PQC Group-based Lecture 3: Computability of Decision and Search Problems in Combinatorial and Computational Groups Theory

Delaram Kahrobaei

CUNY

September 9, 2024

Outline

Combinatorial Group Theory is an area of mathematics lying in the intersection of

- Group Theory
- Mathematical Logic
- Topology
- Theoretical computer Science.

The *hard* one-way functions that are used in group based cryptosystems are based for the most part on algorithmic **group decision and search problems**.
We define some of the decision and search problems in combinatorial group theory, that have been used for non-commutative cryptography.
In general all are algorithmically insolvable.
What is of importance is how difficult, when solvable, they are for a given platform group.

Definition

Decision problems are problems of the following nature: given a property P and an object O, find out whether or not the object O has the property P.

Definition

Witness problems are: given a property P and an object O with the property P, find a proof of the fact that O indeed has the property P.

Definition

Search problems are of the following nature: given a property P and an object O with the property P, find something material establishing the property P.

Definition (Word Decision Problem)

Given a finitely presented group G does there exist an algorithm to decide whether or not a word in the generators is the trivial word?

Definition (Word Search Problem)

Given a finitely presented group G and a w which presents the identity, does there exist an algorithm to find an expression of w as a product of words of the form $f_i^{-1} r_i f_i$ where r_i is a relator of the group G and f_i is a word in the ambient free group.

Definition (Decision Conjugacy Problem)

Given a group G with a finite presentation, does there exist an algorithm to decide whether or not an arbitrary pair of words u and v in the generators of G are conjugate? That is, is there an $x \in G$ such that $x^{-1}ux = v$?

Definition (Conjugator Search Problem)

Let G be a finitely presented group. Given two conjugate words u and v , is there an algorithm to find a z such that $z^{-1}uz = v$?

Definition (Power Decision Conjugacy Problem)

Let G be a finitely presented group. Given two words u and v in the generators of G is there an algorithm to decide whether or not u and v are power conjugated? That is, does there exist $m, n \in \mathbb{Z}$ such that u^m and v^n are non-trivial and conjugate?

Definition (Power Conjugator Search Problem)

Let G be a finitely presented group. Given two power conjugated words u and v, is there an algorithm to find a z such that $z^{-1}u^m z = v^n$?

Definition (Simultaneous Conjugator Search problem)

Let G be a finitely presented group. Given $u_1, \cdots, u_k, v_1, \cdots, v_k \in G$ with $x^{-1}u_i x = v_i$ for each $i \in \{1, 2, \cdots, k\}$, is there an algorithm to find $z \in G$ satisfying $z^{-1}u_i z = v_i$ for each $i \in \{1, 2, ..., k\}$?

The problems are not independent, for example in a finitely presented group G the word problem of G is Turing reducible to the conjugacy problem of G. That is, $WP(G) \leq_T CP(G)$.

Example

If G is a finite group given by a multiplication table presentation, it is easy to describe algorithms for solving Decision and Search Word Problems and Conjugacy problems.

Example

If G is a finitely generated abelian group, then the word and conjugacy problems for G are solvable.

Example

If $F = \langle x_1, \cdots, x_n; \rangle$ is a finitely generated free group: WP(F) is solved by freely reducing. CP(F) is also solvable.

Example

The word problem for nilpotent group is solvable.

Example

Let G be a polycyclic group. Using the fact that every word in G has a normal form, we can conclude that G has solvable word problem.

Example

The word problem for nilpotent group is solvable.

Example

Let G be a polycyclic group. Using the fact that every word in G has a normal form, we can conclude that G has solvable word problem.

One can show that for any finitely presented group such that every word has a normal form then there is an algorithm to solve the word problem for that group.

Definition

A set of objects is Recursive if there is an algorithm for deciding membership in the set.

Definition

A set S of objects is Recursively Enumerable if there is an algorithm for listing all the objects in S.

Lemma

Every recursive set is recursively enumerable.

Lemma

A set S is recursive if and only if both S and its complement are recursively enumerable.

There exists a set which is recursively enumerable but not recursive. This fact is in a sense the source of all undecidability results in mathematics.

Lemma

A set S is recursive if and only if both S and its complement are recursively enumerable.

There exists a set which is recursively enumerable but not recursive. This fact is in a sense the source of all undecidability results in mathematics.

WP and CP are recursively enumerable in the sense that the collection of questions for which the answer is "Yes" is recursively enumerable.

Lemma

WP(G) is recursively enumerable.

Proof.

The set of words w of G such that $w =_G 1$ is recursively enumerable.
For it is the set of words freely equal to a product of conjugates of the given finite set of defining relations and this set can be systematically listed.
Thus WP(G) is recursively enumerable. □

WP and CP are recursively enumerable in the sense that the collection of questions for which the answer is "Yes" is recursively enumerable.

Lemma

$WP(G)$ is recursively enumerable.

Proof.

The set of words w of G such that $w =_G 1$ is recursively enumerable.
For it is the set of words freely equal to a product of conjugates of the given finite set of defining relations and this set can be systematically listed.
Thus WP(G) is recursively enumerable. □

Now WP(G) is recursively solvable (decidable) exactly when the set of words $\{w \in G | w =_G 1\}$ is recursive. So WP(G) is recursively solvable if and only if $\{w \in G | w \neq_G 1\}$ is recursively enumerable.

Definition

A recursive presentation is a presentation of the form $\langle x_1, \cdots, x_n; R_1 = 1, R_2 = 1, \cdots \rangle$ where $R_1, R_2, \cdots$ is a recursively enumerable set of words.

A finitely generated group G is recursively presented if it has a recursive presentation. Of course finitely presented groups are recursively presented but the converse is false.

Definition

A recursive presentation is a presentation of the form $\langle x_1, \cdots, x_n; R_1 = 1, R_2 = 1, \cdots \rangle$ where $R_1, R_2, \cdots$ is a recursively enumerable set of words.

A finitely generated group G is recursively presented if it has a recursive presentation. Of course finitely presented groups are recursively presented but the converse is false.

The word problem and conjugacy problem are defined for recursively presented groups as before and they are still recursively enumerable problems.

Theorem

Conjugacy search problem is always solvable.

Proof.

To see that the conjugacy search problem is always solvable, we use a straightforward algorithm: recursively enumerate all words in the given generators of G, then go over all these words g one at a time, comparing $g^{-1}w_1g$ to w_2 by using the fact that the yes part of the word problem is solvable in any recursively presented group G. The crucial point here is that when we say comparing two elements, we mean initiating the obvious procedure for the yes part of the word problem. However, after initiating such a procedure we do not just sit there waiting for a result because we do not know how long we have to wait (perhaps indefinitely); instead, we move on to the next word, initiate the relevant procedure for the yes part of the word problem, etc. □

Definition

If A and B are two sets of objects, A is Turing reducible to B, $A \leq_T B$ if knowing membership for B can give an algorithm for computing membership for A. Thus the decision problem for A is reducible to that for B.

Definition

Two sets of objects A and B are Turing equivalent $A \equiv_T B$ if each is Turing reducible to the other, that is both $A \leq_T B$ and $B \leq_T A$.

Definition

If A and B are two sets of objects, A is Turing reducible to B, $A \leq_T B$ if knowing membership for B can give an algorithm for computing membership for A. Thus the decision problem for A is reducible to that for B.

Definition

Two sets of objects A and B are Turing equivalent $A \equiv_T B$ if each is Turing reducible to the other, that is both $A \leq_T B$ and $B \leq_T A$.

Lemma

Let G be a finitely presented group. Then the word problem of G is Turing reducible to the conjugacy problem of G. In other words $WP(G) \leq_T CP(G)$.

Lemma

Let G be a finitely generated group given by a recursive presentation

$$G = \langle x_1, \cdots, x_n; R_1 = 1, R_2 = 1, \cdots \rangle.$$

Suppose that H is a finitely generated group with generators $y_1, \cdots, y_m$ and that $\phi : H \to G$ is an injective homomorphism. Then H has a recursive presentation of the form

$$H = \langle y_1, \cdots, y_n; Q_1 = 1, Q_2 = 1, \cdots \rangle$$

where $Q_1, Q_2, \cdots$ is a recursively enumerable set of words in $y_1, \cdots, y_m$. Moreover, $WP(H) \leq_T WP(G)$.

Corollary

For finitely presented groups (respectively finitely generated, recursively presented groups), WP(G), CP(G) are algebraic invariants. That is, for any two presentations Π_1 *and* Π_2 *of the same group on a finite set of generators*

$$WP(\Pi_1) \equiv_T WP(\Pi_2), \text{ and } CP(\Pi_1) \equiv_T CP(\Pi_2).$$

Theorem

(Novikov-Boone) There exists a finitely presented group whose word problem is recursively unsolvable.

Proof.

Let $S \subset N$ be a recursively enumerable set of natural numbers which is not recursive. Define the recursively presented group

$$H_S = \langle a, b, c, d; a^{-i}ba^i = c^{-i}dc^i (\forall i \in S) \rangle$$

Now H_S can be described as the free product with amalgamation of the free group $\langle a, b; \rangle$ and the free group $\langle c, d; \rangle$ amalgamating the subgroup (freely) generated by the left hand sides of the indicated equations with the subgroup (freely) generated by the right hand sides.

$$a^{-i}ba^i c^{-i}d^{-1}c^i =_{H_S} 1$$

if and only if $i \in S$. Thus $S \leq_T WP(H_S)$ and so $WP(H_S)$ is recursively unsolvable. □

Theorem

(Higman Embedding Theorem) A finitely generated group H can be embedded in a finitely presented group if and only if H is recursively presented.

Theorem

(Kharlampovich) There exists a solvable group of class 3 *with unsolvable word problem.*