $\mathbb{Z}_p$, as $\dim(M) > 1$. Now we prove that ϕ cannot be irreducible unless $d \mid d'$. Since it is non-trivial, $\exists \boldsymbol{u} \in \mathbb{Z}_q^\ell$, $\phi(\boldsymbol{u}) \in \mathrm{GL}(\mathbb{Z}_p, d')$ is not identity. Since $\mathrm{ord}(\boldsymbol{u}) = q$, $\mathrm{ord}(\phi(\boldsymbol{u})) = q$ too, which implies that the minimal polynomial of $\phi(\boldsymbol{u})$ is one of the g_i's, $i \in [r]$. Note that the minimal polynomial divides characteristic polynomial. Since the minimal polynomial is irreducible here, it follows that the characteristic polynomial is a power of minimal polynomial. So the degree of the characteristic polynomial of $\phi(\boldsymbol{u})$ is a multiple of d, finishing the proof.

Now we give a justification that this construction gives every non-equivalent irreducible representation exactly once. To see this we go to the algebraic closure of $\mathbb{Z}_p$, and let it be $\overline{\mathbb{Z}_p}$. Over $\overline{\mathbb{Z}_p}$, it can be shown that the irreducible representations of $\mathbb{Z}_q^\ell$ are similar to the case over $\mathbb{C}$. That is, the irreducible representations

of $\mathbb{Z}_q^\ell$ are of dimension 1, and there are q^ℓ irreducible representations in all. Over $\overline{\mathbb{Z}_p}$, the above construction "breaks up" into $d \cdot (\frac{q^\ell-1}{q-1} \cdot r) = q^\ell - 1$ non-trivial different representations. Furthermore, over $\overline{\mathbb{Z}_p}$ it can be seen that the above construction gives nonequivalent representations, which implies that they are not equivalent over $\mathbb{Z}_p$ either.

To get the construction used in Subsection 5.1, we first note that as $x^q - 1$ and its derivative qx^{q-1} have no common roots over $\mathbb{Z}_p$, it follows that Φ_q have $q-1$ distinct roots. Fix $\boldsymbol{M} = \boldsymbol{M}_1$ and consider $\langle \boldsymbol{M} \rangle$. For any $\boldsymbol{N} \in \langle \boldsymbol{M} \rangle$, since its characteristic polynomial (and minimal polynomial) can only be one of g_i's, $\boldsymbol{N}$ must be conjugate to one of $\boldsymbol{M}_i$'s. (As the roots of the characteristic polynomial of $\boldsymbol{M}_i$'s are distinct over $\overline{\mathbb{Z}_p}$, it is enough to compare the characteristic polynomials.) Then consider $\langle \boldsymbol{M} \rangle$ and $\boldsymbol{M}_i$'s over $\overline{\mathbb{Z}_p}$. First, eigenvalues of $\boldsymbol{M}_i$'s form a partition of q-th roots of unity in $\overline{\mathbb{Z}_p}$. Fix an eigenvalue ξ of $\boldsymbol{M} = \boldsymbol{M}_1$, then ξ^j is an eigenvalue of $\boldsymbol{M}^j$. As extension of fields does not change equivalence relation of matrices, $\boldsymbol{M}^j$ is conjugate with some $\boldsymbol{M}_i$. Then $\boldsymbol{M}^j$ must be conjugate to the one with eigenvalue ξ^j. Thus when ξ^j goes over all q-th roots of unity over $\overline{\mathbb{Z}_p}$, $\boldsymbol{M}^j$ goes over all of $\boldsymbol{M}_i$'s (up to equivalence), with each $\boldsymbol{M}_i$ d times. Combining with the fact that when subspaces are different, the irreducible representations constructed cannot be equivalent, we prove Claim 1 and Corollary 3, as above.

Finally, we give another view of the dimension of the irreducible representations of $\mathbb{Z}_q^\ell$ over $\mathbb{Z}_p$. First the trivial representation has dimension 1. Then from the construction it can be seen that for other representations, their dimensions are equal to the degree of the irreducible factors of Φ_q over $\mathbb{Z}_p$. We prove that the degree d is equal to the order of p in multiplicative group $(\mathbb{Z}/q\mathbb{Z})^\times$, namely, d is the smallest integer satisfying $p^d = 1 (\mathrm{mod}\ q)$, and Φ_q can be factorized into distinct irreducibles of degree d. To see this, first note that Φ_q is square free (for any q), this can be proved by showing $X^n - 1$, which is a product of cyclotomic polynomials satisfying $p \nmid n$, is square free. Suppose $X^n - 1 = A(X)^2 B(X)$, with substitution, its derivative can be represented as

$$\begin{aligned} nX^{n-1} &= 2A(X)A'(X)B(X) + A(X)^2 B'(X) \\ &= A(X)C(X). \end{aligned}$$

By rewriting n as

$$\begin{aligned} n &= X \cdot nX^{n-1} - n(X^n - 1) \\ &= A(X)(XC(X) - nA(X)B(X)), \end{aligned}$$

comparing the two sides of the equality, one can derive that $A(X)$ must be constant. Then what remains is to prove that every irreducible of Φ_q is of degree d. Let f be an irreducible factor of Φ_q, and $s = \deg(f)$. Consider the field $\mathbb{K} = \mathbb{Z}_p[X]/f$ with size p^s, and $k^{p^s-1} = 1$, $\forall k \in \mathbb{K}$. $\mathbb{K}$ must contain a q-th primitive root of unity ω. By $\omega^{p^s-1} = 1$ and $\omega^q = 1$, we know that $q | p^s - 1$, thus, $p^s = 1 (\mathrm{mod}\ q)$, and $s \geqslant d$. Conversely, since $q | p^d - 1$, $\omega^{p^d} = \omega$, or equivalently, ω is a root of $X^{p^d} - X$. So the splitting field $\mathbb{S}$ of $X^{p^d} - X$ is a subfield of $\mathbb{K}$ containing ω, but primitivity of ω in $\mathbb{K}$ implies that $\mathbb{S}$ must contain all elements of $\mathbb{K}$. Therefore, $p^d = p^s$, and $d = s$.

We finish this section with examples showing that the irreducible factors of Φ_q over $\mathbb{Z}_p$ are not necessarily of degree 1.

Fact A1.

- $\Phi_3(x) = (x^2 + x + 1)$ *over* $\mathbb{Z}_2$;
- $\Phi_3(x) = (x + 3)(x + 5)$ *over* $\mathbb{Z}_7$;
- $\Phi_{11}(x) = (x^5 + 2x^3 + x^2 + 2x + 2)(x^5 + x^4 + 2x^3 + x^2 + 2)$ *over* $\mathbb{Z}_3$.