Growth functions of braid monoids and generation of random braids

Juan González-Meneses

Universidad de Sevilla

Joint with Volker Gebhardt

GEOMETRIC AND ASYMPTOTIC GROUP THEORY WITH APPLICATIONS (GAGTA-5)

July 11th to 15th, 2011

Manresa (Barcelona, Spain)

Introduction

Most authors working in braid-cryptography or computational braid theory,

when they perform computations using random braids,

they don't use random braids.

They use random words.

Introduction

We will focus on the **positive braid monoid**:

$$B_n^+ = \left\langle \sigma_1, \sigma_2, \ldots, \sigma_{n-1} \;\middle|\; \begin{array}{ll} \sigma_i\sigma_j = \sigma_j\sigma_i & |i-j| > 1 \\ \sigma_i\sigma_j\sigma_i = \sigma_j\sigma_i\sigma_j & |i-j| = 1 \end{array} \right\rangle^+$$

Length of a (positive) braid = Word length

Lattice structure of B_n^+

$a \preccurlyeq b \quad \Leftrightarrow \quad a$ is a prefix of b → Unique gcd's ($\wedge$) and lcm's ($\vee$)

$$\sigma_1 \vee \sigma_2 = \sigma_1\sigma_2\sigma_1$$

$$\sigma_1 \vee \sigma_3 = \sigma_1\sigma_3$$

Introduction

We want to generate a random positive braid of length k.

What if we take the product of k randomly chosen generators?

There is only one word representing the braid $\sigma_1\sigma_1\sigma_1\sigma_1\sigma_1\sigma_1$.

There are 16 words representing the braid $\Delta = \sigma_1\sigma_2\sigma_1\sigma_3\sigma_2\sigma_1$.

The probability of obtaining Δ is 16 times the probability of obtaining $(\sigma_1)^6$.

This becomes more dramatic as n and k grow.

Generating random braids

How do we generate braids with the same probability?

Given a braid α, its Lex-representative, $w(\alpha)$, is the (lexicographically) smallest word representing α.

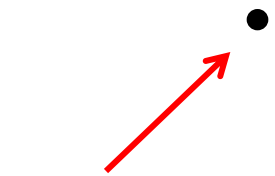
$$w(\sigma_1\sigma_1) = \sigma_1\sigma_1 \qquad w(\sigma_3\sigma_1) = \sigma_1\sigma_3$$

$$\{ \text{ Braids of length } k \} \xleftrightarrow{\text{Bij.}} \{ \text{ Lex-representatives of length } k \}$$

Generating random braids

The situation can be described using a rooted tree.

Example, in B_4^+ :



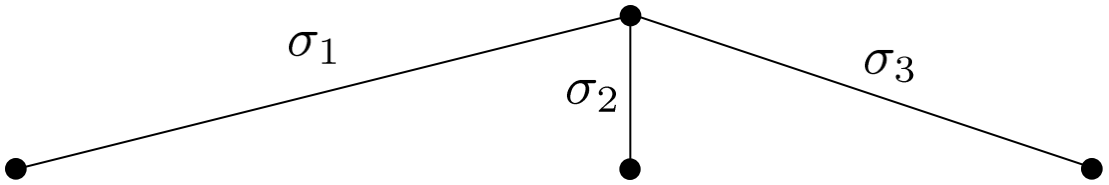
Root.

The only braid of length 0 in B_4^+.

Generating random braids

The situation can be described using a rooted tree.

Example, in B_4^+ :

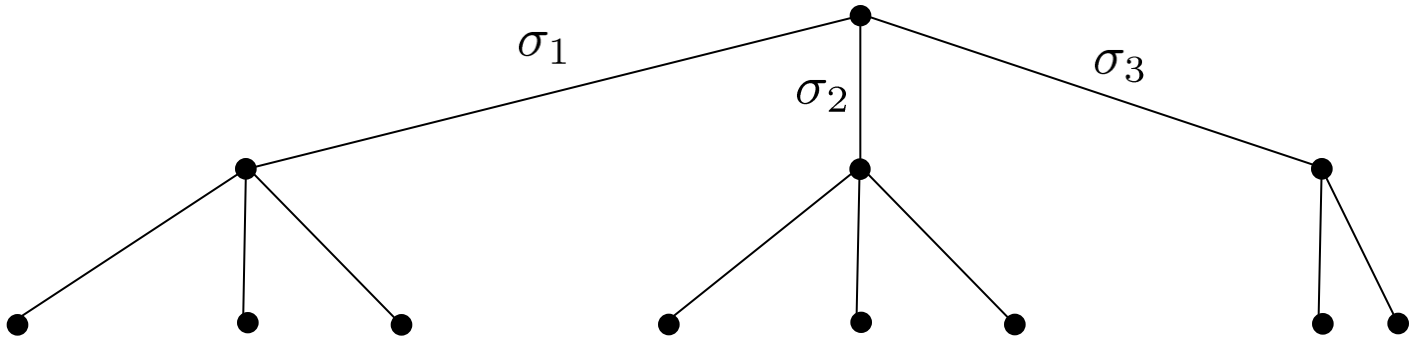


3 braids of length 1 in B_4^+.

Generating random braids

The situation can be described using a rooted tree.

Example, in B_4^+ :



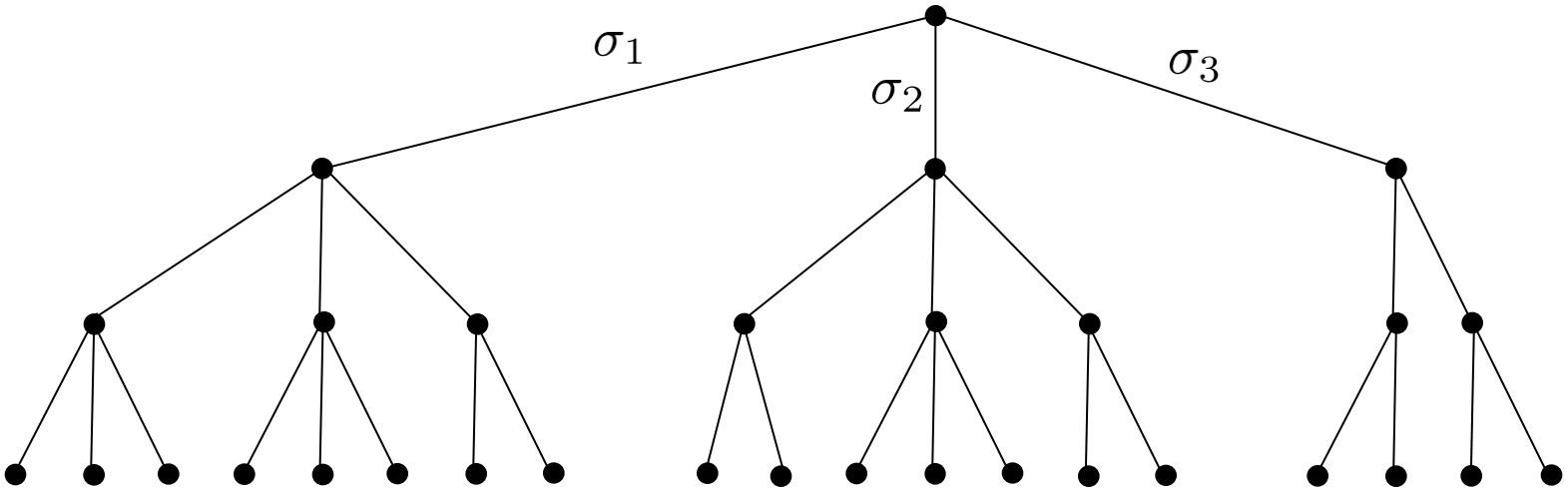
8 braids of length 2 in B_4^+.

The word $\sigma_3\sigma_1$ is not there, as the Lex-representative of $\sigma_3\sigma_1$ is $\sigma_1\sigma_3$.

Generating random braids

The situation can be described using a rooted tree.

Example, in B_4^+ :



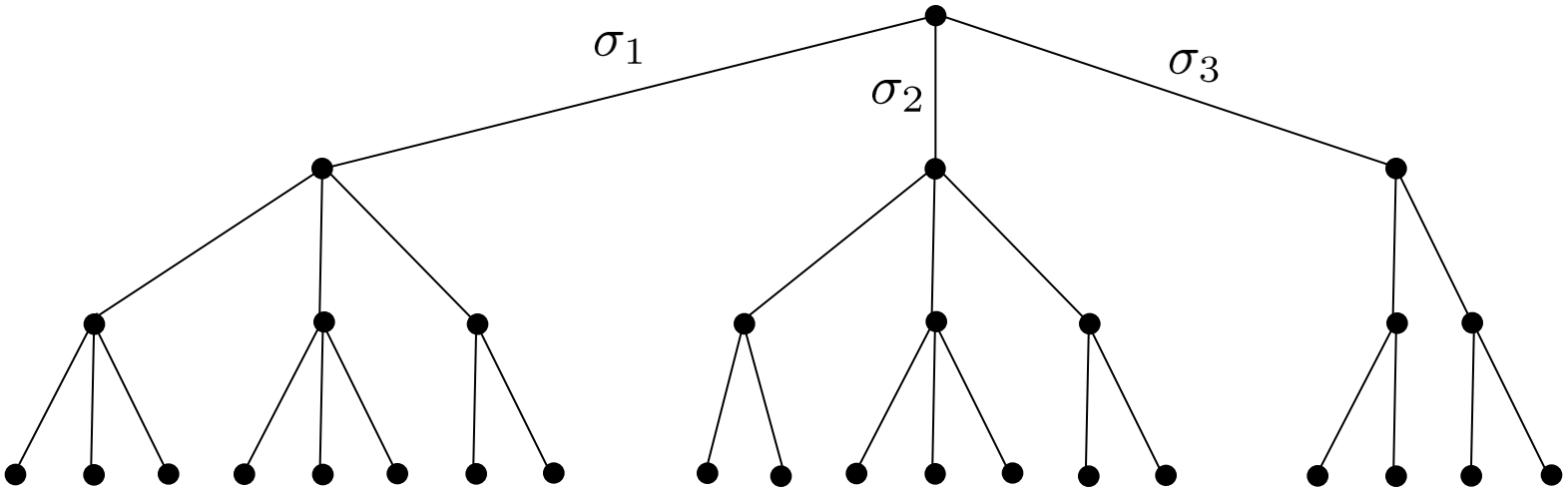
19 braids of length 3 in B_4^+.

There is no subword $\sigma_3\sigma_1$,
neither $\sigma_2\sigma_1\sigma_2$, nor $\sigma_3\sigma_2\sigma_3$.

Generating random braids

The situation can be described using a rooted tree.

Example, in B_4^+ :

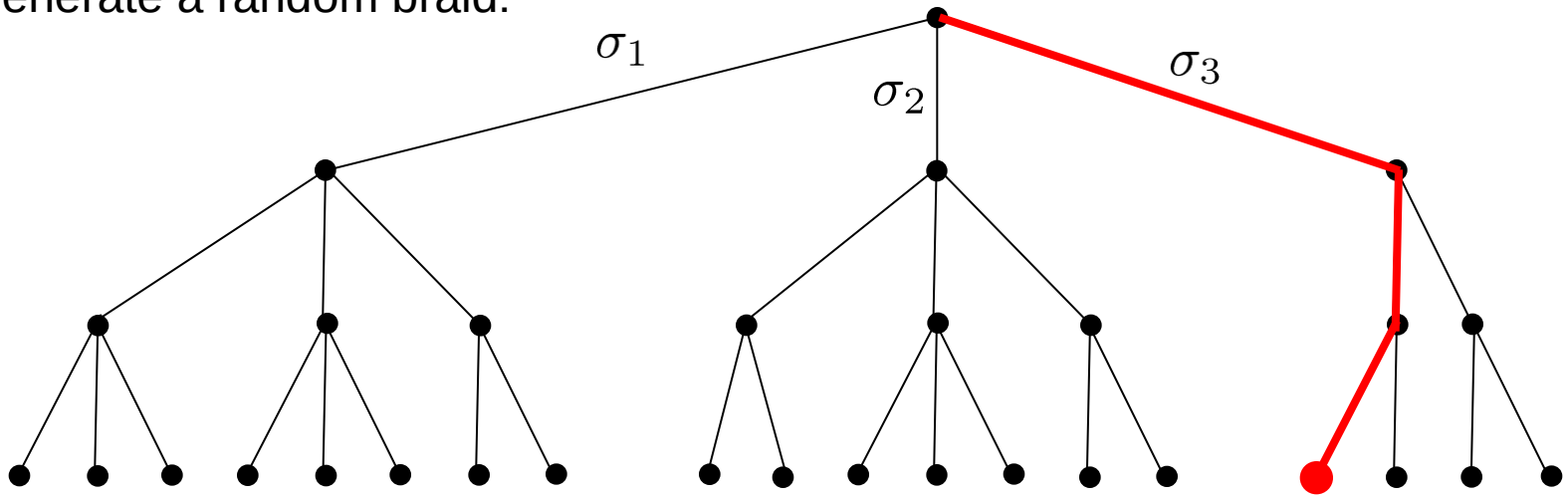


19 braids of length 3 in B_4^+.

There is no subword $\sigma_3\sigma_1$,
neither $\sigma_2\sigma_1\sigma_2$, nor $\sigma_3\sigma_2\sigma_3$.

Generating random braids

To generate a random braid:



1) Compute the number of leaves of the tree: 19

2) Choose a random number between 1 and 19: 16

3) Find the braid corresponding to the 16th leaf: $\sigma_3\sigma_2\sigma_1$

In polynomial time!

Warning: the tree is exponentially big!

Counting braids of given length

$x_{n,k}$ = Number of braids in B_n^+ of length k.

How to compute this number?

Growth function of B_n^+: $$G_n(t) = \sum_{k \geq 0} x_{n,k}\ t^k$$

It is a rational function $\Leftrightarrow$ $\exists$ recurrence relation $x_{n,i} = c_1 x_{n,i-1} + c_2 x_{n,i-2} + \cdots + c_m x_{n,i-m}$ for $i \geq K$

It is the inverse of a polynomial $\Leftrightarrow$ $K = 1$

Deligne (1972): The growth function of B_n^+ is the inverse of a polynomial.

$$G_n(t) = \sum_{k \geq 0} x_{n,k}\ t^k = \frac{1}{H_n(t)}$$

Counting braids of given length

$x_{n,k}$ = Number of braids in B_n^+ of length k.

How to compute this number?

Growth function of B_n^+: $$G_n(t) = \sum_{k \geq 0} x_{n,k}\ t^k$$

It is a rational function $\Leftrightarrow$ $\exists$ recurrence relation $x_{n,i} = c_1 x_{n,i-1} + c_2 x_{n,i-2} + \cdots + c_m x_{n,i-m}$ for $i \geq K$

It is the inverse of a polynomial $\Leftrightarrow$ $K = 1$

Deligne (1972): The growth function of B_n^+ is the inverse of a polynomial.

$$G_n(t) = \sum_{k \geq 0} x_{n,k}\ t^k = \frac{1}{H_n(t)}$$

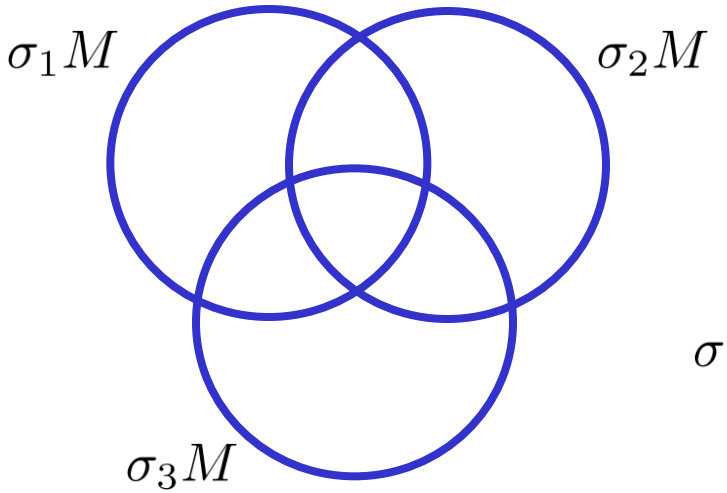
Counting braids of given length

Deligne (1972): The growth function of B_n^+ is the inverse of a polynomial.

$$G_n(t) = \sum_{k \geq 0} x_{n,k}\, t^k = \frac{1}{H_n(t)}$$

Proof: Denote $M = B_n^+$ $\qquad (aM)_k =$ multiples of a of length k

$$x_{n,k} = (\sigma_1 M)_k \cup (\sigma_2 M)_k \cup \cdots \cup (\sigma_{n-1} M)_k$$



$$\sigma_1 M \cap \sigma_2 M = (\sigma_1 \vee \sigma_2) M$$

$$\sigma_1 M \cap \sigma_2 M \cap \sigma_3 M = (\sigma_1 \vee \sigma_2 \vee \sigma_3) M$$

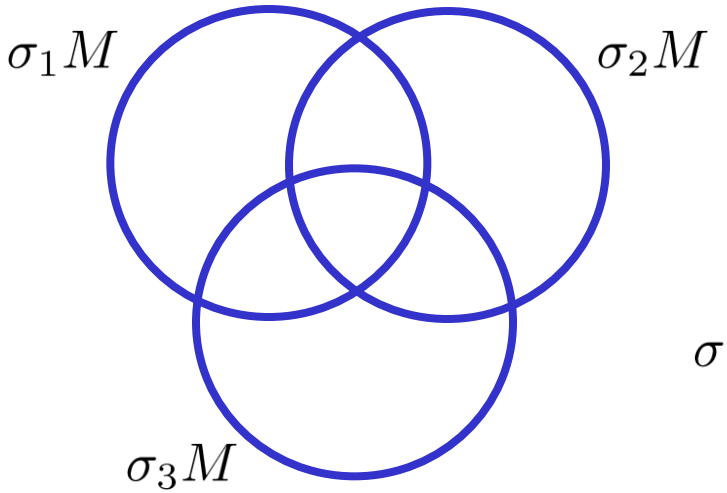
Counting braids of given length

Deligne (1972): The growth function of B_n^+ is the inverse of a polynomial.

$$G_n(t) = \sum_{k \geq 0} x_{n,k}\, t^k = \frac{1}{H_n(t)}$$

Proof: Denote $M = B_n^+$ $\qquad (aM)_k =$ multiples of a of length k

$$x_{n,k} = (\sigma_1 M)_k \cup (\sigma_2 M)_k \cup \cdots \cup (\sigma_{n-1} M)_k$$



$$\sigma_1 M \cap \sigma_2 M = (\sigma_1 \vee \sigma_2) M$$

$$\sigma_1 M \cap \sigma_2 M \cap \sigma_3 M = (\sigma_1 \vee \sigma_2 \vee \sigma_3) M$$

Counting braids of given length

Deligne (1972): The growth function of B_n^+ is the inverse of a polynomial.

$$G_n(t) = \sum_{k \geq 0} x_{n,k}\, t^k = \frac{1}{H_n(t)}$$

Proof: **Inclusion - exclusion principle**

$$x_{n,k} = \sum_{i=1}^{n-1} |(\sigma_i M)_k| - \sum_{i,j} |((\sigma_i \vee \sigma_j) M)_k| + \sum_{i,j,l} |((\sigma_i \vee \sigma_j \vee \sigma_l) M)_k|$$

Each term equals $x_{n,k-i}$ for $1 \leq i \leq \frac{n(n-1)}{2}$

Recurrence relation!

Q.E.D.

Counting braids of given length

Deligne (1972): The growth function of B_n^+ is the inverse of a polynomial.

$$G_n(t) = \sum_{k \geq 0} x_{n,k}\, t^k = \frac{1}{H_n(t)}$$

Proof: **Inclusion - exclusion principle**

$$x_{n,k} = \sum_{i=1}^{n-1} |(\sigma_i M)_k| - \sum_{i,j} |((\sigma_i \vee \sigma_j) M)_k| + \sum_{i,j,l} |((\sigma_i \vee \sigma_j \vee \sigma_l) M)_k|$$

Each term equals $x_{n,k-i}$ for $1 \leq i \leq \frac{n(n-1)}{2}$

Recurrence relation!

Q.E.D.

Counting braids of given length

Deligne (1972): The growth function of B_n^+ is the inverse of a polynomial.

$$G_n(t) = \sum_{k \geq 0} x_{n,k}\, t^k = \frac{1}{H_n(t)}$$

Actually:

$$H_n(t) = \sum_{S \subset \{\sigma_1, \ldots, \sigma_n\}} (-1)^{|S|}\, t^{||\vee S||}$$

Hard to compute!

Bronfman (2001): Recursive formula: $H_n(t) = \sum_{i=1}^{n} (-1)^{i+1} t^{\frac{i(i-1)}{2}} H_{n-i}(t)$

As $H_0(t)$= 1, one can compute $H_n(t)$, and then $x_{n,k}$. (in polynomial time)

We will use another method

Counting braids of given length

Example:

$$\#\left(\text{Lex-representatives of length } k \text{ starting with } \sigma_5\right) =$$

$$\#(\text{Braids of length } k \text{ starting with } \sigma_5 \text{ and not with } \sigma_1, \sigma_2, \sigma_3)$$

$$-\#(\text{Braids of length } k \text{ starting with } \sigma_5 \vee \sigma_4 \text{ and not with } \sigma_1, \sigma_2)$$

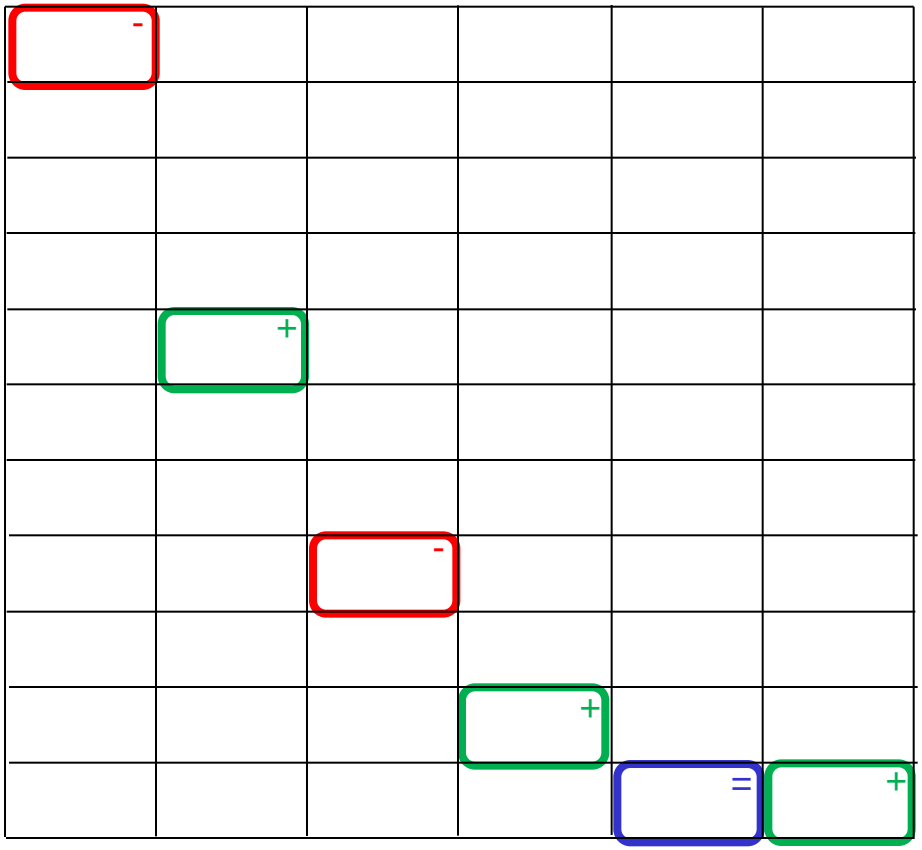
$$+\#(\text{Braids of length } k \text{ starting with } \sigma_5 \vee \sigma_4 \vee \sigma_3 \text{ and not with } \sigma_1)$$

$$-\#(\text{Braids of length } k \text{ starting with } \sigma_5 \vee \sigma_4 \vee \sigma_3 \vee \sigma_2)$$

$$+\#(\text{Braids of length } k \text{ starting with } \sigma_5 \vee \sigma_4 \vee \sigma_3 \vee \sigma_2 \vee \sigma_1)$$

Counting braids of given length

This yields an easy recursive formula:



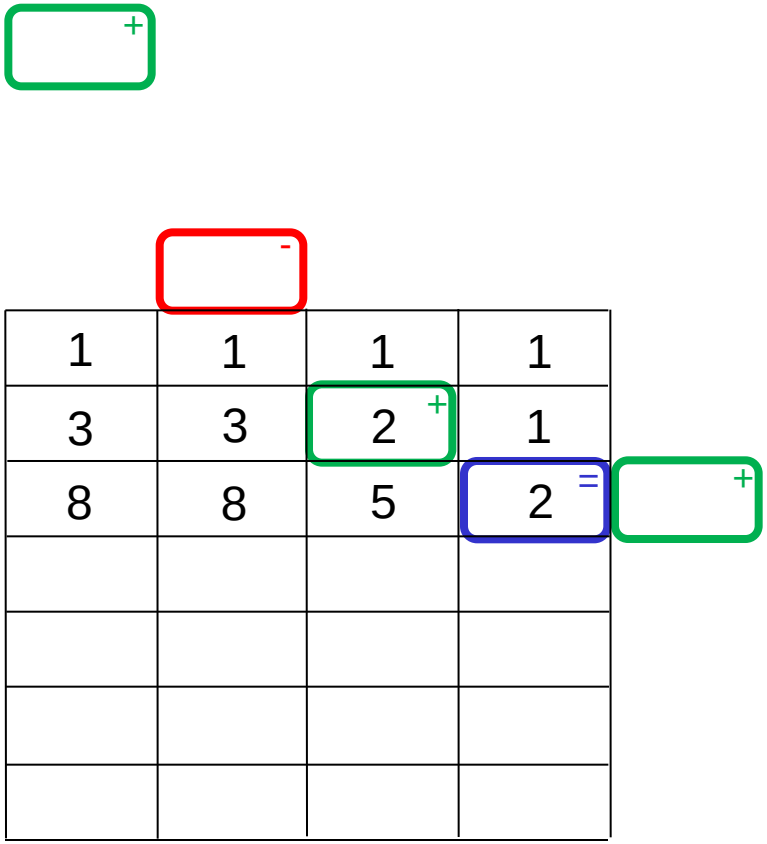
Counting braids of given length

Example, in B_4^+ :

1	1	1 +	1
3	3	2	1 =

Counting braids of given length

Example, in B_4^+ :



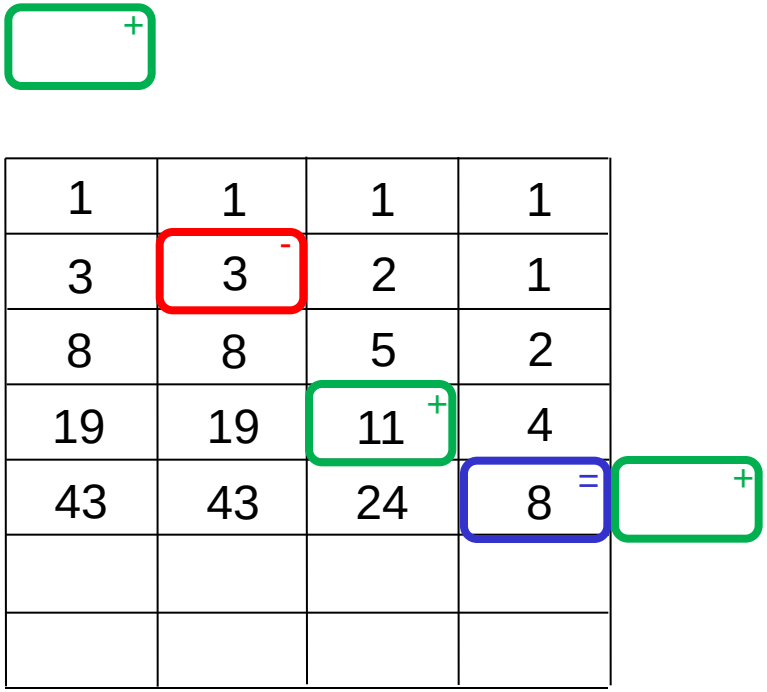
Counting braids of given length

Example, in B_4^+ :

1	1 -	1	1
3	3	2	1
8	8	5 +	2
19	19	11	4 =

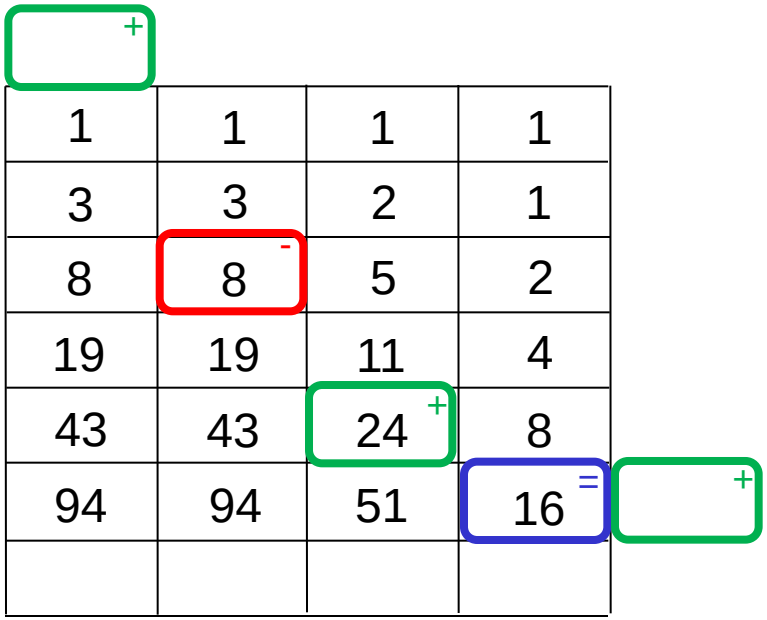
Counting braids of given length

Example, in B_4^+ :



Counting braids of given length

Example, in B_4^+ :



Counting braids of given length

Example, in B_4^+ :

1 +	1	1	1
3	3	2	1
8	8	5	2
19	19 -	11	4
43	43	24	8
94	94	51 +	16
202	202	108	33 =

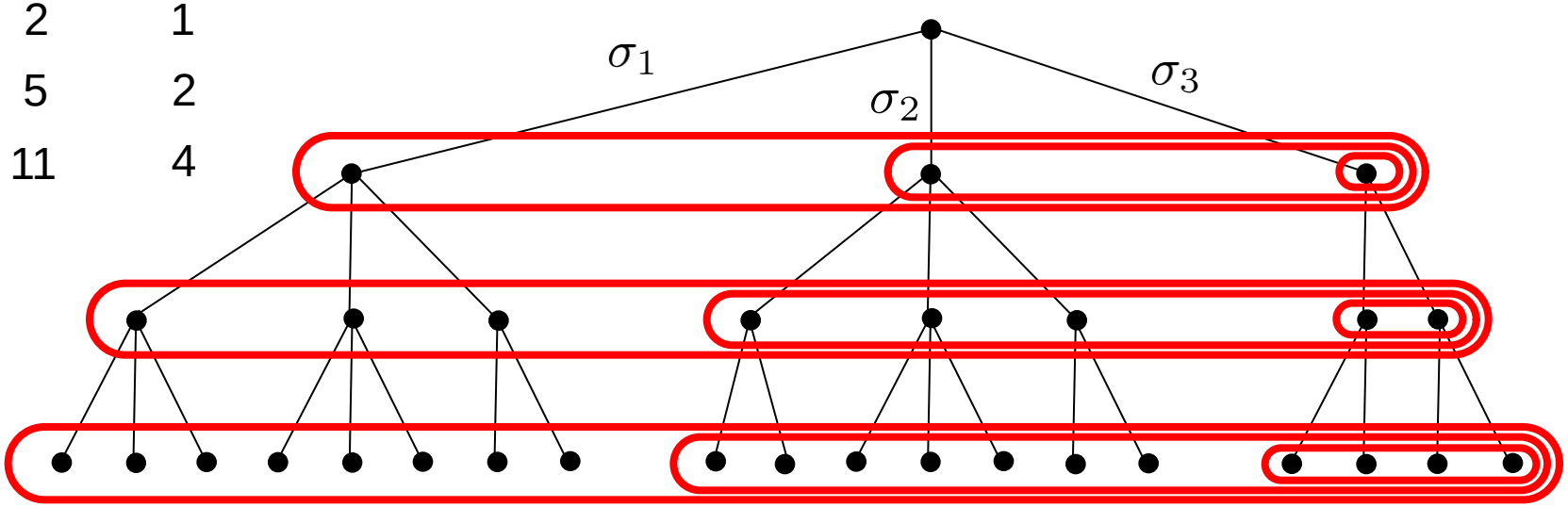
+

⋮

What are we computing?

Counting braids of given length

3	2	1
8	5	2
19	11	4



The first column of our table contains $x_{n,1}$, $x_{n,2}$, $x_{n,3}$,…

Computing k rows, we obtain $x_{n,k}$.

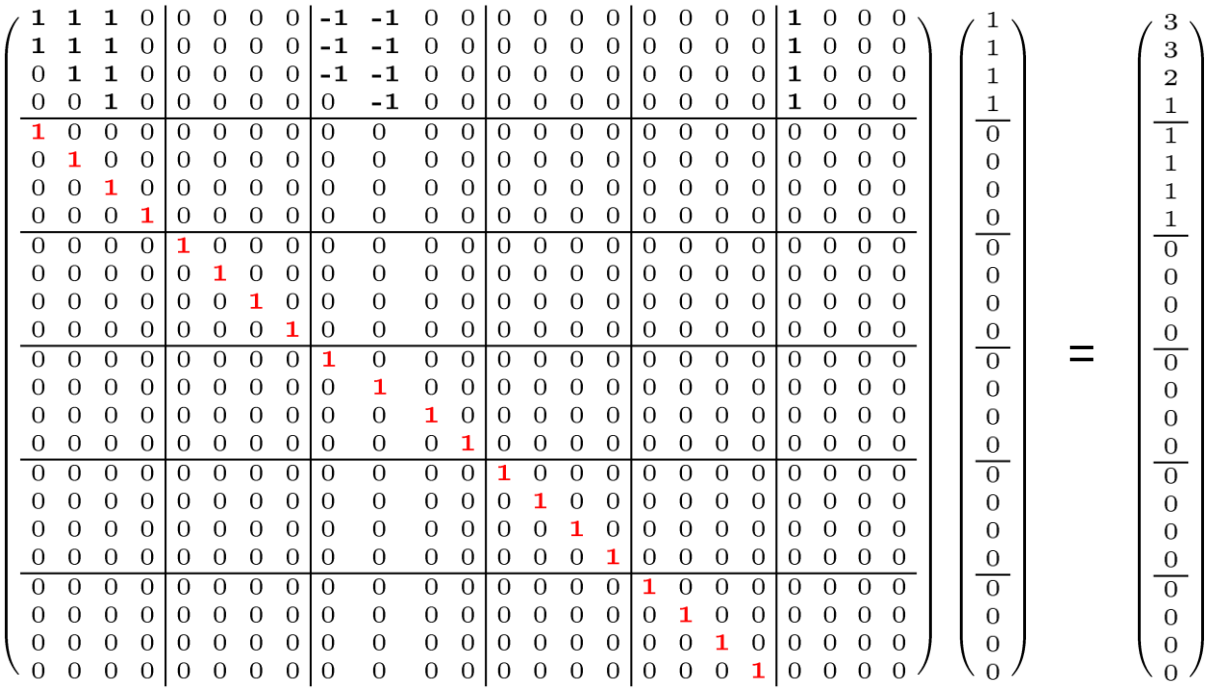
In time $O(k^2n^2)$

Consequence

1	1	1	1
3	3	2	1
8	8	5	2
19	19	11	4
43	43	24	8
94	94	51	16
202	202	108	33

A new line is obtained from the 6 previous ones

by a linear transformation



Multiplying from the left by (1 0 0 0 0 ⋯ 0), one gets $x_{n,k}$

Consequence

$$x_{n,k} = (1\ 0\ \cdots 0)\ A^k \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$G_n(t) = \sum_{k \geq 0} x_{n,k} t^k = (1\ 0\ \cdots 0) \left(\sum_{k \geq 0} A^k t^k \right) \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = (1\ 0\ \cdots 0)\,(I - At)^{-1} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$G_n(t) = \frac{1}{|I - At|}$$

Consequence

$$x_{n,k} = (1\ 0\ \cdots 0)\ A^k \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$G_n(t) = \sum_{k \geq 0} x_{n,k} t^k = (1\ 0\ \cdots 0) \left(\sum_{k \geq 0} A^k t^k \right) \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = (1\ 0\ \cdots 0)\,(I - At)^{-1} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$G_n(t) = \frac{1}{|I - At|}$$

Consequence

$$G_n(t) = \frac{1}{|I - At|}$$

After some easy computations

$$G_8(t) = \begin{vmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -t & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ t^3 & -t & 1 & -1 & 0 & 0 & 0 & 0 \\ -t^6 & t^3 & -t & 1 & -1 & 0 & 0 & 0 \\ t^{10} & -t^6 & t^3 & -t & 1 & -1 & 0 & 0 \\ -t^{15} & t^{10} & -t^6 & t^3 & -t & 1 & -1 & 0 \\ t^{21} & -t^{15} & t^{10} & -t^6 & t^3 & -t & 1 & -1 \\ -t^{28} & t^{21} & -t^{15} & t^{10} & -t^6 & t^3 & -t & 1 \end{vmatrix}^{-1}$$

Exponents are $\binom{i}{2}$

(Bronfman's recursive formula = expansion along the first column)

Other Artin-Tits monoids

Artin-Tits monoid of type B_n

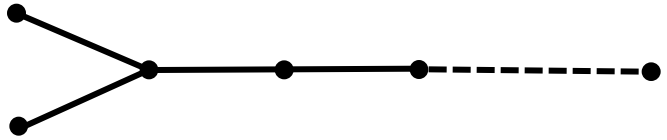
$$G_7(t) = \begin{vmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -t^2 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ t^4 & -t & 1 & -1 & 0 & 0 & 0 & 0 \\ -t^9 & t^3 & -t & 1 & -1 & 0 & 0 & 0 \\ t^{16} & -t^6 & t^3 & -t & 1 & -1 & 0 & 0 \\ -t^{25} & t^{10} & -t^6 & t^3 & -t & 1 & -1 & 0 \\ t^{36} & -t^{15} & t^{10} & -t^6 & t^3 & -t & 1 & -1 \\ -t^{49} & t^{21} & -t^{15} & t^{10} & -t^6 & t^3 & -t & 1 \end{vmatrix}^{-1}$$

Exponents are $\binom{i}{2}$

Exponents are i^2

Other Artin-Tits monoids

Artin-Tits monoid of type D_n



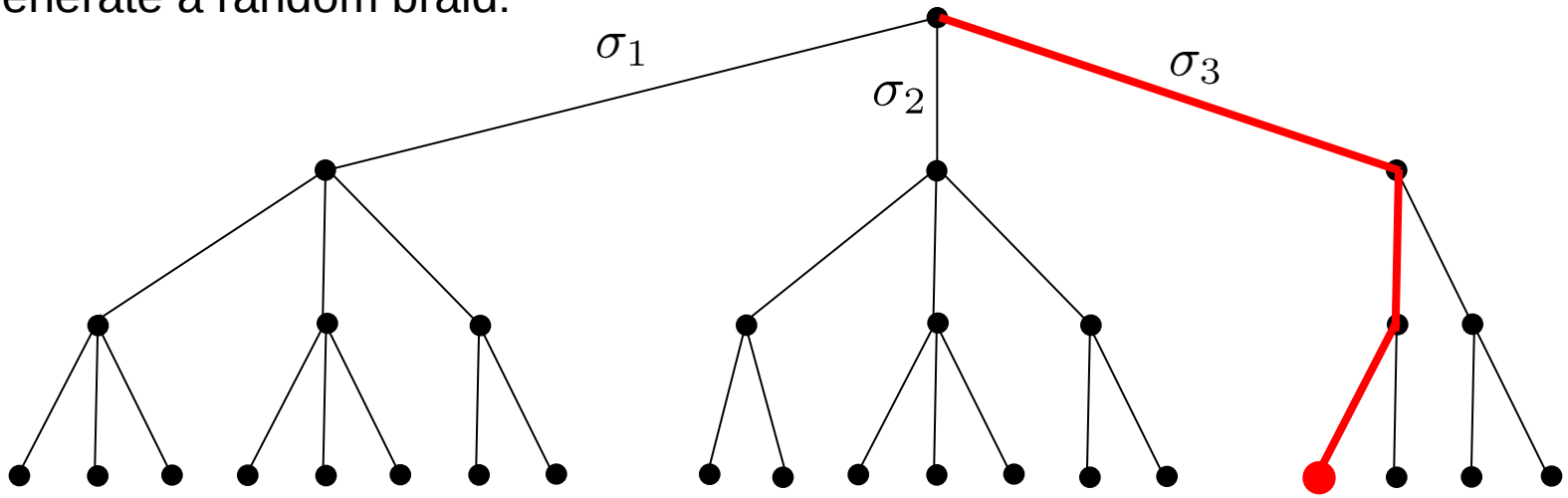
$$G_7(t) = \begin{vmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ t^2 & -2t & 1 & -1 & 0 & 0 & 0 & 0 \\ -t^6 & 2t^3 & -t & 1 & -1 & 0 & 0 & 0 \\ t^{12} & -2t^6 & t^3 & -t & 1 & -1 & 0 & 0 \\ -t^{20} & 2t^{10} & -t^6 & t^3 & -t & 1 & -1 & 0 \\ t^{30} & -2t^{15} & t^{10} & -t^6 & t^3 & -t & 1 & -1 \\ -t^{42} & 2t^{21} & -t^{15} & t^{10} & -t^6 & t^3 & -t & 1 \end{vmatrix}^{-1}$$

Exponents are $\binom{i}{2}$

Exponents are $i(i-1)$

Generating random braids

To generate a random braid:



1) Compute the number of leaves of the tree: 19 ✓

2) Choose a random number between 1 and 19: 16 ✓

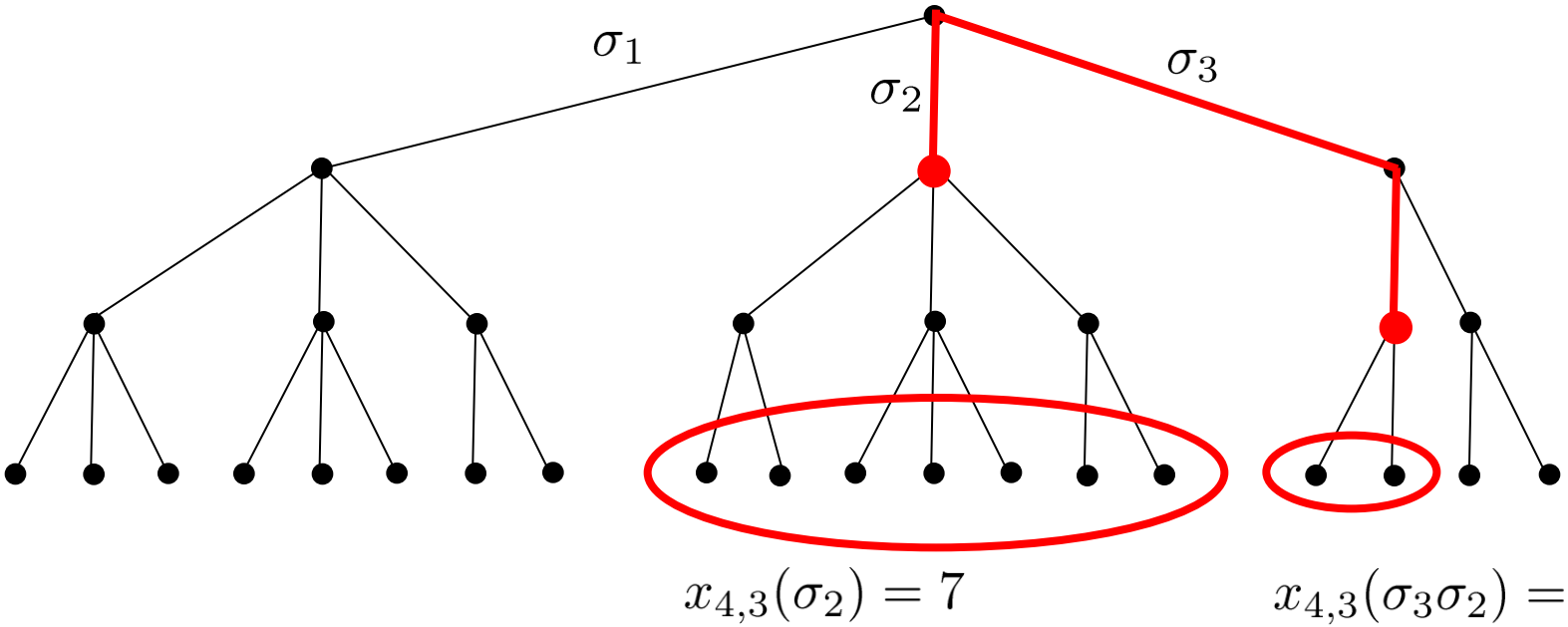
3) Find the braid corresponding to the 16th leaf: $\sigma_3\sigma_2\sigma_1$

Next

Finding the rth braid of length k

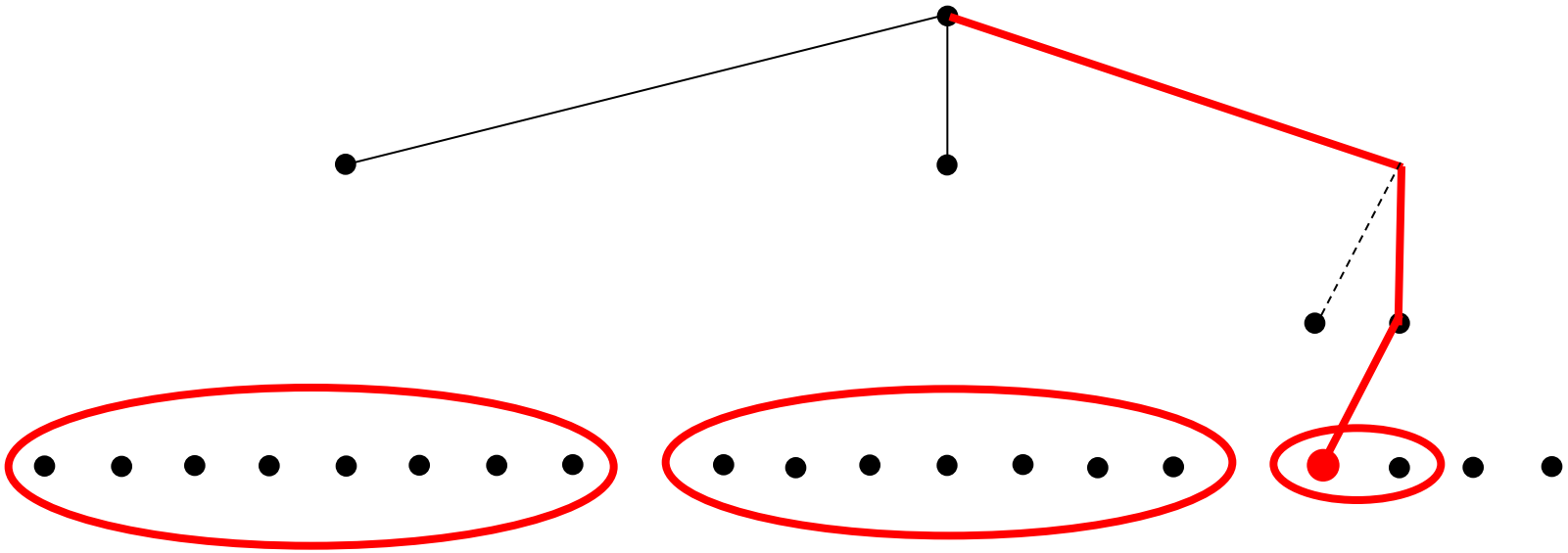
Consider the graph of height k as before.

Given a vertex w, suppose we can compute, in polynomial time, the number of leaves hanging from w: $x_{n,k}(w)$



Finding the rth braid of length k

We want to compute the 16th braid:



$x_{4,3}(\sigma_1) = 8 \qquad x_{4,3}(\sigma_2) = 7$ → The first letter is σ_3.

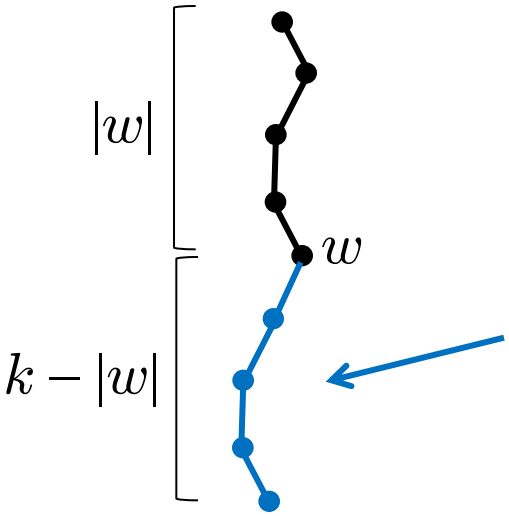
$x_{4,3}(\sigma_3\sigma_1) = 0 \qquad x_{4,3}(\sigma_3\sigma_2) = 2$ → The second letter is σ_2.

$x_{4,3}(\sigma_3\sigma_2\sigma_1) = 1$ → The third letter is σ_1.

Computing hanging leaves

Computing at most $(n-2)k$ times the hanging leaves of a vertex, one computes the r–th braid of length k.

How to compute the hanging leaves?



Lemma:

Can put any braid of length $k-|w|$, except those starting by some forbidden prefixes.

Computing hanging leaves

Example:

Forbidden prefixes for the word $w = \sigma_8\sigma_7\sigma_6\sigma_6\sigma_5\sigma_4\sigma_4\sigma_3\sigma_3\sigma_2\sigma_2\sigma_1\sigma_4\sigma_3\sigma_3$:

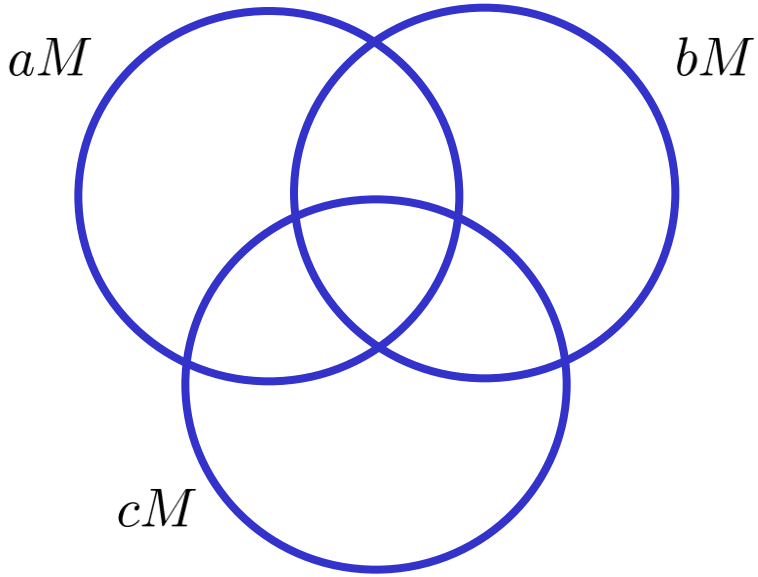
$$F(w) = \{\sigma_1, \quad \sigma_2\sigma_3, \quad \sigma_4\sigma_3, \quad \sigma_5\sigma_4\sigma_3\sigma_2\sigma_1, \quad \sigma_6, \quad \sigma_7\sigma_6\sigma_5\sigma_4\sigma_3\sigma_2\sigma_1, \quad \sigma_8\}.$$

$$x_{n,k}(w) = x_{n,k-|w|} - \left| \bigcup_{\alpha \in F(w)} \alpha M \right|$$

Know how to compute

Computing hanging leaves

By the inclusion-exclusion principle, we can compute $\left| \bigcup_{\alpha \in F(w)} \alpha M \right|$



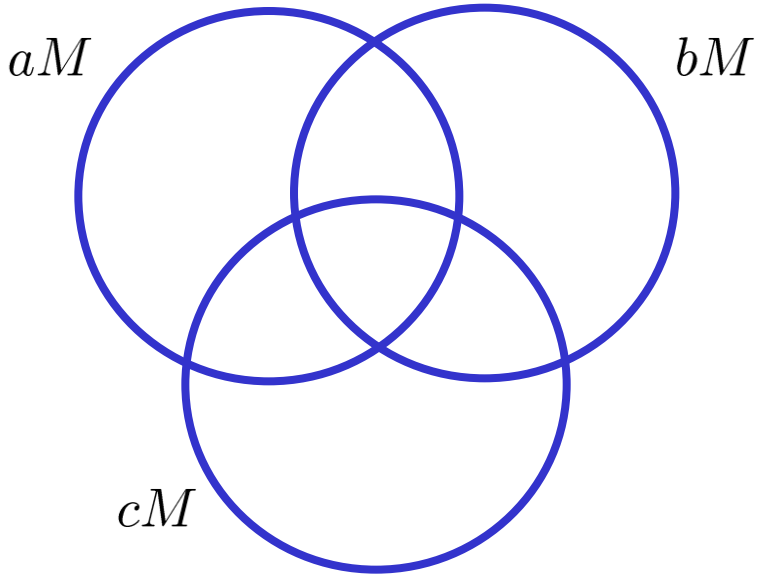
$$aM \cap bM = (a \vee b)M$$

$$aM \cap bM \cap cM = (a \vee b \vee c)M$$

$$\begin{aligned} |aM \cup bM \cup cM| = \quad & |aM| + |bM| + |cM| \\ & -|(a \vee b)M| - |(a \vee c)M| - |(b \vee c)M| \\ & +|(a \vee b \vee c)M| \end{aligned}$$

Computing hanging leaves

By the inclusion-exclusion principle, we can compute $\left| \bigcup_{\alpha \in F(w)} \alpha M \right|$



$$aM \cap bM = (a \vee b)M$$

$$aM \cap bM \cap cM = (a \vee b \vee c)M$$

$$\begin{aligned} |aM \cup bM \cup cM| = \quad & |aM| + |bM| + |cM| \\ & -|(a \vee b)M| - |(a \vee c)M| - |(b \vee c)M| \\ & +|(a \vee b \vee c)M| \end{aligned}$$

Computing hanging leaves

$$\begin{aligned}|aM \cup bM \cup cM| = \quad & |aM| + |bM| + |cM| \\ & -|(a \vee b)M| - |(a \vee c)M| - |(b \vee c)M| \\ & +|(a \vee b \vee c)M|\end{aligned}$$

Good news: Every summand is equal to $\pm\, x_{n,t}$, for some $t \leq k$.

These are the elements in the first column of our table.

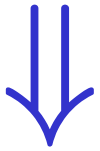
Bad news: There are exponentially many summands.

But we can collect all summands corresponding to elts. $a_1 \vee a_2 \vee \cdots \vee a_s$ with the same length.

Generating random positive braids

As a consequence:

One can compute $x_{n,k}(w)$ in ploynomial space and time with respect to n and k.



Theorem: (GM-Gebhardt, 2011)
There is a procedure to generate a random positive braid in B_n^+ of length k, whose time and space complexity is a polynomial in n and k.

Time $O(k^2 n^4)$?

Generating random positive braids

Time (in ms.) for computing a random positive braid of length k, on n strands.

n \ k	10	20	40	80	160	320	640	1280
5	0.01	0.06	0.28	1.22	4.80	22.40	107.40	566.40
10	0.01	0.10	0.76	5.04	26.00	119.60	513.60	2117.6
20	0.009	0.12	1.14	10.14	79.40	500.60	2911.2	16911.4
40	0.02	0.16	1.38	12.66	118.40	1211.20	10427.2	
80	0.03	0.20	1.48	21.38	272.60	2283	19696.6	

Linux system with an Intel E8400 64-bit CPU (core: 3\,GHz, FSB: 1333\,MHz) and a main memory bandwidth of 6.5\,GB/s (X38 chipset, dual channel DDR2 RAM, memory bus: 1066\,MHz) using a development version of Magma V.2.16.

Open problems

- Generate group elements instead of monoid elements.
- Use another generating set: permutation braids.
- Generalize to other Artin groups of finite type.

Thank you!