DRP F2024 Presentation

Brenda Li & Micky Liu

Dec 2nd, 2024

University of Waterloo

Introduction to Word Problem

Statement of Word Problem in Group Theory

Word Problem in Group Theory

Let G be a finitely presented group give by the presentation $\langle S \mid R \rangle$. Can we decided whether a given word in S is trivial in G?

Statement of Word Problem in Group Theory

Word Problem in Group Theory

Let G be a finitely presented group give by the presentation $\langle S \mid R \rangle$. Can we decided whether a given word in S is trivial in G?

To understand the rigor behind this question, we need to know

1. What is a group? What is a word?

Statement of Word Problem in Group Theory

Word Problem in Group Theory

Let G be a finitely presented group give by the presentation $\langle S \mid R \rangle$. Can we decided whether a given word in S is trivial in G?

To understand the rigor behind this question, we need to know

1. What is a group? What is a word?
2. What is a generator / generating set in a group?

Statement of Word Problem in Group Theory

Word Problem in Group Theory

Let G be a finitely presented group give by the presentation $\langle S \mid R \rangle$. Can we decided whether a given word in S is trivial in G?

To understand the rigor behind this question, we need to know

1. What is a group? What is a word?
2. What is a generator / generating set in a group?
3. What is a group presentation? What is a finitely presented group?

Group

Definition: Group

Let G be a set and $*$ be an operation defined on $G \times G$. We say $G = (G, *)$ is a group if the following are satisfied

1. **Identity**: $\exists 1 \in G$ such that $1 * g = g * 1 = g, \forall g \in G$. In this case, e is said to be the identity of G.
2. **Inverse**: $\forall a \in G, \exists b \in G$ such that $ab = 1$. In this case, b is said to be the inverse of a.
3. **Closure**: $\forall a, b \in G, ab \in G$
4. **Associativity**: $\forall a, b, c \in G, a(bc) = (ab)c$

Group

Definition: Group

Let G be a set and $*$ be an operation defined on $G \times G$. We say $G = (G, *)$ is a group if the following are satisfied

1. **Identity**: $\exists 1 \in G$ such that $1 * g = g * 1 = g, \forall g \in G$. In this case, e is said to be the identity of G.
2. **Inverse**: $\forall a \in G, \exists b \in G$ such that $ab = 1$. In this case, b is said to be the inverse of a.
3. **Closure**: $\forall a, b \in G, ab \in G$
4. **Associativity**: $\forall a, b, c \in G, a(bc) = (ab)c$

For example,

- $\mathbb{Z} \subseteq \mathbb{Q} \subseteq \mathbb{R} \subseteq \mathbb{C}$ with addition

Group

Definition: Group

Let G be a set and $*$ be an operation defined on $G \times G$. We say $G = (G, *)$ is a group if the following are satisfied

1. **Identity**: $\exists 1 \in G$ such that $1 * g = g * 1 = g, \forall g \in G$. In this case, e is said to be the identity of G.
2. **Inverse**: $\forall a \in G, \exists b \in G$ such that $ab = 1$. In this case, b is said to be the inverse of a.
3. **Closure**: $\forall a, b \in G, ab \in G$
4. **Associativity**: $\forall a, b, c \in G, a(bc) = (ab)c$

For example,

- $\mathbb{Z} \subseteq \mathbb{Q} \subseteq \mathbb{R} \subseteq \mathbb{C}$ with addition
- Invertible matrices with matrix multiplication.

Word

Definition: Word

Let S be a set. A word in S is a finite string of symbols in S and their inverses. We can "stick" two words together by concatenation. We can also a "simplify" a word by canceling out the symbols and the adjacent inverses.

Definition: Word

Let S be a set. A word in S is a finite string of symbols in S and their inverses. We can "stick" two words together by concatenation. We can also a "simplify" a word by canceling out the symbols and the adjacent inverses.

Take $S = \{a, b\}$ and consider two words

- $s_1 = abab^{-1}a^{-1}, s_2 = ababa$

Word

Definition: Word

Let S be a set. A word in S is a finite string of symbols in S and their inverses. We can "stick" two words together by concatenation. We can also a "simplify" a word by canceling out the symbols and the adjacent inverses.

Take $S = \{a, b\}$ and consider two words

- $s_1 = abab^{-1}a^{-1}, s_2 = ababa$
- $s_1 s_2 = abab^{-1}a^{-1}ababa = abaaba$

Relators and Presentation

Question: What does $\langle S \mid R \rangle$ mean?

Relators and Presentation

Question: What does $\langle S \mid R \rangle$ mean?

- $\langle S \rangle$ gives the set of all simplified words in S with concatenation.

Relators and Presentation

Question: What does $\langle S \mid R \rangle$ mean?

- $\langle S \rangle$ gives the set of all simplified words in S with concatenation.
- The set of R is the additional "hidden information" or "toolbox" for us to further simplify a word in S.

Relators and Presentation

Question: What does $\langle S \mid R \rangle$ mean?

- $\langle S \rangle$ gives the set of all simplified words in S with concatenation.
- The set of R is the additional "hidden information" or "toolbox" for us to further simplify a word in S.

Group Presentation

G has the presentation $\langle S \mid R \rangle$ if every element of G is a word in S, where we can simplify the word based on the rules in R.

Relators and Presentation

Question: What does $\langle S \mid R \rangle$ mean?

- $\langle S \rangle$ gives the set of all simplified words in S with concatenation.
- The set of R is the additional "hidden information" or "toolbox" for us to further simplify a word in S.

Group Presentation

G has the presentation $\langle S \mid R \rangle$ if every element of G is a word in S, where we can simplify the word based on the rules in R.

Theorem: Existence of Group Presentation

Every group G admits a group presentation $\langle S \mid R \rangle$ where S is a set and $R \subseteq G$

Relators and Presentation

Question: What does $\langle S \mid R \rangle$ mean?

- $\langle S \rangle$ gives the set of all simplified words in S with concatenation.
- The set of R is the additional "hidden information" or "toolbox" for us to further simplify a word in S.

Group Presentation

G has the presentation $\langle S \mid R \rangle$ if every element of G is a word in S, where we can simplify the word based on the rules in R.

Theorem: Existence of Group Presentation

Every group G admits a group presentation $\langle S \mid R \rangle$ where S is a set and $R \subseteq G$

Definition: Finitely Presented

A group G is said to be finitely presented if it admits a presentation $\langle S \mid R \rangle$ where S, R are finite sets.

Example

- Cyclic group of order n

$$C_n := \{1, a, a^2, a^3, \ldots, a^{n-1}\}$$

Example

- Cyclic group of order n

$$C_n := \{1, a, a^2, a^3, \dots, a^{n-1}\}$$

  - This is also known $\mathbb{Z}_n$, the set of integer modulo n

Example

- Cyclic group of order n

$$C_n := \{1, a, a^2, a^3, \ldots, a^{n-1}\}$$

  - This is also known $\mathbb{Z}_n$, the set of integer modulo n
  - $C_n = \langle a \mid a^n = 1 \rangle$

Example

- Cyclic group of order n

$$C_n := \{1, a, a^2, a^3, \ldots, a^{n-1}\}$$

  - This is also known $\mathbb{Z}_n$, the set of integer modulo n
  - $C_n = \langle a \mid a^n = 1 \rangle$

- Dihedral group of order 8

$$D_8 = \{1, a, a^2, a^3, a^4, ba, ba^2, ba^3\}$$

Example

- Cyclic group of order n

$$C_n := \{1, a, a^2, a^3, \dots, a^{n-1}\}$$

  - This is also known $\mathbb{Z}_n$, the set of integer modulo n
  - $C_n = \langle a \mid a^n = 1 \rangle$
- Dihedral group of order 8

$$D_8 = \{1, a, a^2, a^3, a^4, ba, ba^2, ba^3\}$$

  - Geometric Interpretation: Group of all symmetries on a square

Example

- Cyclic group of order n

$$C_n := \{1, a, a^2, a^3, \ldots, a^{n-1}\}$$

  - This is also known $\mathbb{Z}_n$, the set of integer modulo n
  - $C_n = \langle a \mid a^n = 1 \rangle$

- Dihedral group of order 8

$$D_8 = \{1, a, a^2, a^3, a^4, ba, ba^2, ba^3\}$$

  - Geometric Interpretation: Group of all symmetries on a square
  - $D_8 = \langle a, b \mid a^2 = b^4 = (ab)^2 = 1 \rangle$

Revisit to Word Problems

Word Problem in Group Theory

Let G be a finitely presented group given by the presentation $\langle S \mid R \rangle$.
Can we decided whether a given word in S is trivial in G?

Revisit to Word Problems

Word Problem in Group Theory

Let G be a finitely presented group given by the presentation $\langle S \mid R \rangle$. Can we decided whether a given word in S is trivial in G?

Equivalent Description of Word Problem

Let G be a finitely presented group given by the presentation $\langle S \mid R \rangle$. Can we decided whether two words in S, say s_1, s_2, are equivalent in G?

Revisit to Word Problems

Word Problem in Group Theory

Let G be a finitely presented group given by the presentation $\langle S \mid R \rangle$.
Can we decided whether a given word in S is trivial in G?

Equivalent Description of Word Problem

Let G be a finitely presented group given by the presentation $\langle S \mid R \rangle$.
Can we decided whether two words in S, say s_1, s_2, are equivalent in G?

A naive (intuitive but not so rigorous) example:

- Social media account?

Revisit to Word Problems

Word Problem in Group Theory

Let G be a finitely presented group given by the presentation $\langle S \mid R \rangle$. Can we decided whether a given word in S is trivial in G?

Equivalent Description of Word Problem

Let G be a finitely presented group given by the presentation $\langle S \mid R \rangle$. Can we decided whether two words in S, say s_1, s_2, are equivalent in G?

A naive (intuitive but not so rigorous) example:

- Social media account?
- Mysterious Question: Is "miicky" and "miiiicky" the same person?

Revisit to Word Problems

Word Problem in Group Theory

Let G be a finitely presented group given by the presentation $\langle S \mid R \rangle$. Can we decided whether a given word in S is trivial in G?

Equivalent Description of Word Problem

Let G be a finitely presented group given by the presentation $\langle S \mid R \rangle$. Can we decided whether two words in S, say s_1, s_2, are equivalent in G?

A naive (intuitive but not so rigorous) example:

- Social media account?
- Mysterious Question: Is "miicky" and "miiiicky" the same person?
- $S = \{m, i, c, k, y\}$, $R = \{micky\}$

Solvable Word Problem

Solvability of Word Problem

Let $G = \langle S \mid R \rangle$. We say G has a **solvable** word problem if there exists an algorithm such that, given any word s_1, s_2 in S, we can determine if $s_1 = s_2$

Solvable Word Problem Example

Definition: Free Group

A group G is said to be a free group if it admits a presentation $\langle S \mid \emptyset \rangle$

Solvable Word Problem Example

Definition: Free Group

A group G is said to be a free group if it admits a presentation $\langle S \mid \emptyset \rangle$

Theorem

Word problem on free group $G = \langle S \mid \emptyset \rangle$ is solvable.

Solvable Word Problem Example

- The cyclic group of order n: $C_n = \langle a \mid a^n = 1 \rangle$
    - Any word has the form $a^k, k \in \mathbb{Z}$.

Solvable Word Problem Example

- The cyclic group of order n: $C_n = \langle a \mid a^n = 1 \rangle$
  - Any word has the form $a^k, k \in \mathbb{Z}$.
  - In C_7, consider the words
    $s_1 = 17, s_2 = 27 \implies s_1 = [3], s_2 = [6] \implies s_1 \neq s_2$

Solvable Word Problem Example

- The cyclic group of order n: $C_n = \langle a \mid a^n = 1 \rangle$
  - Any word has the form $a^k, k \in \mathbb{Z}$.
  - In C_7, consider the words
    $s_1 = 17, s_2 = 27 \implies s_1 = [3], s_2 = [6] \implies s_1 \neq s_2$
  - Algorithm: For $s_1 = a^{k_1}, s_2 = a^{k_2}, s_1 = s_2 \iff k_1 \equiv k_2 \mod n$

Solvable Word Problem Example

- The cyclic group of order n: $C_n = \langle a \mid a^n = 1 \rangle$
  - Any word has the form $a^k, k \in \mathbb{Z}$.
  - In C_7, consider the words $s_1 = 17, s_2 = 27 \implies s_1 = [3], s_2 = [6] \implies s_1 \neq s_2$
  - Algorithm: For $s_1 = a^{k_1}, s_2 = a^{k_2}, s_1 = s_2 \iff k_1 \equiv k_2 \mod n$
- The dihedral group of order 8: $D_8 = \langle a, b \mid a^2 = b^4 = (ab)^2 = 1 \rangle$
  - Consider two words $s_1 = a, s_2 = a^4bab^5$

Solvable Word Problem Example

- The cyclic group of order n: $C_n = \langle a \mid a^n = 1 \rangle$
  - Any word has the form $a^k, k \in \mathbb{Z}$.
  - In C_7, consider the words $s_1 = 17, s_2 = 27 \implies s_1 = [3], s_2 = [6] \implies s_1 \neq s_2$
  - Algorithm: For $s_1 = a^{k_1}, s_2 = a^{k_2}, s_1 = s_2 \iff k_1 \equiv k_2 \mod n$
- The dihedral group of order 8: $D_8 = \langle a, b \mid a^2 = b^4 = (ab)^2 = 1 \rangle$
  - Consider two words $s_1 = a, s_2 = a^4bab^5$
  - By considering the relators, we have $s_2 = a^4bab^5 = a(a)^2(ab)^2b^4 = a = s_1 \implies s_1 = s_2$

Solvable Word Problem Example

- The cyclic group of order n: $C_n = \langle a \mid a^n = 1 \rangle$
  - Any word has the form $a^k, k \in \mathbb{Z}$.
  - In C_7, consider the words $s_1 = 17, s_2 = 27 \implies s_1 = [3], s_2 = [6] \implies s_1 \neq s_2$
  - Algorithm: For $s_1 = a^{k_1}, s_2 = a^{k_2}, s_1 = s_2 \iff k_1 \equiv k_2 \bmod n$
- The dihedral group of order 8: $D_8 = \langle a, b \mid a^2 = b^4 = (ab)^2 = 1 \rangle$
  - Consider two words $s_1 = a, s_2 = a^4bab^5$
  - By considering the relators, we have $s_2 = a^4bab^5 = a(a)^2(ab)^2b^4 = a = s_1 \implies s_1 = s_2$
  - In fact, word problem on D_8 is solvable.

Solvable Word Problem Example

- The cyclic group of order n: $C_n = \langle a \mid a^n = 1 \rangle$
  - Any word has the form $a^k, k \in \mathbb{Z}$.
  - In C_7, consider the words $s_1 = 17, s_2 = 27 \implies s_1 = [3], s_2 = [6] \implies s_1 \neq s_2$
  - Algorithm: For $s_1 = a^{k_1}, s_2 = a^{k_2}, s_1 = s_2 \iff k_1 \equiv k_2 \bmod n$
- The dihedral group of order 8: $D_8 = \langle a, b \mid a^2 = b^4 = (ab)^2 = 1 \rangle$
  - Consider two words $s_1 = a, s_2 = a^4bab^5$
  - By considering the relators, we have $s_2 = a^4bab^5 = a(a)^2(ab)^2b^4 = a = s_1 \implies s_1 = s_2$
  - In fact, word problem on D_8 is solvable.
  - $\forall n \in \mathbb{N}$, word problem on $D_{2n} = \langle a, b \mid a^2 = b^n = (ab)^2 = 1 \rangle$ is solvable.

Solvable Word Problem Example

- Does every finitely presented group $G = \langle S \mid R \rangle$ have a solvable word problem?

Solvable Word Problem Example

- Does every finitely presented group $G = \langle S \mid R \rangle$ have a solvable word problem?
    - **NO**

Solvable Word Problem Example

- Does every finitely presented group $G = \langle S \mid R \rangle$ have a solvable word problem?
  - **NO**
- For a formal discussion on **unsolvability**, we need
  - What is an **algorithm**? - Turing Machine
  - What does it mean that a suitable algorithm **doesn't exist**? - Halting and Decidability

Unsolvability of Word Problems

Connection: Solvable Word Problem to Algorithms

Let $G = \langle S \mid R \rangle$ be a finitely presented group.

Recall: We say that G has a *solvable word problem* if there exists an algorithm that, given any word $w \in S$, determines whether $w =_G 1$.

Implication:

If the word problem for G is solvable, there exists an algorithm (i.e., a finite and well-defined computational procedure) that always halts and determines whether $w = 1$.

Connection: Algorithms to Turing Machines

Definition: A *Turing machine* is a mathematical model of computation that provides a precise definition of an algorithm.

- Tape: The tape is divided into squares, each of which can be blank or contain a symbol.
- Tape head: The head can move along the tape, reading, writing, erasing, and changing internal states.
- Control mechanism: The head's control mechanism stores instructions from a finite set.
- Internal states: The machine can only be in one of a finite number of internal states at any given time.

Continuation on Turing Machine

Key property:
Algorithms are represented by Turing machines: if an algorithm exists to solve a problem, a corresponding Turing machine can simulate it and in this process halts.
Halting: A Turing machine *halts* if it reaches a state where no further steps are defined. Halting indicates that the algorithm has completed its computation and produced an output. **(Terminal state)**

Implication

Thus, solvability $\iff$ computability $\iff$ a corresponding Turing machine always halts for inputs representing words in G.

Key Note: The solvability of the word problem guarantees termination because the input w is finite, and the algorithm is assumed to reduce w to a canonical form in a finite number of steps.

However, determining whether a Turing machine halts or not is not very feasible, is it?

To address this, modular machines are used to break complex computations into smaller, well-defined modules.

Modular Machine

Definition: A modular machine M is a computational model similar to a Turing machine but designed with modular transitions. A computation on M is a finite sequence of configurations:

$$(\alpha, \beta) = (\alpha_1, \beta_1) \to (\alpha_2, \beta_2) \to \ldots \to (\alpha_t, \beta_t) = (\bar{\alpha}, \bar{\beta}),$$

where $(\bar{\alpha}, \bar{\beta})$ is a terminal configuration.

Theorem:

- Any Turing machine T can be simulated by a modular machine M.
- Modular machines and Turing machines have equivalent definitions of halting and undecidability.

Undecidability and Modular Machines

Corollary: Let M be a modular machine. The set of halting configurations is

$$H_M = \{(\alpha, \beta) \mid (\alpha, \beta) \xrightarrow{M} (0,0)\}.$$

H_M is undecidable, as it directly corresponds to the undecidability of the halting problem for Turing machines.

Implication: The equivalence between modular machines and Turing machines shows that modular machines are no more powerful than Turing machines but offer a practical framework for modularizing complex computations.

Group Relations Based on Transitions

To encode the behavior of the modular machine into a group G_0, we interpret each transition $(\alpha, \beta) \to (\alpha', \beta')$ as a group relation.

- Define a group generator $g_{(\alpha,\beta)}$ for each configuration (α, β).
- Each transition $(\alpha, \beta) \to (\alpha', \beta')$ corresponds to a relation:

$$g_{(\alpha,\beta)} = g_{(\alpha',\beta')} \cdot t,$$

  where t is a "transition element" introduced to distinguish between different states.

For a terminal configuration $(\bar{\alpha}, \bar{\beta})$, we enforce the relation:

$$g_{(\bar{\alpha},\bar{\beta})} =_G 1,$$

where 1 is the identity element in G_0.

The group defined by these relations captures the modular machine's behavior.

HNN Extensions

HNN extensions are a subsequent step, extending the base group G_0 into a new group G' that encodes recursion and higher-order complexity.

Definition: An HNN (Higman-Neumann-Neumann) extension is a way of constructing new groups by adding a stable letter p, where $p \in G'$ and $G \subseteq G'$. Furthermore, this p is found with the property that $p^{-1}ap = \phi(a), \forall a \in A$, essentially brings the presentation

$$G' = \langle G, p \mid p^{-1}ap = \phi(a),\ \forall a \in A \rangle = \mathrm{HNN}(G, A, B, \phi),$$

where A and B are isomorphic subgroups of G and $\phi : A \to B$ is an isomorphism.

This result of G' can be further extended to families of $\{A_i\}_{i \in I}, \{B_i\}_{i \in I}, \{\phi_i\}_{i \in I}$

Modular Machines and HNN Extensions

Process:

1. Define group relations based on transitions in the modular machine.
2. Construct an HNN extension to encode the machine's behavior into the group.

Unsolvable Word Problems

Key Theorem: There exist finitely presented groups with unsolvable word problems. In other words, no algorithm can determine whether a given word is equivalent to the identity in such groups.

Proof Sketch:

- The proof uses a reduction from the halting problem, a classic undecidable problem.
- By encoding Turing machines into groups (via constructions such as HNN extensions), one can create groups whose word problem corresponds to the halting problem.

Implications:

- The word problem in group theory is not uniformly solvable for all finitely presented groups.
- Decidability of the word problem depends on the specific group.

Conclusion: Unsolvable word problems highlight deep connections between group theory and computability.

Interesting Facts

From HNN construction, word problem is not always solvable :(

Interesting Facts

From HNN construction, word problem is not always solvable :(

- A unsolvable word problem can also be very harmless looking

Interesting Facts

From HNN construction, word problem is not always solvable :(

- A unsolvable word problem can also be very harmless looking
- Shrink the size of S, R in an unsolvable word problem

Interesting Facts

From HNN construction, word problem is not always solvable :(

- A unsolvable word problem can also be very harmless looking
- Shrink the size of S, R in an unsolvable word problem
- A weaker version of word problem: with the same setup of word problem, can we determine whether two words being conjugate? i.e. ($s_1 = g s_2 g^{-1}$ for some $g \in G$)

Interesting Facts

From HNN construction, word problem is not always solvable :(

- A unsolvable word problem can also be very harmless looking
- Shrink the size of S, R in an unsolvable word problem
- A weaker version of word problem: with the same setup of word problem, can we determine whether two words being conjugate? i.e. ($s_1 = gs_2g^{-1}$ for some $g \in G$)
    - **NO** :(

References

Dummit, S. D., Foote, M. R. (2004). *Abstract Algebra*, Third Edition (3rd ed.). John Wiley and Sons

Touikan, N, *HNN extensions and dual tracks*. https://ntouikan.ext.unb.ca/MATH6022/IntroCGGT/html_output/section-16.html

Cravitz, W., *An introduction to word problem for groups* https://math.uchicago.edu/~may/REU2021/REUPapers/Cravitz.pdf