Polynomial-time isomorphism test of groups that are tame extensions

Joshua A. Grochow Youming Qiao

Santa Fe Institute

University of Technology Sydney

December 11, 2015 @ Nagoya, ISAAC 2015

Group isomorphism problem

Problem (Group isomorphism test (GROUPI))

Given the Cayley tables of two groups of order n, decide whether they are isomorphic or not.

Group isomorphism problem

Problem (Group isomorphism test (GROUPI))

Given the Cayley tables of two groups of order n, decide whether they are isomorphic or not.

- Easy $n^{\log n + O(1)}$-time algorithm (Felsch and Neubüser, 1970; Miller, 1978);
- Classical $n^{1/2 \log n}$, quantum $n^{1/3 \log n}$ (Rosenbaum, 2013);
- Reducible to graph isomorphism (GRAPHI).

Group isomorphism problem

Problem (Group isomorphism test (GROUPI))

Given the Cayley tables of two groups of order n, decide whether they are isomorphic or not.

- Easy $n^{\log n + O(1)}$-time algorithm (Felsch and Neubüser, 1970; Miller, 1978);
- Classical $n^{1/2 \log n}$, quantum $n^{1/3 \log n}$ (Rosenbaum, 2013);
- Reducible to graph isomorphism (GRAPHI).

One motivation:

- Very recently L. Babai announced that graph isomorphism can be solved in time $n^{(\log n)^c}$ for $c \geq 2$;
- In one of the talks he suggested that GROUPI is a bottleneck to put GRAPHI in P.

Some recent results

Polynomial-time algorithms for:

Abelian groups $O(n)$-time (Kavitha, 2007);

Groups with no abelian normal subgroups
Babai et al. (2011) and Babai et al. (2012);

Groups with abelian Sylow towers
Le Gall (2009), Qiao et al. (2011), and Babai and Qiao (2012);

p-groups of genus 2; quotients of generalized Heisenberg groups
Lewis and Wilson (2012) and Brooksbank et al. (2015).

And a group class with $n^{O(\log \log n)}$-time algorithm:

Central-radical groups Grochow and Qiao (2014).

Why these group classes?

- Groups with no abelian normal subgroups;
- Groups with abelian Sylow towers;
- p-groups of genus 2 and quotients of generalized Heisenberg groups;
- Central-radical groups.

A possible explanation for successes over these group classes?

In Grochow and Qiao (2014) we provide some explanation from the perspective of *extension theory of groups*.

A strategy for group isomorphism...

A divide and conquer strategy

Given two groups G_1 and G_2, consider the following recipe...

1. Agree on some characteristic (normal) subgroup $\mathcal{S}$.
   - e.g. center, commutator subgroup, etc.
2. Slice into the normal parts and the quotient parts.
   - To get $\mathcal{S}(G_i)$ and $G_i/\mathcal{S}(G_i)$.

A divide and conquer strategy

Given two groups G_1 and G_2, consider the following recipe...

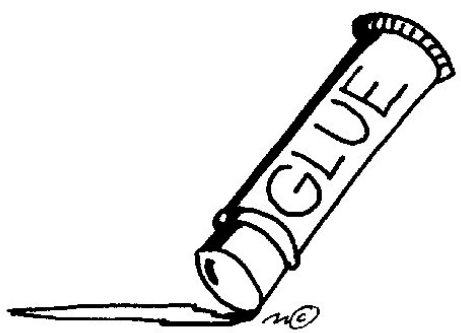
1. Agree on some characteristic (normal) subgroup $\mathcal{S}$.
   - e.g. center, commutator subgroup, etc.
2. Slice into the normal parts and the quotient parts.
   - To get $\mathcal{S}(G_i)$ and $G_i/\mathcal{S}(G_i)$.
3. (Divide) Test isomorphism of the two parts respectively.
   - If both parts are isomorphic respectively, identify the normal part by A and quotient part by Q, continue.
   - Otherwise `not isomorphic`.
4. (Conquer) ...?

A divide and conquer strategy

Given two groups G_1 and G_2, consider the following recipe...

1. Agree on some characteristic (normal) subgroup $\mathcal{S}$.
   - e.g. center, commutator subgroup, etc.
2. Slice into the normal parts and the quotient parts.
   - To get $\mathcal{S}(G_i)$ and $G_i/\mathcal{S}(G_i)$.
3. (Divide) Test isomorphism of the two parts respectively.
   - If both parts are isomorphic respectively, identify the normal part by A and quotient part by Q, continue.
   - Otherwise `not isomorphic`.
4. (Conquer) ...?

After step 3, we call G_1 and G_2 *extensions* of A by Q.
Q: How do the normal part A, and the quotient part Q glue together?



How to conquer?

. . . G_1 and G_2 are extensions of A by Q. For simplicity in the following we assume A is *abelian*.

How to conquer?

. . . G_1 and G_2 are extensions of A by Q. For simplicity in the following we assume A is *abelian*.
By extension theory, two functions arise as the "glue."

Action The conjugation action of Q on A; a homom. $Q \to \mathrm{Aut}(A)$;

2-cocycle How different is from semidirect product; a function $Q \times Q \to A$ satisfying the 2-cocycle identity.

$\mathrm{Aut}(A) \times \mathrm{Aut}(Q)$ acts naturally on the actions and the 2-cocycles.

How to conquer?

. . . G_1 and G_2 are extensions of A by Q. For simplicity in the following we assume A is *abelian*.
By extension theory, two functions arise as the "glue."

Action The conjugation action of Q on A; a homom. $Q \to \mathrm{Aut}(A)$;

2-cocycle How different is from semidirect product; a function $Q \times Q \to A$ satisfying the 2-cocycle identity.

$\mathrm{Aut}(A) \times \mathrm{Aut}(Q)$ acts naturally on the actions and the 2-cocycles.

Lemma (Folklore, cf. Grochow and Qiao (2014))

$G_1 \cong G_2$ if and only if actions and 2-cocycles are the same up to the action of $\mathrm{Aut}(A) \times \mathrm{Aut}(Q)$.

An algorithmic problem about extensions

If the normal subgroup is elementary abelian ($\cong \mathbb{Z}_p^d$)...

Problem (Extension pseudo-congruence problem)

Given two groups that are extensions of $\mathbb{Z}_p^d$ by Q, and $\mathrm{Aut}(Q)$ by a set of generators, decide whether the two extensions are the same under $\mathrm{Aut}(\mathbb{Z}_p^d) \times \mathrm{Aut}(Q)$ in time $\mathrm{poly}(|Q|, p^d)$.

An algorithmic problem about extensions

If the normal subgroup is elementary abelian ($\cong \mathbb{Z}_p^d$)...

Problem (Extension pseudo-congruence problem)

Given two groups that are extensions of $\mathbb{Z}_p^d$ by Q, and $\mathrm{Aut}(Q)$ by a set of generators, decide whether the two extensions are the same under $\mathrm{Aut}(\mathbb{Z}_p^d) \times \mathrm{Aut}(Q)$ in time $\mathrm{poly}(|Q|, p^d)$.

- Solving this problem will solve group isomorphism in general (Cannon and Holt, 2003);
- For $Q = \mathbb{Z}_p^e$ and central extensions, this is p-group isomorphism and considered difficult.

Classification problems in mathematics

In mathematics, an important theme is to classify certain objects. Formally, for a group action, we want to find canonical objects in each orbit.

Classification problems in mathematics

In mathematics, an important theme is to classify certain objects. Formally, for a group action, we want to find canonical objects in each orbit.

Space The set of $n \times n$ matrices, $M(n, \mathbb{C})$;

Group action $A \in \mathrm{GL}(n, \mathbb{C})$ sends $B \in M(n, \mathbb{C})$ to ABA^{-1};

Canonical form (1) B is a direct sum of Jordan blocks; (2) Each Jordan block is determined by the size and the eigenvalue.

Classification problems in mathematics

In mathematics, an important theme is to classify certain objects. Formally, for a group action, we want to find canonical objects in each orbit.

Space The set of $n \times n$ matrices, $M(n, \mathbb{C})$;

Group action $A \in \mathrm{GL}(n, \mathbb{C})$ sends $B \in M(n, \mathbb{C})$ to ABA^{-1};

Canonical form (1) B is a direct sum of Jordan blocks; (2) Each Jordan block is determined by the size and the eigenvalue.

On the other hand, consider a similar problem:

Space The set of pairs of $n \times n$ matrices, $M(n, \mathbb{C}) \oplus M(n, \mathbb{C})$;

Group action $A \in \mathrm{GL}(n, \mathbb{C})$ sends $(B, C) \in M(n, \mathbb{C}) \oplus M(n, \mathbb{C})$ to (ABA^{-1}, ACA^{-1});

Canonical form A long-standing open problem; believed to be intractable.

The tame-wild dichotomy

Definition

A classification problem is *tame*, if the indecomposables of dimension d come from a finite number of 1-parameter families. It is *wild* if it "contains" the problem of classifying pairs of matrices under simultaneous conjugation.

Theorem (Drozd, 1970's)

The classification problem for representations of associative algebras over algebraically-closed fields are either tame or wild.

The tame setting

(We consider extensions of $\mathbb{Z}_p^d$ by Q.)

Theorem (Grochow and Qiao (2015))

If the group algebra $\overline{\mathbb{F}_p}Q$ is tame, then the extension pseudo-congruence problem can be solved.

The tame setting

(We consider extensions of $\mathbb{Z}_p^d$ by Q.)

Theorem (Grochow and Qiao (2015))

If the group algebra $\overline{\mathbb{F}_p}Q$ is tame, then the extension pseudo-congruence problem can be solved.

$\overline{\mathbb{F}_p}Q$ is tame, iff the Sylow p-subgroup of Q is:

- cyclic. (Finite; Higman (1954).)
- p=2 and dihedral, semi-dihedral, or generalized quaternion. (Tame and not finite; Bondarenko (1975), Ringel (1975), Bondarenko and Drozd (1982) and Crawley-Boevey (1989).)

Othercases are wild (Kruglyak (1963) and Brenner (1970)).

The difference b/w tame and wild

Theorem

Let $n(Q, p, d)$ be the number of indecomposable modules of Q over $\mathbb{F}_p$ of dimension d.

- *If $\overline{\mathbb{F}_p}Q$ is tame, then $n(Q, p, d) \leq \mathrm{poly}(|Q|, p^d)$.*
- *(J. Rickard) If wild, then $n(Q, p, d) = p^{\Omega(d^2)}$.*

The difference b/w tame and wild

Theorem

Let $n(Q, p, d)$ be the number of indecomposable modules of Q over $\mathbb{F}_p$ of dimension d.

- *If $\overline{\mathbb{F}_p}Q$ is tame, then $n(Q, p, d) \leq \mathrm{poly}(|Q|, p^d)$.*
- *(J. Rickard) If wild, then $n(Q, p, d) = p^{\Omega(d^2)}$.*

Some remarks:

- Does not follow from the definition of tame/wild because of finite fields.
- Rather, this is about determining the number of 1-parameter families and finite cases.

The difference b/w tame and wild

Theorem

Let $n(Q, p, d)$ be the number of indecomposable modules of Q over $\mathbb{F}_p$ of dimension d.

- *If $\overline{\mathbb{F}_p}Q$ is tame, then $n(Q, p, d) \leq \mathrm{poly}(|Q|, p^d)$.*
- *(J. Rickard) If wild, then $n(Q, p, d) = p^{\Omega(d^2)}$.*

Some remarks:

- Does not follow from the definition of tame/wild because of finite fields.
- Rather, this is about determining the number of 1-parameter families and finite cases.
- Finite case is known by Higman (1954).
- Wild case by explicit construction.
- Tame case by examining the explicit classification as in Crawley-Boevey (1989).

The cohomology aspect

Theorem

Let $m(Q, p, d)$ be the order of the 2-cohomology group of Q w.r.t. any fixed $\mathbb{F}_p Q$ module of dimension d. If $\overline{\mathbb{F}_p Q}$ is tame, then $m(Q, p, d) \leq p^{3d}$.

The cohomology aspect

Theorem

Let $m(Q, p, d)$ be the order of the 2-cohomology group of Q w.r.t. any fixed $\mathbb{F}_p Q$ module of dimension d. If $\overline{\mathbb{F}_p Q}$ is tame, then $m(Q, p, d) \leq p^{3d}$.

The algorithm: given two 2-cocycles $f, g : Q \times Q \to \mathbb{Z}_p^d$ w.r.t. $\mathbb{F}_p Q$ module M:

1. Compute $J \leq \mathrm{Aut}(\mathbb{Z}_p^d) \times \mathrm{Aut}(Q)$ that preserves M;
2. View the given two 2-cocycles as two points in $H^2(Q, M)$;
   - The problem reduces to test if some $\alpha \in J$ that sends f to g.
3. Apply the pointwise transporter algorithm.
   - Runs in time $\mathrm{poly}(|H^2(Q, M)|)$.

The cohomology aspect

Theorem

Let $m(Q, p, d)$ be the order of the 2-cohomology group of Q w.r.t. any fixed $\mathbb{F}_p Q$ module of dimension d. If $\overline{\mathbb{F}_p Q}$ is tame, then $m(Q, p, d) \leq p^{3d}$.

The algorithm: given two 2-cocycles $f, g : Q \times Q \to \mathbb{Z}_p^d$ w.r.t. $\mathbb{F}_p Q$ module M:

1. Compute $J \leq \mathrm{Aut}(\mathbb{Z}_p^d) \times \mathrm{Aut}(Q)$ that preserves M;
2. View the given two 2-cocycles as two points in $H^2(Q, M)$;
   - The problem reduces to test if some $\alpha \in J$ that sends f to g.
3. Apply the pointwise transporter algorithm.
   - Runs in time $\mathrm{poly}(|H^2(Q, M)|)$.

(Ingredients from permutation group algorithms (Luks, 1991) and routines about 2-cohomology classes (Grochow and Qiao, 2014)).)

The last slide. . .

In this work, we show:

- A concrete example on how the tame-wild dichotomy affects the efficiency of an algorithm for group isomorphism test.
- The bounds rely critically on the known descriptions of indecomposables for semi-dihedral groups.

The last slide. . .

In this work, we show:

- A concrete example on how the tame-wild dichotomy affects the efficiency of an algorithm for group isomorphism test.
- The bounds rely critically on the known descriptions of indecomposables for semi-dihedral groups.

Question for further study:

- Go into the wild!

The last slide. . .

In this work, we show:

- A concrete example on how the tame-wild dichotomy affects the efficiency of an algorithm for group isomorphism test.
- The bounds rely critically on the known descriptions of indecomposables for semi-dihedral groups.

Question for further study:

- Go into the wild!

The algorithm

For the action aspect: given two $\mathbb{F}_p Q$ modules M and N of dimension d. Let R be the set of indecomposables of $\mathbb{F}_p Q$ of dimension $\leq d$.

The algorithm

For the action aspect: given two $\mathbb{F}_p Q$ modules M and N of dimension d. Let R be the set of indecomposables of $\mathbb{F}_p Q$ of dimension $\leq d$.

1. Decompose M and N into indecomposables, and group them by isomorphism types;
   - $M = L_1^3 \oplus L_2^3 \oplus L_3^2$, and $N = L_1^2 \oplus L_2^3 \oplus L_3^3$.
2. The induced action of $\mathrm{Aut}(Q)$ permutes the indecomposables;
   - The problem reduces to test whether there exists $\alpha \in \mathrm{Aut}(Q)$ s.t. $\alpha(\{L_1, L_2\}) = \{L_2, L_3\}$ and $\alpha(\{L_3\}) = \{L_1\}$.
3. For $S, T \subseteq \Omega$, test whether there exists $\alpha(S) = T$ is the setwise transporter problem. Solvable in time $\mathrm{poly}(|R|, 2^{|S|})$.

The algorithm

For the action aspect: given two $\mathbb{F}_p Q$ modules M and N of dimension d. Let R be the set of indecomposables of $\mathbb{F}_p Q$ of dimension $\leq d$.

1. Decompose M and N into indecomposables, and group them by isomorphism types;
   - $M = L_1^3 \oplus L_2^3 \oplus L_3^2$, and $N = L_1^2 \oplus L_2^3 \oplus L_3^3$.
2. The induced action of $\mathrm{Aut}(Q)$ permutes the indecomposables;
   - The problem reduces to test whether there exists $\alpha \in \mathrm{Aut}(Q)$ s.t. $\alpha(\{L_1, L_2\}) = \{L_2, L_3\}$ and $\alpha(\{L_3\}) = \{L_1\}$.
3. For $S, T \subseteq \Omega$, test whether there exists $\alpha(S) = T$ is the setwise transporter problem. Solvable in time $\mathrm{poly}(|R|, 2^{|S|})$.

(Ingredients from computational representation theory (Chistov et al., 1997; Brooksbank and Luks, 2008) and permutation group algorithms (Luks, 1999; Babai and Qiao, 2012).)