ANNOUNCEMENTS MAR 25

- Cameras on
- HW due Thu 3:30
- Office Hours Fri 2-3, Tue 11-12
- Progress report Apr 2 ~ 1 page
- First draft Apr 9
- Talk to me about makeup points!

Today

- Howson's thm
- Regular languages
- Automata

Howson's Thm

Thm 7.32 (1954)

If G, H f.g. subgps of F_∞ then $G \cap H$ is f.g.

Original proof: algebraic topology

Languages

$S = \{x_1, \dots, x_n\}$ "alphabet"

$S^* = \{$words of finite length in $S\}$

Any subset $L \subseteq S^*$ is called a language

examples

① $L = \{a^i b^j : i, j > 0\} \subseteq \{a, b\}^*$

② Consider $H = \langle a^2, b \rangle \leq F_2$

$L =$ reduced words in a, b, a^{-1}, b^{-1} corresponding to elts of H.

$\subseteq \{a, b, a^{-1}, b^{-1}\}^*$

Language examples

① $L = \{a^i b^j : i,j > 0\} \subseteq \{a,b\}^*$

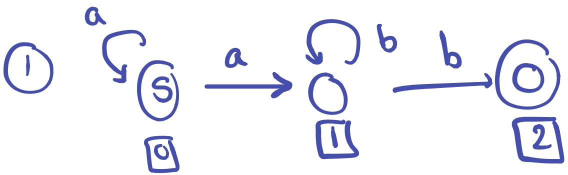
② Consider $H = \langle a^2, b \rangle \leq F_2$

L = reduced words in a, b, a^{-1}, b^{-1} corresponding to elts of H.

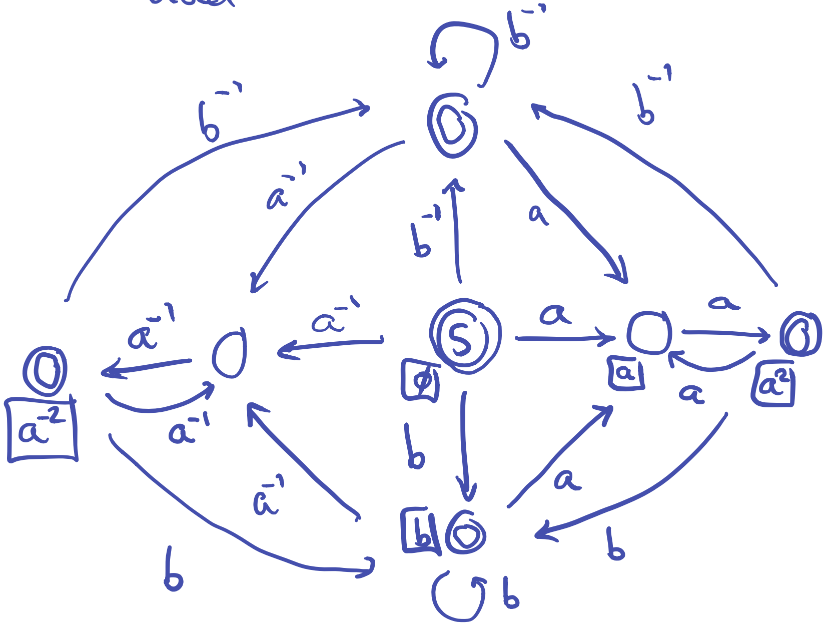
$\subseteq \{a, b, a^{-1}, b^{-1}\}^*$

Automaton examples

①



② Roughly states correspond to last letter used.



Deterministic FSA

FSA with

- one start state
- no edges w/ empty label
- ≤ 1 edge with a given letter starting from each vertex

$\rightsquigarrow$ regular languages

Complete: $= 1$ in 3^{rd} bullet.

Tidying up a FSA

last time $\rightarrow$ **Lemma 1.** L accepted by det FSA

$\Rightarrow$ L accepted by complete det FSA

Lemma 2. L acc by non-det FSA

$\Rightarrow$ L acc by det FSA.

In other words: FSAs, det FSAs, compl. det FSAs all give same languages, i.e. regular lang's.

<u>Lemma 2</u>. L acc by non-det FSA

$\Longrightarrow$ L acc by det FSA.

<u>Pf</u>. Given FSA M, want to make it satisfy the 3 bullet pts without changing the accepted lang.

We'll just do 3rd bullet:

- ≤ 1 edge with a given letter starting from each vertex

Let D be FSA with

<u>Vertices</u> $V(D) = P(V(M)) \setminus \emptyset$

<u>Edges</u> Let $U = \{v_1, \ldots, v_k\} \in V(D)$
$v_i \in V(M)$

3rd bullet

For each $a \in S$ (= alphabet)

Make an a-edge from U to

$$V = \bigcup_{i=1}^{k} \{v \in V(M) : \exists \text{ a-edge from } v_i \text{ to } v\}$$

1st bullet!

<u>Start state</u> {start states in M}

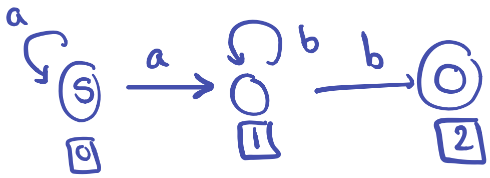
<u>Accept states</u> elts of $P(V(M))$ cont. accept st.

Key part of defn of D:

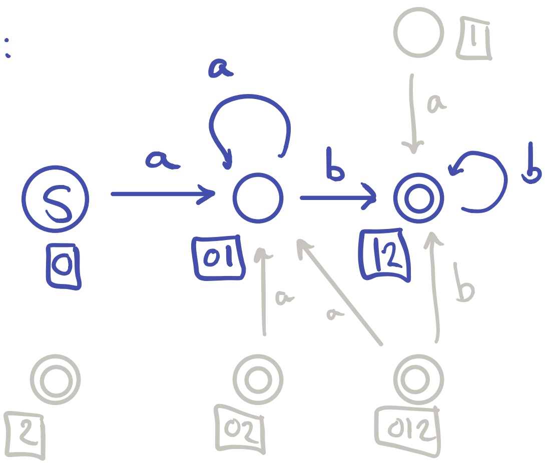
Make an a-edge from U to

$$V = \bigcup_{i=1}^{k} \{ v \in V(M) : \exists \text{ a-edge from } v_i \text{ to } v \}$$

M:



D:



Can cut the chaff

$\square$

Automaton version of Howson's Thm

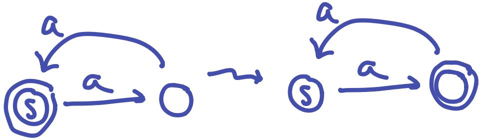
Thm 7.11 Say $K, L \subseteq S^*$ are reg. languages. Then so are:

1. $S^* \setminus K$
2. $K \cup L$
3. ** $K \cap L$
4. $KL = \{w_K w_L : w_K \in K, w_L \in L\}$
5. $L^* = L \cup LL \cup LLL \cup \cdots$

** reg. lang. is automaton version of f.g.

Pf. 1. Toggle accept/non-accept states

example. $L = \{a^i : i \text{ even}\}$



2. Say M_K, M_L FSA for K, L then $M_K \cup M_L$ is a FSA for $K \cup L$. Apply the 2 lemmas.

3. $K \cap L = S^* \setminus ((S \setminus K) \cup (S \setminus L))$

Apply 1 & 2

Regular vs. finite gen.

Thm. S = fin. gen. set for G

Then $H \leq G$ is fin. gen

$\iff$ H is image of reg. lang. $L \subseteq (S^{\pm 1})^*$.

under $\pi : (S^{\pm})^* \longrightarrow H$

Idea of Pf: Generators for H

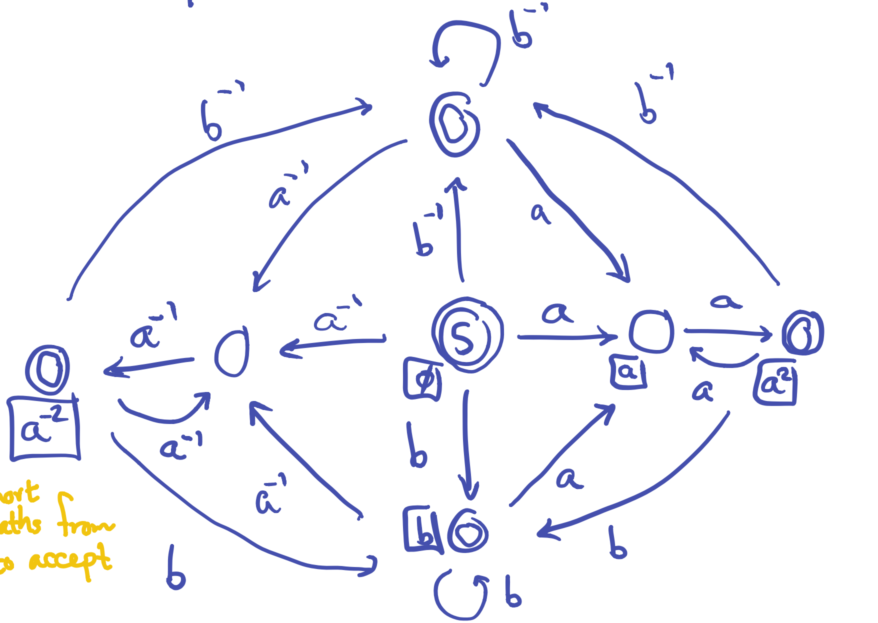
$\longleftrightarrow$ circuits in M

and short paths from start to accept

finitely many since M is finite.

$\square$

Example. $H = \langle a^2, b \rangle$



Circuits: $a^2, a^{-2}, b, b^{-1}, ba^2, b^{-1}a^2, b^{-1}a^{-2}, ba^{-2}$

& b, a^2 generate H!

Freely reducing a language

Lemma 3. L = reg. lang over $S^{\pm 1}$

R = lang obtained from L by freely reducing.

Then R is regular.

Pf. Say L given by FSA M.

If we see

$$\begin{tikzcd} \circ \arrow[r, "s"] \arrow[rr, bend right] & \circ \arrow[r, "s^{-1}"] & \circ \end{tikzcd}$$

add empty edge

$\rightsquigarrow M'$

M' accepts all the words M did <u>plus</u> their freely reduced versions.

Let K = language of <u>all</u> freely reduced words in $S^{\pm 1}$

K is regular (exercise)

& $R = K \cap L(M')$

By Thm, R regular $\square$

Pf of Howson's thm

H, K fin gen. subgps of F_n

H, K are images of reg lang's L_H, L_K by Thm.

By Lemma 3 we may assume L_H, L_K consist of freely red. words, which are exactly elts of H, K (need a free gp for this!)

Other Thm $\Rightarrow$ $\underbrace{L_H \cap L_K}_{\text{elts of } H \cap K}$ regular.

Thm $\Rightarrow$ $H \cap K$ fin. gen.

$\square$

We can now convert $\mathcal{M}_\varepsilon$ into a deterministic automaton, $\mathcal{D}$. The states of $\mathcal{D}$ consist of all the subsets of $V(\mathcal{M}_\varepsilon)$. The single start state of $\mathcal{D}$ is the subset of $V(\mathcal{M}_\varepsilon)$ consisting of all the start states of $\mathcal{M}_\varepsilon$. The accept states of $\mathcal{D}$ are the subsets of $V(\mathcal{M}_\varepsilon)$ that contain at least one accept state of $\mathcal{M}_\varepsilon$. In $\mathcal{D}$ there is an edge from U to U' labelled by x if, for each $v \in U$, there is an edge labelled x from v to some $v' \in U'$, and U' is entirely composed of such vertices. That is, there is an edge labelled x from U to the vertex corresponding to the set

$$U' = \{v' \in V(\mathcal{M}_\varepsilon) \mid v' \text{ is at the end of an edge labelled } x \text{ that begins at some } v \in U\}.$$