13/03/26

Complexity working group.

Algorithmic problems in Group theory.

[Rotman - An Introduction to the Theory of groups ; chap 12 (~50 pages)]

I. Recursively enumerable and recursive sets.
II. Group presentations and the word problem.
III. Some applications/generalizations.

We fix an alphabet S, Ω the set of positive words on S (we can add a blank symbol if needed).

Rq: The set of nonempty word is a semigroup, i.e endowed with an associative composition law.

$$\text{Set} \supseteq \underset{\text{comp law}}{\text{Magma}} \supseteq \underset{\text{associative}}{\text{Semigroup}} \supseteq \underset{\text{id. elt}}{\text{Monoid}} \supseteq \underset{\text{inverability}}{\text{group}}.$$

I. Recursively enumerable and recursive sets.

1) Recursively enumerable sets

Def: We say that a Turing machine T computes $w \in \Omega$ if T running with w as input terminates. (after a finite # of steps).

Exple: supp $S = S \sqcup S^{-1}$ so that words are signed words in S.

(E1): If length$(w) \leq 0, 1$ go to state 3. and at the 1st letter of the word. Otherwise, underline the first adjacent pair of letters, if any, of the form $s_i s_i^{-1}$ or $s_i^{-1} s_i$. If no such pair exist, underline the final two lett. Then go to Step (E2) at the beginning of the word

(E2): If the underlined letter are either $s_i s_i^{-1}$ or $s_i^{-1} s_i$, erase it and proceed to step (E1) at the beginning of the word. Otherwise, proceed to step (3) at the beginning of the word.

(E3): If w is empty, write $w=1$ and stop. otherwise $w \neq 1$ and stop.

This machine T computes every word $w \in \Omega$.

Def: The set $e(T) = \{\omega \in \Sigma^* \mid T \text{ computes } \omega\}$ is said to be recursively enumerable (r.e.).

If a set E is r.e., we can try a first strategy to determine whether a word $\omega \in E$ or not.

→ Run T on ω. If this stops then $\omega \in E$.

Of course, if $\omega \notin E$, then we are going to wait for long...

2) An example.

We can easily build an alphabet S with symbols $<, !, >, ;, =$, + letters such that a finite presentation is an element of Σ^*.

<	a	,	b	!	a	b	=	b	a	>

We can check whether a finite word is indeed a finite presentation. So we can ensure that Σ^* is $\Sigma^* =$ finite presentations.

The Todd Coxeter algorithm is a Turing machine whose input is a finite pres and such that

- Output $= |G|$ if $|G|$ is finite, - run indefinitely otherwise.

Thus $e(T) =$ "finite presentations of finite groups" But we will see later that the question "is a finitely presented group finite" is undecidable.

3) Recursive set and the halting problem.

Def: A subset $E \subseteq \Sigma^*$ is recursive if both E and its complement are r.e.

Prop: Let $E \subseteq \Sigma^*$, the following are equivalent

(i) E is recursive

(ii) $\exists$ Turing machine T such that $T(\omega) = \begin{cases} 1 \text{ if } \omega \in E \\ 0 \text{ if } \omega \notin E \end{cases}$

dem: (ii) $\Rightarrow$ (i) is rather clear. (i) $\Rightarrow$ (ii) Let T_1, T_2 such that $e(T_1) = E$ $e(T_2) = \Sigma^* \setminus E$. make $T =$ run T_1, T_2 in parallel on ω, and output dep. on who terminates ②

Prop: Tasse?
(i) Every recursively enumerable set is enumerable
(ii) There exists a TM H solving the halting problem

Dem: Let E be r.e., attached to a machine T. We have

(ii)⇒(i) $\omega \in E \Leftrightarrow H(T,\omega) = 1$

Thus E is recursive.

(i)⇒(ii). Enumerate $(T_i)_{i\in\mathbb{N}}$ the Turing machines. $\forall i$, $\exists A_i$ s.t.

$A_i(\omega) = \begin{cases} 1 \text{ if } T_i(\omega) \text{ terminates} \\ 0 \text{ otherwise.} \end{cases}$ (by assumption this exists.)

Then create H this way.

Tele step j: Run one step of $A_0(\omega), A_1(\omega) \dots A_j(\omega)$.

→ $S(A_0)$; $S(A_0)\,S(A_1)$; $S A_0\, S A_1\, S A_2$; ... (words)

If A_i terminates, write the result on the ith piece of ribbon.

We then solve the halting problem by:
① Test $T = T_i$ on i if it works.
② Run the above algo until something is written on the ith piece of ribbon
③ Return that. □

By last week we deduce that there are $\aleph_0$ which are recursively enumerable while not being recursive; + there are non recursively enumerable sets by countability of Turing machines.

Cor: There is a Turing machine T for which the halting problem is undecidable.

II. Group presentation and word problem.

1). Semigroup of a Turing machine.

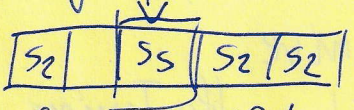
Like groups, semigroups can be defined by presentations with generators and relations (it is in fact easier).

It is (rather) easy to build a semigroup presentation starting from a Turing Machine T.

$Q = q_0 \dots q_R$ (q_0 = stop state, q_1 = start state).

$S = s_0 \dots s_M$ (s_0 = blank symbol)

The possible situations can be modeled by words:

$s_2 s_0 q_1 s_5 s_2 s_2$. <-> 

We are in state q_1 reading the next character on the tape, i.e s_5.

All the possible transition functions can then be interpreted as relations between word.

eg: * If in state q_i reading s_j, then move to the right and enter state q_k = $q_i s_j = s_j q_k$.

* ——— + change to s_0 ———

——— = $q_i s_j = s_0 q_k$

Def: Semigroup $\Gamma(T)$ attached to a Turing machine T

$\Gamma(T) = \langle q, h, S, Q \mid R(T) \rangle$.

with $R(T)$ indicate all the moves, h signifies beginning and end of an actual input, q represents "were finished"

eg: $h q_i s_i = h q_e s_0 s_i$ if we needed to move to the left in situation $q_i s_i$.

$+ h \dots q_0 \dots h = q$ "if we reached the terminal state, we stop".

A possible situation would be $h \underset{\in S}{\sigma} q_i \underset{\in S}{\tau} h$. "We are in 1 state, in 1 position".

Prop: If w is an S-word, then T terminates on w if and only if $h q_1 w h = q$ in $\Gamma(T)$.

all relations preserve "being a possible situation".

Thus solving the word problem on $\Gamma(T)$ implies solving the Termination problem on T.

Thm (Markov Post 1947)
There is a finitely presented semigroup with undecidable word problem

2) What about groups?

Prop: Let $G=(S,R)$ be a finite presentation. The word problem for G is solvable (by def) if $\{\omega \in S^* \mid \omega = 1 \text{ in } G\}$ is recursive.
This set is r.e

dem: Let $\{\omega_0 \dots \omega_i\} = S^* \simeq \mathbb{N}$, and $\{\rho_0 \dots \rho_i\} = \text{Words in } R \simeq \mathbb{N}$.
The set $\{\omega_i \rho_j \omega_i^{-1}\}$ is in bij with $\mathbb{N}$ too. Given ω, we can then "$\Pi_0 \dots \Pi_i \dots$"
do While $\omega \neq \Pi_i$, do $i := i+1$;

Theo (Novikov-Boone Britton ... S4-S7 S8 S9 ...)
$\exists$ finite group presentation w/ undecidable word problem.

dem: (20 pages → just the idea here). Build a group $B(T)$ from the semigroup $\Gamma(T)$, and show that solving the word problem for $B(T)$ implies (with additional gen k,t) solving it for $\Gamma(T)$.

Cor: There exist another group B^Δ (extension of B) which has solvable Word problem but unsolvable conjugacy problem

Theo: (Magnus 32) If R is a singleton (G is a "one relator group") then G has solvable word problem
+ we can determine the center of G!

3) A detail: the Higman embedding theorem.

Def: A presentation $R = \langle a_1 \dots a_m \mid \omega = 1, \omega \in E \rangle$, where each ω is a positive word, is called recursive if E is r.e.
↳ recursively presented group.

Thm (Higman 61)
Every recursively presented group R embeds in a finitely presented gp
(proof indep in theory from NBB, but the proof in Rotman uses it).

Prop: Higman $\Rightarrow$ NBB

Idea: Let $F = \text{Free}(a,b)$. The derived subgroup F' is free with basis $\{\omega_0 \dots \omega_n \dots\}$.
Let $E \subseteq \mathbb{N}$ be a r.e. not recursive set, then

$$G = \langle a, b, p \mid p^{-1}\omega_m p = \omega_n \ \forall n \in E \rangle.$$

is recursively presented: it embeds in a finitely pres group $G^* \not\cong G^*$
has solvable word problem, then so does G, but this implies that
E is recursive. #.

III. Some application/generalizations.

Thm (12.29). There exists a finitely presented group U containing isomorphic copies of every finitely presented group as a subgroup.

Thm (Boone Higman 74) 12.30
A finitely generated group G has solvable word problem iff G embeds in a simple subgroup of some finitely presented group.

3) Adyan Rabin theorem

lemma (Rabin 1958) 12.31 Let $G = \langle \Sigma, R \rangle$ be a finite presentation, there exists a map $P : \Sigma^* \to \{\text{finite presentations}\}$ such that

(i) If $w \neq 1$ in G, then $G \leq$ group pres by $R(w)$

(ii) If $w = 1$, then $R(w)$ presents the trivial group

Def : Soit $\mathcal{P}$ une propriété de théorie des groupes (invariante par isomorphisme)
On dit que $\mathcal{P}$ est de Markov si :

(i) $\exists\, G_+$ finiment présenté ayant la propriété $\mathcal{P}$.

(ii) $\exists\, G_-$ finiment présenté ne se plongeant dans aucun groupe finiment présenté ayant la propriété $\mathcal{P}$

Eg : Trivial ; finite ; p-group ; abelian ; solvable ; nilpotent ; torsion free ; having a solvable word problem ; being simple ; having solvable conjugacy problem ; being a Gromov group.

thm (Adyan Rabin 58) (12.32)
Any Markov property is undecidable for a fin. pres. grp

dem : Let $\mathcal{B}$ be a fin. pres group. Let $P = G_- * \mathcal{B}$, and let $R(w)$ be the groups given by Rabin's lemma, for w word in the gens of P.
Define $Q(w) = G_+ * R(w)$.
Let w be a word in the gens of $\mathcal{B}$.

* $w \neq 1$ in $\mathcal{B}$, then $G_- \leq P \leq R(w) \leq Q(w)$ and $Q(w)$ does not have property $\mathcal{P}$.
* $w = 1 \Rightarrow Q(w) \simeq G_+$ has property $\mathcal{P}$.

a decision procedure for $\mathcal{P}$ solves the word problem in $\mathcal{B}$. This is not always possible #.

Cor: The isomorphism problem for f.p. groups is undecidable.

Thm: Let H be a f.p. pres group H. The following are equivalent
* H is a universal f.p. pres group
* H satisfies no Markov property

dem: Assume H is universal, G_- embeds in H, and thus H does not satisfy $\mathcal{Z}$.
- The property "not being universal" is a Markov property (by existence).