Whitehead automorphisms:

(a) W_n (Weyl group of Aut F_n)

(b) $a \in X \uplus X^{-1}$ $\quad X = \{x_1, \dots, x_n\}$ basis of F_n

$A \subset X \cup X^{-1}$

with $a \in A$ $\quad a^{-1} \notin A$

$$(A, a)(x_j) = \begin{cases} x_j & x_j = a^{\pm 1} \\ a^{\alpha_j} x_j a^{-\beta_j} & x_j \neq a^{\pm 1} \end{cases}$$

$$\alpha_j = \begin{cases} 1 & x_j \in A \\ 0 & x_j \notin A \end{cases} \qquad \beta_j = \begin{cases} 1 & x_j^{-1} \in A \\ 0 & x_j^{-1} \notin A \end{cases}$$

Claim: Whitehead automorphisms generate Aut F_n.

We will actually give an algorithm for writing a given automorphism of F_n as a product of Whitehead automorphisms

Set up to graphically describe an automorphism:

As before we call R_n the graph with 1 vertex and n edge pairs with a choice $\mathcal{v}$ of orientation and a labeling $x_1, \dots, x_n$ of the edges of $\mathcal{v}$. There is a natural identification of $\pi_1(R_n, *)$ with $F(x_1, x_2, \dots, x_n)$.

For today's lecture a <u>homotopy equivalence between two connected graphs</u> Γ, Δ is a graph map f such that f induces an isomorphism $f: \pi_1(\Gamma, v) \to \pi_1(\Delta, f(v))$ for some vertex v (and therefore all vertices) of Γ.

(Compare the remarks at bottom of page 5.4 and top of 5.5)

Given a graph Γ and a homotopy equivalence $f : \Gamma \to R_n$, we know that for any vertex v of Γ $\pi_1(\Gamma, v)$ is isomorphic to $F(x_1, \dots, x_n)$ $(=: F_n)$. Once we make this isomorphism explicit f induces a well defined automorphism of F_n. (Compare for a particular example page 5.4)

Remember from chapter 2 (see 2.18) that we get an explicit isomorphism $\pi_1(\Gamma, v) \to F_n$ by

choosing a vertex (basepoint) v_0 of Γ

(*) a maximal tree T in Γ

an orientation of the edges in $\Gamma \setminus T$

a labeling $e_1, \dots, e_n$ of these edges.

Γ is here as always connected.

The isomorphism then maps $\pi_1(\Gamma, v_0)$ to F_n by mapping the reduced path

$[v_0, \iota(e_i)]_T \, e_i \, [\tau(e_i), v_0]_T$ to the reduced word x_i,

Given in addition the homotopy equivalence the associated automorphism is then

$$[x_i] \to \left[f[v_0, \iota(e_i)]_T \, f(e_i) \, f[\tau(e_i), v_0]_T \right]$$

$$\in \pi_1(R_n, *) = F_n$$

Therefore we make the following

<u>5.2 Definition.</u> A (based) marking of a connected finite graph Γ is a choice (*). A graph with a marking and homotopy equivalence $f : \Gamma \to R_n$ we call a marked graph

<u>Remark.</u> Later we also consider markings without base points. That is the reason for thinking of our markings as based markings

The upshot of all this is

5.3 A marked graph $\Gamma_M = \{\Gamma, f, b, T, e_1, \dots, e_n\}$ determines a well defined automorphism of F_n and any automorphism of F_n is determined by a marked graph (Compare page 5.4)

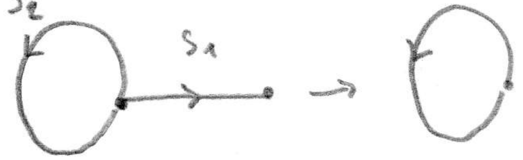
Since Γ is finite by 3.5 there is a sequence $p_1, \dots, p_r$ of foldings $\Gamma_1 = \Gamma \xrightarrow{p_1} \Gamma_2 \to \cdots \xrightarrow{p_r} \Gamma_0$ and an immersion $\Gamma_0 \xrightarrow{g} R_n$ such that

$f = g \circ p_r \circ \cdots \circ p_1$

We know that the p_i are surjective on π_1. Since f is a homotopy equivalence, all p_i must also be homotopy equivalences, consequently, also g is a homotopy equivalence.

We have investigated the effect of foldings on π_1 (see 3.20) If the folding is a homotopy equivalence, there are 2 possibilities pictured in the graph to be folded.

with $\tau(s_1) \neq \tau(s_2)$



with $\tau(s_1) \neq \iota(s_1)$

we will analyse what projections do with respect to automorphisms later.

For immersions we have:

5.4 Proposition. Let $f: \Gamma \longrightarrow R_n$ be a homotopy equivalence and immersion with Γ finite. Then f is an isomorphism. In particular Γ has only one vertex and any autom. of R_n given by any marking of Γ is an element of W_n.

Proof. By 4.10 we can add edges to Γ and extend f to obtain a covering $f': \Gamma' \longrightarrow R_n$

Clearly $f'(\pi_1(\Gamma')) \supset f(\pi_1(\Gamma)) = \pi_1(R_n)$

Since f' is a covering this means that

$f': \pi_1(\Gamma') \xrightarrow{\cong} \pi_1(R_n)$. But $f: \pi_1(\Gamma) \xrightarrow{\cong} \pi_1(R_n)$

So we cannot add any edges, i.e. $\Gamma' = \Gamma$, $f' = f$;

Thus f is a degree 1 covering i.e. an isomorphism. $\square$

So it remains to analyze what happens at foldings.

~~and~~

Scheme of the proof of 5.2.

We start with $(\Gamma, f, b, T, e_1, \dots, e_n)$ where

$f = g \circ p_r \circ \dots \circ p_1$ Call the associated autom. α_r.

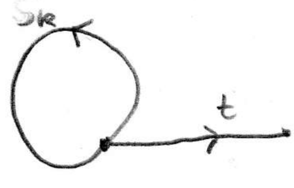
We do induction on r, starting with $r = 0$. By 5.4 we are done.

At the inductive step we change from $g \circ p_r \circ \dots \circ p_1$ to $g \circ p_r \circ \dots \circ p_2$ (and then set $p_i' = p_{i+1}$, $i = 1, \dots, r-1$) and from Γ to $p_1(\Gamma)$, b to $p_1(b)$, and some tree T_1' in $p_1(\Gamma)$ and orienting and labeling edges $e_1', \dots, e_n'$ of $p_1(\Gamma) \setminus T_1$. Let the associated automorphism of F_n be denoted by α_{r-1}. We will show that

$\alpha_r = \alpha_{r-1} \circ w$ where w is a product of 1, 2 or three Whitehead automorphisms, depending on how many of the folding edges with different initial and endpoints lie in T. So we have the following cases:

1. 

with a) both b) one c) none of t_1, t_2 are in T

2. 

with a) t in T b) t not in T

Before we start we use the following

<u>TRP</u>: (Tree Recognition Principle). A connected graph Δ is a tree iff. $\# V_\Delta = 1 + \#\{\text{edge pairs of } \Delta\}$

$$(= 1 + \tfrac{1}{2}\# E_\Delta)$$

<u>Proof</u>. By induction starting with $\# E_\Delta = 0$. □

<u>Case 1.a</u> Set $T' = p_1(T)$. $p_1(T)$ is connected has one edgepair and one vertex less than T. So T' is a tree. Then $p_1(e_1), \ldots, p_1(e_n) =: e_1', \ldots, e_n'$ are edges of an orientation of $p_1(\Gamma) - T'$. For b' we take, as said before, $p_1(b)$. Our new htpy equivalence is

$$f' = g \circ p_r \circ \cdots \circ p_2 \quad \text{so that } f = f' \circ p_1 .$$

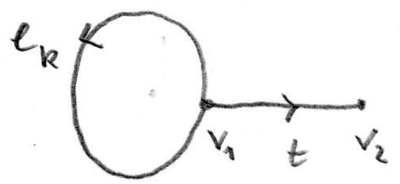
We have to determine the autom. α_{r-1} associated to $(\Gamma', f', b', T', e_1', \ldots, e_n')$. It maps

x_i to the image under f' of $[b', \iota(e_i')]_{T'} e_i' [\tau(e_i'), b']_{T'}$, by construction this is the image of f' of the image of p_1 of the path $\underbrace{[b, \iota(e_i)]_T \, e_i \, [\tau(e_i), b]_T}_{\gamma_i}$

i.e. the image of γ_i under $f' \circ p_1 = f$. Therefore,

$\alpha_{r-1} = \alpha_r$, i.e. $\alpha_r = \alpha_{r-1} \circ id_{F_n}$. □ Case 1.a

Case 2a. (Then the folding loop edge with some orientation is e_k for some $k \in \{1,\dots,n\}$



Orient the nonloop edge so that p_1 maps t and e_k to the same edge in $p_1(\Gamma)$.

Every edge s in T has an orientation determined by b: it is positive if $\iota(s)$ is closer to b in T than $\tau(s)$. Otherwise it is negative.

$$\text{Let} \quad \varepsilon = \begin{cases} +1 & t \text{ is positive} \\ -1 & t \text{ is negative} \end{cases}$$

Let $T' = p_1(T \setminus \{t, \bar{t}\})$. Since $p_1(v_2) = p(v_1)$ we lose one vertex and one edge pair. So T' is a maximal tree of $p_1(\Gamma)$.

Again we take: $e_i' = p_1 e_i$, $i = 1, \dots, n$; $b' = p_1(b)$.

We want to determine $\alpha_{r-1}^{-1} \alpha_r$.

α_{r-1} maps x_i to $f_1\left([b', \iota(e_i')]_{T'} \, e_i' \, [\tau(e_i'), b']_{T'}\right)$

where $f_1 = g \circ p_r \circ \dots \circ p_2$

α_r maps x_i to $f_1 \circ p_1\left([b, \iota(e_i)]_T \, e_i \, [\tau(e_i), b]_T\right)$

so that $\alpha_{r-1} \alpha_r$ maps x_i to $w_i(x_1, x_2, \dots, x_n)$ if

$p_1\left([b, \iota(e_i)]_T \, e_i \, [\tau(e_i), b]_T\right) = w_i(s_1, \dots, s_n)$ where

$$s_j = [b', \iota(e_j')]_{T'} \, e_j' \, [\tau(e_j'), b']_{T'}$$

By definition $e_j' = p_1(e_j)$, so we need to describe

$p_1[b, \iota(e_j)]_T$ and $p_1[\tau(e_j), b]_T$

If $p_1([b, v]_T) \in T'$ then (up to homotopy in T')

$$p_1([b,v]_T) = [b', p_1(v)]_{T'}$$

Any edge of T different from t is mapped by p_1 to an edge of T', and $p_1(t) = e'_k$. If t is positive, i.e. $\varepsilon = +1$, then in any $[b, c(e_i)]_T$ containing t or $\bar{t}$ only t occurs, while in $[z(e_i), b]_T$ if t or $\bar{t}$ occur it will be $\bar{t}$. For $\varepsilon = -1$ it is the other way around. Thus, if we set

$a = x_k^{\varepsilon}$ and define A by $a \in A$, $a^{-1} \notin A$,

$x_i \in A$ iff t or $\bar{t}$ occurs in $[b, c(e_i)]_T$

$x_i^{-1} \in A$ iff t or $\bar{t}$ occurs in $[z(e_i), b]_T$

then

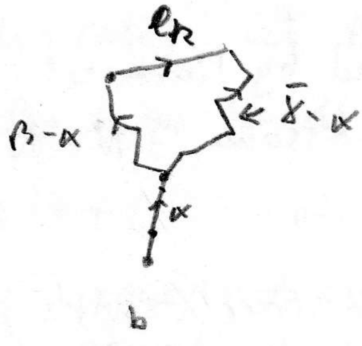
$$\alpha_{r-1}\alpha_r = (A, a) \qquad \square \text{ case 2a}$$

<u>Cases 1b and 2b.</u>

Then in case 1b one of the 2 edges is <u>not</u> in T the other one is, and in 2b the edge t is <u>not</u> in T.

Call the edge not in T in both cases t.

Then there exists k such that e_k equals t or $\bar{t}$. Let $\beta = [b, c(e_k)]_T$, $\gamma = [z(e_k), b]_T$ and let α be the common part of β and $\bar{\gamma}$

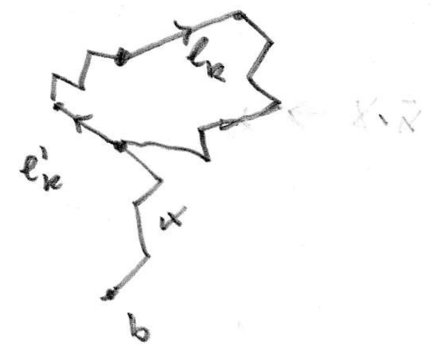


There are again two cases.

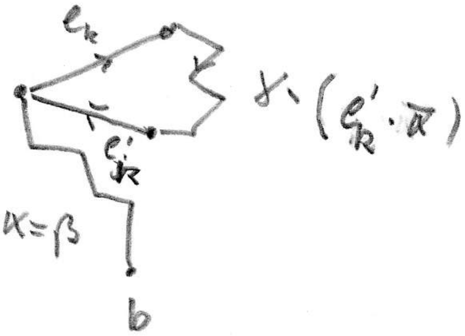
$\beta \setminus \alpha \neq \emptyset$. Then let e'_k be the first edge of $\beta \setminus \alpha$

$\beta \setminus \alpha = \emptyset$. Then $\gamma \setminus \bar{\alpha} \neq \emptyset$ (since $\iota(e_k) \neq \tau(e_k)$). Let e'_k be the last edge of $\gamma \setminus \bar{\alpha}$

1st case



2nd case



Now let $T' = T \cup \{e_k, \bar{e}_k\} \setminus \{e'_k, \bar{e}'_k\}$ which is again a maximal tree. Again, let α_r be the automorphism of F_n associated to $\{\Gamma, f, b, T, e_1, \dots, e_n\}$ and α'_r the automorphism associated to $\{\Gamma, f, b, T', e_1, \dots, e_{k-1}, e'_k, e_{k+1}, \dots, e_n\}$

We have to show that ${\alpha'_r}^{-1} \alpha_r$ is a Whitehead automorphism. Under α_r x_j gets mapped to the image under f of

$$[b, \iota(e_j)]_T \, e_j \, [\tau(e_j), b]_T$$

Under α'_r x_j gets mapped to the image under f of

$$[b, \iota(e_j)]_{T'} \, e_j \, [\tau(e_j), b]_{T'} \quad \text{for } j \neq k \text{ and}$$

to $[b, \iota(e'_k)]_{T'} \, e'_k \, [\tau(e'_k), b]_{T'}$

Let us look first at $j = k$

1st case

$$[b, \iota(e_k')]_{T'} = \alpha = [b, \iota(e_k')]_T$$

$[\tau(e_k'), b]_{T'} = [\tau(e_k'), \iota(e_k)]_T \, e_k \, [\tau(e_k), b]$ so that

$$[b, \iota(e_k')]_{T'} \, e_k' \, [\tau(e_k'), b]_{T'} =$$

$[b, \iota(e_k)]_T \, e_k \, [\tau(e_k), b]_T$, i.e. $\alpha_r(x_k) = \alpha_r'(x_k)$.

2nd case

$$[b, \iota(e_k')]_{T'} = [b, \iota(e_k)]_T \, e_k \, [\tau(e_k), \iota(e_k')]_T$$

and $[\tau(e_k'), b]_{T'} = [\tau(e_k'), b]_T$, so that again

$\alpha_r(x_k) = \alpha_r'(x_k)$

Now we look at $j \neq k$. Then $e_j \in (\Gamma \setminus T) \cap (\Gamma \setminus T')$

1st case:

If $[b, \iota(e_j)]_T$ contains e_k' or $\bar{e}_k'$ it will contain e_k

and $[b, \iota(e_j)]_{T'} \underset{\substack{\uparrow \\ \text{homotopy} \\ \text{in } T'}}{\simeq} \underbrace{[b, \iota(e_k')]_T}_{\alpha} \cdot \underbrace{[\iota(e_k'), \tau(e_k)]_T}_{\bar{\gamma} \cdot \alpha} \cdot \bar{e}_k$

$$\cdot [\iota(e_k), \tau(e_k')]_T \cdot [\tau(e_k'), \iota(e_j)]_T$$

$$\simeq \bar{\gamma} \, \bar{e}_k \, \bar{\beta} \cdot [b, \iota(e_j)]_T$$

$$\simeq [\beta \cdot e_k \gamma]^{-1} \cdot [b, \iota(e_j)]_T$$

2^{nd} case

If $[b, \iota(e_j)]_T$ contains e_k' or $\overline{e_k'}$ then it contains $\overline{e_k'}$

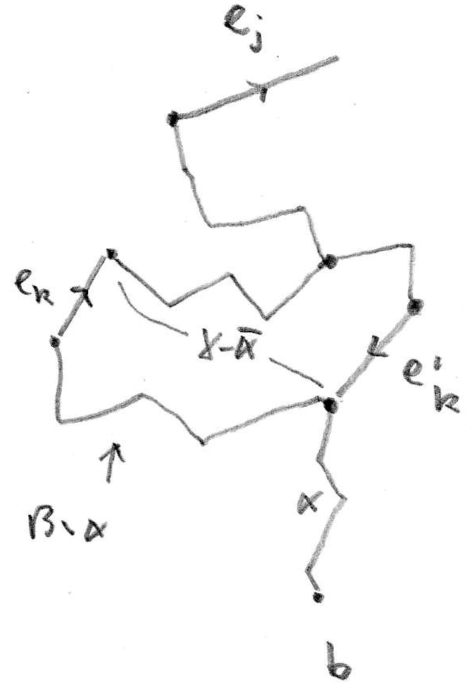
and

$$[b, \iota(e_j)]_{T'} \simeq \underbrace{[b, \tau(e_k')]_T}_{\alpha} \cdot \underbrace{[\tau(e_k'), \iota(e_k)]_T}_{\beta \cdot \alpha}$$

$$\cdot e_k \cdot [\tau(e_k), \iota(e_k)]_T \cdot$$

$$[\iota(e_k), \iota(e_j)]_T$$

$$\simeq \beta \cdot e_k \cdot \gamma \cdot [b, \iota(e_j)]_T$$



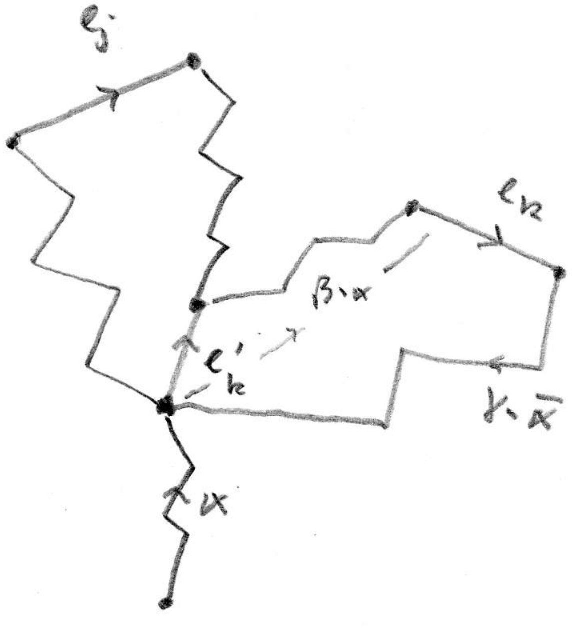
One obtains similar expressions if $e'_k, \overline{e'_k}$ is part of $[\tau(e_j), b]_T$:

1^{st} case

$$[\tau(e_j), b]_{T'} \simeq [\tau(e_j), \tau(e'_k)]_T \, [\tau(e'_k), \iota(e_k)] \cdot e_k \cdot$$

$$[\tau(e_k), \iota(e'_k)]_T \cdot [\iota(e'_k), b]_T$$

$$\simeq [\tau(e_j), b]_T \cdot [\beta e_k \gamma]$$



and in 2^{nd} case

$$[\tau(e_j), b]_{T'} \simeq [\tau(e_j), b]_T \cdot [\,\overline{\beta e_k \gamma}\,].$$

If $[b, \iota(e_j)]_T$ or $[\tau(e_j), b]_T$ do not contain $e'_k, \overline{e'_k}$ obviously the path does not change, if we replace T by T'

To put it together:

In the first case we have

$$\alpha_r : \quad x_j \to [b, \iota(e_j)]_T \, e_j \, [\tau(e_j), b]_T \xrightarrow{\ f\ }$$

$$\alpha_r' : \quad x_j \to [b, \iota(e_j)]_{T'} \, e_j \, [\tau(e_j, b)]_{T'} \xrightarrow{\ f\ }$$

$$= \underbrace{[\beta e_k \gamma]^{-\alpha_j}}_{(f^{-1}\alpha_r(x_k))^{-\alpha_j}} \, [b, \iota(e_j)]_T \, e_j \, [\tau(e_j), b]_T \, \underbrace{[\beta e_k \gamma]^{\beta_j}}_{(f^{-1}\alpha_r(x_k))^{\beta_j}}$$

$$\text{where } \alpha_j = \begin{cases} 1 & \text{if } e_k' \text{ or } e_k \text{ is in } [b, \iota(e_j)]_T \\ 0 & \text{if not} \end{cases}$$

$$\beta_j = \begin{cases} 1 & \text{if } e_k' \text{ or } e_k \text{ is in } [\tau(e_j), b]_T \\ 0 & \text{if not.} \end{cases}$$

Thus $\alpha_r'^{-1} \alpha_r(x_j) = x_k^{\alpha_j} \, x_j \, x_k^{-\beta_j}$

In the second case we have

$$\alpha_r'^{-1} \alpha_r(x_j) = (x_k^{-1})^{\alpha_j} \, x_j \, (x_k^{-1})^{-\beta_j}.$$ Thus

$\alpha_r'^{-1}\alpha_r = (A, a)$ with $a = \begin{cases} x_k & 1^{st} \text{ case} \\ x_k^{-1} & 2^{nd} \text{ case} \end{cases}$

and $x_k \in A$, $x_j \in A$ if $e_k' \overset{or}{\vee} \bar{e}_k$ is in $[b, \iota(e_j)]_T$ cases 1b, 2b
$x_j^{-1} \in A$ if $e_k' \vee \bar{e}_k$ is in $[\tau(e_j), b]_T$. $\square$

Last case: 1c.

In the treatment of cases 2b, 1b we have seen, what happens when we change the tree T not containing one of our non-loop folding edges to a tree T' containing it.

All that was required, was that the folding edge t to be put into the tree T' had distinct end points. In case 1.c both folding edges have distinct endpoints. So we first go from T to T' to include one of them. Let us say this is t_1 in

Clearly, we can pass from T' to T'' to include t_2. But when doing this, we have to remove some edge pair from T', and if we are not lucky, we may have to remove t_1 to move t_2 into the tree. Now looking at T', assuming t_2 is not in T'. For a somewhat simpler picture we made a fixed joice of e_k' in our argument for cases 1b, 2b. But we are free to choose e_k' from any edge of $\beta \setminus \alpha$, $\bar{\gamma} \setminus \alpha$.

So we are in trouble if $(\beta \setminus \alpha) \cup (\bar{\gamma} \setminus \alpha)$ consists of exactly one edge, and this one is t_1 or $\bar{t}_1$.

β, γ are the shortest paths in T' going to the endpoints of t_2. From the picture above we see that the last edge in T' in the path from b to $\tau(t_2)$ cannot be t_1. We choose this edge for e_k or $\bar{e}_k$. $\square$