Outer space and Automorphisms of free groups

LECTURE 2

We were considering the question:

$F_n = F\langle a_1, \ldots a_n \rangle$, $\{w_1, \ldots, w_n\}$ distinct elements of F_n

when is $a_i \longmapsto w_i$ an automorphism?

eg $\left\{ \begin{array}{l} a \longmapsto ab \\ b \longmapsto ba^{-1}bc^{-1} \\ c \longmapsto cab \end{array} \right.$ <u>Is this an automorphism?</u>

<u>Whitehead</u> gave an algorithm for answering the question, using $M_n = \# S^1 \times S^2$ as a model for F_n, homeomorphisms as a model for automorphisms

<u>Nielsen</u> had shown that the following automorphisms of $F\langle a_1, \ldots, a_n \rangle$ generate $\mathrm{Aut}\, F_n$ (and hence $\mathrm{Out}(F_n)$):

$$\rho_{ij}: \begin{cases} a_i \longmapsto a_i a_j \\ a_k \longmapsto a_k & k \neq i \end{cases}$$

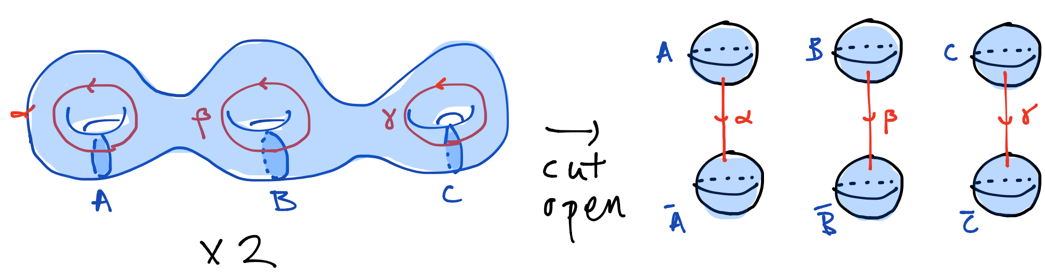
$$\lambda_{ij}: \begin{cases} a_i \longmapsto a_j a_i \\ a_k \longmapsto a_k & k \neq i \end{cases}$$

$\varepsilon_1 : a_1 \longmapsto a_1^{-1}$ generate $\mathrm{Aut}(F_n)$ ($\therefore$ hence $\mathrm{Out}(F_n)$)

Assume this for now

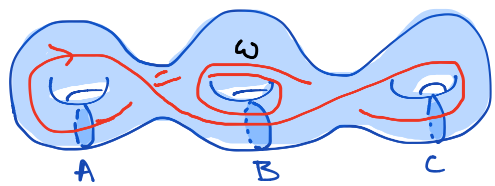
<u>Claim</u> $\pi_0 \,\mathrm{Homeo}\, M_n \longrightarrow \mathrm{Out}(F_n)$ is surjective

By Nielsen, it suffices to realize $\rho_{ij}, \lambda_{ij}, \varepsilon_1$ on M_n.



α, β and γ generate $\pi_1(M_n)$ (we're ignoring basepts)

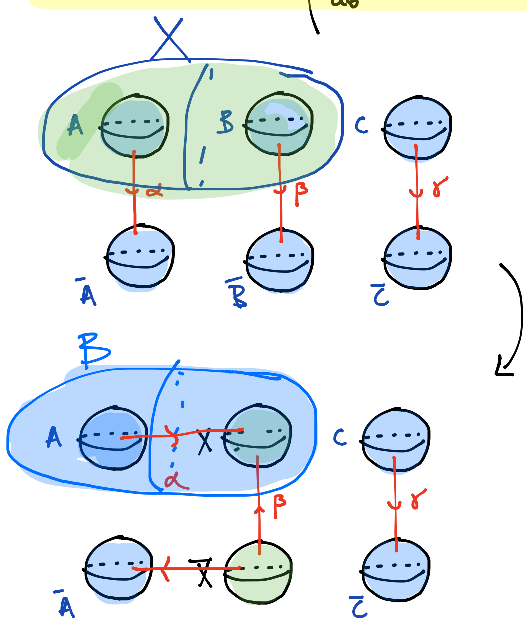
w another loop ($=$ elt of $\pi_1(M_n)$)



$w = b c^{-1} b a^{-1}$

you can read off w as a cyclic word in F_3 by looking at which spheres it punctures (in which direction

To realize $\rho_{ab}: a \longmapsto ab$



Choose a diffeomorphism sending $A \to A$, $B \leftrightarrow X$, $C \to C$

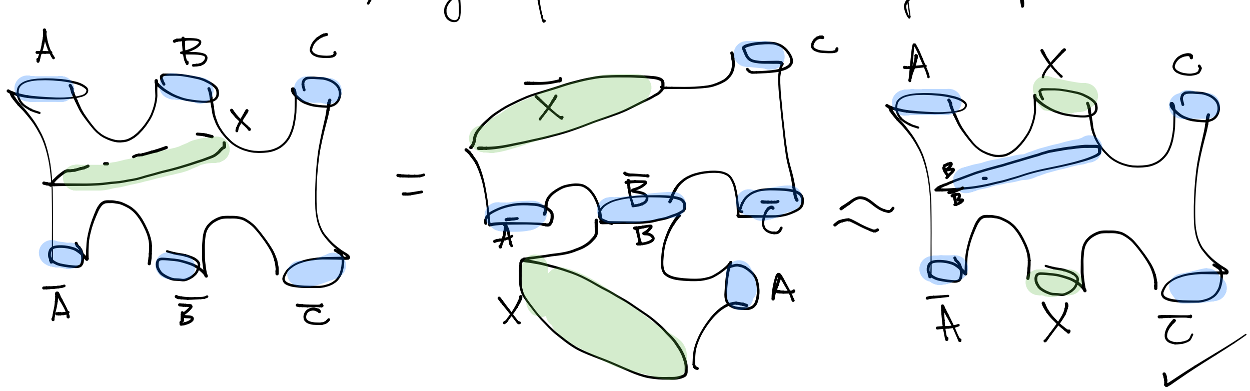
"X is the new B"

This sends $a \mapsto ab^{-1}$

$b \mapsto b^{-1}$

$c \mapsto c$

To see the diffeo, may help to look at <u>half</u> the picture:

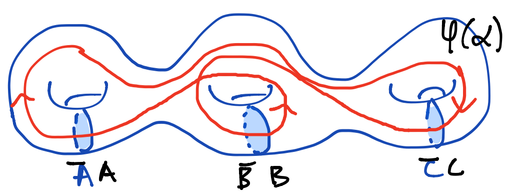


Now look at a general automorphism

Let $\varphi \in \mathrm{Out}\, F_n$, represent φ by a diffeomorphism of M_n

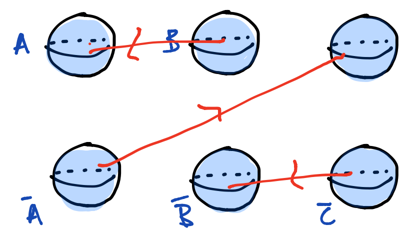
Image of α is a loop

one view:



$\varphi(a) = c^{-1} b a^{-1}$

another view:



Looks like a graph if you squint

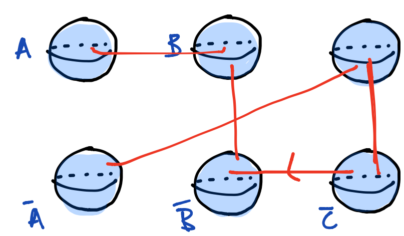
Called the <u>star graph</u> of $\varphi(a)$.

Put $\varphi(a), \varphi(b), \varphi(c)$ in the same graph

= star graph $St(\varphi)$

eg

$$\boxed{\begin{aligned} \varphi(a) &= c^{-1} b a \\ \varphi(b) &= b \\ \varphi(c) &= c \end{aligned}}$$



<u>Notes</u>: ① Valence (X) = valence $(\bar{X})$ = # of occurrences of the letter X in the cyclically reduced w_i.

② $St(\varphi)$ uses all vertices:

If it misses A (and $\bar{A}$) no w_i contains a or a^{-1}

so $\{w_i\}$ isn't a basis

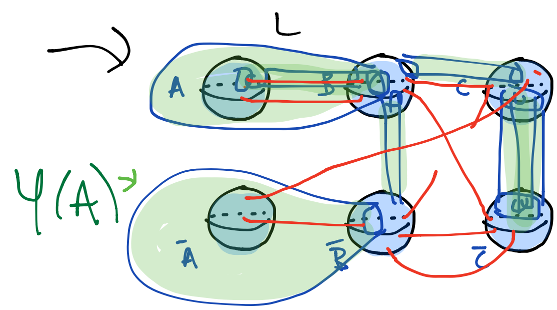
Lemma φ an automorphism $\Rightarrow$ $St(\varphi)$ has a cut vertex or is disconnected

Pf: Put the image $\varphi(A)$ in the picture as well:

First suppose $\varphi(A)$ intersects some $X = A, B$ or C essentially. Make the $\cap$'s with A B & C transverse (so $\varphi(A) \cap (A \cup B \cup C)$ is a union of circles) and minimal. These circles cut $\varphi(A)$ into planar surfaces, including ≥ 2 disks.

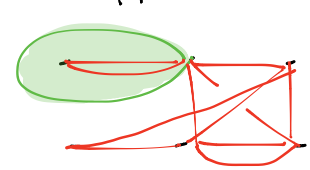
$St(\varphi) = \varphi(\alpha) \cup \varphi(\beta) \cup \varphi(\gamma)$ intersects $\varphi(A)$ in only one point, so one of the disks is missed!

This disk misses $St(\varphi)$ $\longrightarrow$



The disk intersects exactly one X or $\bar{X}$, corresp to a vertex of $St(\varphi)$. There are vertices on both sides (since otherwise you could homotope the disk away) and the graph doesn't connect any vertices on opposite sides

ie X is a cut vertex for $St(\varphi)$:

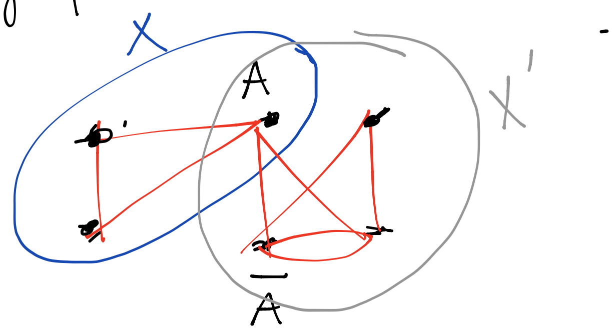


Exercise: What happens if $\varphi(A)$ doesn't intersect any X? Can you still find a cut vertex?

Now for the algorithm:
Define the complexity of $\varphi = \sum_{v \in St(\varphi)} \text{valence}(v)$

Claim I can find a diffeomorphism of M_n which decreases the complexity of the star graph.

pf: Let X be a sphere separating the star graph at the cut vertex:



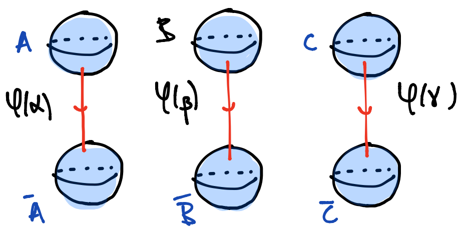
We can assume X separates A from $\bar{A}$:
(if not, use the other sphere)
Choose a diffeomorphism sending $A \leftrightarrow X$, fixing all other spheres

The old star graph intersects X less than it intersects A. So the new star graph has lower complexity. ✓

The algorithm: Keep reducing complexity

Minimal complexity $= 3$: $\varphi(\alpha)$ has to intersect some $A, B,$ or C since it is a non-trivial loop in $\pi_1 M_n$

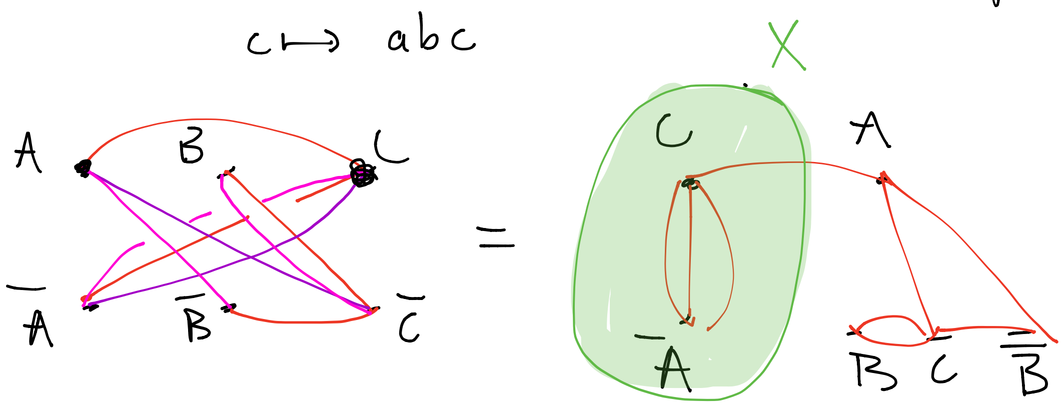
Same for $\varphi(b)$ and $\varphi(c)$



If it is exactly one in each case, then $\{\varphi(\alpha), \varphi(\beta), \varphi(\gamma)\} = \{\alpha^{\pm}, \beta^{\pm}, \gamma^{\pm}\}$ and φ could be an automorphism

Example:
$a \mapsto a\bar{c}bc$
$b \mapsto ca$
$c \mapsto abc$

Is this an automorphism?

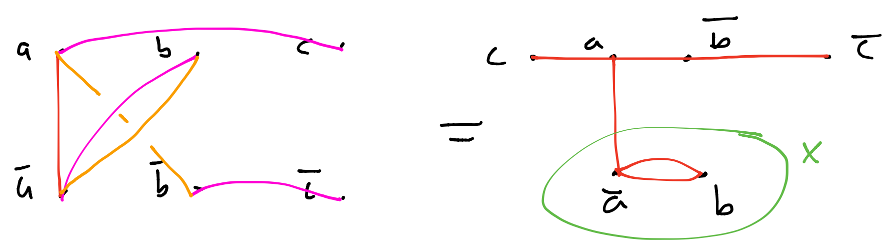


$(\bar{a}^1 \longmapsto \bar{a}^1 c)$ (replace C by X)

So $a \longmapsto \bar{c}^1 a$ should shorten it.

$$\begin{array}{llllll}
a \longmapsto a\bar{c}bc & \longmapsto & \bar{c}^1 a \bar{c}^1 b c & \sim & a\bar{c}^1 b & \\
b \longmapsto ca & \longmapsto & c\,\bar{c}^1 a & = & a & \text{complexity} \\
c \longmapsto abc & \longmapsto & \bar{c}^1 abc & \sim & bc & \underline{\underline{6}}
\end{array}$$

New star graph is



replace $a \leftrightarrow x$

corresps to $\quad c \mapsto c,\ a \longmapsto a \quad b \longmapsto b\bar{a}^1$

$x \mapsto cxc^{-1}$ $\qquad b \mapsto b\bar{a}^1$ $\qquad x \mapsto \bar{a}^1 x a$

$$\begin{array}{lllllll}
\bar{c}^1 a \bar{c}^1 b c & \longmapsto & a\bar{c}^1 b & \longmapsto & a\bar{c}^1 b \bar{a}^1 & \to \bar{c}^1 b & = \bar{c}^1 b \\
a & \longmapsto & cac^{-1} & \longmapsto & cac^{-1} & \to \bar{a}^1 c a \bar{c}^1 a & \sim a \\
\bar{c}^1 abc & \longmapsto & ab & \longmapsto & ab\bar{a}^1 & \to b & \sim b
\end{array}$$

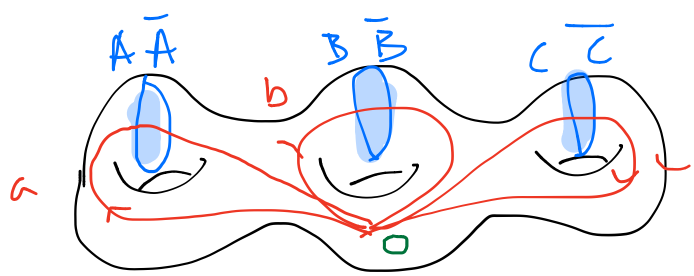
$c \mapsto bc^{-1}$

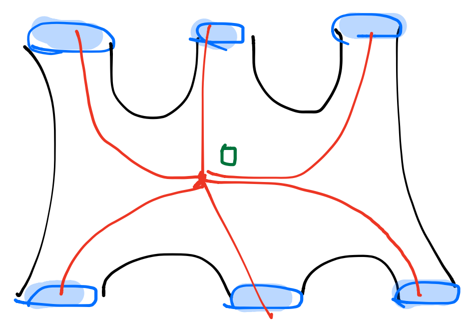
$$\begin{array}{lll}
\bar{c}^1 b \longmapsto c b^{-1} b & = c & \\
\bar{a}^1 c a \bar{c}^1 a \mapsto \bar{a}^1 b \bar{c}^1 a c b^{-1} a = \bar{a}^1 b \bar{c}^1\, a\, c b^{-1} a & & \text{complexity} = \underline{\underline{3}} \\
b \longmapsto b & = b & \text{(minimal!}
\end{array}$$

In this way you reduce to a map sending each generator to a conjugate of another generator. You then need to decide whether that is an automorphism.

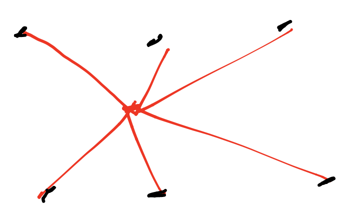
In our case this is <u>not</u> an automorphism! Can't make "a" with this basis, since any word with $\underline{\underline{1}}$ a in it has at least $\underline{\underline{3}}$ a's in it.

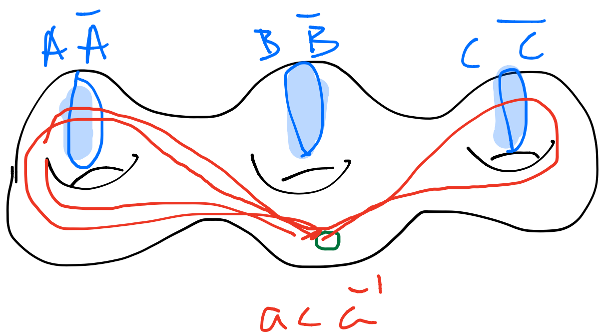
If you want actual automorphisms, need to put a basept in the star graph – slightly more complicated, but the same idea

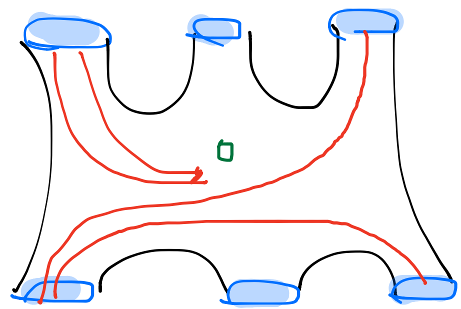




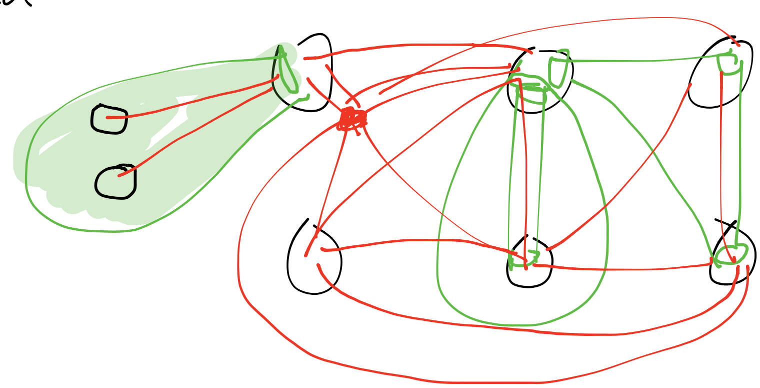
(Now it really <u>does</u> look like a star graph!)







Still true that $St^+(\varphi)$ intersects $\varphi(A)$ in one pt., and $\varphi(A)$ has at least two disk components, so one disk is missed



So graph $St^+\varphi$ has a cut vertex other than 0.

Whitehead refined this to an algorithm:
Given $u_1, \ldots, u_k$, $w_1, \ldots, w_k$ words in the a_i
Is there an automorphism φ with $\varphi(u_i) = w_i$?

<u>Exercise</u>: Use the Whitehead algorithm to decide whether these are automorphisms:

$$\begin{cases} a \mapsto ab \\ b \mapsto ba^{-1}bc^{-1} \\ c \mapsto cab \end{cases}$$

$$\begin{cases} a \mapsto ab \\ b \mapsto bcab \\ c \mapsto b^{-1}a^{-1}c \end{cases}$$

Now write down three random words w_1, w_2, w_3 and use Whitehead's algorithm.

Fast forward 40 years

What about Nielsen's theorem that the P_{ij}, λ_{ij} and ε_1 generate $\mathrm{Aut}(F_n)$?

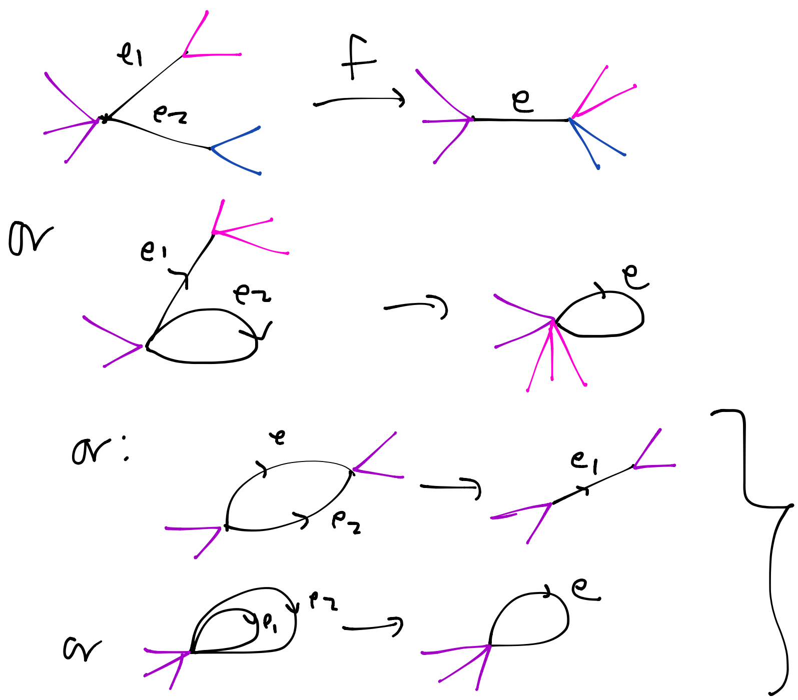
Stallings gave an elegant proof using the graph model for F_n. It also gives an alternate to Whitehead's algorithm.

Def X, Y graphs. A map $f: X \to Y$ is a graph morphism if

1. vertices $\mapsto$ vertices
2. Can subdivide edges of X so that each edge is either collapsed to a vertex or sent linearly to an edge of Y.

Def A Stallings fold is a graph morphism which identifies two edges emanating from the same vertex of X but makes no other identifications

Examples:



singular folds

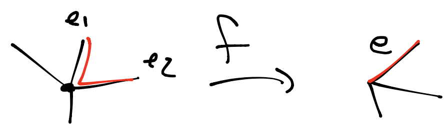
– reduce rk π_1

Non-singular folds are homotopy equivalences.

<u>Lemma</u>: If $f: X \to Y$ a graph morphism is not locally injective, then either

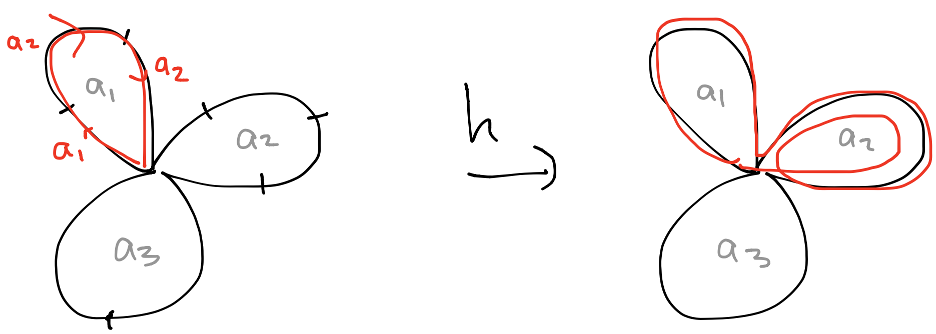
1. some edge collapses or
2. Two edges coming out of the same vertex have the same image

Pf Suppose X is subdivided, and no edge collapses
Linear on edges $\Rightarrow$ loc. injective on edges
Suppose not loc. injective at some vertex:



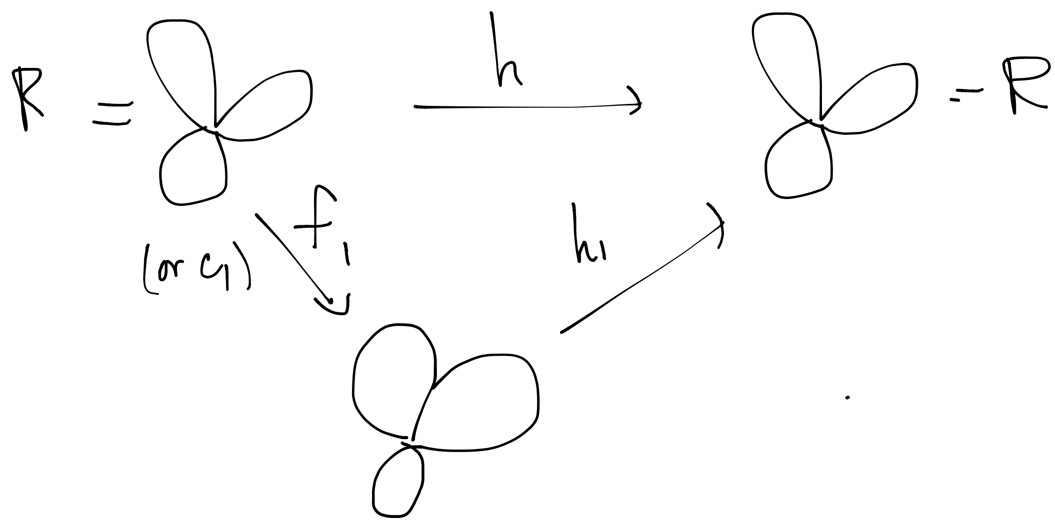
Now let $\varphi \in Out(F_n)$ be an (outer) automorphism
Represent it by a graph morphism which is a homotopy equivalence. h

eg $a_i \longmapsto w_i$



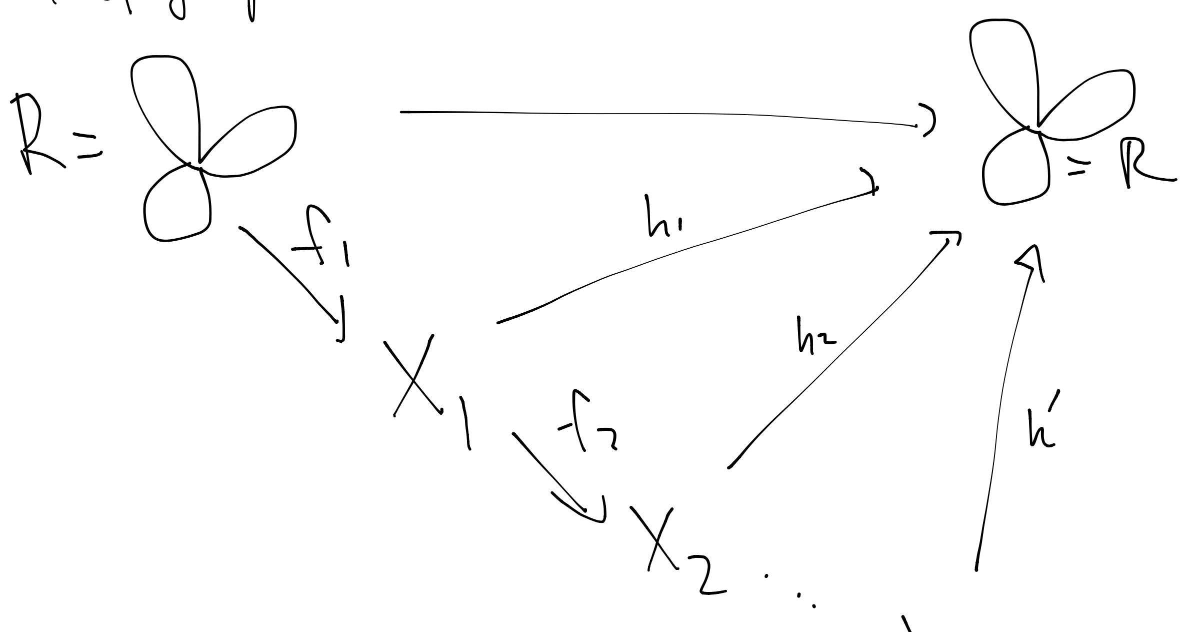
$a_1 \longmapsto a_1 a_2^2$
$a_2 \longmapsto w_2$
$a_3 \longmapsto w_3$

If h is not locally injective, fold or collapse an edge.



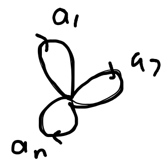
Get a new graph morphism h_1, also a homotopy equivalence. (f_* injective $\Rightarrow$ f is a h.e $\Rightarrow$ h_1 is a he)

Keep going:



until you get to a locally injective $h' : X \longrightarrow R$

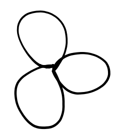
Claim h' is a homeomorphism.

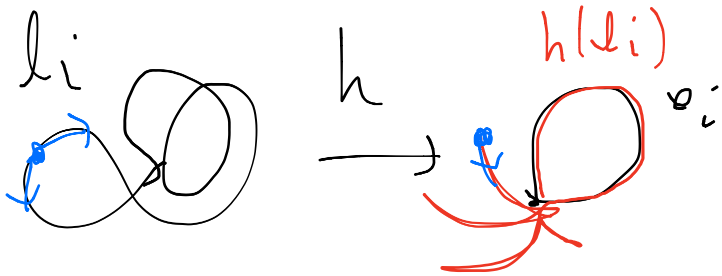
Pf: $X \xrightarrow{h'}$ 

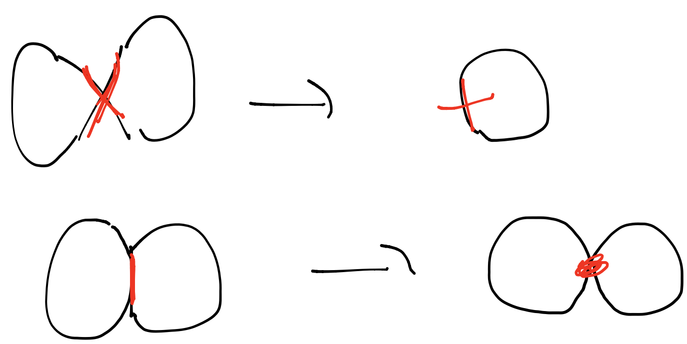
h' a homotopy equivalence $\Rightarrow$ some loop in X is sent to e_i : $h'(l_i) \simeq e_i$

h locally injective $\Rightarrow$

- $h'(l_i) \subseteq e_i$
- l_i is simple
- $l_i \cap l_j$ contains no edges
- $\bigcup l_i = X$
- $l_i \cap l_j = pt$

$\Rightarrow X =$ 





etc.

<u>Claim</u>: This procedure gives a way of factoring φ as a product of $\rho_{ij}, \lambda_{ij}, \sigma, \varepsilon_i$

$$X_0 = R \xrightarrow{f_1} X_1 \xrightarrow{f_2} X_2 \longrightarrow \cdots \xrightarrow{f_r} X_k \xrightarrow{\approx} R$$

signed permutation

Want to identify F_n with $\pi_1 X_i$ for all i

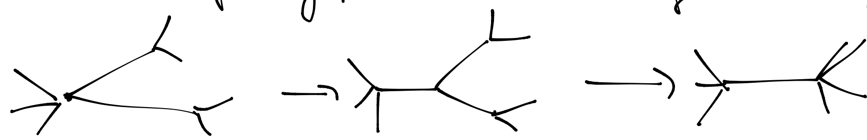
Usual way to do this: choose a maximal tree $T_i \subset X_i$, orient and label the rest of the edges with generators a_i of F_n

$$\begin{tikzcd}
X_0 = R \arrow[r, "f_1"] \arrow[d] & X_1 \arrow[r, "f_2"] \arrow[d] & X_2 \arrow[r] \arrow[d] & \cdots \arrow[r, "f_r"] & X_k \arrow[r, "\approx"] \arrow[d, "\approx"] & R \arrow[d, "\approx"] \\
R_n \arrow[r, bend left, "\varphi_1"] \arrow[rrrrr, bend right, "\varphi"'] & R_n \arrow[r, bend left, "\varphi_2"] & R_n \arrow[r, bend left, "\varphi_3"] & \cdots \arrow[r, bend left, "\varphi_r"] & R_n \arrow[r, "\approx"] \arrow[r, bend left, "h"] & R_n
\end{tikzcd}$$

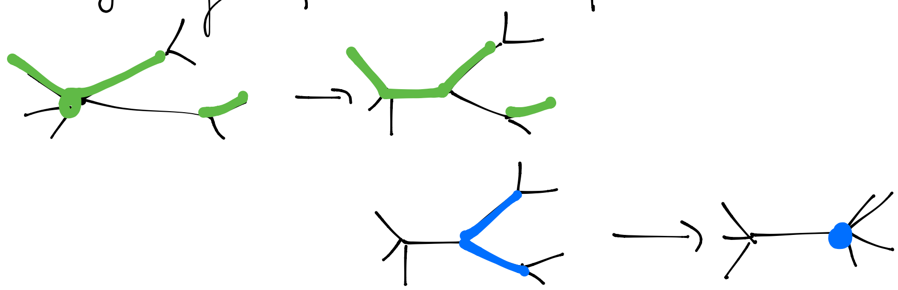
$\varphi = h \circ \varphi_r \circ \cdots \circ \varphi_1$

Each vertical arrow is "collapse T_i"

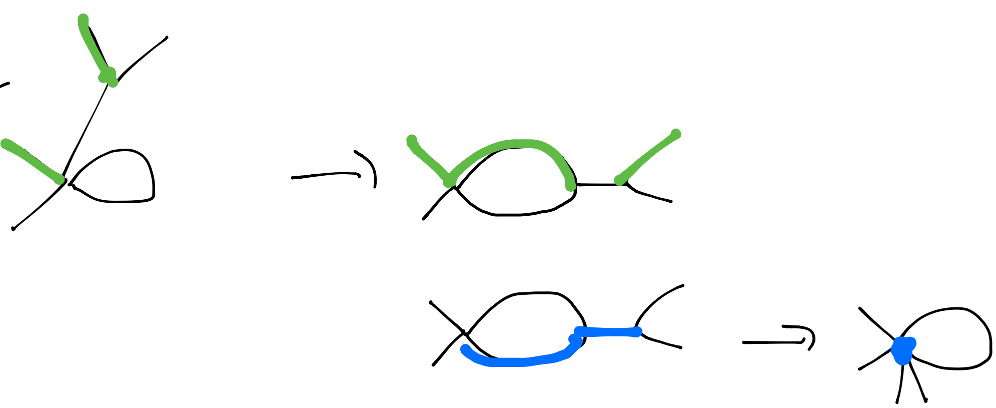
To keep track of what's happening, convenient to: think of a fold as occuring in 2 stages: fold halfway, then the rest of the way



If you have a maximal tree in X, first half grows the tree, second half collapses 2 edges of X, neither a loop:



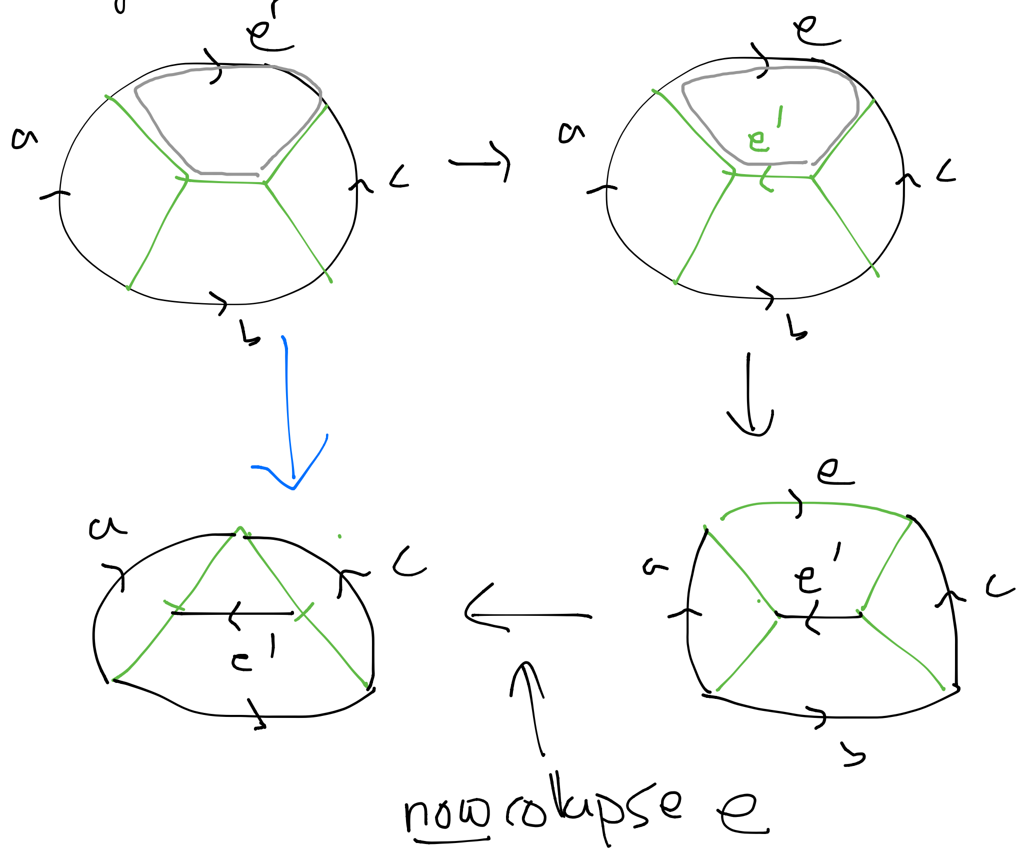
or



So, given a tree $T_i \subset X_i$, first grow it.
Then we want to collapse a couple of edges

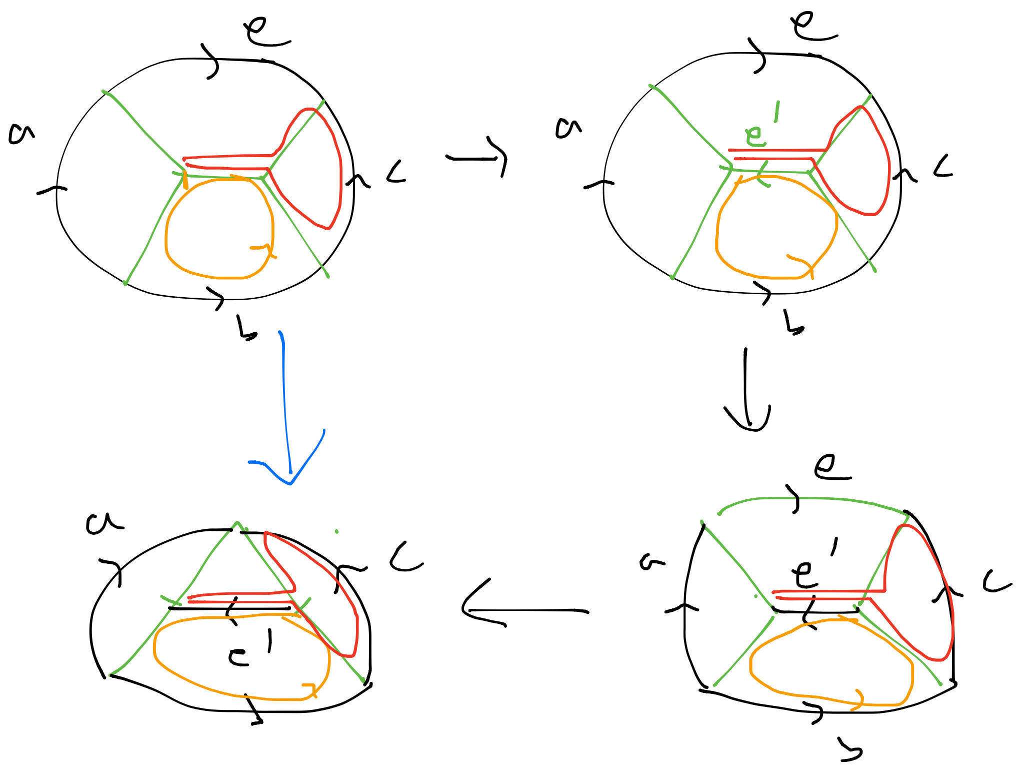
If an edge is in T_i, no problem, get a new tree, nothing has happened to F_n

If you want to collapse $e \notin T_i$:
replace e by an (oriented) edge e' of T_i



Different choices of e' give different isos $\pi_1 X_i \cong F_n$

Effect on F_n:



$a \mapsto a$
$b \mapsto be$
$c \mapsto e^{-1}ce$
$e \mapsto e$

multiplies some generators on right or left by $e^{\pm 1}$

$= \rho_{ij}$ or λ_{ij} or both