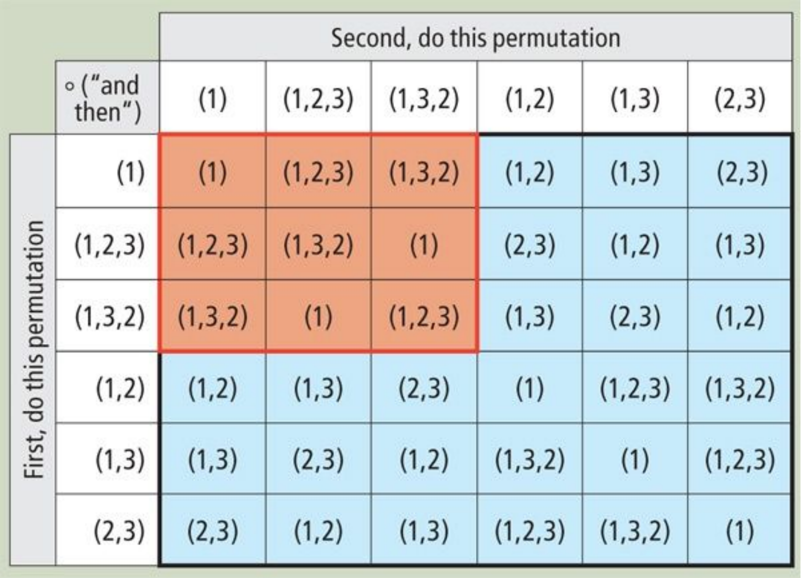
The Word Problem

Group Presentations

- A group is a set of elements G = {g1, g2…} and a binary operation (.): GxG->G such that
  - Identity: There exists an identity element 1 such that 1.g=g.1=g for all g in G
  - Associativity: (a.b).c=a.(b.c) for all a,b,c in G
  - Inverses: for all g, there exists h such that gh = hg = 1
- Knowing the set of elements and the operation gives us complete information about the group

Hence, a common way to represent groups is using a multiplication table



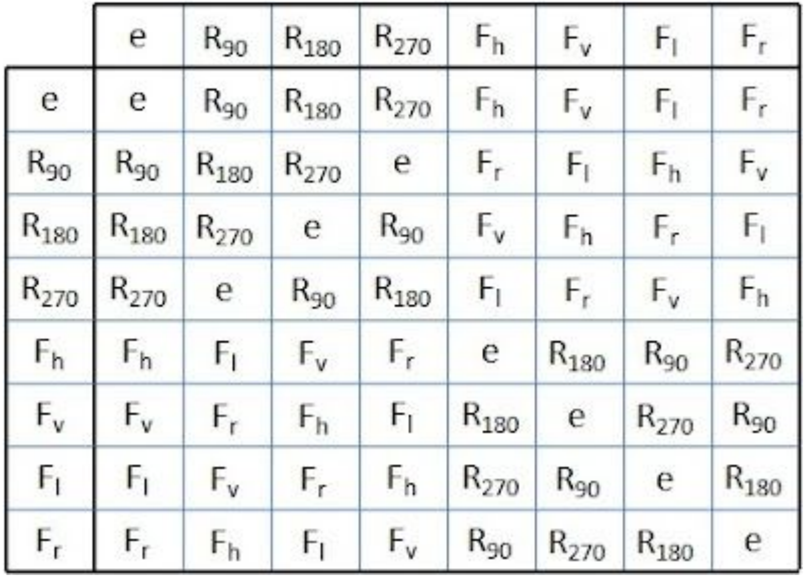
		Second, do this permutation					
	∘ (“and then”)	(1)	(1,2,3)	(1,3,2)	(1,2)	(1,3)	(2,3)
First, do this permutation	(1)	(1)	(1,2,3)	(1,3,2)	(1,2)	(1,3)	(2,3)
	(1,2,3)	(1,2,3)	(1,3,2)	(1)	(2,3)	(1,2)	(1,3)
	(1,3,2)	(1,3,2)	(1)	(1,2,3)	(1,3)	(2,3)	(1,2)
	(1,2)	(1,2)	(1,3)	(2,3)	(1)	(1,2,3)	(1,3,2)
	(1,3)	(1,3)	(2,3)	(1,2)	(1,3,2)	(1)	(1,2,3)
	(2,3)	(2,3)	(1,2)	(1,3)	(1,2,3)	(1,3,2)	(1)

Multiplication table for Symmetric Group S4

Group Presentations

- A group is a set of elements G = {g1, g2…} and a binary operation (.): GxG->G such that
    - Identity: There exists an identity element 1 such that 1.g=g.1=g
    - Associativity: (a.b).c=a.(b.c)
    - Inverses: for all g, there exists h such that gh = hg = 1
- Knowing the set of elements and the operation gives us complete information about the group

Hence, a common way to represent groups is using a multiplication table



	e	R_{90}	R_{180}	R_{270}	F_h	F_v	F_l	F_r
e	e	R_{90}	R_{180}	R_{270}	F_h	F_v	F_l	F_r
R_{90}	R_{90}	R_{180}	R_{270}	e	F_r	F_l	F_h	F_v
R_{180}	R_{180}	R_{270}	e	R_{90}	F_v	F_h	F_r	F_l
R_{270}	R_{270}	e	R_{90}	R_{180}	F_l	F_r	F_v	F_h
F_h	F_h	F_l	F_v	F_r	e	R_{180}	R_{90}	R_{270}
F_v	F_v	F_r	F_h	F_l	R_{180}	e	R_{270}	R_{90}
F_l	F_l	F_v	F_r	F_h	R_{270}	R_{90}	e	R_{180}
F_r	F_r	F_h	F_l	F_v	R_{90}	R_{270}	R_{180}	e

Multiplication table for Dihedral Group D4

Group Presentations

However, this description of a group is very large and has a lot of redundant information. We cannot describe infinite groups at all with this method.

What are some alternate Description of Groups?

- As embeddings in General Linear Groups → Representation Theory
- Group Presentations → Combinatorial Group Theory

Group Presentations

Definition 1: A presentation of a group G consists of

(1) A set of generators g1, g2… such that every element of G can be written as a product (a “word”) of the generators and their inverses
(2) A set of relations a1=b1, a2=b2, a3=b3… where each ai, bi is a word over the generators and their inverses

Group presentation is depicted as: <g1, g2… | a1=b1, a2=b2…> or simply <g1, g2…| a1b1^-1, a2b2^-1…>

Group Presentations

The group given by a presentation is the set of all words in the generators and their inverses, modulo the equivalence relation given by applying gigi^-1=1 and the relations in any part of the word, with multiplication given by [w1][w2]=[w1w2]

- Identity: [1][w]=[w]=[w][1]
- Associativity: ([w1][w2])[w3]=[w1w2][w3]=[w1w2w3]=[w1]([w2w3])=[w1]([w2][w3])
- Inverse: [w][w^-1]=[1]=[w^-1][w]

Group Presentations

Every group has a group presentation:

- Let the set of generators be all the elements in the group
- Let the relations be those given by the multiplication table

We will consider only groups with finite presentations for the remainder of the talk.

Group Presentations

Example 1: $\langle x \mid x^3 =1 \rangle$

- There are three equivalence classes/elements: $[1]$, $[x]$, $[x^2]$
- Isomorphic to $Z/3Z$

Group Presentations

Example 2: <x, y | xy=yx>

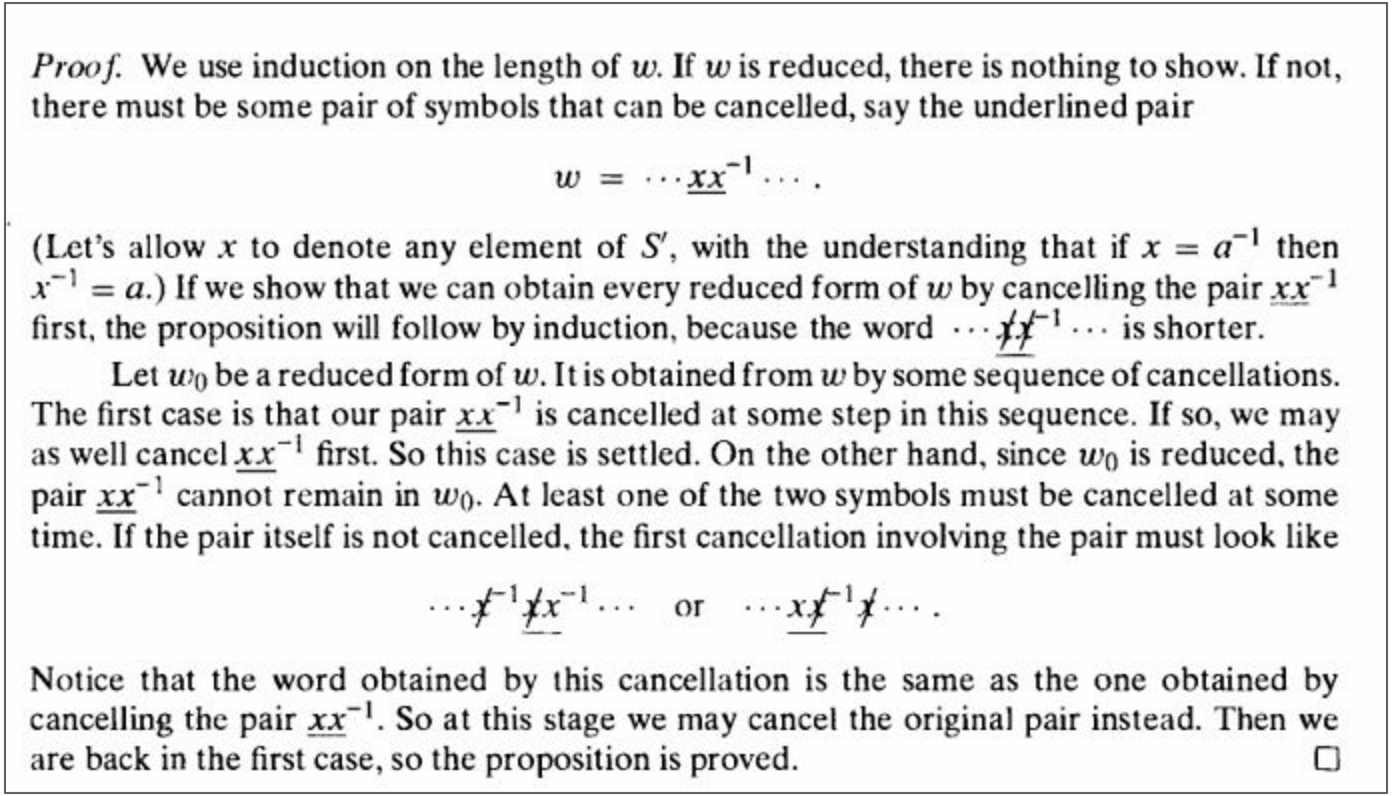
- In any word, we can rearrange the ‘x’s and ‘y’s to get a word of the form x^my^n
- [x^m1y^n1] and [x^m2y^n2] are distinct for (m1, n1)=/=(m2, n2)
- Group is Z^2!
- {x^my^n} gives a ‘normal form’ for the group

Normal Form: A choice of word in each equivalence class

Group Presentations

Example 3: <x1, x2, … xn>

- Known as the Free group on n elements or Fn
- Claim: The set of words S without any xixi^-1 present in them (‘freely reduced’) give a normal form for Fn
- Proof:
    - Any word w is equivalent to an element in S: Repeatedly cancel any xixi^-1 present in w. Cancellation reduces length each time, so the process stops at some point.
    - ‘Diamond’ Argument + A word cannot reduce to two distinct words in S: proof by induction…



Proof. We use induction on the length of w. If w is reduced, there is nothing to show. If not, there must be some pair of symbols that can be cancelled, say the underlined pair

$$w = \cdots \underline{xx}^{-1} \cdots .$$

(Let's allow x to denote any element of S', with the understanding that if $x = a^{-1}$ then $x^{-1} = a$.) If we show that we can obtain every reduced form of w by cancelling the pair $\underline{xx}^{-1}$ first, the proposition will follow by induction, because the word $\cdots \underline{\cancel{x}\cancel{x}}^{-1} \cdots$ is shorter.

Let w_0 be a reduced form of w. It is obtained from w by some sequence of cancellations. The first case is that our pair $\underline{xx}^{-1}$ is cancelled at some step in this sequence. If so, we may as well cancel $\underline{xx}^{-1}$ first. So this case is settled. On the other hand, since w_0 is reduced, the pair $\underline{xx}^{-1}$ cannot remain in w_0. At least one of the two symbols must be cancelled at some time. If the pair itself is not cancelled, the first cancellation involving the pair must look like

$$\cdots \cancel{x}^{-1}\underline{\cancel{x}x}^{-1} \cdots \quad \text{or} \quad \cdots \underline{x\cancel{x}}^{-1}\cancel{x} \cdots .$$

Notice that the word obtained by this cancellation is the same as the one obtained by cancelling the pair $\underline{xx}^{-1}$. So at this stage we may cancel the original pair instead. Then we are back in the first case, so the proposition is proved. □

Group Presentations

Definition 2: Given a group presentation <g1, g2..| r1, r2…>, the group G represented by the presentation is the Free Group on {g1, g2…} = <g1, g2…> modulo the Normal Subgroup generated by r1^-1, r2^-1…

Group Presentations

Example 4: Dihedral Group Dn

- With generators being a reflection and a rotation:

  <r, f | r^n=1, f^2=1, rf=fr^-1>

- With generators being two reflections:

  <f1, f2| f1^2=1, f2^2=1, (f1f2)^n=1>

The Word Problem

Q: Given a finite presentation, it’s not always so easy to tell what the group it represents actually is. In general, how easily can we determine properties of the group? For instance, is there an algorithm that takes in a group presentation and…

- … tells us if the group is finite?
- … tells us if the group is trivial?
- … gives us the multiplication table for the group, if the group is finite?

The Word Problem

Q: Given a finite presentation, it’s not always so easy to tell what the group it represents actually is. In general, how easily can we determine properties of the group? For instance, is there an algorithm that takes in a group presentation and…

- … tells us if the group is finite? No!
- … tells us if the group is trivial? No!
- … gives us the multiplication table for the group, if the group is finite? Yes! (via Tedd-Coxeter coset enumeration)

We will return to the first two questions later…

The Word Problem

Q: For any group presentation, is there an algorithm that…

- … takes in a word and determines if its trivial?
- … takes in two words and determines if they are equal?
- … takes in two words and determines if they are conjugates?

The Word Problem

Q: For any group presentation, is there an algorithm that…

- … takes in a word and determines if its trivial? - No, called The Word Problem
- … takes in two words and determines if they are equal? No
- … takes in two words and determines if they are conjugates? NO

The Word Problem

It is easy to do computations in Free Group, and we showed that any group G can be written as a quotient of a free group. Hence, it would very useful to know when a word is trivial in G. This motivates the question:

For every finite group presentation, does there exist an algorithm that takes in a word w in its generators (and inverses) and returns whether w is trivial in G?

This is called the Word Problem for G.

The Word Problem

In all the groups we discussed so far, the Word Problem was decidable (note that for finite groups and groups with computable normal forms, the word problem is decidable)

Using the Halting Problem, we will explicitly exhibit a group G for which the word problem is undecidable.

Word Problem for Semigroups

Semigroups: similar to groups, but elements need not have inverses.

We can use a similar idea of presentations with generators and relations for semigroups, with some modifications:

- We consider only positive words (i.e. words with no inverses) on the generators
- Two words w and w’ are equal if there is a finite sequence w = w1 -> w2 -> … wn = w’ where wi -> wi+1 is gotten by replacing aj with bj or replacing bj with aj for some relation aj = bj.

Word Problem for Semigroups

Note that semigroups have a lot less ‘cancellation’ resulting from the relations than in groups.

<a,b,c | ab=ac> : b = c in the group, but not in the semigroup

<x| x^3=x>: Has three elements when viewed as a semigroup (1, x, x^2) but only two (1,x) when viewed as a group.

Malcev’s Group: <a, b, c, d, u, v, x, y | au=bv, cu=dv, cx=dy>: ax=by in a group but not in a semigroup

Word Problem for Semigroups

- The word problem for semigroups is therefore about determining whether two words are equal rather than determining whether a word is trivial (which are equivalent in a group).

Using the Halting Problem, we will construct a Semigroup S with an undecidable word problem, i.e.:

There is a word w in S such that there is NO algorithm which takes as input a word w’ from S and determines whether w = w’

Word Problem for Semigroups

Recall our conventions for Turing Machines:

- Has alphabet s1, s2… sm and states q0, q1, q2… qn
- Represent tape position using SqiS’, where S is the tape contents to the left of the current position, S’ is the tape contents to the right and including the current position, and qi is the current state.
- Can either move left and go to a new state OR move right and go to a new state OR change current alphabet and go to a new state.
- Only ‘halt’ state is q0.
- s0 is the letter corresponding to a blank space.

Word Problem for Semigroups

We want to reduce the halting problem for Turing Machines to the word problem for Semigroups. We would like to do this by ‘simulating’ Turing Machines using semigroups, i.e. for any turing machine T, we want to come up with a semigroup S(T) such that:

1. S contains words corresponding to the possible configurations of T, including a special word q for the ‘halt’ configuration
2. If we go from one configuration to another in T, then their corresponding words should be equal

Since we showed that there exists a Turing Machine T with an undecidable halting problem, this will give us a semigroup S and a word w such that the Word Problem for S, q is undecidable.

Word Problem for Semigroups

Attempt at Construction:

- Generating set:
    - s0, s1… sm; q0, q1… qn and q (for the halt configuration).
- Relation for replacing the current letter:
    - qi sj = qi’ sj’
- Relation for moving to the left:
    - sj qi sk = qi’ sj sk
- Relation for moving to the right:
    - qi sj = sj qi’

- Issue when we reach the end of the tape. Introduce new generator ‘h’ that serves as an end-marker and let word corresponding to configuration c of the Turing Machine be hch

Word Problem for Semigroups

Final Construction:

- Generating set: s0, s1… sm; q0, q1… qn; q, h
- Relations for replacing the current letter:
  - qi sj = qi’ sj’
- Relation for moving to the left:
  - sj qi sk = qi’ sj sk
  - h qi sk = h qi’ s0 sk
- Relation for moving to the right:
  - qi sj sk = sj qi’ sk
  - qi sj h = sj qi’ s0 h
- Relations for halt configuration:
  - q0sj = q0
  - sjq0h = q0h
  - hq0h = q

Word Problem for Semigroups

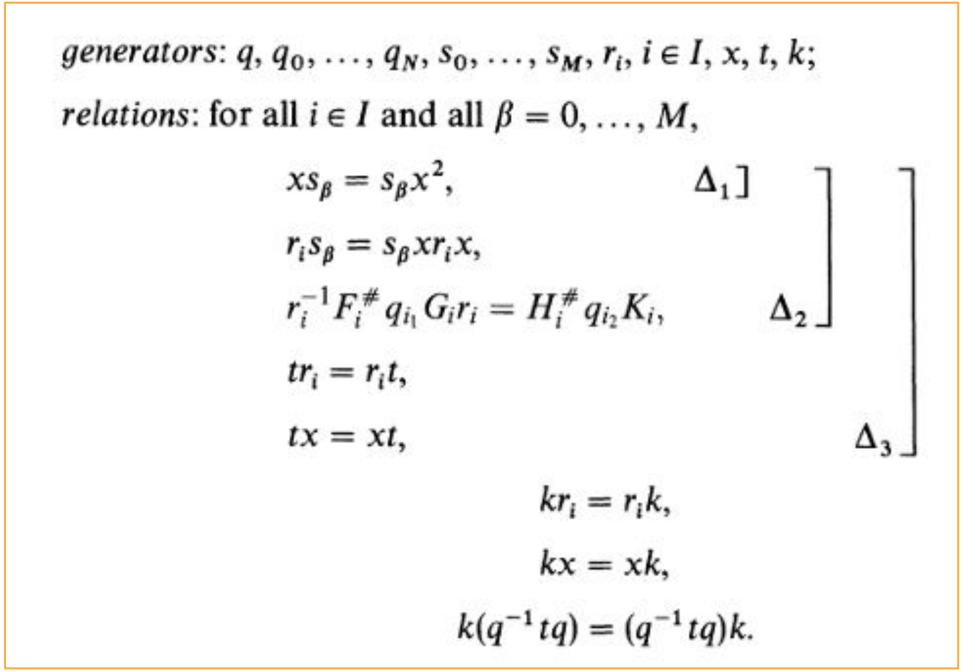
Want to prove T halts on c iff hch = q in S(T)

- (=>): Follows from construction
- (<=): Consider the sequence hch = w1 = w2 = … wn = q as described previously.
  - We may assume every word appears only once.
  - By induction, every wi for i<=n-1 must be of the form hcih for some possible configuration ci
  - At cn-1 we are in halt state q0
  - Either ci -> ci+1 or ci <- ci+1 for all i<=n-2, and cn-2 -> cn-1 since cn-1 is a halt state.
  - Since T is deterministic, we have c1 -> c2 -> … cn-1 and this concludes the proof.

Word Problem for Groups

- As we have seen before, we cannot directly use the same presentation to construct a group with unsolvable word problem, as we would get a lot of unwanted cancellation.
- How can we stop this?

Word Problem for Groups



generators: $q, q_0, \ldots, q_N, s_0, \ldots, s_M, r_i, i \in I, x, t, k;$

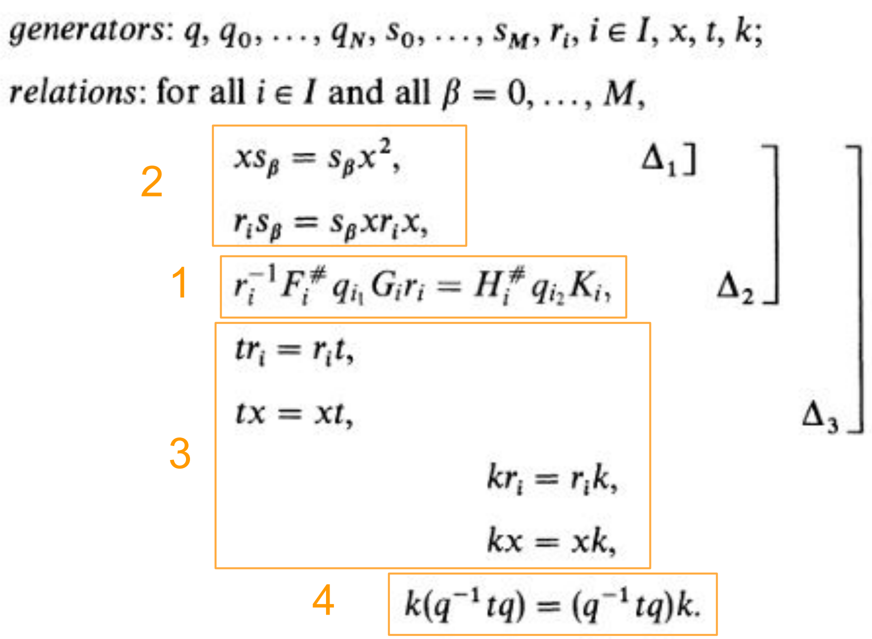
relations: for all $i \in I$ and all $\beta = 0, \ldots, M,$

$$\begin{aligned}
xs_\beta &= s_\beta x^2, && \Delta_1] \\
r_i s_\beta &= s_\beta x r_i x, && \\
r_i^{-1} F_i^{\#} q_{i_1} G_i r_i &= H_i^{\#} q_{i_2} K_i, && \Delta_2] \\
tr_i &= r_i t, && \\
tx &= xt, && \Delta_3] \\
kr_i &= r_i k, && \\
kx &= xk, && \\
k(q^{-1}tq) &= (q^{-1}tq)k. &&
\end{aligned}$$

Notation: sM = h, (g1g2...gn)# = g1^-1g2^-1…gn^-1 (not inverse of the entire word!), Fiqi1Gi = Hiqi2Gi are the relations from S(T) Represent configuration SqiS' by g(c) = (hS)#qi(S'h)

- Problem: 'Too much cancellation'
- Idea: Add a new generator ri for each relation ai=bi from the semigroup. Set ai = ri bi ri^-1
- New problem: cannot proceed after the first step
- Idea: Add an x with relations such that x, ri can now partially commute with tape alphabet
- We have, c -> c' => g(c) -> Lg(c')R, where L, R are words in x, ri
- Add t, k to detect whether a word is of the form LqR or not.
- **Claim: T halts on c iff [k, g(c)^-1tg(c)] = 1**

Word Problem for Groups



generators: $q, q_0, \ldots, q_N, s_0, \ldots, s_M, r_i, i \in I, x, t, k;$

relations: for all $i \in I$ and all $\beta = 0, \ldots, M,$

2

$xs_\beta = s_\beta x^2,$ $\Delta_1]$

$r_i s_\beta = s_\beta x r_i x,$

1 $r_i^{-1} F_i^{\#} q_{i_1} G_i r_i = H_i^{\#} q_{i_2} K_i,$ $\Delta_2]$

$tr_i = r_i t,$

$tx = xt,$ $\Delta_3]$

3

$kr_i = r_i k,$

$kx = xk,$

4 $k(q^{-1}tq) = (q^{-1}tq)k.$

(=>):

Lemma 1: c -> c' in T => g(c) = Lg(c')R, where L, R are words in x, ri

This is easily seen from applying (1) and then applying (2) repeatedly

Lemma 2: g = LqR => [k, g^{-1}tg] = 1

This is easily seen by noting that (3) is saying that t, k commute with k and ri and hence L and R. We can use this fact to get some cancellation and then apply (4).

HNN Extensions

For the proof of (<=), we will have to look at HNN Extensions.

If $G = \langle S \mid R \rangle$ has two subgroups A, B with isomorphism given by f, then the HNN extension of G (with respect to A, B, f) is $G^* = \langle S, t \mid R, tat^{-1} = f(a) \text{ for all } a \text{ in } A \rangle$

- G is called the ‘base’, A and B the ‘associated subgroups’, and t the ‘stable letter’.

We can generalise this idea by allowing multiple stable letters $t_1, \dots t_i$ corresponding to some $(A_1, B_1, f_1) \dots (A_i, B_i, f_i)$.

HNN Extensions

Baumslag-Solitar groups are an example of HNN Extensions.

- BS(m,n) = <a, b | ba^mb^-1 = a^n>
- This is an HNN Extension with Base <a>, stable letter b and associated subgroups <a^m>, <a^n>

HNN Extensions have nice properties that make them useful in many combinatorial group theroetic proofs, embedding theorems etc. and they also have a nice topological interpretation that makes them useful in Basse-Serre Theory. You can read about this by looking up graphs on groups :)

HNN Extensions

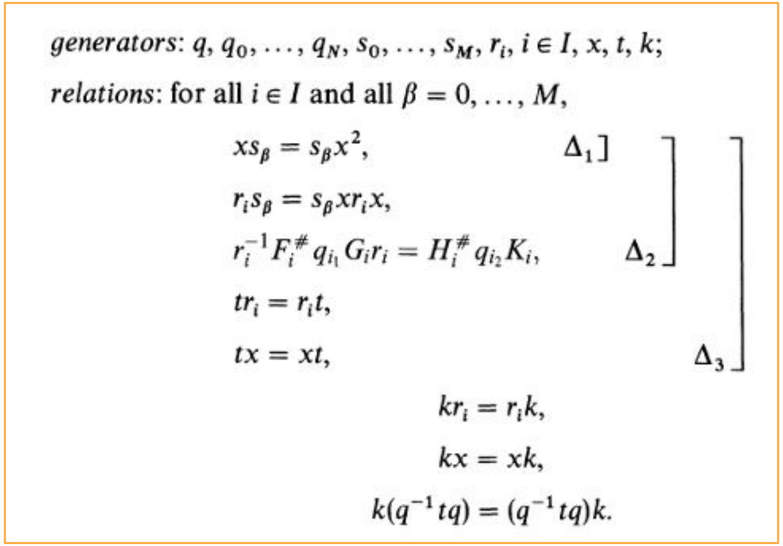
There are two properties of HNN Extensions that we are interested in:

1. **Britton's Lemma:** If a word $w = g_0t^{\pm1}g_1t^{\pm1}... t^{\pm1}g_n = 1$ then either $n=0$, $g_0=1$ or w contains some tg_it^{-1} with g_i in A or $t^{-1}g_it$ with g_i in B. This kind of tg_it^{-1} or $t^{-1}g_it$ is called a 'pinch' and a word that doesn't contain a pinch is said to be t-reduced. The analogous statement for HNN Extension by multiple stable letters holds.

2. G is embedded in its HNN Extension G* (by the obvious embedding g -> g). Note that this is a corollary of Britton's Lemma.

Word Problem for Groups

(<=): Notice that our construction is a sequence of HNN Extensions. That is,



generators: $q, q_0, \ldots, q_N, s_0, \ldots, s_M, r_i, i \in I, x, t, k$;

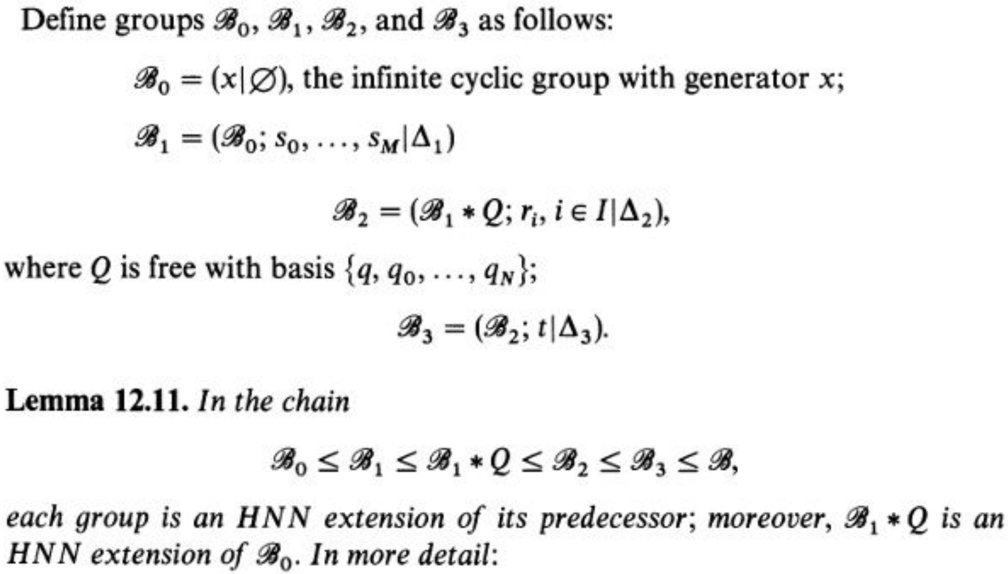
relations: for all $i \in I$ and all $\beta = 0, \ldots, M$,

$$\begin{array}{llll}
xs_\beta = s_\beta x^2, & \Delta_1] & & \\
r_i s_\beta = s_\beta x r_i x, & & & \\
r_i^{-1} F_i^\# q_{i_1} G_i r_i = H_i^\# q_{i_2} K_i, & & \Delta_2] & \\
tr_i = r_i t, & & & \\
tx = xt, & & & \Delta_3]
\end{array}$$

$$kr_i = r_i k,$$

$$kx = xk,$$

$$k(q^{-1}tq) = (q^{-1}tq)k.$$



Define groups $\mathscr{B}_0$, $\mathscr{B}_1$, $\mathscr{B}_2$, and $\mathscr{B}_3$ as follows:

$\mathscr{B}_0 = (x|\varnothing)$, the infinite cyclic group with generator x;

$\mathscr{B}_1 = (\mathscr{B}_0; s_0, \ldots, s_M|\Delta_1)$

$$\mathscr{B}_2 = (\mathscr{B}_1 * Q; r_i, i \in I|\Delta_2),$$

where Q is free with basis $\{q, q_0, \ldots, q_N\}$;

$$\mathscr{B}_3 = (\mathscr{B}_2; t|\Delta_3).$$

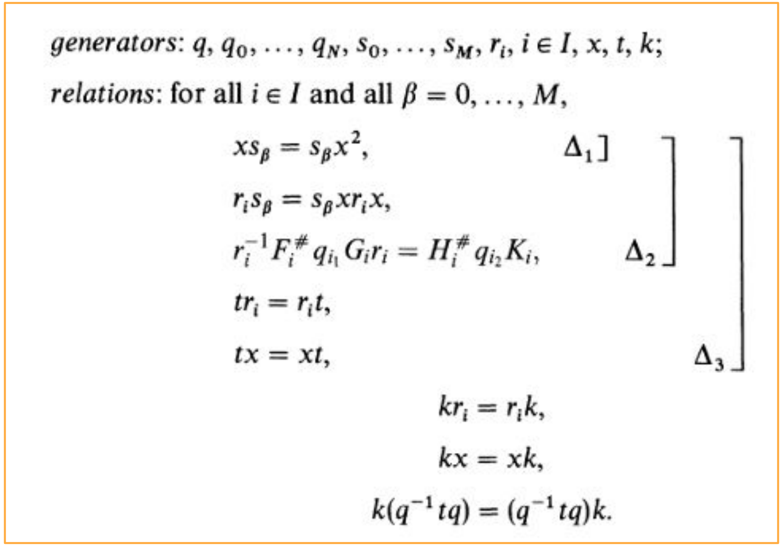
Lemma 12.11. *In the chain*

$$\mathscr{B}_0 \leq \mathscr{B}_1 \leq \mathscr{B}_1 * Q \leq \mathscr{B}_2 \leq \mathscr{B}_3 \leq \mathscr{B},$$

each group is an HNN extension of its predecessor; moreover, $\mathscr{B}_1 * Q$ *is an HNN extension of* $\mathscr{B}_0$. *In more detail:*

Word Problem for Groups

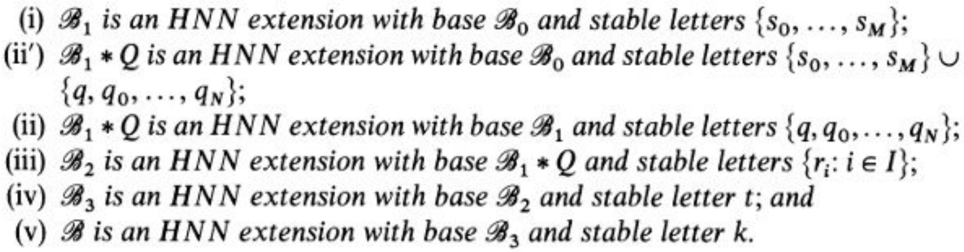
(<=): Notice that our construction is a sequence of HNN Extensions. That is,



generators: $q, q_0, \ldots, q_N, s_0, \ldots, s_M, r_i, i \in I, x, t, k;$

relations: for all $i \in I$ and all $\beta = 0, \ldots, M$,

$$\begin{aligned}
xs_\beta &= s_\beta x^2, && \Delta_1 \\
r_i s_\beta &= s_\beta x r_i x, && \\
r_i^{-1} F_i^{\#} q_{i_1} G_i r_i &= H_i^{\#} q_{i_2} K_i, && \Delta_2 \\
tr_i &= r_i t, && \\
tx &= xt, && \Delta_3 \\
kr_i &= r_i k, && \\
kx &= xk, && \\
k(q^{-1}tq) &= (q^{-1}tq)k. &&
\end{aligned}$$



(i) *$\mathscr{B}_1$ is an HNN extension with base $\mathscr{B}_0$ and stable letters $\{s_0, \ldots, s_M\}$;*

(ii′) *$\mathscr{B}_1 * Q$ is an HNN extension with base $\mathscr{B}_0$ and stable letters $\{s_0, \ldots, s_M\} \cup \{q, q_0, \ldots, q_N\}$;*

(ii) *$\mathscr{B}_1 * Q$ is an HNN extension with base $\mathscr{B}_1$ and stable letters $\{q, q_0, \ldots, q_N\}$;*

(iii) *$\mathscr{B}_2$ is an HNN extension with base $\mathscr{B}_1 * Q$ and stable letters $\{r_i\colon i \in I\}$;*

(iv) *$\mathscr{B}_3$ is an HNN extension with base $\mathscr{B}_2$ and stable letter t; and*

(v) *$\mathscr{B}$ is an HNN extension with base $\mathscr{B}_3$ and stable letter k.*

Word Problem for Groups

We break the rest of the proof into 3 parts:

1. Suppose g is a word of the form g(c) for some configuration c (i.e. g is of the form X#qiY, where X and Y are words in si and h), and [k, $g^{-1}tg$] = 1 in B. Then, g = LqR in B_2, where L, R are freely reduced words in x, ri.

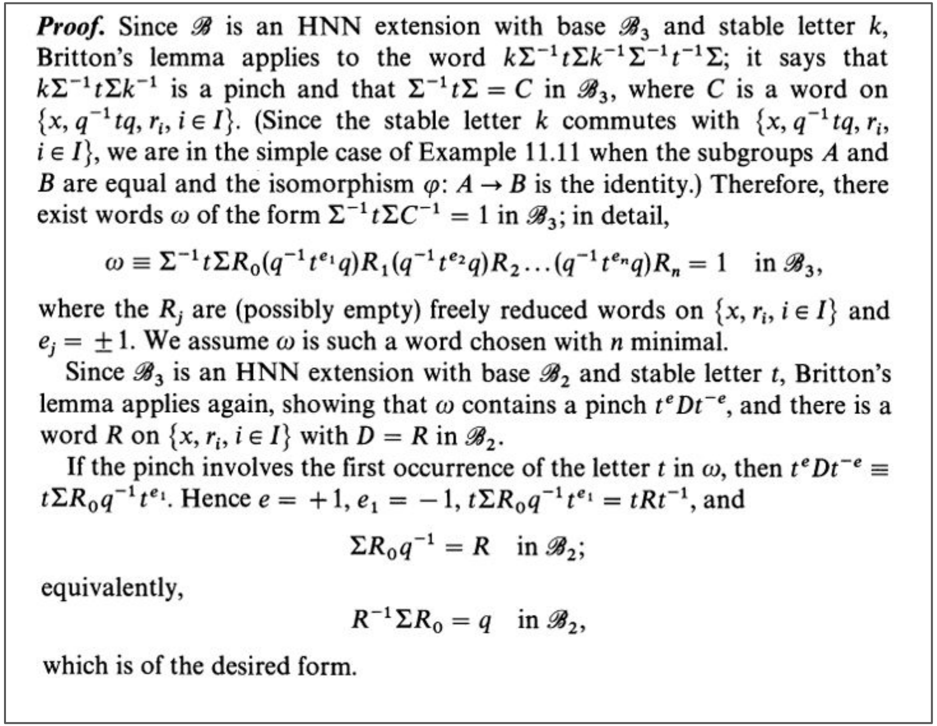
2. If g = X#qiY= LqR (with conditions as described above), then L^{-1}X#qi and qRY^{-1} are ri - reduced.

3. If L1 and L2 are ri reduced words in x, ri; X, Y are freely reduced words in si and h, and X#qiY= LqR in B_2, then both X and Y are positive and XqjY = q in S(T) (note: we already have that X, Y are freely reduced and positive)

We will only illustrate how HNN Extensions are used for the proof of (1). They can be used similarly to prove (2) and (3) with some case-work.

Word Problem for Groups

Proof of (1):



Proof. Since $\mathscr{B}$ is an HNN extension with base $\mathscr{B}_3$ and stable letter k, Britton's lemma applies to the word $k\Sigma^{-1}t\Sigma k^{-1}\Sigma^{-1}t^{-1}\Sigma$; it says that $k\Sigma^{-1}t\Sigma k^{-1}$ is a pinch and that $\Sigma^{-1}t\Sigma = C$ in $\mathscr{B}_3$, where C is a word on $\{x, q^{-1}tq, r_i, i \in I\}$. (Since the stable letter k commutes with $\{x, q^{-1}tq, r_i, i \in I\}$, we are in the simple case of Example 11.11 when the subgroups A and B are equal and the isomorphism $\varphi\colon A \to B$ is the identity.) Therefore, there exist words ω of the form $\Sigma^{-1}t\Sigma C^{-1} = 1$ in $\mathscr{B}_3$; in detail,

$$\omega \equiv \Sigma^{-1}t\Sigma R_0(q^{-1}t^{e_1}q)R_1(q^{-1}t^{e_2}q)R_2\dots(q^{-1}t^{e_n}q)R_n = 1 \quad \text{in } \mathscr{B}_3,$$

where the R_j are (possibly empty) freely reduced words on $\{x, r_i, i \in I\}$ and $e_j = \pm 1$. We assume ω is such a word chosen with n minimal.

Since $\mathscr{B}_3$ is an HNN extension with base $\mathscr{B}_2$ and stable letter t, Britton's lemma applies again, showing that ω contains a pinch t^eDt^{-e}, and there is a word R on $\{x, r_i, i \in I\}$ with $D = R$ in $\mathscr{B}_2$.

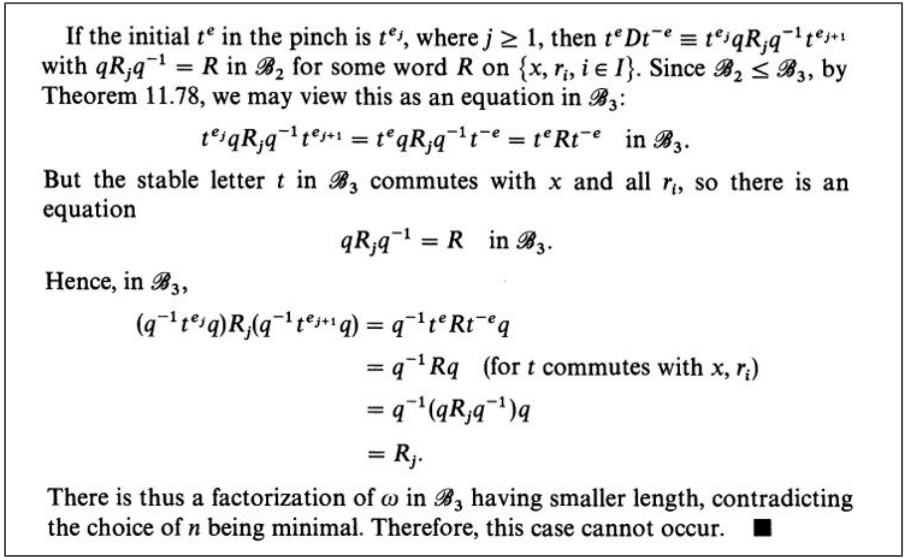
If the pinch involves the first occurrence of the letter t in ω, then $t^eDt^{-e} \equiv t\Sigma R_0q^{-1}t^{e_1}$. Hence $e = +1$, $e_1 = -1$, $t\Sigma R_0q^{-1}t^{e_1} = tRt^{-1}$, and

$$\Sigma R_0 q^{-1} = R \quad \text{in } \mathscr{B}_2;$$

equivalently,

$$R^{-1}\Sigma R_0 = q \quad \text{in } \mathscr{B}_2,$$

which is of the desired form.



If the initial t^e in the pinch is t^{e_j}, where $j \geq 1$, then $t^eDt^{-e} \equiv t^{e_j}qR_jq^{-1}t^{e_{j+1}}$ with $qR_jq^{-1} = R$ in $\mathscr{B}_2$ for some word R on $\{x, r_i, i \in I\}$. Since $\mathscr{B}_2 \leq \mathscr{B}_3$, by Theorem 11.78, we may view this as an equation in $\mathscr{B}_3$:

$$t^{e_j}qR_jq^{-1}t^{e_{j+1}} = t^eqR_jq^{-1}t^{-e} = t^eRt^{-e} \quad \text{in } \mathscr{B}_3.$$

But the stable letter t in $\mathscr{B}_3$ commutes with x and all r_i, so there is an equation

$$qR_jq^{-1} = R \quad \text{in } \mathscr{B}_3.$$

Hence, in $\mathscr{B}_3$,

$$\begin{aligned}(q^{-1}t^{e_j}q)R_j(q^{-1}t^{e_{j+1}}q) &= q^{-1}t^eRt^{-e}q\\ &= q^{-1}Rq \quad \text{(for } t \text{ commutes with } x, r_i)\\ &= q^{-1}(qR_jq^{-1})q\\ &= R_j.\end{aligned}$$

There is thus a factorization of ω in $\mathscr{B}_3$ having smaller length, contradicting the choice of n being minimal. Therefore, this case cannot occur. ■

A generalisation: Adian-Rabin Theorem

A **Markov Property** P of finitely presentable groups is one for which:

- P is preserved under isomorphism
- There exists a finitely presentable group with property P
- There exists a finitely presentable group which cannot be embedded in any group with property P

Almost all properties we are interested in are Markov Properties: trivial, finite, abelian, simple, torsion, torsion-free, free, solvable word problem, solvable conjugacy problem etc.

A generalisation: Adian-Rabin Theorem

Adian-Rabin Theorem: If P is a Markov Property, there does not exist an algorithm that takes in a finite presentation and determines whether the group given by the presentation has property P .

This means that the problem of deciding whether a group given by a finite presentation has any of the properties mentioned on the previous slide are all undecidable.

This extremely powerful theorem can be proved fairly easily using the group we have constructed.

Groups with Solvable Word Problem

- Free Group
- Free Abelian Group
- Finite Groups
- Groups with Computable Normal Form

(as we have seen)

- Free Product of groups with solvable word problem

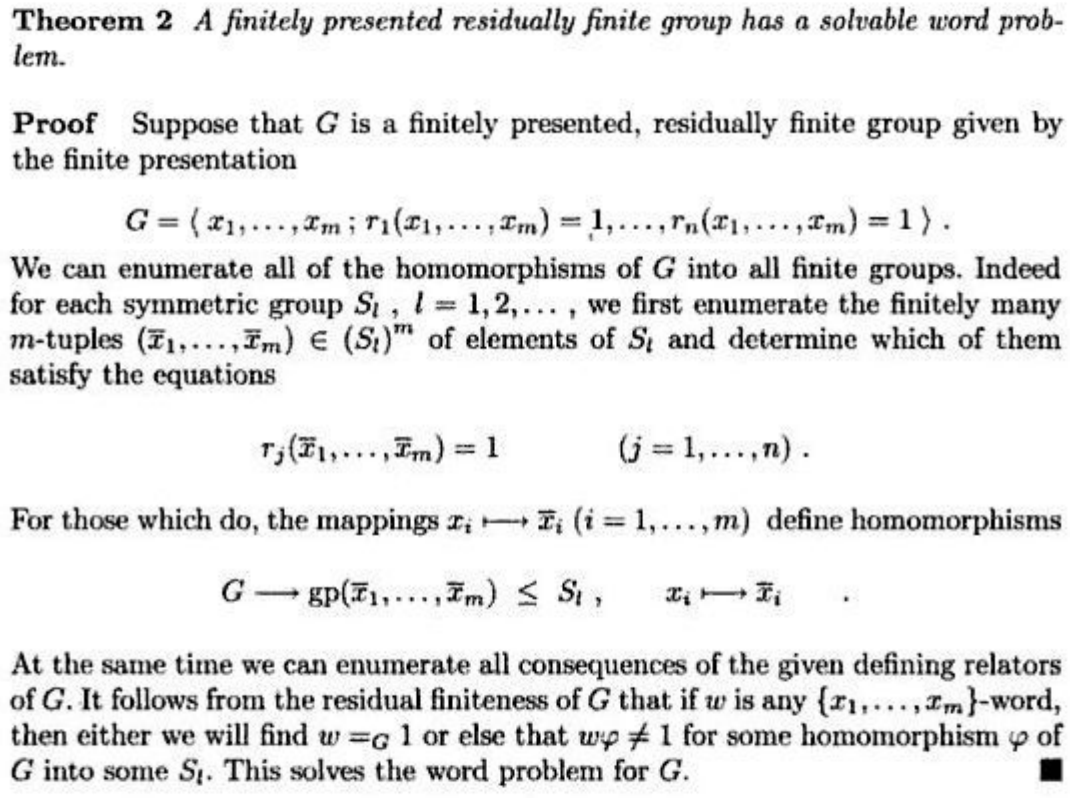
Groups with Solvable Word Problem: Residually Finite Groups

Residually Finite Groups are groups G where for every non-trivial element g in G, there exists a finite index normal subgroup N of G not containing g.

Another way to state this property is that the intersection of all finite index normal subgroups is trivial.

These groups have interesting properties and turn up a lot in Combinatorial Group Theory. Finitely presented residually finite groups have a solvable word problem.

Groups with Solvable Word Problem: Residually Finite Groups



Theorem 2 *A finitely presented residually finite group has a solvable word problem.*

Proof Suppose that G is a finitely presented, residually finite group given by the finite presentation

$$G = \langle\, x_1, \ldots, x_m \,;\, r_1(x_1, \ldots, x_m) = 1, \ldots, r_n(x_1, \ldots, x_m) = 1 \,\rangle \,.$$

We can enumerate all of the homomorphisms of G into all finite groups. Indeed for each symmetric group S_l , $l = 1, 2, \ldots$, we first enumerate the finitely many m-tuples $(\overline{x}_1, \ldots, \overline{x}_m) \in (S_l)^m$ of elements of S_l and determine which of them satisfy the equations

$$r_j(\overline{x}_1, \ldots, \overline{x}_m) = 1 \qquad\qquad (j = 1, \ldots, n) \,.$$

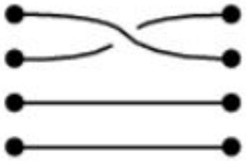
For those which do, the mappings $x_i \longmapsto \overline{x}_i$ $(i = 1, \ldots, m)$ define homomorphisms

$$G \longrightarrow \mathrm{gp}(\overline{x}_1, \ldots, \overline{x}_m) \;\leq\; S_l \,, \qquad x_i \longmapsto \overline{x}_i \qquad .$$

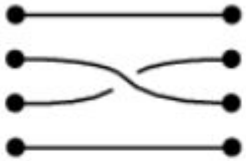
At the same time we can enumerate all consequences of the given defining relators of G. It follows from the residual finiteness of G that if w is any $\{x_1, \ldots, x_m\}$-word, then either we will find $w =_G 1$ or else that $w\varphi \neq 1$ for some homomorphism φ of G into some S_l. This solves the word problem for G. ■

Groups with Solvable Word Problems: Braid Groups

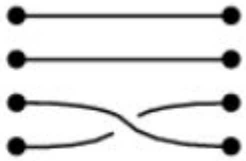
$$B_n = \left\langle \sigma_1, \ldots, \sigma_{n-1} \;\middle|\; \begin{array}{cl} \sigma_i\sigma_j = \sigma_j\sigma_i, & |i-j| > 1 \\ \sigma_i\sigma_j\sigma_i = \sigma_j\sigma_i\sigma_j, & |i-j| = 1 \end{array} \right\rangle$$



σ_1



σ_2



σ_3

Solving the word problem for braid groups
tells us when a braid can be undone

Groups with Solvable Word Problems: Braid Groups

Garside Normal Form:

- The relations are all positive, so the group can be seen as a semigroup
- Consider the element $\Delta = \sigma_1(\sigma_2\sigma_1)...(\sigma_{n-1}\sigma_{n-2}\ldots\sigma_1)$
- Δ has the property that $\sigma_i\Delta = \Delta\sigma_{n-i}$ and, for any i, Δ can be rewritten as σ_iX_i where X_i is some positive word
- Now, in a word from the group we can replace any σ_i^-1 by $X_i\Delta$^-1. Then, using, $\sigma_i\Delta$^-1 = Δ^-1σ_{n-i}, we can write the element as Δ^pA for some positive word A which does not have Δ as a prefix.
- There is a unique way to do this, hence this is a normal form.

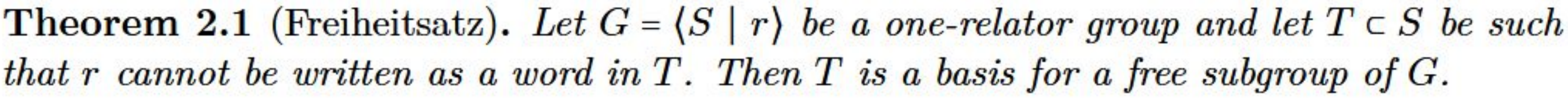
Groups with Solvable Word Problem: One-Relator Groups

One-Relator Groups are groups with only one relation.

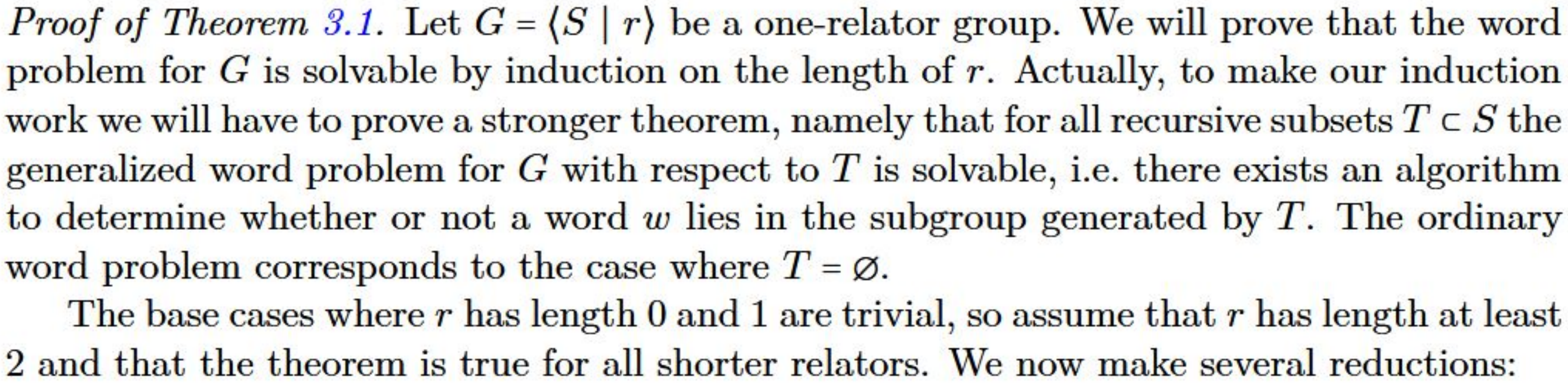
An example of one-relator groups are **Surface Groups**, the Fundamental Groups of surfaces. For a surface of genus g, the fundamental group is <a1, b1, a2, b2… ag, bg| [a1, b1]...[ag, bg]>. There is only one relation.

One-Relator Groups have solvable word problem.

Groups with Solvable Word Problem: One-Relator Groups



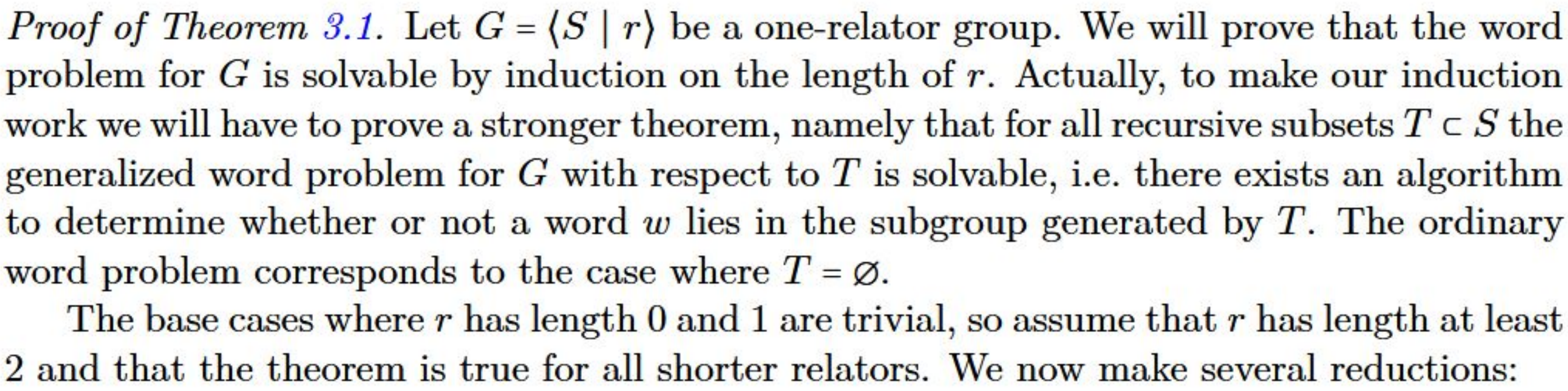
Theorem 2.1 (Freiheitsatz). *Let $G = \langle S \mid r \rangle$ be a one-relator group and let $T \subset S$ be such that r cannot be written as a word in T. Then T is a basis for a free subgroup of G.*



Proof of Theorem 3.1. Let $G = \langle S \mid r \rangle$ be a one-relator group. We will prove that the word problem for G is solvable by induction on the length of r. Actually, to make our induction work we will have to prove a stronger theorem, namely that for all recursive subsets $T \subset S$ the generalized word problem for G with respect to T is solvable, i.e. there exists an algorithm to determine whether or not a word w lies in the subgroup generated by T. The ordinary word problem corresponds to the case where $T = \varnothing$.

The base cases where r has length 0 and 1 are trivial, so assume that r has length at least 2 and that the theorem is true for all shorter relators. We now make several reductions:

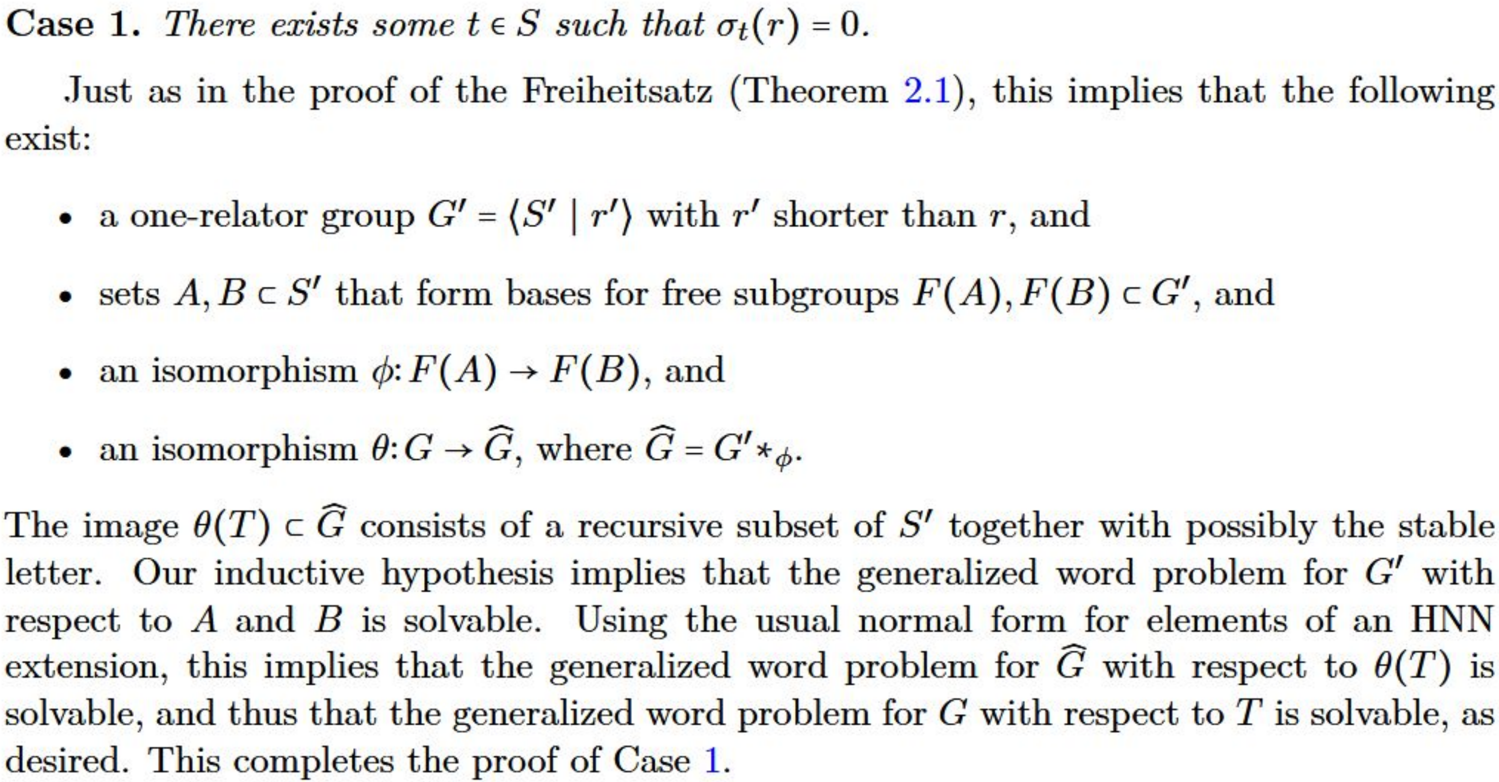
Groups with Solvable Word Problem: One-Relator Groups



Proof of Theorem 3.1. Let $G = \langle S \mid r \rangle$ be a one-relator group. We will prove that the word problem for G is solvable by induction on the length of r. Actually, to make our induction work we will have to prove a stronger theorem, namely that for all recursive subsets $T \subset S$ the generalized word problem for G with respect to T is solvable, i.e. there exists an algorithm to determine whether or not a word w lies in the subgroup generated by T. The ordinary word problem corresponds to the case where $T = \varnothing$.

The base cases where r has length 0 and 1 are trivial, so assume that r has length at least 2 and that the theorem is true for all shorter relators. We now make several reductions:

Groups with Solvable Word Problem: One-Relator Groups



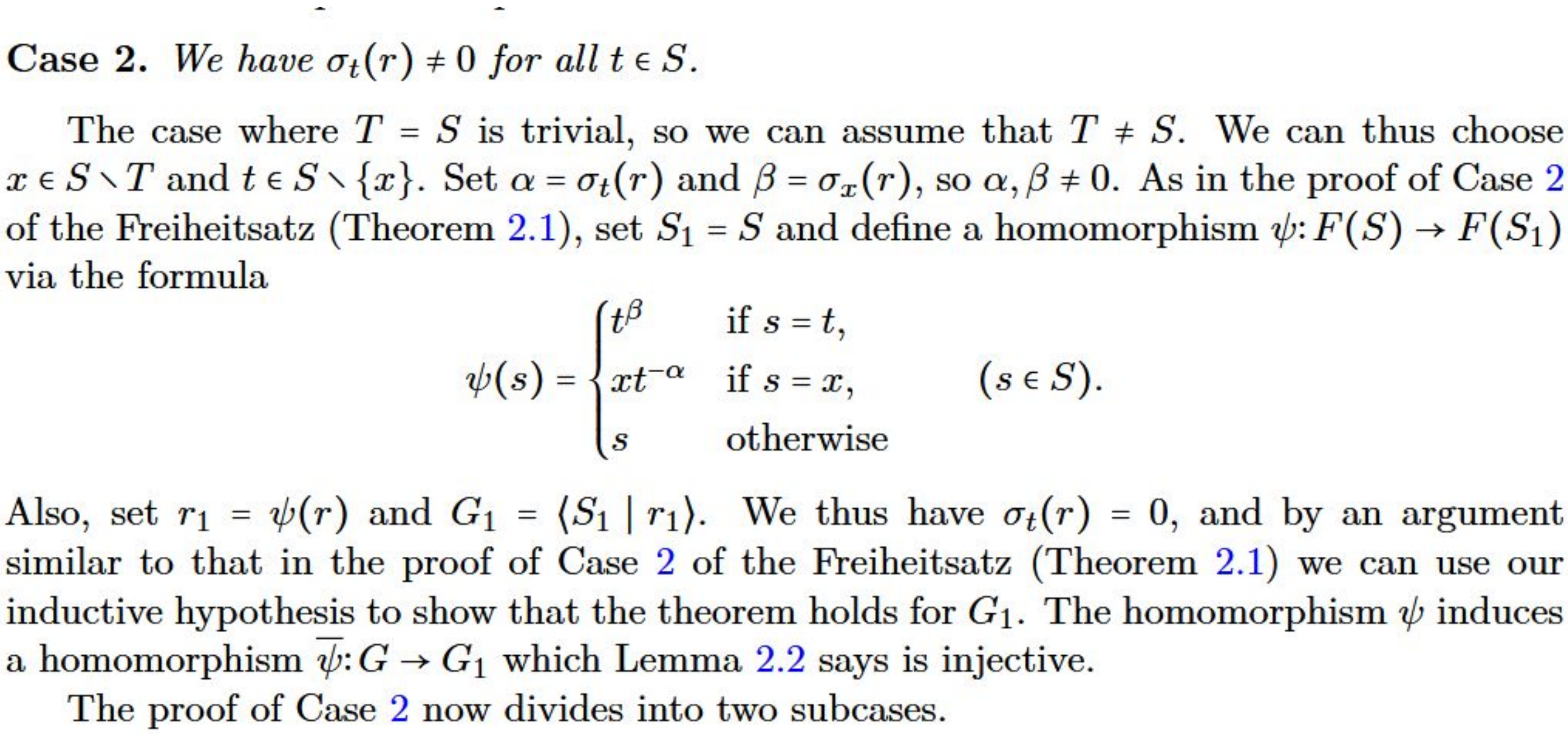
Case 1. *There exists some $t \in S$ such that $\sigma_t(r) = 0$.*

Just as in the proof of the Freiheitsatz (Theorem 2.1), this implies that the following exist:

- a one-relator group $G' = \langle S' \mid r' \rangle$ with r' shorter than r, and
- sets $A, B \subset S'$ that form bases for free subgroups $F(A), F(B) \subset G'$, and
- an isomorphism $\phi\colon F(A) \to F(B)$, and
- an isomorphism $\theta\colon G \to \widehat{G}$, where $\widehat{G} = G' *_{\phi}$.

The image $\theta(T) \subset \widehat{G}$ consists of a recursive subset of S' together with possibly the stable letter. Our inductive hypothesis implies that the generalized word problem for G' with respect to A and B is solvable. Using the usual normal form for elements of an HNN extension, this implies that the generalized word problem for $\widehat{G}$ with respect to $\theta(T)$ is solvable, and thus that the generalized word problem for G with respect to T is solvable, as desired. This completes the proof of Case 1.

Groups with Solvable Word Problem: One-Relator Groups



Case 2. *We have $\sigma_t(r) \neq 0$ for all $t \in S$.*

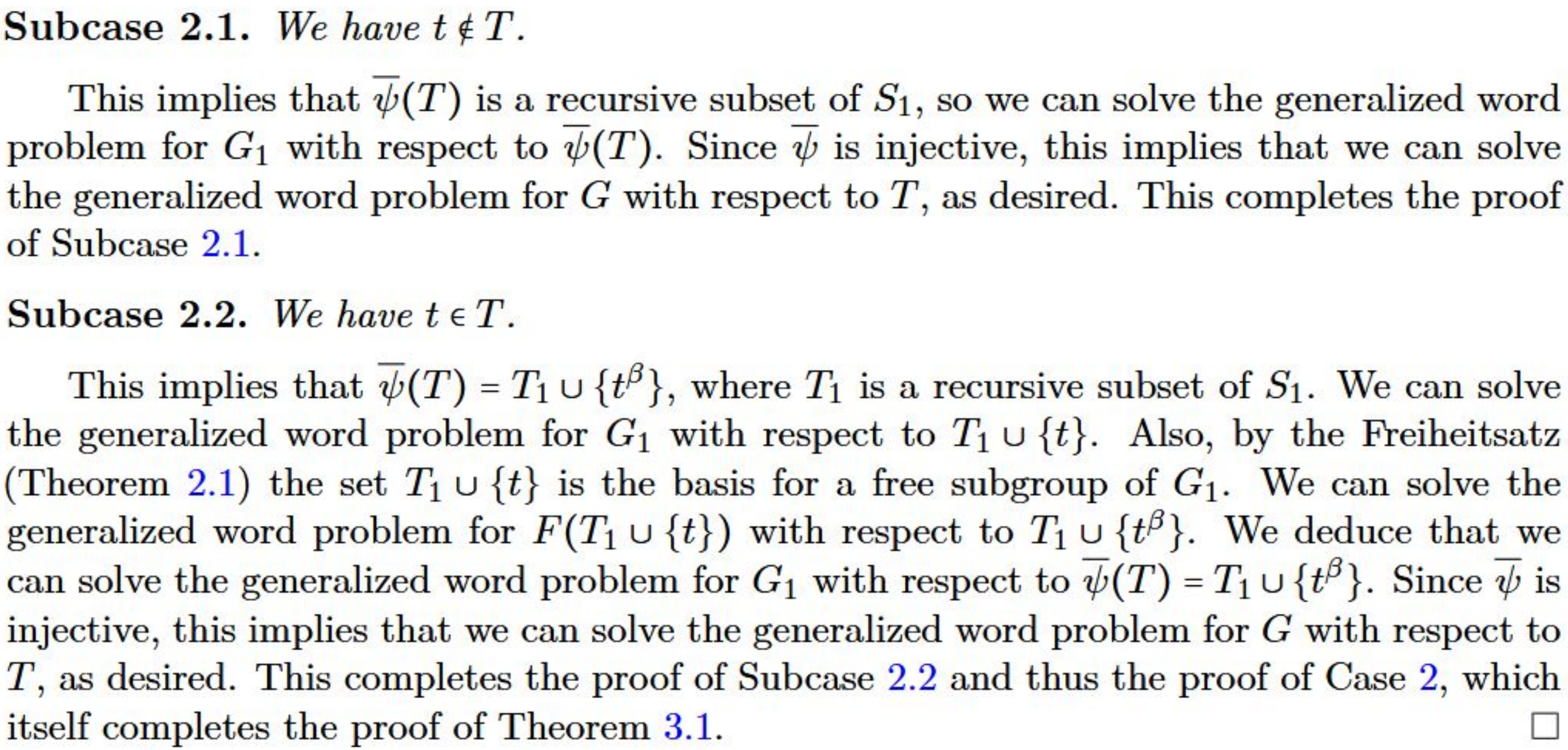
The case where $T = S$ is trivial, so we can assume that $T \neq S$. We can thus choose $x \in S \smallsetminus T$ and $t \in S \smallsetminus \{x\}$. Set $\alpha = \sigma_t(r)$ and $\beta = \sigma_x(r)$, so $\alpha, \beta \neq 0$. As in the proof of Case 2 of the Freiheitsatz (Theorem 2.1), set $S_1 = S$ and define a homomorphism $\psi: F(S) \to F(S_1)$ via the formula

$$\psi(s) = \begin{cases} t^{\beta} & \text{if } s = t, \\ xt^{-\alpha} & \text{if } s = x, \\ s & \text{otherwise} \end{cases} \qquad (s \in S).$$

Also, set $r_1 = \psi(r)$ and $G_1 = \langle S_1 \mid r_1 \rangle$. We thus have $\sigma_t(r) = 0$, and by an argument similar to that in the proof of Case 2 of the Freiheitsatz (Theorem 2.1) we can use our inductive hypothesis to show that the theorem holds for G_1. The homomorphism ψ induces a homomorphism $\overline{\psi}: G \to G_1$ which Lemma 2.2 says is injective.

The proof of Case 2 now divides into two subcases.

Groups with Solvable Word Problem: One-Relator Groups



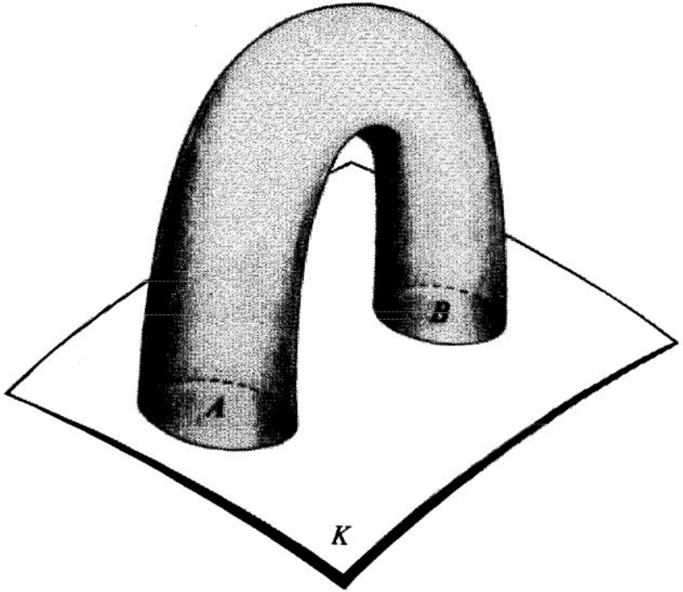
Subcase 2.1. *We have $t \notin T$.*

This implies that $\overline{\psi}(T)$ is a recursive subset of S_1, so we can solve the generalized word problem for G_1 with respect to $\overline{\psi}(T)$. Since $\overline{\psi}$ is injective, this implies that we can solve the generalized word problem for G with respect to T, as desired. This completes the proof of Subcase 2.1.

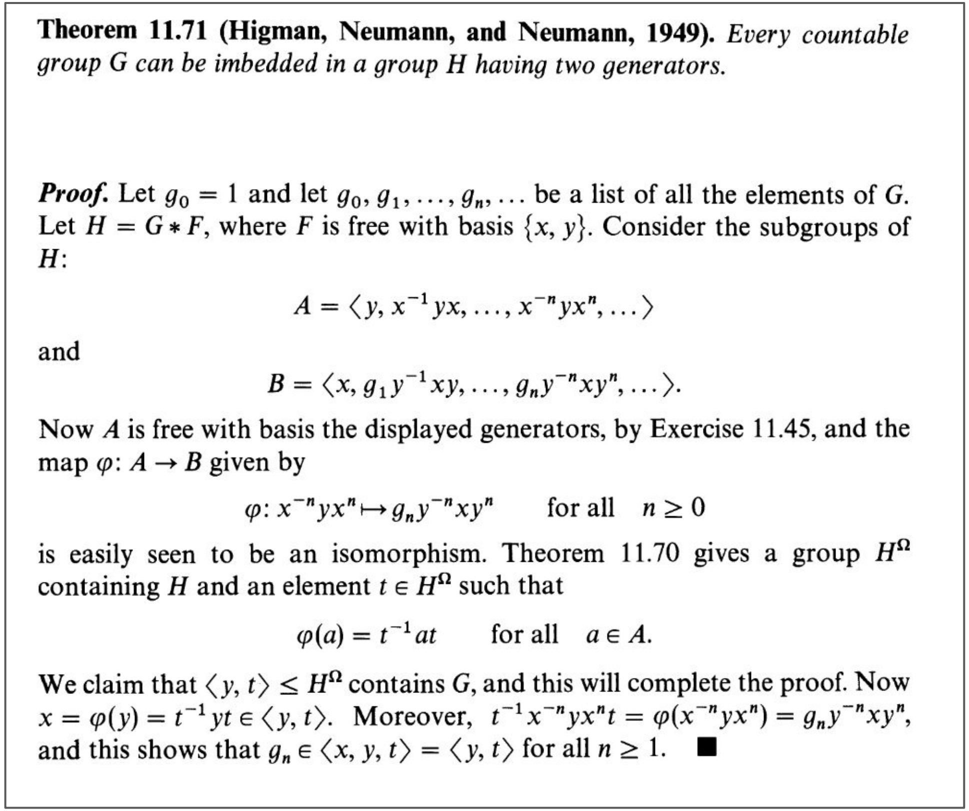
Subcase 2.2. *We have $t \in T$.*

This implies that $\overline{\psi}(T) = T_1 \cup \{t^\beta\}$, where T_1 is a recursive subset of S_1. We can solve the generalized word problem for G_1 with respect to $T_1 \cup \{t\}$. Also, by the Freiheitsatz (Theorem 2.1) the set $T_1 \cup \{t\}$ is the basis for a free subgroup of G_1. We can solve the generalized word problem for $F(T_1 \cup \{t\})$ with respect to $T_1 \cup \{t^\beta\}$. We deduce that we can solve the generalized word problem for G_1 with respect to $\overline{\psi}(T) = T_1 \cup \{t^\beta\}$. Since $\overline{\psi}$ is injective, this implies that we can solve the generalized word problem for G with respect to T, as desired. This completes the proof of Subcase 2.2 and thus the proof of Case 2, which itself completes the proof of Theorem 3.1. □

More on HNN Extensions: Graphs on Groups



More on HNN Extensions: A Nice Result :)



Theorem 11.71 (Higman, Neumann, and Neumann, 1949). *Every countable group G can be imbedded in a group H having two generators.*

Proof. Let $g_0 = 1$ and let $g_0, g_1, \ldots, g_n, \ldots$ be a list of all the elements of G. Let $H = G * F$, where F is free with basis $\{x, y\}$. Consider the subgroups of H:

$$A = \langle y, x^{-1}yx, \ldots, x^{-n}yx^n, \ldots \rangle$$

and

$$B = \langle x, g_1 y^{-1}xy, \ldots, g_n y^{-n}xy^n, \ldots \rangle.$$

Now A is free with basis the displayed generators, by Exercise 11.45, and the map $\varphi\colon A \to B$ given by

$$\varphi\colon x^{-n}yx^n \mapsto g_n y^{-n}xy^n \qquad \text{for all} \quad n \geq 0$$

is easily seen to be an isomorphism. Theorem 11.70 gives a group H^Ω containing H and an element $t \in H^\Omega$ such that

$$\varphi(a) = t^{-1}at \qquad \text{for all} \quad a \in A.$$

We claim that $\langle y, t \rangle \leq H^\Omega$ contains G, and this will complete the proof. Now $x = \varphi(y) = t^{-1}yt \in \langle y, t \rangle$. Moreover, $t^{-1}x^{-n}yx^n t = \varphi(x^{-n}yx^n) = g_n y^{-n}xy^n$, and this shows that $g_n \in \langle x, y, t \rangle = \langle y, t \rangle$ for all $n \geq 1$. ■

References

- An Introduction to the Theory of Groups: J.J. Rotman
- Combinatorial Group Theory: Lyndon and Schupp
- Combinatorial Group Theory: Magnus, Karrass, Solitar
- One-relator groups: notes by Andew Putman
- Basic results on braid groups: González-Meneses