On Isomorphism Testing of Groups with Normal Hall Subgroups

Youming Qiao, Jayalal Sarma, Bangsheng Tang

On Isomorphism Testing of Groups with Normal Hall Subgroups*

Youming Qiao[1], Jayalal Sarma M.N.[2], and Bangsheng Tang[1]

1 **Institute for Theoretical Computer Science, Tsinghua University, Beijing, China**
jimmyqiao86,megatang@gmail.com

2 **Department of Computer Science & Engineering, Indian Institute of Technology Madras, Chennai, India**
jayalal@cse.iitm.ac.in

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 in [9], 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. 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 in [3]. 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 and phrases Group Isomorphism Problem, Normal Hall Subgroups, Computational Complexity

Digital Object Identifier 10.4230/LIPIcs.STACS.2011.567

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 $\phi : G \to H$

* 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. All the authors are supported in part by the National Natural Science Foundation of China Grant 60553001, 61073174, 61033001 and the National Basic Research Program of China Grant 2007CB807900, 2007CB807901.

28th Symposium on Theoretical Aspects of Computer Science (STACS'11).
Editors: Thomas Schwentick, Christoph Dürr; pp. 567–578

Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

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 [8, 1]. For permutation groups given as generators, the best upper bound known [6] 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) [13], thus giving an upper bound of $\mathrm{NP} \cap \mathrm{coAM}$. However, unlike many other isomorphism-type problems, a reduction in the reverse direction is not known[13]. A recent work [7] shows that GrI can not 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 $2^{\tilde{O}(\sqrt{n})}$ [5], where n is the size of the graph. For groups of size n with b generators, in [16] 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 [14], who gave an algorithm running in $O(\log^2 n)$ space. However, whether a polynomial time algorithm exists is still open.

1.1 Progress for testing isomorphism of restricted classes of groups

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 [19] first gave an $O(n^2)$ algorithm, which was improved to $O(n \log n)$ by Vikas [24] and finally to $O(n)$ by Kavitha [11]. Little is known beyond abelian groups until 2008, when Le Gall [9] 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 [25, 26] on the structure of p-groups are noteworthy.

Recently, Kayal and Nezhmetdinov [12] and Wilson [27] address the problem of finding the factors of a group under the direct product operation (Wilson [27] 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 [12], 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 [12] and [27], 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 products, which is already considered in [9]. 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 [9], 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: a 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 [9], we begin with formalizing the strategy for isomorphism testing discussed in Section 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 1.1.** (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. Our next result, which has been discovered by Taunt [23] in the context of construction of finite groups, is the formulation of a necessary and sufficient condition of the original groups being isomorphic in Theorem 4.1. That condition involves the actions associated with the semidirect products, and the isomorphisms of the normal and complement parts. It is not listed here, partly due to its technicality, but the main reason is that as discussed, we need to turn our focus to the case when the factors of semidirect product are efficiently testable. The following notations will help us to 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 groups[1], symmetric

[1] For readers unfamiliar with this fact, c.f. the first theorem in [15], and note that a simple abelian group must be a cyclic group with prime order.

groups and cyclic groups. When the specific number of generators is not of our concern, we will simply write $\mathcal{B}$. $\mathcal{C} = \mathcal{B}_1$ is the class of cyclic groups. Finally, we let $\mathcal{K}$ be a variable taking values from the class of groups with known efficient isomorphism testing/computing procedure. 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 [9] 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). $\mathcal{H}(\mathcal{A}, \mathcal{B})$ improves the class $\mathcal{H}(\mathcal{A}, \mathcal{C})$ studied in [9].

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. In Theorem 4.1, when the normal subgroup is an elementary abelian group $\mathbb{Z}_p^k$, p a prime, it naturally gives rise to the following algorithmic problem in representation theory of finite groups which may be of independent interest, which we call 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 $\text{poly}(|H|, p^d)$.

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

▶ **Theorem 1.2.** *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 generalization[2] 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 1.3.** *For groups from $\mathcal{H}(\mathcal{E}, \mathcal{E})$,* AutoInducedRepEquiv *reduces to generalized code isomorphism problem.*

In a recent work [3], 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 1.4.** *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. Applying a technique in [9], 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 Section 8.9, [10] (see Section 4.2).

[2] See Section 5 for specific points of generalization.

▶ **Theorem 1.5.** *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 1.1. 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 1.2, elaborate on the framework, and show that how a technique from [9] 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 isomorphism, the reductions (Theorem 1.3) and show how to test isomorphism of $\mathcal{H}(\mathcal{A},\mathcal{E})$. Due to the page constraints, we only give sketches of proofs for some propositions. We refer the interested readers to a full version of this article for the detailed proofs and complete algorithms.

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 [18] 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 $\mathbb{Z}_p^k$. Every abelian group can be decomposed as direct product of cyclic groups by the fundamental theorem of abelian groups.

For a group G, we say that G is the semidirect product of N by H, for $N\lhd G$ and $H\leq G$, 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 2.1.** (Schur-Zassenhaus theorem, c.f. [18]) *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 $H\leq G$ 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 [20] 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

[3] Note 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 $G=N\rtimes_\tau H$.

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 $\forall g \in G$, $\phi_g(L) = L$. $\vec{0}$ 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: $(\phi \oplus \rho)_g(u + v) := \phi_g(u) + \rho_g(v)$ 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 coprime 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.

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

▶ **Theorem 2.2.** (Maschke's theorem. Adaptation of [20], page 6, Theorem 1) *Let $\phi : G \to \mathrm{GL}(\mathbb{F}, d)$ be a representation,* $\gcd(|G|, \mathrm{char}(\mathbb{F})) = 1$. *$W \leq V$ 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 2.2 suggest the following procedure to decompose a representation into its irreducible components. Let $\phi : G \to \mathrm{GL}(V)$ be a representation. For every $v \in V$, test if $\langle gv \mid g \in G \rangle$ generates V. If so, it is an irreducible representation. Otherwise, for a specific v, $\langle gv \mid g \in G \rangle$ is a sub-representation W. Then Theorem 2.2 helps to identify a sub-representation 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 $\phi : G \to \mathrm{GL}(V)$, its irreducible components 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 efficient (c.f. [17] and Chapter 7.4 of [10]). Given two irreducible representations, there is an efficient algorithm to determine whether they are equivalent (c.f.[10], 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 [21]. For computing canonical normal form of a linear transformation, Steel's algorithm [22] in time $O(n^4)$ suffices.

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 1.1, 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.

Proposition 3 follows from that for a specific Hall divisor m, if the normal Hall subgroup of m exists then it is generated by $\langle g^{n/m} \mid g \in G \rangle$. Proof of Proposition 4 follows from the constructive proof of Schur-Zassenhaus theorem [18], which can be rephrased as a recursive algorithm. The base case of the algorithm is abelian groups, for which a complement can be found starting with an arbitrary set of representatives. When the input is not abelian, the algorithm branches into two cases depending on whether the normal subgroup is minimal. The case using the Hall condition is when the normal subgroup is minimal, and we use the Frattini argument and second isomorphism theorem to reduce to an instance of smaller size.

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 [23] in the context of construction of finite groups, though he did not apply it to normal Hall subgroups explicitly.

▶ **Theorem 4.1.** *(Theorem 3.3, [23]) 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, $\forall h \in H_1$,*

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

4.1 Proof of Theorem 1.2

Theorem 1.2 states that isomorphism of $\mathcal{H}(\mathcal{E}, \mathcal{K})$ is equivalent to AutoInducedRepEquiv. In this section 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. Then to test isomorphism of the original group, we first use known isomorphism procedure for normal and complement parts. Given the isomorphisms of the normal and complement parts, the only task left is to test Equation 1, which, by composing the isomorphisms of the normal and complement parts, becomes AutoInducedRepEquiv naturally.

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 4.1, 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 A 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 1.1, 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 Equation 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 enumerated. 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). It is also noted that when the normal subgroup is $\prod \mathcal{E}$, the idea of treating the representations over the factors separately does not work in general unless an automorphism of the complements is fixed as a priori. 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

In [9], Le Gall presented a technique that reduces testing conjugation of automorphisms of an abelian group to that of linear mappings, 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.

▶ **Lemma 4.2.** (Le Gall's technique) *For a given abelian p-group A, and a generating set $S \subseteq A$, let ϕ_1 and ϕ_2 be two automorphisms of A, given by listing the images of the generating set. If $p \nmid |\phi_1| = |\phi_2|$, there exists an efficiently-computable map $\Lambda_p : \mathrm{Aut}(A) \to \mathrm{GL}(\mathbb{Z}_p, |S|)$, such that ϕ_1 and ϕ_2 are conjugate if and only if $\Lambda_p(\phi_1)$ and $\Lambda_p(\phi_2)$ are conjugate.*

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 4.1, 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 ψ, for every $h \in H$. Noting that $p \nmid |H|$, Lemma 4.2 tells that this happens if and only if $\Lambda_p(\tau(h))$ and $\Lambda_p(\gamma(\phi(h)))$ are conjugate. Thus composing Λ_p with τ and γ, noting that $\Lambda_p \circ \tau$ and $\Lambda_p \circ \gamma$ send H to $\mathrm{GL}(\mathbb{Z}_p, k)$, we reduce the case of $\mathcal{H}(\mathcal{A}_p, \mathcal{K})$ to $\mathcal{H}(\mathbb{Z}_p^k, \mathcal{K})$. To go from $\mathcal{H}(\mathcal{A}, \mathcal{K})$ to $\mathcal{H}(\prod \mathcal{E}, \mathcal{K})$ we just need to consider the factors of $\prod \mathcal{E}$ separately and apply the appropriate Λ_p.

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. For $\mathbb{F}^n$, a linear code of dimension d is a subspace of dimension d. A generating matrix of a code C of dimension d is a d by n matrix with row vectors being a basis of

C. With abuse of notation we will also use C to denote the generating matrix of the code C. Two codes C and 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 G and an n by n permutation matrix P such that $GCP = D$.

▶ **Theorem 5.1.** *([3]) For C and D be two linear codes given as generating matrices, their isomorphism can be tested, and the coset of isomorphism be computed, in time $(2 + o(1))^n$.*

We generalizes code isomorphism problem slightly to get:

▶ Problem 2. *(Generalized code isomorphism problem)* Given two matrices $d' \times n$ matrices C' and D' over the field $\mathbb{F}$, and a permutation group $S \leq S_n$, if there exists $G \in \mathrm{GL}(\mathbb{F}, d')$ and a permutation matrix $P \in S$, such that $GC'P = D'$.

The generalized code isomorphism problem generalizes code isomorphism problem in two ways: first we do not require row vectors of C' and D' to be linearly independent. Secondly the permutation matrix P must come from a certain permutation group S. Its solution in singly exponential time can be viewed as a corollary to Theorem 5.1, by applying a coset intersection running in singly exponential time[2].

▶ **Corollary 5.2.** *Given two $d' \times n$ matrices C' and D', and a permutation group S, whether C' and D' are isomorphic can be tested, the coset of permutation matrices be computed, in time $(2 + o(1))^n$.*

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

In this section, 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 standard textbooks for more details. 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 $M \in \mathrm{GL}(\mathbb{Z}_p, d)$ be the companion matrix of g_1.[4] For $v \in \mathbb{Z}_q^\ell$, $v \neq \vec{0}$, we define $v^* : \mathbb{Z}_q^\ell \to \mathbb{Z}_q$ by mapping $v^*(u) = (v, u)$ (the inner product of v and u). Now define $f_v : \mathbb{Z}_q^\ell \to \mathrm{GL}(\mathbb{Z}_p, d)$ by sending $u \to M^{v^*(u)}$. To unify notation let $f_{\vec{0}} : \mathbb{Z}_q^\ell \to \mathbb{Z}_p$ be the trivial representation. Then f_v gives an irreducible representation of $\mathbb{Z}_q^\ell$ over $\mathbb{Z}_p$, and $\{f_v \mid v \in V\}$ is the set of all irreducible representations. However, f_v and f_u may be equivalent, for $u, v \in V$, as described in the following claim.

▶ Claim 1. Let f_v and f_u be two irreducible representations of $\mathbb{Z}_q^\ell$ over $\mathbb{Z}_p$ induced from $v, u \in \mathbb{Z}_q^\ell$, $v, u \neq \vec{0}$ as above. f_v and f_u are equivalent if and only if $u = sv$ for $s \in \mathbb{Z}_q$, and M^s and M are conjugate.

▶ **Corollary 5.3.** *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 $\mathbb{Z}_p$ are completely reducible. Suppose $\tau = f_{v_1}^{k_1} \oplus \cdots \oplus f_{v_t}^{k_t}$, for $v_i \in V$, $i \in [t]$, $k_1 \geq \cdots \geq k_t \geq 1$. Note that t is bounded by $1 + \lfloor (k-1)/d \rfloor$ or k/d, depending on whether the trivial representation exists or not. We will assume when a representation is decomposed as such, the multiplicities of irreducible components are arranged to be non-increasing. For

[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 $i \in [r]$.

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, at the cost of losing uniqueness.

▶ **Definition 5.4.** 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 $v \in \mathcal{L}_\tau(w)$ such that f_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. By Corollary 5.3, the number of different indexing tuples of L_τ is bounded by $d^{k/d} \le (e^{1/e})^k < 2^k$. (Note that we do not need to consider $f_{\vec{0}}$.)

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 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_v by $\phi \in \mathrm{GL}(\mathbb{Z}_q, l)$ has a nice form: $(f_v \circ \phi)(u) = f_v(\phi(u)) = M^{v^*(\phi(u))} = M^{(\phi^T(v))^*(u)} = f_{\phi^T(v)}(u)$. That is $f_v \circ \phi = f_{\phi^T(v)}$. 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 $\tau \circ \phi = f^{k_1}_{\phi^T(v_1)} \oplus \cdots \oplus f^{k_t}_{\phi^T(v_t)}$. 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 1.3

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 1.2 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 1.3, 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 $\tau = f^{k_1}_{v_1} \oplus \cdots \oplus f^{k_t}_{v_t}$ and $\gamma = f^{\ell_1}_{u_1} \oplus \cdots \oplus f^{\ell_{t'}}_{u_{t'}}$ 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 pre-determined 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 can not 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 $\mathcal{L}_{\tau\circ\phi}$ is to apply 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 5.1, 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 $\mathcal{L}_\tau^{\phi^T} = \mathcal{L}_\gamma$ 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

$$(\{v_1, \ldots, v_{s_1}\}, \{v_{s_1+1}, \ldots, v_{s_2}\}, \ldots, \{v_{s_{k-1}+1}, \ldots, v_{s_k}\}),$$

in which $s_1 \leq s_2 \leq \cdots \leq s_k = t$. Similarly expand $\mathcal{L}_\gamma$ as

$$(\{u_1, \ldots, u_{s_1}\}, \{u_{s_1+1}, \ldots, u_{s_2}\}, \ldots, \{u_{s_{k-1}+1}, \ldots, u_{s_k}\}).$$

$\mathcal{L}_\tau^{\phi^T}$ is just $(\{\phi(v_1), \ldots, \phi(v_{s_1})\}, \{\phi(v_{s_1+1}), \ldots, \phi(v_{s_2})\}, \ldots, \{\phi(v_{s_{k-1}+1}), \ldots, \phi(v_{s_k})\})$, $\mathcal{L}_\tau^{\phi^T} = \mathcal{L}_\gamma$ can be formulated as finding $\phi \in \mathrm{GL}(\mathbb{Z}_q, \ell)$ and $\sigma \in S_{s_1} \times S_{s_2-s_1} \times \cdots \times S_{s_k-s_{k-1}}$ such that $\phi(v_1, \ldots, v_t)\sigma = (u_1, \ldots, u_t)$. This is just generalized code isomorphism problem with the permutation group $S_{s_1} \times S_{s_2-s_1} \times \cdots \times S_{s_k-s_{k-1}}$, whose the 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 1.3.

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

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 ith 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 $\mathcal{L}_{\tau_i}^{\phi^T} = \mathcal{L}_{\gamma_i}$, 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 $\mathcal{H}(\mathcal{A}, \mathcal{E})$.

Acknowledgement

Part of the work was done while Youming Qiao was visiting the University of Chicago, and he would like to thank Laci Babai and Sasha Razborov for hosting him. Youming 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 [3].

References

1 S.I. Adian. The unsolvability of certain algorithmic problems in the theory of groups. *Trudy Moskov. Math. Obshch*, 6:231–298, 1957.
2 László Babai. Coset intersection in moderately exponential time. *Chicago J. Theoret. Comp. Sci.*, 2010.
3 László Babai. Equivalence of linear codes. Manuscript, 2010. See [4].
4 László Babai, Paolo Codenotti, Joshua Grochow, and Youming Qiao. Towards efficient algorithm for semisimple group isomorphism. Manuscript, 2010. To appear at SODA 11.
5 László Babai and Eugene M. Luks. Canonical labeling of graphs. In *STOC '83: Proceedings of the 15th Annual ACM Symposium on Theory of Computing*, pages 171–183, New York, NY, USA, 1983. ACM.

6 László Babai and Endre Szemerédi. On the complexity of matrix group problems i. In *FOCS*, pages 229–240, 1984.

7 Arkadev Chattopadhyay, Jacobo Toran, and Fabian Wagner. Graph isomorphism is not AC^0 reducible to Group Isomorphism. In *Proceedings of FSTTCS 2010 (To Appear)*, July 2010. Technical report available at ECCC : TR10-117.

8 Max Dehn. über unendliche diskontinuierliche gruppen. *Mathematische Annalen*, 71:116–144, 1911.

9 Francois Le Gall. Efficient isomorphism testing for a class of group extensions. In Susanne Albers and Jean-Yves Marion, editors, *26th International Symposium on Theoretical Aspects of Computer Science (STACS 2009)*, volume 3 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 625–636, Dagstuhl, Germany, 2009. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

10 Derek F. Holt, Bettina Eick, and Eamonn A. O'Brien. *Handbook of computational group theory*. Chapman and Hall/CRC, London, 2005.

11 T. Kavitha. Linear time algorithms for abelian group isomorphism and related problems. *J. Comput. Syst. Sci.*, 73(6):986–996, 2007.

12 Neeraj Kayal and Timur Nezhmetdinov. Factoring groups efficiently. In *Proceedings of the 36th ICALP (2009)*, pages 585–596, 2009. Also availabe as ECCC Tech Report TR08-074.

13 J. Köbler, U. Schöning, and J. Torán. *The Graph Isomorphism Problem: Its Structural Complexity*. Progress in Theoretical Computer Science. Birkhauser, Boston, 1993.

14 Richard J. Lipton, Lawrence Snyder, and Y. Zalcstein. The complexity of word and isomorphism problems for finite groups. Technical report, John Hopkins, 1976.

15 Federico Menegazzo. The number of generators of a finite group. *Irish Math. Soc. Bulletin*, 50:117–128, 2003.

16 Gary L. Miller. On the $n \log n$ isomorphism technique (a preliminary report). In *STOC '78: Proceedings of the 10th Annual ACM Symposium on Theory of Computing*, pages 51–58, New York, NY, USA, 1978. ACM.

17 Lajos Rónyai. Computing the structure of finite algebras. *J. Symb. Comput.*, 9(3):355–373, 1990.

18 Joseph J. Rotman. *An Introduction to the Theory of Groups (4th Ed.)*. Springer-Verlag, 1995.

19 C. Savage. An $O(n^2)$ algorithm for abelian group isomorphism. Technical report, North Carolina State University, 1980.

20 Jean Pierre Serre. *Linear representations of finite groups*. Springer-Verlag, New York, 1977. Translated from the second French edition by Leonard L. Scott, Graduate Texts in Mathematics, Vol. 42.

21 Victor Shoup. On the deterministic complexity of factoring polynomials over finite fields. *Inform. Process. Lett*, 33:261–267, 1990.

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

23 D. R. Taunt. Remarks on the isomorphism problem in theories of construction of finite groups. *Mathematical Proceedings of the Cambridge Philosophical Society*, 51:16–24, 1955.

24 Narayan Vikas. An $O(n)$ algorithm for abelian p-group isomorphism and an $O(n \log n)$ algorithm for abelian group isomorphism. *J. Comput. Syst. Sci.*, 53(1):1–9, 1996.

25 James B. Wilson. Decomposing p-groups via jordan algebras. *J. Algebra*, 322:2642–2679, 2009.

26 James B. Wilson. Finding central decompositions of p-groups. *J. Group Theory*, 12:813–830, 2009.

27 James B. Wilson. Finding direct product decompositions in polynomial time, May 2010. arXiv:1005.0548.