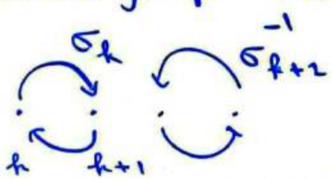
Erice, 9/6/22

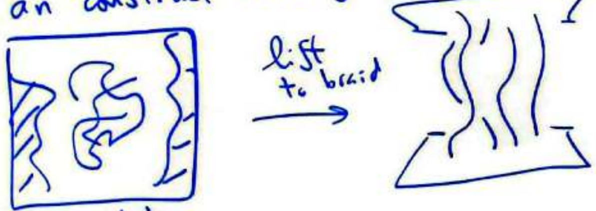
Braid group: B_n



Relations: $\sigma_1\sigma_2\sigma_1 = \sigma_2\sigma_1\sigma_2$

$\sigma_1\sigma_3 = \sigma_3\sigma_1$

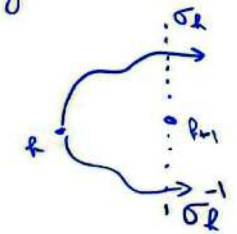
Can construct braid from orbit data



NOT closed

Float data

Assign generators for each crossing along a projection line.



Obtain sequence $\alpha = \sigma_2\sigma_1^{-1}\sigma_3\sigma_1^{2}\sigma_2^{-1}\cdots$

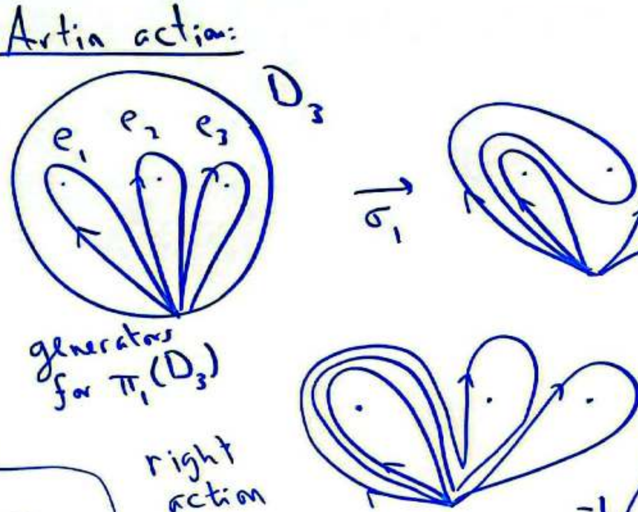
If closed, change of projection is conjugation $\alpha' = \beta\alpha\alpha^{-1}$

Growth of braid: dilatation of associated mapping class of homeos $D_n \to D_n$.

Asymptotic growth rate of loop "caught" on braid

How to compute this?
Dynnikov coordinates, etc.

Artin action:



generators for $\pi_1(D_3)$

right action

$e_1\sigma_1 = e_1e_2e_1^{-1}$

$e_2\sigma_1 = e_1$

$e_3\sigma_1 = e_3$

$e_1e_2\sigma_1 = (e_1\sigma_1)(e_2\sigma_1)$

$e_1\sigma_1\sigma_2 = (e_1\sigma_1)\sigma_2$

"word length" in π_1 grows at the same rate as the growth of braid.

So just count symbols!

$e_1\sigma_1\sigma_2^{-1} = (e_1e_2e_1^{-1})\sigma_2^{-1}$
$= (e_1\sigma_2^{-1})(e_2\sigma_2^{-1})(e_1^{-1}\sigma_2^{-1})$
$= e_1(e_3)(e_1^{-1})$

Now apply to each symbol → grows exponentially.
Very hard to keep track of in practice.

Better: "Abelianize"

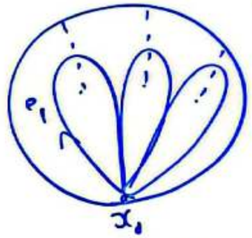
$e_1\sigma_1 = \cancel{e_1} + e_2 - \cancel{e_1}$ oops!

$e_2\sigma_1 = e_1$

$e_3\sigma_1 = e_3$

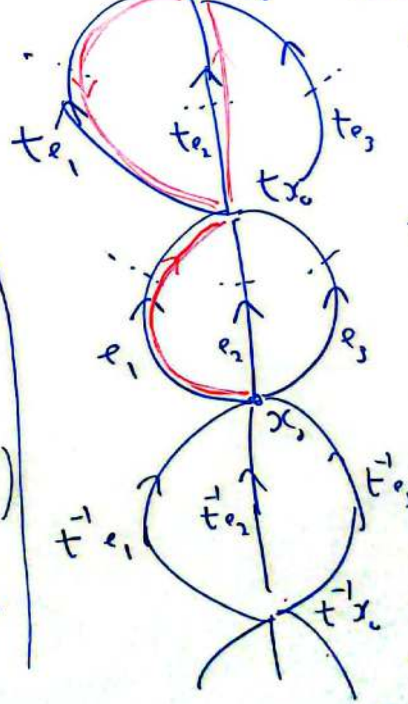
Linear representation, but of symmetric group! No growth.

Keep a bit more info. Use a branched cover of D_3. Make 3 "cuts".



Crossing a cut takes you up/down a "floor" (deck)

Loop e_1 becomes an arc



Schematic representation as a "1-skeleton".

$e_1\sigma_1 = e_1(te_2)(te_1)^{-1}$

t is a symbolic parameter that labels the deck.

Abelianize:

$e_1\sigma_1 = e_1 + te_2 - te_1$
$= (1-t)e_1 + te_2$

Linear representation, but not symmetric group.

Burau representation:

$e_1\sigma_1 = (1-t)e_1 + e_2$

$e_2\sigma_1 = e_1$

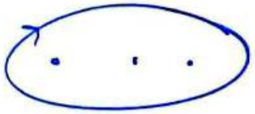
$e_3\sigma_1 = e_3$

Matrix: $[\sigma_1] = \left(\begin{array}{cc|c} 1-t & 1 & 0 \\ 1 & 0 & 0 \\ \hline 0 & 0 & 1 \end{array}\right)$

$[\sigma_2] = \left(\begin{array}{c|cc} 1 & 0 & 0 \\ \hline 0 & 1-t & 1 \\ 0 & 1 & 0 \end{array}\right)$

etc. for more strings.

$e_1 + te_2 + t^2e_3 + \dots + t^{n-1}e_n$ is invariant.



corresponds to loop around all the punctures. (lifted)

Change basis: $u_1 = e_1 - e_2$

$u_2 = e_2 - e_3$

$\vdots$

$u_{n-1} = e_{n-1} - e_n$

Get reduced Burau representation (more familiar)

$$[\sigma_k] = I_{k-2} \oplus \begin{pmatrix} 1 & t & 0 \\ 0 & \boxed{-t} & 0 \\ 0 & 1 & 1 \end{pmatrix} \oplus I_{n-k-2}$$

in k,k position.

So far t is a symbolic parameter.

Can pick specific values: $t^m = 1 \rightarrow$ finite cover

$t=1$: no cover

$t=-1$: double-cover

Thm: (Fried / Kolev) braid α with Burau rep $[\alpha]$

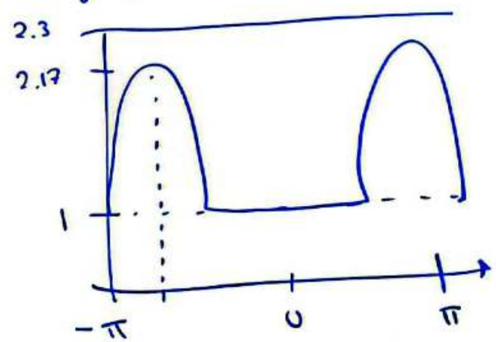
growth $\lambda \geq \sup_{|t|=1} \operatorname{spr}[\alpha]$

This is a fast and easy bound on the growth.

If bound is exact, then it must be so for $t=-1$ Boyland/Harrington

(cover coincides with "orientation double cover")

But $t=-1$ not always best choice



$t = e^{i\theta}$

$\alpha = \sigma_1\sigma_2\sigma_3^{-1}$

Not much is known about the "quality" of this bound.

See also Lawrence-Krammer representation, which can sometimes improve bound. (example by Lynch)

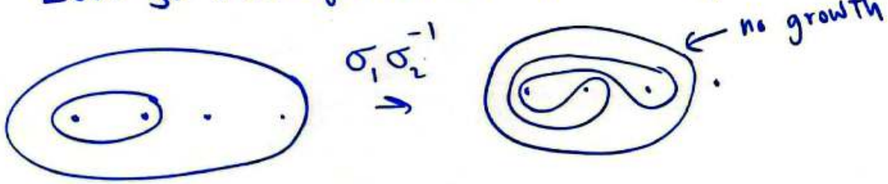
What else is Burau good for?

Yeung, Cohen-Steiner, Desbrun (2020)

Use Burau eigensystem to quantify "coherence"

reducible braid
(one of the 3
cases in TN th.)

Related to detection of "Lagrangian coherent structures"
Allshouse + J-LT → using Dynnikov loop coords.
Look for reducing curves as loops with slow growth.

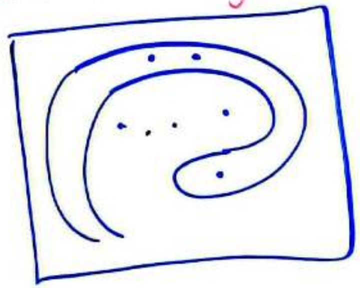


YCD had the insight that the eigenvector
structure of [α] captures coherence with
piecewise-constant eigenvector entries.
Use Burau param t near 1 with $\arg t < \frac{1}{n}$
to guarantee unitarity and existence of e-vectors.
FAST! Room for refinement.

example: modified Duffing oscillator

+ rotation

chaotic
+ invariant regions



Algos detect
the fact that trajectories
belong to different dynamical
regions.