Experiments on growth series of braid groups

Jean Fromentin

Université du Littoral
Calais, France

Words

Let S be a finite generating set of a semigroup M.

Words

Let S be a finite generating set of a semigroup M.

- Definitions :
  - the empty word is denoted by ε ;
  - a word on the alphabet S is an S-word ;
  - the set of all S-words is denoted by S^* ;
  - for $u \in S^*$: $|u|$ its length and $\overline{u}$ the element of M it represents ;
  - two S-words u and v are equivalent, denoted by $u \equiv v$, if they represent the same element in M, *i.e.*, $\overline{u} = \overline{v}$.

Words

Let S be a finite generating set of a semigroup M.

- Definitions :
  - the empty word is denoted by ε ;
  - a word on the alphabet S is an S-word ;
  - the set of all S-words is denoted by S^* ;
  - for $u \in S^*$: $|u|$ its length and $\overline{u}$ the element of M it represents ;
  - two S-words u and v are equivalent, denoted by $u \equiv v$, if they represent the same element in M, *i.e.*, $\overline{u} = \overline{v}$.

- Examples : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.

Words

Let S be a finite generating set of a semigroup M.

- Definitions :
  - the empty word is denoted by ε ;
  - a word on the alphabet S is an S-word ;
  - the set of all S-words is denoted by S^* ;
  - for $u \in S^*$: $|u|$ its length and $\overline{u}$ the element of M it represents ;
  - two S-words u and v are equivalent, denoted by $u \equiv v$, if they represent the same element in M, *i.e.*, $\overline{u} = \overline{v}$.

- Examples : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.
  - $\overline{aa} = (1\ 2) \circ (1\ 2) = \mathbf{1}_{\mathfrak{S}_3} = \overline{\varepsilon}$ and so $aa \equiv \varepsilon$ (and also $bb \equiv \varepsilon$).

Words

Let S be a finite generating set of a semigroup M.

- Definitions :
  - the empty word is denoted by ε ;
  - a word on the alphabet S is an S-word ;
  - the set of all S-words is denoted by S^* ;
  - for $u \in S^*$: $|u|$ its length and $\overline{u}$ the element of M it represents ;
  - two S-words u and v are equivalent, denoted by $u \equiv v$, if they represent the same element in M, *i.e.*, $\overline{u} = \overline{v}$.

- Examples : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.
  - $\overline{aa} = (1\ 2) \circ (1\ 2) = \mathbf{1}_{\mathfrak{S}_3} = \overline{\varepsilon}$ and so $aa \equiv \varepsilon$ (and also $bb \equiv \varepsilon$).
  - $\overline{aba} = (1\ 2) \circ (2\ 3) \circ (1\ 2) = (1\ 2) \circ (1\ 3\ 2) = (1\ 3)$,

    $\overline{bab} = (2\ 3) \circ (1\ 2) \circ (2\ 3) = (2\ 3) \circ (1\ 2\ 3) = (1\ 3)$

    and so $aba \equiv bab$.

Spherical growth series

Let S be a finite generating set of a semigroup M.

- Definition : The S-length of an element $x \in M$, denoted $|x|_S$, is the length of a shortest S-word representing x.

Spherical growth series

Let S be a finite generating set of a semigroup M.

- Definition : The S-length of an element $x \in M$, denoted $|x|_S$, is the length of a shortest S-word representing x.
  - $|x|_S$ corresponds to the distance from $\mathbf{1}_M$ to x in $\mathrm{Cay}(M, S)$.

Spherical growth series

Let S be a finite generating set of a semigroup M.

- Definition : The S-length of an element $x \in M$, denoted $|x|_S$, is the length of a shortest S-word representing x.
  - $|x|_S$ corresponds to the distance from $\mathbf{1}_M$ to x in $\mathrm{Cay}(M, S)$.

- Definition : For any $\ell \in \mathbb{N}$, we put

$$s(M, S; \ell) = \mathrm{card}\left(\{x \in M \text{ s.t. } |x|_S = \ell\}\right).$$

Spherical growth series

Let S be a finite generating set of a semigroup M.

- Definition : The S-length of an element $x \in M$, denoted $|x|_S$, is the length of a shortest S-word representing x.

  - $|x|_S$ corresponds to the distance from $\mathbf{1}_M$ to x in $\mathrm{Cay}(M, S)$.

- Definition : For any $\ell \in \mathbb{N}$, we put

$$s(M, S; \ell) = \mathrm{card}\left(\{x \in M \text{ s.t. } |x|_S = \ell\}\right).$$

The spherical growth series of M w.r.t. S is

$$\mathcal{S}(M, S) = \sum_{x \in M} t^{|x|_S} = \sum_{\ell \in \mathbb{N}} s(M, S; \ell)\, t^{\ell}.$$

Spherical growth series

Let S be a finite generating set of a semigroup M.

- **Definition** : The S-length of an element $x \in M$, denoted $|x|_S$, is the length of a shortest S-word representing x.

  - $|x|_S$ corresponds to the distance from $\mathbf{1}_M$ to x in $\mathrm{Cay}(M, S)$.

- **Definition** : For any $\ell \in \mathbb{N}$, we put

$$s(M, S; \ell) = \mathrm{card}\left(\{x \in M \text{ s.t. } |x|_S = \ell\}\right).$$

  The spherical growth series of M w.r.t. S is

$$\mathcal{S}(M, S) = \sum_{x \in M} t^{|x|_S} = \sum_{\ell \in \mathbb{N}} s(M, S; \ell)\, t^{\ell}.$$

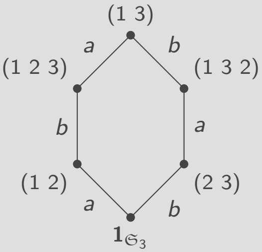
  - $\mathrm{card}\,(M) = \mathcal{S}(M, S)|_{t=1}$.

Spherical growth series – Example

- Example : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.

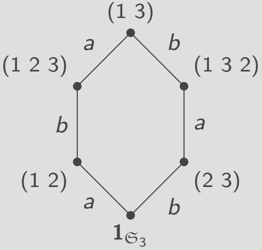
Spherical growth series – Example

- Example : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.



Spherical growth series – Example

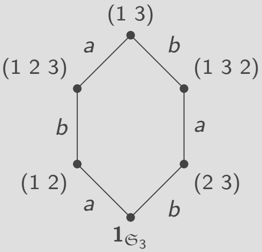
- Example : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.



$|\mathbf{1}_{\mathfrak{S}_3}|_S = 0$

Spherical growth series – Example

- Example : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.

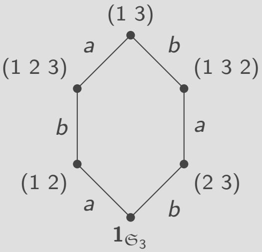


$$|\mathbf{1}_{\mathfrak{S}_3}|_S = 0$$

$$|(1\ 2)|_S = |(2\ 3)|_S = 1$$

Spherical growth series – Example

- Example : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.



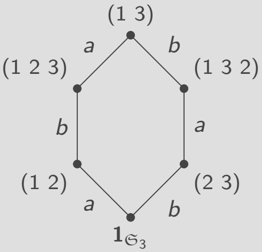
$$|\mathbf{1}_{\mathfrak{S}_3}|_S = 0$$

$$|(1\ 2)|_S = |(2\ 3)|_S = 1$$

$$|(1\ 2\ 3)|_S = |(1\ 3\ 2)|_S = 2$$

Spherical growth series – Example

- Example : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.



$$|\mathbf{1}_{\mathfrak{S}_3}|_S = 0$$

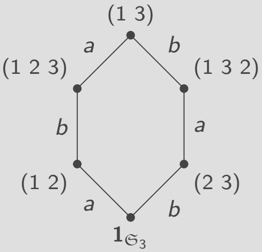
$$|(1\ 2)|_S = |(2\ 3)|_S = 1$$

$$|(1\ 2\ 3)|_S = |(1\ 3\ 2)|_S = 2$$

$$|(1\ 3)|_S = 3$$

Spherical growth series – Example

- Example : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.



$$|\mathbf{1}_{\mathfrak{S}_3}|_S = 0$$
$$|(1\ 2)|_S = |(2\ 3)|_S = 1$$
$$|(1\ 2\ 3)|_S = |(1\ 3\ 2)|_S = 2$$
$$|(1\ 3)|_S = 3$$

- Example :

$$s(\mathfrak{S}_3, S, \ell) = \begin{cases} 1 & \text{if } \ell = 0 \text{ or } \ell = 3, \\ 2 & \text{if } \ell = 1 \text{ or } \ell = 2, \\ 0 & \text{if } \ell \geqslant 4. \end{cases}$$

and so $\mathcal{S}(\mathfrak{S}_3, S) = 1 + 2t + 2t^2 + t^3$.

Geodesic growth series

Let S be a finite generating set of a semigroup M.

- Definition : An S-word u is said to be geodesic if $|u| = |\overline{u}|_S$ holds.

Geodesic growth series

Let S be a finite generating set of a semigroup M.

- Definition : An S-word u is said to be geodesic if $|u| = |\overline{u}|_S$ holds.
  - Geodesic S-words correspond to geodesic paths on $\mathrm{Cay}(M, S)$.

Geodesic growth series

Let S be a finite generating set of a semigroup M.

- **Definition** : An S-word u is said to be geodesic if $|u| = |\overline{u}|_S$ holds.
  - Geodesic S-words correspond to geodesic paths on $\mathrm{Cay}(M, S)$.

- **Definition** : For any $\ell \in \mathbb{N}$, we put

$$g(M, S; \ell) = \mathrm{card}\left(\{u \in S^* \text{ s.t. } |u| = |\overline{u}|_S = \ell\}\right).$$

Geodesic growth series

Let S be a finite generating set of a semigroup M.

- **Definition** : An S-word u is said to be geodesic if $|u| = |\overline{u}|_S$ holds.
  - Geodesic S-words correspond to geodesic paths on $\mathrm{Cay}(M, S)$.

- **Definition** : For any $\ell \in \mathbb{N}$, we put

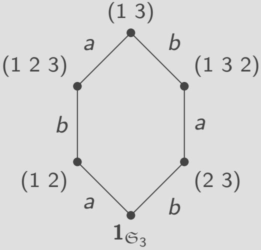
$$g(M, S; \ell) = \mathrm{card}\left(\{u \in S^* \text{ s.t. } |u| = |\overline{u}|_S = \ell\}\right).$$

The geodesic growth series of M w.r.t. S is

$$\mathcal{G}(M, S) = \sum_{\substack{u \in S^* \\ |u| = |\overline{u}|_S}} t^{|u|} = \sum_{\ell \in \mathbb{N}} g(M, S; \ell)\, t^{\ell}.$$

Geodesic growth series – Example

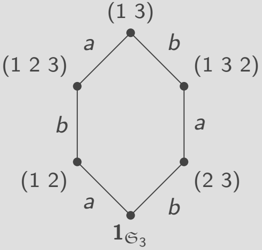
- Example : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.



Geodesic S-words are

Geodesic growth series – Example

- Example : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.

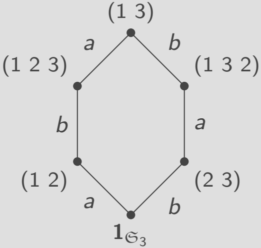


Geodesic S-words are

- ε

Geodesic growth series – Example

- Example : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.

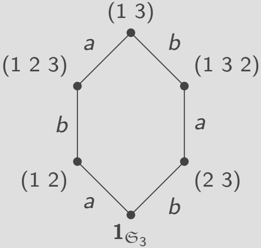


Geodesic S-words are

- ε
- a and b

Geodesic growth series – Example

- Example : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.

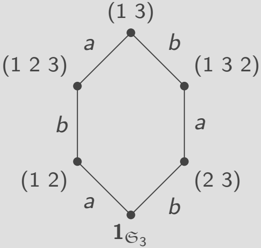


Geodesic S-words are

- ε
- a and b
- ab and ba

Geodesic growth series – Example

- Example : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.

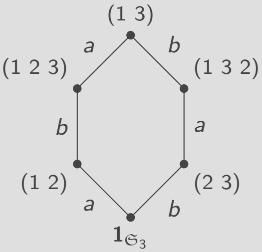


Geodesic S-words are

- ε
- a and b
- ab and ba
- aba and bab

Geodesic growth series – Example

- Example : $M = (\mathfrak{S}_3, \circ)$, $a = (1\ 2)$, $b = (2\ 3)$ and $S = \{a, b\}$.



Geodesic S-words are

- ε
- a and b
- ab and ba
- aba and bab

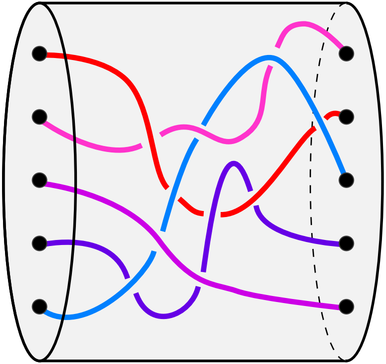
- Example :

$$g(\mathfrak{S}_3, S; \ell) = \begin{cases} 1 & \text{if } \ell = 0 \\ 2 & \text{if } \ell = 1 \text{ or } \ell = 2 \text{ or } \ell = 3, \\ 0 & \text{if } \ell \geqslant 4. \end{cases}$$

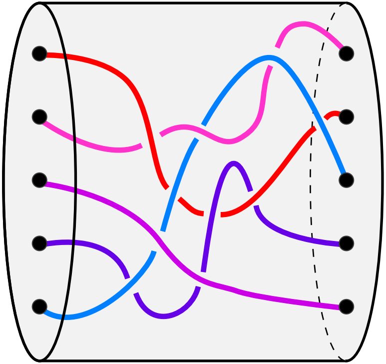
and so $\mathcal{G}(\mathfrak{S}_3, S) = 1 + 2t + 2t^2 + 2t^3$.

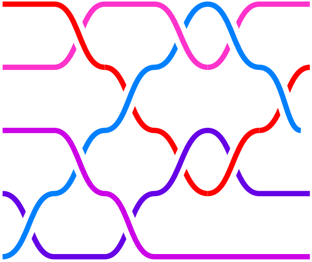
Braid words

Braid words

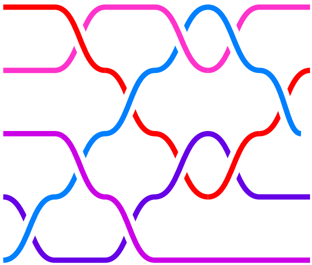


Braid words

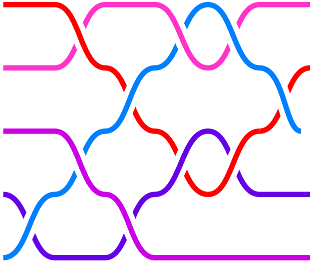


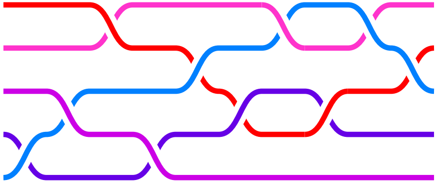


Braid words

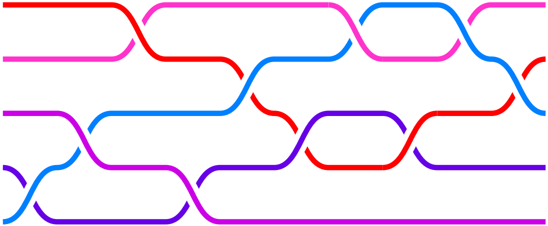


Braid words

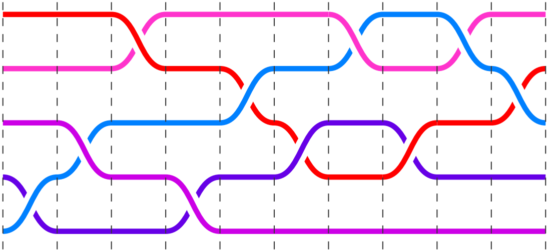




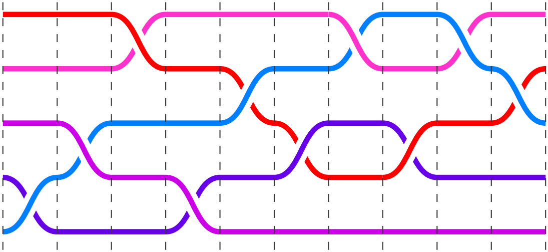
Braid words



Braid words

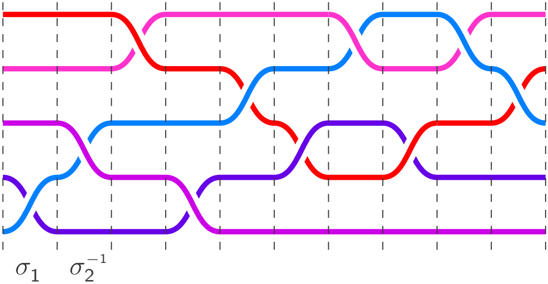


Braid words

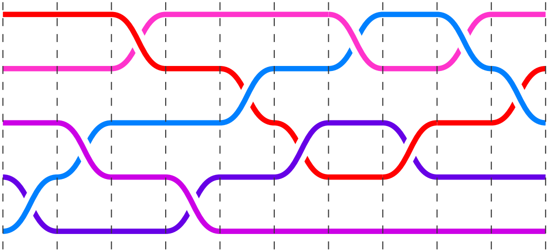


σ_1

Braid words

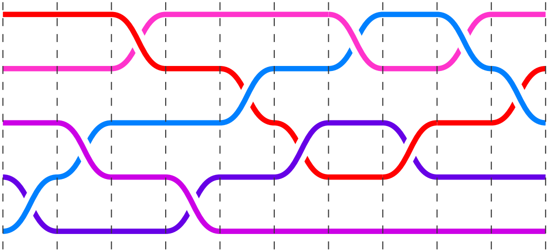


Braid words



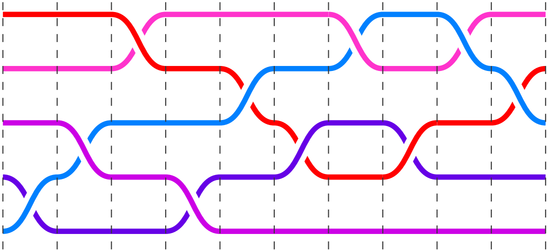
$\sigma_1 \quad \sigma_2^{-1} \quad \sigma_4^{-1}$

Braid words



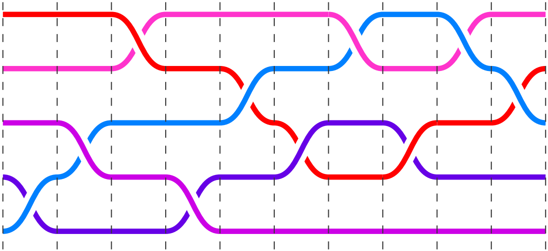
$\sigma_1 \quad \sigma_2^{-1} \quad \sigma_4^{-1} \quad \sigma_1^{-1}$

Braid words



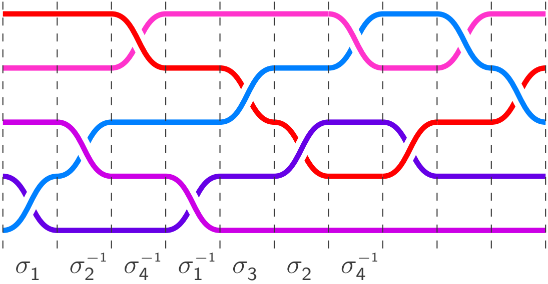
$\sigma_1 \quad \sigma_2^{-1} \quad \sigma_4^{-1} \quad \sigma_1^{-1} \quad \sigma_3$

Braid words

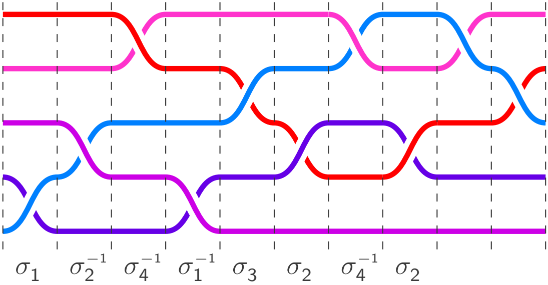


$\sigma_1 \quad \sigma_2^{-1} \quad \sigma_4^{-1} \quad \sigma_1^{-1} \quad \sigma_3 \quad \sigma_2$

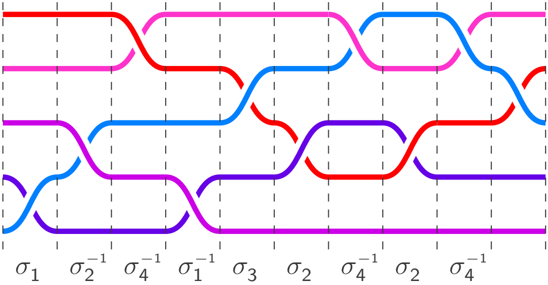
Braid words



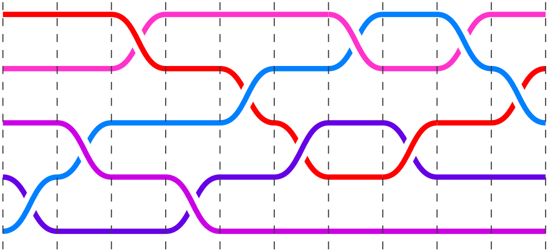
Braid words


$\sigma_1 \quad \sigma_2^{-1} \quad \sigma_4^{-1} \quad \sigma_1^{-1} \quad \sigma_3 \quad \sigma_2 \quad \sigma_4^{-1} \quad \sigma_2$

Braid words

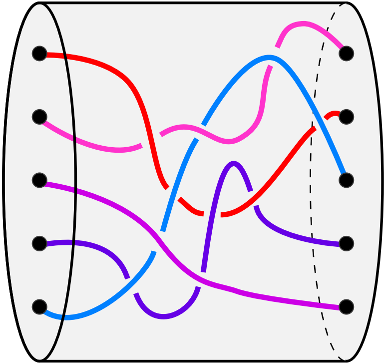

$\sigma_1 \quad \sigma_2^{-1} \quad \sigma_4^{-1} \quad \sigma_1^{-1} \quad \sigma_3 \quad \sigma_2 \quad \sigma_4^{-1} \quad \sigma_2 \quad \sigma_4^{-1}$

Braid words



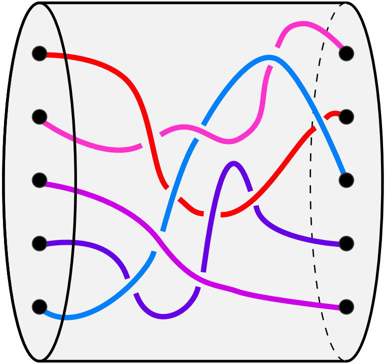
$$\sigma_1 \; \sigma_2^{-1} \; \sigma_4^{-1} \; \sigma_1^{-1} \; \sigma_3 \; \sigma_2 \; \sigma_4^{-1} \; \sigma_2 \; \sigma_4^{-1} \; \sigma_3^{-1}$$

Braid words



$$\longrightarrow \quad \sigma_1\sigma_2^{-1}\sigma_4^{-1}\sigma_1^{-1}\sigma_3\sigma_2\sigma_4^{-1}\sigma_2\sigma_4^{-1}\sigma_3^{-1}$$

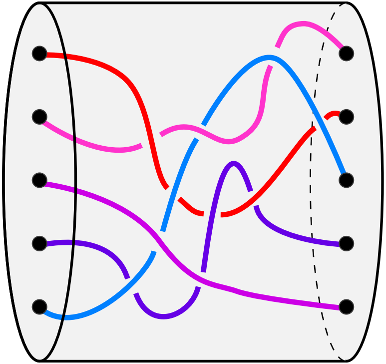
Braid words



$\longrightarrow \quad \sigma_1\sigma_2^{-1}\sigma_4^{-1}\sigma_1^{-1}\sigma_3\sigma_2\sigma_4^{-1}\sigma_2\sigma_4^{-1}\sigma_3^{-1}$

$\longrightarrow \quad \sigma_4^{-1}\sigma_1\sigma_2^{-1}\sigma_3\sigma_1^{-1}\sigma_4^{-1}\sigma_2\sigma_4^{-1}\sigma_2\sigma_3^{-1}$

Braid words



$$\longrightarrow \quad \sigma_1\sigma_2^{-1}\sigma_4^{-1}\sigma_1^{-1}\sigma_3\sigma_2\sigma_4^{-1}\sigma_2\sigma_4^{-1}\sigma_3^{-1}$$

$$\longrightarrow \quad \sigma_4^{-1}\sigma_1\sigma_2^{-1}\sigma_3\sigma_1^{-1}\sigma_4^{-1}\sigma_2\sigma_4^{-1}\sigma_2\sigma_3^{-1}$$

$$\longrightarrow \quad \cdots$$

Artin's presentation of braid groups

- Theorem (E. Artin 1925) : The braid group B_n is presented by

$$\left\langle \sigma_1, \ldots, \sigma_{n-1} \;\middle|\; \begin{array}{rcll} \sigma_i\sigma_j & \equiv & \sigma_j\sigma_i & \text{for } |i-j| \geqslant 2 \\ \sigma_i\sigma_j\sigma_i & \equiv & \sigma_j\sigma_i\sigma_j & \text{for } |i-j| = 1 \end{array} \right\rangle \tag{1}$$

Artin's presentation of braid groups

- Theorem (E. Artin 1925) : The braid group B_n is presented by

$$\left\langle \sigma_1, \ldots, \sigma_{n-1} \;\middle|\; \begin{array}{rcll} \sigma_i\sigma_j & \equiv & \sigma_j\sigma_i & \text{for } |i-j| \geqslant 2 \\ \sigma_i\sigma_j\sigma_i & \equiv & \sigma_j\sigma_i\sigma_j & \text{for } |i-j| = 1 \end{array} \right\rangle \tag{1}$$

- Definition : For all $n \geqslant 2$, we denote by Σ_n^+ the set $\{\sigma_1, \ldots, \sigma_{n-1}\}$ and by Σ_n the set $\Sigma_n^+ \sqcup (\Sigma_n^+)^{-1} = \{\sigma_1^{\pm 1}, \ldots, \sigma_{n-1}^{\pm 1}\}$.

Artin's presentation of braid groups

- **Theorem** (E. Artin 1925) : The braid group B_n is presented by

$$\left\langle \sigma_1, \ldots, \sigma_{n-1} \;\middle|\; \begin{array}{rcll} \sigma_i\sigma_j & \equiv & \sigma_j\sigma_i & \text{for } |i-j| \geqslant 2 \\ \sigma_i\sigma_j\sigma_i & \equiv & \sigma_j\sigma_i\sigma_j & \text{for } |i-j| = 1 \end{array} \right\rangle \tag{1}$$

- **Definition** : For all $n \geqslant 2$, we denote by Σ_n^+ the set $\{\sigma_1, \ldots, \sigma_{n-1}\}$ and by Σ_n the set $\Sigma_n^+ \sqcup (\Sigma_n^+)^{-1} = \{\sigma_1^{\pm 1}, \ldots, \sigma_{n-1}^{\pm 1}\}$.

  - As a group B_n is generated by Σ_n^+.

Artin's presentation of braid groups

- **Theorem** (E. Artin 1925) : The braid group B_n is presented by

$$\left\langle \sigma_1, \ldots, \sigma_{n-1} \;\middle|\; \begin{array}{rcll} \sigma_i\sigma_j & \equiv & \sigma_j\sigma_i & \text{for } |i-j| \geqslant 2 \\ \sigma_i\sigma_j\sigma_i & \equiv & \sigma_j\sigma_i\sigma_j & \text{for } |i-j| = 1 \end{array} \right\rangle \tag{1}$$

- **Definition** : For all $n \geqslant 2$, we denote by Σ_n^+ the set $\{\sigma_1, \ldots, \sigma_{n-1}\}$ and by Σ_n the set $\Sigma_n^+ \sqcup (\Sigma_n^+)^{-1} = \{\sigma_1^{\pm 1}, \ldots, \sigma_{n-1}^{\pm 1}\}$.

  - As a group B_n is generated by Σ_n^+.
    - But, as a semigroup, it is generated by Σ_n with relations (1) together with $\sigma_i\,\sigma_i^{-1} \equiv \sigma_i\,\sigma_i^{-1} \equiv \varepsilon$ for all $i \in \{1, \ldots, n\}$.

Artin's presentation of braid groups

- **Theorem** (E. Artin 1925) : The braid group B_n is presented by

$$\left\langle \sigma_1, \ldots, \sigma_{n-1} \;\middle|\; \begin{array}{rcll} \sigma_i\sigma_j & \equiv & \sigma_j\sigma_i & \text{for } |i-j| \geqslant 2 \\ \sigma_i\sigma_j\sigma_i & \equiv & \sigma_j\sigma_i\sigma_j & \text{for } |i-j| = 1 \end{array} \right\rangle \tag{1}$$

- **Definition** : For all $n \geqslant 2$, we denote by Σ_n^+ the set $\{\sigma_1, \ldots, \sigma_{n-1}\}$ and by Σ_n the set $\Sigma_n^+ \sqcup (\Sigma_n^+)^{-1} = \{\sigma_1^{\pm 1}, \ldots, \sigma_{n-1}^{\pm 1}\}$.

  - As a group B_n is generated by Σ_n^+.
    - But, as a semigroup, it is generated by Σ_n with relations (1) together with $\sigma_i\, \sigma_i^{-1} \equiv \sigma_i\, \sigma_i^{-1} \equiv \varepsilon$ for all $i \in \{1, \ldots, n\}$.

- **Definition** : B_n^+ is the submonoid of B_n generated by Σ_n^+.

Artin's presentation of braid groups

- **Theorem** (E. Artin 1925) : The braid group B_n is presented by

$$\left\langle \sigma_1, \ldots, \sigma_{n-1} \;\middle|\; \begin{array}{rcll} \sigma_i\sigma_j & \equiv & \sigma_j\sigma_i & \text{for } |i-j| \geqslant 2 \\ \sigma_i\sigma_j\sigma_i & \equiv & \sigma_j\sigma_i\sigma_j & \text{for } |i-j| = 1 \end{array} \right\rangle \tag{1}$$

- **Definition** : For all $n \geqslant 2$, we denote by Σ_n^+ the set $\{\sigma_1, \ldots, \sigma_{n-1}\}$ and by Σ_n the set $\Sigma_n^+ \sqcup (\Sigma_n^+)^{-1} = \{\sigma_1^{\pm 1}, \ldots, \sigma_{n-1}^{\pm 1}\}$.

  - As a group B_n is generated by Σ_n^+.
    - But, as a semigroup, it is generated by Σ_n with relations (1) together with $\sigma_i\,\sigma_i^{-1} \equiv \sigma_i\,\sigma_i^{-1} \equiv \varepsilon$ for all $i \in \{1, \ldots, n\}$.

- **Definition** : B_n^+ is the submonoid of B_n generated by Σ_n^+.

  - B_n is the group of fractions of B_n^+.

Artin's presentation of braid groups

- **Theorem** (E. Artin 1925) : The braid group B_n is presented by

$$\left\langle \sigma_1, \ldots, \sigma_{n-1} \;\middle|\; \begin{array}{rcll} \sigma_i\sigma_j & \equiv & \sigma_j\sigma_i & \text{for } |i-j| \geqslant 2 \\ \sigma_i\sigma_j\sigma_i & \equiv & \sigma_j\sigma_i\sigma_j & \text{for } |i-j| = 1 \end{array} \right\rangle \tag{1}$$

- **Definition** : For all $n \geqslant 2$, we denote by Σ_n^+ the set $\{\sigma_1, \ldots, \sigma_{n-1}\}$ and by Σ_n the set $\Sigma_n^+ \sqcup (\Sigma_n^+)^{-1} = \{\sigma_1^{\pm 1}, \ldots, \sigma_{n-1}^{\pm 1}\}$.

  - As a group B_n is generated by Σ_n^+.
    - But, as a semigroup, it is generated by Σ_n with relations (1) together with $\sigma_i\,\sigma_i^{-1} \equiv \sigma_i\,\sigma_i^{-1} \equiv \varepsilon$ for all $i \in \{1, \ldots, n\}$.

- **Definition** : B_n^+ is the submonoid of B_n generated by Σ_n^+.

  - B_n is the group of fractions of B_n^+.
    - As a semigroup B_n^+ is presented by (1).

Artin growth series of braid groups

- Theorem (L. Sabalka 2004) :

$$\mathcal{G}(B_3, \Sigma_3) = \frac{t^4 + 3t^3 + t + 1}{(t^2 + 2t - 1)(t^2 + t - 1)}.$$

Artin growth series of braid groups

- Theorem (L. Sabalka 2004) :

$$\mathcal{G}(B_3, \Sigma_3) = \frac{t^4 + 3t^3 + t + 1}{(t^2 + 2t - 1)(t^2 + t - 1)}.$$

  - ▶ Construction of an explicit deterministic finite states automaton.

Artin growth series of braid groups

- Theorem (L. Sabalka 2004) :

$$\mathcal{G}(B_3, \Sigma_3) = \frac{t^4 + 3t^3 + t + 1}{(t^2 + 2t - 1)(t^2 + t - 1)}.$$

  - Construction of an explicit deterministic finite states automaton.

- Using Knuth-Bendix methods by D. Holt, D.B.A Epstein and S. Rees, he also obtains

$$\mathcal{S}(B_3, \Sigma_3) = \frac{(t+1)(2t^3 - t^2 + t - 1)}{(t-1)(2t-1)(t^2 + t - 1)}.$$

Artin growth series of braid groups

- Theorem (L. Sabalka 2004) :
$$\mathcal{G}(B_3, \Sigma_3) = \frac{t^4 + 3t^3 + t + 1}{(t^2 + 2t - 1)(t^2 + t - 1)}.$$
  - Construction of an explicit deterministic finite states automaton.

- Using Knuth-Bendix methods by D. Holt, D.B.A Epstein and S. Rees, he also obtains
$$\mathcal{S}(B_3, \Sigma_3) = \frac{(t + 1)(2t^3 - t^2 + t - 1)}{(t - 1)(2t - 1)(t^2 + t - 1)}.$$

- In her PhD, M. Albenque computes the first 13 terms of $s(B_4, \Sigma_4; \ell)$.

Artin growth series of braid groups

- Theorem (L. Sabalka 2004) :

$$\mathcal{G}(B_3, \Sigma_3) = \frac{t^4 + 3t^3 + t + 1}{(t^2 + 2t - 1)(t^2 + t - 1)}.$$

  - Construction of an explicit deterministic finite states automaton.

- Using Knuth-Bendix methods by D. Holt, D.B.A Epstein and S. Rees, he also obtains

$$\mathcal{S}(B_3, \Sigma_3) = \frac{(t+1)(2t^3 - t^2 + t - 1)}{(t-1)(2t-1)(t^2 + t - 1)}.$$

- In her PhD, M. Albenque computes the first 13 terms of $s(B_4, \Sigma_4; \ell)$.

  - She obtains $s(B_4, \Sigma_4; 12) = 2\,975\,728$.

Artin growth series of braid groups

- Theorem (L. Sabalka 2004) :

$$\mathcal{G}(B_3, \Sigma_3) = \frac{t^4 + 3t^3 + t + 1}{(t^2 + 2t - 1)(t^2 + t - 1)}.$$

  - Construction of an explicit deterministic finite states automaton.

- Using Knuth-Bendix methods by D. Holt, D.B.A Epstein and S. Rees, he also obtains

$$\mathcal{S}(B_3, \Sigma_3) = \frac{(t+1)(2t^3 - t^2 + t - 1)}{(t-1)(2t-1)(t^2 + t - 1)}.$$

- In her PhD, M. Albenque computes the first 13 terms of $s(B_4, \Sigma_4; \ell)$.

  - She obtains $s(B_4, \Sigma_4; 12) = 2\,975\,728$.

- Open question : Do $\mathcal{G}(B_n, \Sigma_n)$ and $\mathcal{S}(B_n, \Sigma_n)$ are rationals for $n \geqslant 4$?

Artin growth series of braid semigroups

- Fact : As every Σ_n^+-word is geodesic, $\mathcal{G}(B_n^+, \Sigma_n^+)$ is irrelevant.

Artin growth series of braid semigroups

- Fact : As every Σ_n^+-word is geodesic, $\mathcal{G}(B_n^+, \Sigma_n^+)$ is irrelevant.

- Theorem (A. Bronfman 2001) : We have

$$\mathcal{S}(B_n^+, \Sigma_n^+) = \frac{1}{P_n(t)}$$

where $P_n(t)$ is given by

$$P_n(t) = \sum_{i=1}^{n} (-1)^{i+1} t^{\frac{i(i-1)}{2}} P_{n-i}(t)$$

with $P_0(t) = P_1(t) = 1$.

Artin growth series of braid semigroups

- Fact : As every Σ_n^+-word is geodesic, $\mathcal{G}(B_n^+, \Sigma_n^+)$ is irrelevant.

- Theorem (A. Bronfman 2001) : We have

$$\mathcal{S}(B_n^+, \Sigma_n^+) = \frac{1}{P_n(t)}$$

where $P_n(t)$ is given by

$$P_n(t) = \sum_{i=1}^{n} (-1)^{i+1} t^{\frac{i(i-1)}{2}} P_{n-i}(t)$$

with $P_0(t) = P_1(t) = 1$.

  - Generalized to positive braid semigroups of types B and D by M. Albenque and P. Nadeau in 2009 using Viennot's heap of pieces.

Artin growth series of braid semigroups

- Fact : As every Σ_n^+-word is geodesic, $\mathcal{G}(B_n^+, \Sigma_n^+)$ is irrelevant.

- Theorem (A. Bronfman 2001) : We have
$$\mathcal{S}(B_n^+, \Sigma_n^+) = \frac{1}{P_n(t)}$$
where $P_n(t)$ is given by
$$P_n(t) = \sum_{i=1}^{n} (-1)^{i+1} t^{\frac{i(i-1)}{2}} P_{n-i}(t)$$
with $P_0(t) = P_1(t) = 1$.

  - Generalized to positive braid semigroups of types B and D by M. Albenque and P. Nadeau in 2009 using Viennot's heap of pieces.
  - And for all spherical Artin–Tits semigroups by R. Flores and J. González-Meneses in 2018.

Dual generators

- Definition : For $1 \leqslant p \leqslant q$ we put

$$a_{p,q} = \sigma_p \dots \sigma_{q-2}\ \sigma_{q-1}\ \sigma_{q-2}^{-1} \dots \sigma_p^{-1}.$$

Dual generators

- Definition : For $1 \leqslant p \leqslant q$ we put
$$a_{p,q} = \sigma_p \dots \sigma_{q-2}\ \sigma_{q-1}\ \sigma_{q-2}^{-1} \dots \sigma_p^{-1}.$$

- Example :

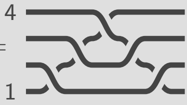
$$a_{1,4} =$$

Dual generators

- Definition : For $1 \leqslant p \leqslant q$ we put

$$a_{p,q} = \sigma_p \ldots \sigma_{q-2}\ \sigma_{q-1}\ \sigma_{q-2}^{-1} \ldots \sigma_p^{-1}.$$

- Example :

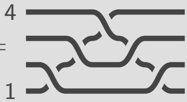
$a_{1,4} =$  $\approx$

Dual generators

- Definition : For $1 \leqslant p \leqslant q$ we put

$$a_{p,q} = \sigma_p \ldots \sigma_{q-2}\ \sigma_{q-1}\ \sigma_{q-2}^{-1} \ldots \sigma_p^{-1}.$$

- Example :

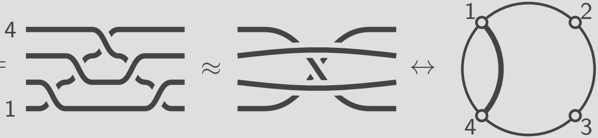
$a_{1,4} =$  $\approx$  $\leftrightarrow$

Dual generators

- **Definition** : For $1 \leqslant p \leqslant q$ we put

$$a_{p,q} = \sigma_p \dots \sigma_{q-2}\ \sigma_{q-1}\ \sigma_{q-2}^{-1} \dots \sigma_p^{-1}.$$

- **Example** :

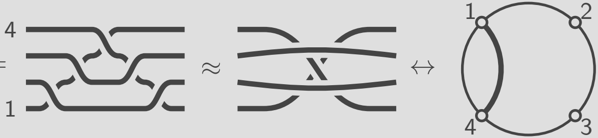
$a_{1,4} =$ 

Dual generators

- **Definition** : For $1 \leqslant p \leqslant q$ we put

$$a_{p,q} = \sigma_p \ldots \sigma_{q-2}\ \sigma_{q-1}\ \sigma_{q-2}^{-1} \ldots \sigma_p^{-1}.$$

- **Example** :

$a_{1,4} =$ 

- **Definition** : For all $n \geqslant 2$, we put

$$\Sigma_n^{+*} = \{a_{p,q} \,|\, 1 \leqslant p < q \leqslant n\}$$

and $\Sigma_n^* = \Sigma_n^{+*} \sqcup (\Sigma_n^{+*})^{-1}$.

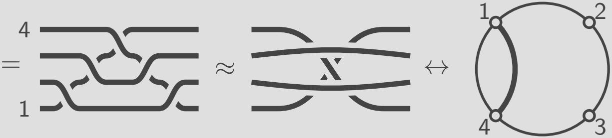
Dual generators

- Definition : For $1 \leqslant p \leqslant q$ we put

$$a_{p,q} = \sigma_p \ldots \sigma_{q-2}\ \sigma_{q-1}\ \sigma_{q-2}^{-1} \ldots \sigma_p^{-1}.$$

- Example :

$a_{1,4} =$



- Definition : For all $n \geqslant 2$, we put

$$\Sigma_n^{+*} = \{a_{p,q} \,|\, 1 \leqslant p < q \leqslant n\}$$

and $\Sigma_n^* = \Sigma_n^{+*} \sqcup (\Sigma_n^{+*})^{-1}$.

▸ As a semigroup, B_n is generated by Σ_n^*.

Dual presentation

- Theorem (J. Birman, K. H. Ko, S. J. Lee 1998) : In terms of Σ_n^{+*}, the group B_n is presented by the relations

Dual presentation

- Theorem (J. Birman, K. H. Ko, S. J. Lee 1998) : In terms of Σ_n^{+*}, the group B_n is presented by the relations

$$a_{p,q}a_{r,s} \equiv a_{r,s}a_{p,q} \quad \text{for } [p,q] \text{ and } [r,s] \text{ disjoint or nested,}$$

Dual presentation

- Theorem (J. Birman, K. H. Ko, S. J. Lee 1998) : In terms of Σ_n^{+*}, the group B_n is presented by the relations

$$a_{p,q} a_{r,s} \equiv a_{r,s} a_{p,q} \quad \text{for } [p,q] \text{ and } [r,s] \text{ disjoint or nested},$$
$$a_{p,q} a_{q,r} \equiv a_{q,r} a_{p,r} \equiv a_{p,r} a_{p,q} \quad \text{for } 1 \leqslant p < q < r \leqslant n.$$

Dual presentation

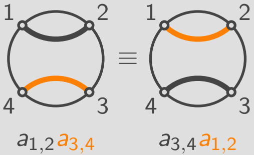
- Theorem (J. Birman, K. H. Ko, S. J. Lee 1998) : In terms of Σ_n^{+*}, the group B_n is presented by the relations

$$a_{p,q}a_{r,s} \equiv a_{r,s}a_{p,q} \quad \text{for } [p,q] \text{ and } [r,s] \text{ disjoint or nested},$$
$$a_{p,q}a_{q,r} \equiv a_{q,r}a_{p,r} \equiv a_{p,r}a_{p,q} \quad \text{for } 1 \leqslant p < q < r \leqslant n.$$

Dual presentation

- Theorem (J. Birman, K. H. Ko, S. J. Lee 1998) : In terms of Σ_n^{+*}, the group B_n is presented by the relations

$$a_{p,q}a_{r,s} \equiv a_{r,s}a_{p,q} \quad \text{for } [p,q] \text{ and } [r,s] \text{ disjoint or nested},$$
$$a_{p,q}a_{q,r} \equiv a_{q,r}a_{p,r} \equiv a_{p,r}a_{p,q} \quad \text{for } 1 \leqslant p < q < r \leqslant n.$$

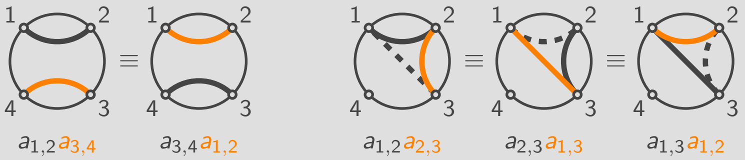


Dual presentation

- Theorem (J. Birman, K. H. Ko, S. J. Lee 1998) : In terms of Σ_n^{+*}, the group B_n is presented by the relations

$$a_{p,q}a_{r,s} \equiv a_{r,s}a_{p,q} \quad \text{for } [p,q] \text{ and } [r,s] \text{ disjoint or nested,}$$

$$a_{p,q}a_{q,r} \equiv a_{q,r}a_{p,r} \equiv a_{p,r}a_{p,q} \quad \text{for } 1 \leqslant p < q < r \leqslant n.$$

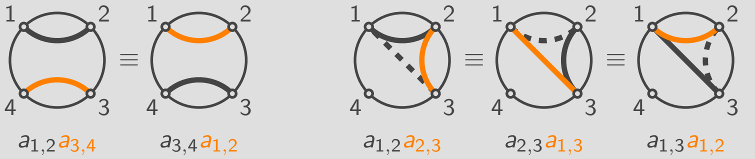


Dual presentation

- Theorem (J. Birman, K. H. Ko, S. J. Lee 1998) : In terms of Σ_n^{+*}, the group B_n is presented by the relations

$$a_{p,q}a_{r,s} \equiv a_{r,s}a_{p,q} \quad \text{for } [p,q] \text{ and } [r,s] \text{ disjoint or nested},$$

$$a_{p,q}a_{q,r} \equiv a_{q,r}a_{p,r} \equiv a_{p,r}a_{p,q} \quad \text{for } 1 \leqslant p < q < r \leqslant n.$$

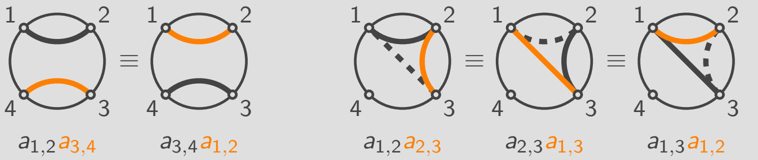


- Definition : We denote by B_n^{+*} the submonoid of B_n generated by Σ_n^{+*}.

Dual presentation

- **Theorem** (J. Birman, K. H. Ko, S. J. Lee 1998) : In terms of Σ_n^{+*}, the group B_n is presented by the relations

$$a_{p,q}a_{r,s} \equiv a_{r,s}a_{p,q} \quad \text{for } [p,q] \text{ and } [r,s] \text{ disjoint or nested},$$
$$a_{p,q}a_{q,r} \equiv a_{q,r}a_{p,r} \equiv a_{p,r}a_{p,q} \quad \text{for } 1 \leqslant p < q < r \leqslant n.$$



- **Definition** : We denote by B_n^{+*} the submonoid of B_n generated by Σ_n^{+*}.
  - The group of fractions of B_n^{+*} is also B_n.

Dual growth series

- Theorem (M. Albenque, P. Nadeau 2009) :

$$\mathcal{S}(B_n^{+*}, \Sigma_n^{+*}) = \left[\sum_{k=0}^{n-1} \frac{(n-1+k)!(-t)^k}{(n-1-k)!k!(k+1)!} \right]^{-1}$$

Dual growth series

- Theorem (M. Albenque, P. Nadeau 2009) :

$$\mathcal{S}(B_n^{+*}, \Sigma_n^{+*}) = \left[\sum_{k=0}^{n-1} \frac{(n-1+k)!(-t)^k}{(n-1-k)!k!(k+1)!} \right]^{-1}$$

  - $\mathcal{G}(B_n^{+*}, \Sigma_n^{+*})$ is irrelevant as every Σ_n^{+*}-word is geodesic.

Dual growth series

- Theorem (M. Albenque, P. Nadeau 2009) :

$$\mathcal{S}(B_n^{+*}, \Sigma_n^{+*}) = \left[\sum_{k=0}^{n-1} \frac{(n-1+k)!(-t)^k}{(n-1-k)!k!(k+1)!}\right]^{-1}$$

  - $\mathcal{G}(B_n^{+*}, \Sigma_n^{+*})$ is irrelevant as every Σ_n^{+*}-word is geodesic.
  - They obtain similar results for dual braid monoids of type B and D.

Dual growth series

- Theorem (M. Albenque, P. Nadeau 2009) :
$$\mathcal{S}(B_n^{+*}, \Sigma_n^{+*}) = \left[\sum_{k=0}^{n-1} \frac{(n-1+k)!(-t)^k}{(n-1-k)!k!(k+1)!}\right]^{-1}$$
  - $\mathcal{G}(B_n^{+*}, \Sigma_n^{+*})$ is irrelevant as every Σ_n^{+*}-word is geodesic.
  - They obtain similar results for dual braid monoids of type B and D.

- Except for $n = 2$, which is obvious, I can't found any result on $\mathcal{S}(B_n, \Sigma_n^*)$ or $\mathcal{G}(B_n, \Sigma_n^*)$ in the litterature.

Dual growth series

- Theorem (M. Albenque, P. Nadeau 2009) :

$$\mathcal{S}(B_n^{+*}, \Sigma_n^{+*}) = \left[\sum_{k=0}^{n-1} \frac{(n-1+k)!(-t)^k}{(n-1-k)!k!(k+1)!}\right]^{-1}$$

  - $\mathcal{G}(B_n^{+*}, \Sigma_n^{+*})$ is irrelevant as every Σ_n^{+*}-word is geodesic.
  - They obtain similar results for dual braid monoids of type B and D.

- Except for $n = 2$, which is obvious, I can't found any result on $\mathcal{S}(B_n, \Sigma_n^*)$ or $\mathcal{G}(B_n, \Sigma_n^*)$ in the litterature.

- Fact : We have $\Sigma_2^* = \Sigma_2^+ = \{\sigma_1^{\pm 1}\}$ and

$$\mathcal{S}(B_2, \{\sigma_1^{\pm 1}\}) = \mathcal{G}(B_2, \{\sigma_1^{\pm 1}\}) = \mathcal{G}(\mathbb{Z}, \{\pm 1\}) = \frac{1+t}{1-t}.$$

Roadmap

For the sequel, S_n will denotes either Σ_n or Σ_n^*.

Roadmap

For the sequel, S_n will denotes either Σ_n or Σ_n^*.

- Construct an algorithmic framework for computing
  - $s(B_n, S_n; \ell) = \operatorname{card}\left(\{\beta \in B_n \text{ s.t. } |\beta|_{S_n} = \ell\}\right)$,
  - $g(B_n, S_n; \ell) = \operatorname{card}\left(\{u \in S_n^* \text{ s.t. } |u| = |\overline{u}|_{S_n} = \ell\}\right)$.

Roadmap

For the sequel, S_n will denotes either Σ_n or Σ_n^*.

- Construct an algorithmic framework for computing
  - $s(B_n, S_n; \ell) = \operatorname{card}(\{\beta \in B_n \text{ s.t. } |\beta|_{S_n} = \ell\})$,
  - $g(B_n, S_n; \ell) = \operatorname{card}(\{u \in S_n^* \text{ s.t. } |u| = |\overline{u}|_{S_n} = \ell\})$.

  ▸ Efficient as possible, in particular in the case of $n \leqslant 4$.

Roadmap

For the sequel, S_n will denotes either Σ_n or Σ_n^*.

- Construct an algorithmic framework for computing
  - $s(B_n, S_n; \ell) = \operatorname{card}(\{\beta \in B_n \text{ s.t. } |\beta|_{S_n} = \ell\})$,
  - $g(B_n, S_n; \ell) = \operatorname{card}(\{u \in S_n^* \text{ s.t. } |u| = |\overline{u}|_{S_n} = \ell\})$.

  - Efficient as possible, in particular in the case of $n \leqslant 4$.
  - Here we focus on $s(B_n, S_n; \ell)$.

Roadmap

For the sequel, S_n will denotes either Σ_n or Σ_n^*.

- Construct an algorithmic framework for computing
  - $s(B_n, S_n; \ell) = \mathrm{card}\left(\{\beta \in B_n \text{ s.t. } |\beta|_{S_n} = \ell\}\right)$,
  - $g(B_n, S_n; \ell) = \mathrm{card}\left(\{u \in S_n^* \text{ s.t. } |u| = |\overline{u}|_{S_n} = \ell\}\right)$.

  - Efficient as possible, in particular in the case of $n \leqslant 4$.
  - Here we focus on $s(B_n, S_n; \ell)$.

- Compute as many terms as possible of $s(B_n, S_n; \ell)$ and $g(B_n, S_n; \ell)$ with $n = 4$ for $S_n = \Sigma_n$ and $n = 3, 4$ for $S_n = \Sigma_n^*$.

Roadmap

For the sequel, S_n will denotes either Σ_n or Σ_n^*.

- Construct an algorithmic framework for computing
  - $s(B_n, S_n; \ell) = \mathrm{card}\left(\{\beta \in B_n \text{ s.t. } |\beta|_{S_n} = \ell\}\right)$,
  - $g(B_n, S_n; \ell) = \mathrm{card}\left(\{u \in S_n^* \text{ s.t. } |u| = |\overline{u}|_{S_n} = \ell\}\right)$.

  - Efficient as possible, in particular in the case of $n \leqslant 4$.
  - Here we focus on $s(B_n, S_n; \ell)$.

- Compute as many terms as possible of $s(B_n, S_n; \ell)$ and $g(B_n, S_n; \ell)$ with $n = 4$ for $S_n = \Sigma_n$ and $n = 3, 4$ for $S_n = \Sigma_n^*$.

- Try to guess rational values for
  - $\mathcal{S}(B_4, \Sigma_4)$ and $\mathcal{G}(B_4, \Sigma_4)$,
  - $\mathcal{S}(B_3, \Sigma_3^*)$ and $\mathcal{G}(B_4, \Sigma_3^*)$,
  - $\mathcal{S}(B_4, \Sigma_4^*)$ and $\mathcal{G}(B_4, \Sigma_4^*)$.

Representative sets

S_n denotes either Σ_n or Σ_n^*.

- Definition : $B_n(S_n; \ell) = \{\beta \in B_n \text{ s.t. } |\beta|_{S_n} = \ell\}$.

Representative sets

S_n denotes either Σ_n or Σ_n^*.

- **Definition** : $B_n(S_n;\ell) = \{\beta \in B_n \text{ s.t. } |\beta|_{S_n} = \ell\}$.
  - $s(B_n, S_n;\ell) = \mathrm{card}\,(B_n(S_n;\ell))$.

Representative sets

S_n denotes either Σ_n or Σ_n^*.

- **Definition** : $B_n(S_n;\ell) = \{\beta \in B_n \text{ s.t. } |\beta|_{S_n} = \ell\}$.
  - $s(B_n, S_n; \ell) = \text{card}\,(B_n(S_n;\ell))$.

- **Definition** : A set W of S_n-words **represents** a subset X of B_n, denoted $W \vdash X$, whenever
  – words occuring in W are **geodesics** ;
  – each braid of X has a **unique representative** in W.

Representative sets

S_n denotes either Σ_n or Σ_n^*.

- **Definition** : $B_n(S_n;\ell) = \{\beta \in B_n \text{ s.t. } |\beta|_{S_n} = \ell\}$.
  - $s(B_n, S_n; \ell) = \text{card}\,(B_n(S_n;\ell))$.

- **Definition** : A set W of S_n-words represents a subset X of B_n, denoted $W \vdash X$, whenever
  – words occuring in W are geodesics ;
  – each braid of X has a unique representative in W.
  - For all $\ell \in \mathbb{N}$, we aim to construct a set $W_\ell \vdash B_n(S_n;\ell)$.

Representative sets

S_n denotes either Σ_n or Σ_n^*.

- **Definition** : $B_n(S_n;\ell) = \{\beta \in B_n \text{ s.t. } |\beta|_{S_n} = \ell\}$.
  - $s(B_n, S_n;\ell) = \operatorname{card}(B_n(S_n;\ell))$.

- **Definition** : A set W of S_n-words represents a subset X of B_n, denoted $W \vdash X$, whenever
  – words occuring in W are geodesics ;
  – each braid of X has a unique representative in W.
  - For all $\ell \in \mathbb{N}$, we aim to construct a set $W_\ell \vdash B_n(S_n;\ell)$.
    - $s(B_n, S_n;\ell) = \operatorname{card}(W_\ell)$.

Representative sets

S_n denotes either Σ_n or Σ_n^*.

- **Definition** : $B_n(S_n;\ell) = \{\beta \in B_n \text{ s.t. } |\beta|_{S_n} = \ell\}$.

  - $s(B_n, S_n; \ell) = \operatorname{card}(B_n(S_n;\ell))$.

- **Definition** : A set W of S_n-words **represents** a subset X of B_n, denoted $W \vdash X$, whenever
  – words occuring in W are **geodesics** ;
  – each braid of X has a **unique representative** in W.

  - For all $\ell \in \mathbb{N}$, we aim to construct a set $W_\ell \vdash B_n(S_n;\ell)$.
    - $s(B_n, S_n; \ell) = \operatorname{card}(W_\ell)$.

- **Examples** : We have $B_n(S_n; 0) = \{1\}$ and so $\{\varepsilon\} \vdash B_n(S_n; 0)$.
  With a little more work we obtain $S_n \vdash B_n(S_n; 1)$.

Representative sets

S_n denotes either Σ_n or Σ_n^*.

- Definition : $B_n(S_n;\ell) = \{\beta \in B_n \text{ s.t. } |\beta|_{S_n} = \ell\}$.
  - $s(B_n, S_n; \ell) = \operatorname{card}(B_n(S_n;\ell))$.

- Definition : A set W of S_n-words represents a subset X of B_n, denoted $W \vdash X$, whenever
  – words occuring in W are geodesics ;
  – each braid of X has a unique representative in W.
  - For all $\ell \in \mathbb{N}$, we aim to construct a set $W_\ell \vdash B_n(S_n;\ell)$.
    - $s(B_n, S_n; \ell) = \operatorname{card}(W_\ell)$.

- Examples : We have $B_n(S_n; 0) = \{1\}$ and so $\{\varepsilon\} \vdash B_n(S_n; 0)$.
  With a little more work we obtain $S_n \vdash B_n(S_n; 1)$.

- Question : How to determine if a given S_n-word u is geodesic ?

Representative sets

S_n denotes either Σ_n or Σ_n^*.

- Definition : $B_n(S_n;\ell) = \{\beta \in B_n \text{ s.t. } |\beta|_{S_n} = \ell\}$.
  - $s(B_n, S_n; \ell) = \operatorname{card}(B_n(S_n;\ell))$.

- Definition : A set W of S_n-words represents a subset X of B_n, denoted $W \vdash X$, whenever
  – words occuring in W are geodesics ;
  – each braid of X has a unique representative in W.
  - For all $\ell \in \mathbb{N}$, we aim to construct a set $W_\ell \vdash B_n(S_n;\ell)$.
    - $s(B_n, S_n; \ell) = \operatorname{card}(W_\ell)$.

- Examples : We have $B_n(S_n; 0) = \{1\}$ and so $\{\varepsilon\} \vdash B_n(S_n; 0)$. With a little more work we obtain $S_n \vdash B_n(S_n; 1)$.

- Question : How to determine if a given S_n-word u is geodesic ?
  - No good algorithms.

Representative sets : inductive construction

Assume $\ell \geqslant 2$.

- Lemma : Let u be a geodesic S_n-word of length $\ell - 1$ and $\alpha \in S_n$.
  If the word $u\,\alpha$ is not geodesic, then there exists a geodesic S_n-word w of length $\ell - 2$ satisfying $w \equiv u\,\alpha$.

Representative sets : inductive construction

Assume $\ell \geqslant 2$.

- **Lemma** : Let u be a geodesic S_n-word of length $\ell - 1$ and $\alpha \in S_n$. If the word $u\,\alpha$ is not geodesic, then there exists a geodesic S_n-word w of length $\ell - 2$ satisfying $w \equiv u\,\alpha$.

- **Definition** : We say that an S_n-word u appears in a subset W of S_n^* whenever u is equivalent to a word of W, denoted $u \lhd W$.

Representative sets : inductive construction

Assume $\ell \geqslant 2$.

- Lemma : Let u be a geodesic S_n-word of length $\ell - 1$ and $\alpha \in S_n$. If the word $u\,\alpha$ is not geodesic, then there exists a geodesic S_n-word w of length $\ell - 2$ satisfying $w \equiv u\,\alpha$.

- Definition : We say that an S_n-word u appears in a subset W of S_n^* whenever u is equivalent to a word of W, denoted $u \lhd W$.

- We now tackle the construction of a representative set of $B_n(S_n;\ell)$.

Representative sets : inductive construction

Assume $\ell \geqslant 2$.

- **Lemma** : Let u be a geodesic S_n-word of length $\ell - 1$ and $\alpha \in S_n$. If the word $u\,\alpha$ is not geodesic, then there exists a geodesic S_n-word w of length $\ell - 2$ satisfying $w \equiv u\,\alpha$.

- **Definition** : We say that an S_n-word u appears in a subset W of S_n^* whenever u is equivalent to a word of W, denoted $u \lhd W$.

- We now tackle the construction of a representative set of $B_n(S_n; \ell)$. Assume we have constructed :

  $$W_{\ell-2} \vdash B_n(S_n; \ell-2) \text{ and } W_{\ell-1} \vdash B_n(S_n; \ell-1).$$

  We then construct

Representative sets : inductive construction

Assume $\ell \geqslant 2$.

- Lemma : Let u be a geodesic S_n-word of length $\ell - 1$ and $\alpha \in S_n$.
  If the word $u\,\alpha$ is not geodesic, then there exists a geodesic S_n-word w of length $\ell - 2$ satisfying $w \equiv u\,\alpha$.

- Definition : We say that an S_n-word u appears in a subset W of S_n^* whenever u is equivalent to a word of W, denoted $u \lhd W$.

- We now tackle the construction of a representative set of $B_n(S_n;\ell)$.
  Assume we have constructed :

  $W_{\ell-2} \vdash B_n(S_n;\ell{-}2)$ and $W_{\ell-1} \vdash B_n(S_n;\ell{-}1)$.

  We then construct

  – $W''_\ell = \{u\,\alpha \text{ for } (u,\alpha) \in W_{\ell-1} \times S_n\}$,

Representative sets : inductive construction

Assume $\ell \geqslant 2$.

- Lemma : Let u be a geodesic S_n-word of length $\ell - 1$ and $\alpha \in S_n$. If the word $u\,\alpha$ is not geodesic, then there exists a geodesic S_n-word w of length $\ell - 2$ satisfying $w \equiv u\,\alpha$.

- Definition : We say that an S_n-word u appears in a subset W of S_n^* whenever u is equivalent to a word of W, denoted $u \lhd W$.

- We now tackle the construction of a representative set of $B_n(S_n;\ell)$. Assume we have constructed :

  $W_{\ell-2} \vdash B_n(S_n;\ell-2)$ and $W_{\ell-1} \vdash B_n(S_n;\ell-1)$.

  We then construct

  - $W''_\ell = \{u\,\alpha \text{ for } (u,\alpha) \in W_{\ell-1} \times S_n\}$,
  - W'_ℓ by keeping words u of W''_ℓ that do not appear in $W_{\ell-2}$,

Representative sets : inductive construction

Assume $\ell \geqslant 2$.

- **Lemma** : Let u be a geodesic S_n-word of length $\ell - 1$ and $\alpha \in S_n$. If the word $u\,\alpha$ is not geodesic, then there exists a geodesic S_n-word w of length $\ell - 2$ satisfying $w \equiv u\,\alpha$.

- **Definition** : We say that an S_n-word u appears in a subset W of S_n^* whenever u is equivalent to a word of W, denoted $u \lhd W$.

- We now tackle the construction of a representative set of $B_n(S_n;\ell)$. Assume we have constructed :

  $W_{\ell-2} \vdash B_n(S_n;\ell-2)$ and $W_{\ell-1} \vdash B_n(S_n;\ell-1)$.

  We then construct

  - $W_\ell'' = \{u\,\alpha \text{ for } (u,\alpha) \in W_{\ell-1} \times S_n\}$,
  - W_ℓ' by keeping words u of W_ℓ'' that do not appear in $W_{\ell-2}$,
  - W_ℓ from W_ℓ' by keeping only one word in each $\equiv$-classes.

A first algorithm

- Algorithm $(W_{\ell-2} \vdash B_n(S_n; \ell-2),\ W_{\ell-1} \vdash B_n(S_n; \ell-1))$:

for $u \in W_{\ell-1}$ **do**
    for $\alpha \in S_n$ **do**
        $v \leftarrow u\,\alpha$
        if $v \not\trianglelefteq W_{\ell-2}$ and $v \not\trianglelefteq W_\ell$ **then**
            $W_\ell \leftarrow W_\ell \sqcup \{v\}$ *A new braid* $\overline{v}$ *of* $B_n(S_n; \ell)$ *is found.*
        end if
    end for
end for
return W_ℓ

A first algorithm

- Algorithm $(W_{\ell-2} \vdash B_n(S_n; \ell-2),\ W_{\ell-1} \vdash B_n(S_n; \ell-1))$:

  for $u \in W_{\ell-1}$ **do**
      for $\alpha \in S_n$ **do**
          $v \leftarrow u\,\alpha$
          if $v \not\triangleleft W_{\ell-2}$ and $v \not\triangleleft W_\ell$ **then**
              $W_\ell \leftarrow W_\ell \sqcup \{v\}$ *A new braid $\overline{v}$ of $B_n(S_n; \ell)$ is found.*
          end if
      end for
  end for
  return W_ℓ

- Question : How to test if a S_n-word u appears in a subset W of S_n^* ?

Representative sets implementation

- A S_n-word u is naturally represented as an `array` of size $|u|$.

Representative sets implementation

- A S_n-word u is naturally represented as an `array` of size $|u|$.

Les W be a finite subset of S_n^* and u be a S_n-word. Does u appear in W ?

Representative sets implementation

- A S_n-word u is naturally represented as an `array` of size $|u|$.

Les W be a finite subset of S_n^* and u be a S_n-word. Does u appear in W ?

- A first idea : Representing W by an `array` and use Garside normal form to detect equivalences.

Representative sets implementation

- A S_n-word u is naturally represented as an `array` of size $|u|$.

Les W be a finite subset of S_n^* and u be a S_n-word. Does u appear in W ?

- A first idea : Representing W by an `array` and use Garside normal form to detect equivalences.
  - $u \lhd W$ requires at most $O(|W|)$ test of equivalence $\equiv$,

Representative sets implementation

- A S_n-word u is naturally represented as an `array` of size $|u|$.

Les W be a finite subset of S_n^* and u be a S_n-word. Does u appear in W ?

- A first idea : Representing W by an `array` and use Garside normal form to detect equivalences.
  - $u \lhd W$ requires at most $O(|W|)$ test of equivalence $\equiv$,
  - Garside normal form does not preserve geodesic words.

Representative sets implementation

- A S_n-word u is naturally represented as an `array` of size $|u|$.

Les W be a finite subset of S_n^* and u be a S_n-word. Does u appear in W ?

- A first idea : Representing W by an `array` and use Garside normal form to detect equivalences.
  - $u \lhd W$ requires at most $O(|W|)$ test of equivalence $\equiv$,
  - Garside normal form does not preserve geodesic words.

- A second idea : Representing W by an `ordered array` using braid ordering introduced by P. Dehornoy.

Representative sets implementation

- A S_n-word u is naturally represented as an `array` of size $|u|$.

Les W be a finite subset of S_n^* and u be a S_n-word. Does u appear in W ?

- A first idea : Representing W by an `array` and use Garside normal form to detect equivalences.
  - $u \lhd W$ requires at most $O(|W|)$ test of equivalence $\equiv$,
  - Garside normal form does not preserve geodesic words.

- A second idea : Representing W by an `ordered array` using braid ordering introduced by P. Dehornoy.
  - $u \lhd W$ requires at most $O(\log(|W|))$ comparisons.

Representative sets implementation

- A S_n-word u is naturally represented as an `array` of size $|u|$.

Les W be a finite subset of S_n^* and u be a S_n-word. Does u appear in W ?

- A first idea : Representing W by an `array` and use Garside normal form to detect equivalences.
  - $u \lhd W$ requires at most $O(|W|)$ test of equivalence $\equiv$,
  - Garside normal form does not preserve geodesic words.

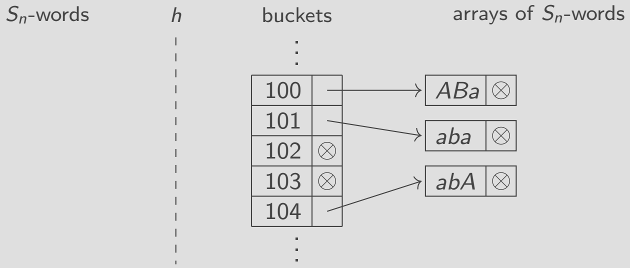
- A second idea : Representing W by an `ordered array` using braid ordering introduced by P. Dehornoy.
  - $u \lhd W$ requires at most $O(\log(|W|))$ comparisons.
  - Better but we can do more in this direction.

Use of a hash table

- Assume we have a map $h : S_n^* \to \mathbb{N}$ s.t. $h(u) = h(v)$ whenever $u \equiv v$.

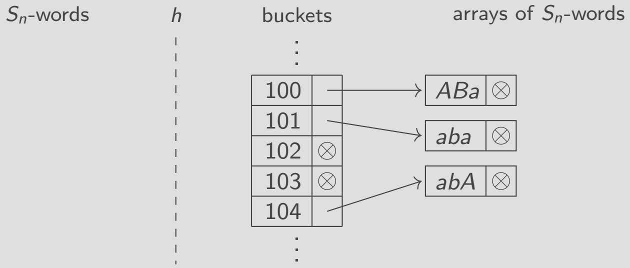
Use of a hash table

- Assume we have a map $h : S_n^* \to \mathbb{N}$ s.t. $h(u) = h(v)$ whenever $u \equiv v$.
  We can then represent a subset W of S_n^* using a hash table :



Use of a hash table

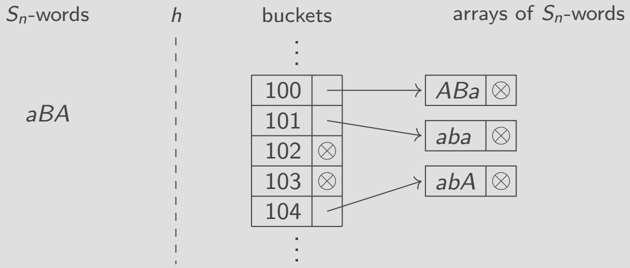
- Assume we have a map $h : S_n^* \to \mathbb{N}$ s.t. $h(u) = h(v)$ whenever $u \equiv v$.
  We can then represent a subset W of S_n^* using a hash table :



We want to insert aBA, bab, Aba if they do not appear in W.

Use of a hash table

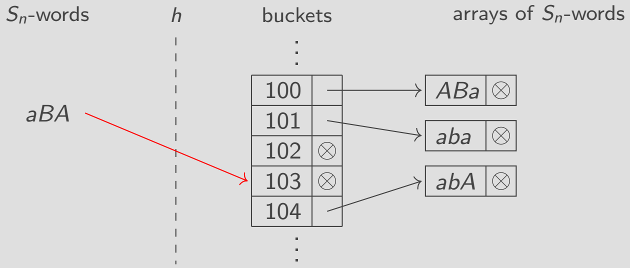
- Assume we have a map $h : S_n^* \to \mathbb{N}$ s.t. $h(u) = h(v)$ whenever $u \equiv v$. We can then represent a subset W of S_n^* using a hash table :



We want to insert aBA, bab, Aba if they do not appear in W.

Use of a hash table

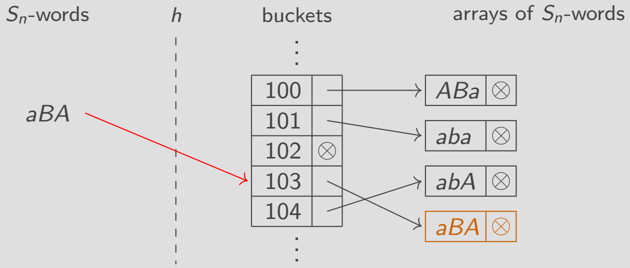
- Assume we have a map $h : S_n^* \to \mathbb{N}$ s.t. $h(u) = h(v)$ whenever $u \equiv v$.
  We can then represent a subset W of S_n^* using a hash table :



We want to insert aBA, bab, Aba if they do not appear in W.

Use of a hash table

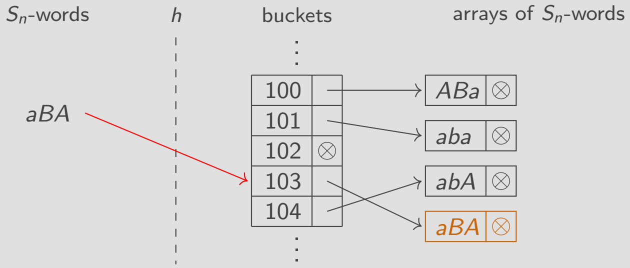
- Assume we have a map $h : S_n^* \to \mathbb{N}$ s.t. $h(u) = h(v)$ whenever $u \equiv v$.
  We can then represent a subset W of S_n^* using a hash table :



We want to insert aBA, bab, Aba if they do not appear in W.

Use of a hash table

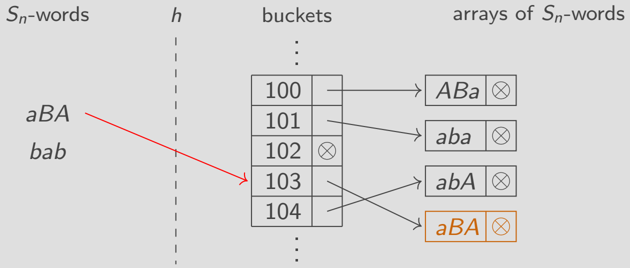
- Assume we have a map $h : S_n^* \to \mathbb{N}$ s.t. $h(u) = h(v)$ whenever $u \equiv v$.
  We can then represent a subset W of S_n^* using a hash table :



We want to insert aBA, bab, Aba if they do not appear in W.

Use of a hash table

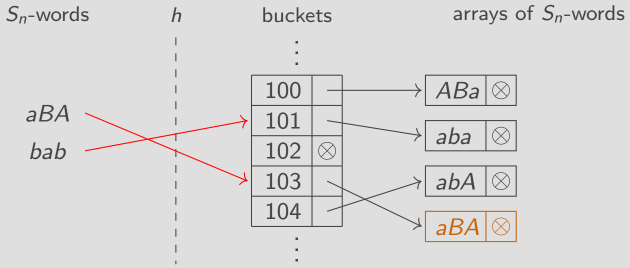
- Assume we have a map $h : S_n^* \to \mathbb{N}$ s.t. $h(u) = h(v)$ whenever $u \equiv v$.
  We can then represent a subset W of S_n^* using a hash table :



We want to insert aBA, bab, Aba if they do not appear in W.

Use of a hash table

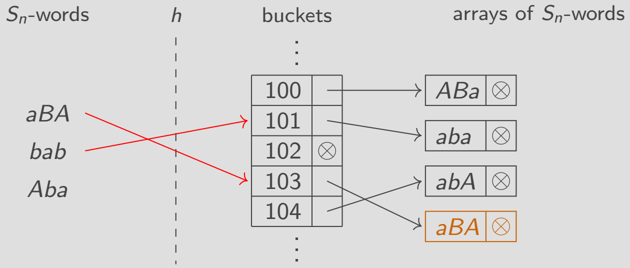
- Assume we have a map $h : S_n^* \to \mathbb{N}$ s.t. $h(u) = h(v)$ whenever $u \equiv v$.
  We can then represent a subset W of S_n^* using a hash table :



We want to insert aBA, bab, Aba if they do not appear in W.

Use of a hash table

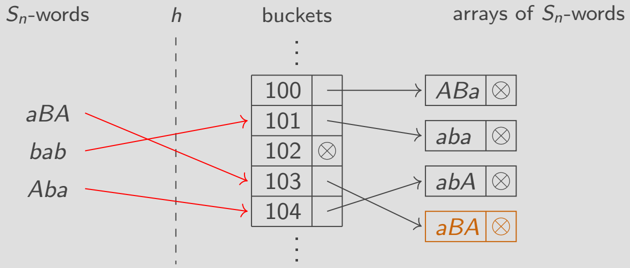
- Assume we have a map $h : S_n^* \to \mathbb{N}$ s.t. $h(u) = h(v)$ whenever $u \equiv v$.
  We can then represent a subset W of S_n^* using a hash table :



We want to insert aBA, bab, Aba if they do not appear in W.

Use of a hash table

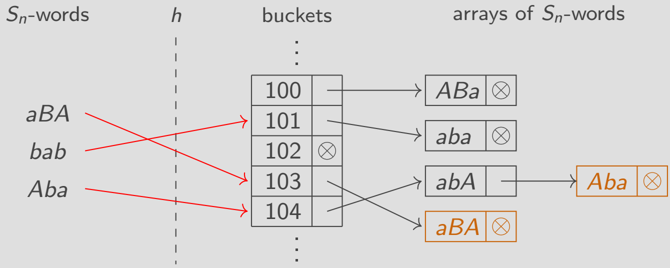
- Assume we have a map $h : S_n^* \to \mathbb{N}$ s.t. $h(u) = h(v)$ whenever $u \equiv v$.
  We can then represent a subset W of S_n^* using a hash table :



We want to insert aBA, bab, Aba if they do not appear in W.

Use of a hash table

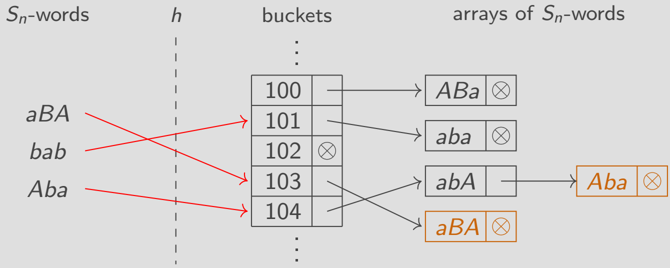
- Assume we have a map $h : S_n^* \rightarrow \mathbb{N}$ s.t. $h(u) = h(v)$ whenever $u \equiv v$. We can then represent a subset W of S_n^* using a hash table :



We want to insert aBA, bab, Aba if they do not appear in W.

Use of a hash table

- Assume we have a map $h : S_n^* \to \mathbb{N}$ s.t. $h(u) = h(v)$ whenever $u \equiv v$.
  We can then represent a subset W of S_n^* using a hash table :

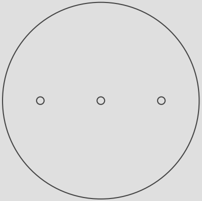


We want to insert aBA, bab, Aba if they do not appear in W.

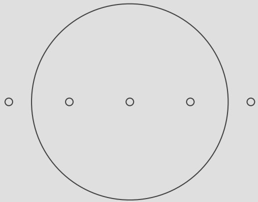
- In average case, insertion has a constant time complexity and a linear one in worst case.

Dynnikov's coordinates

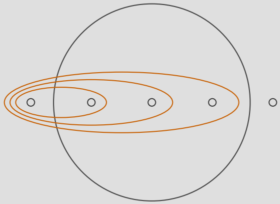
Dynnikov's coordinates



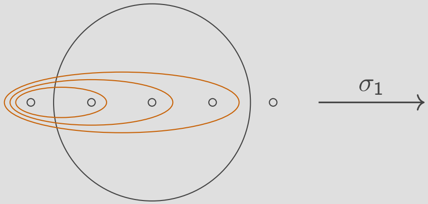
Dynnikov's coordinates



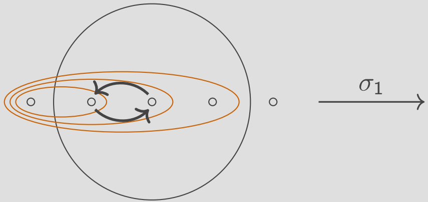
Dynnikov's coordinates



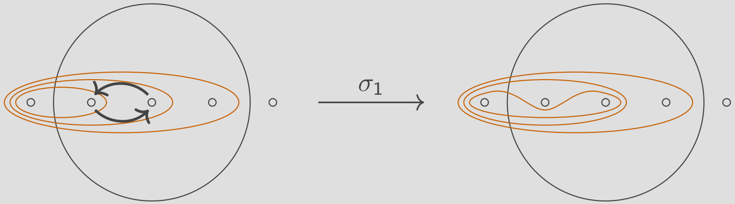
Dynnikov's coordinates



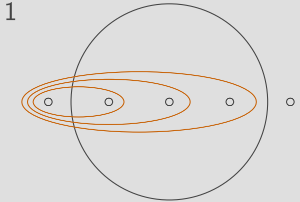
Dynnikov's coordinates

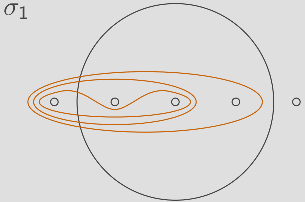


Dynnikov's coordinates



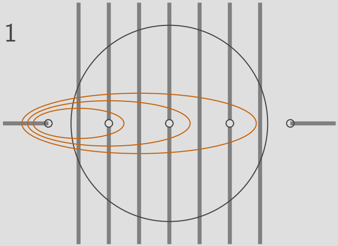
Dynnikov's coordinates



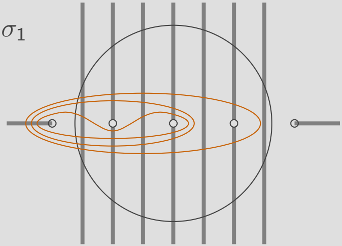


Dynnikov's coordinates

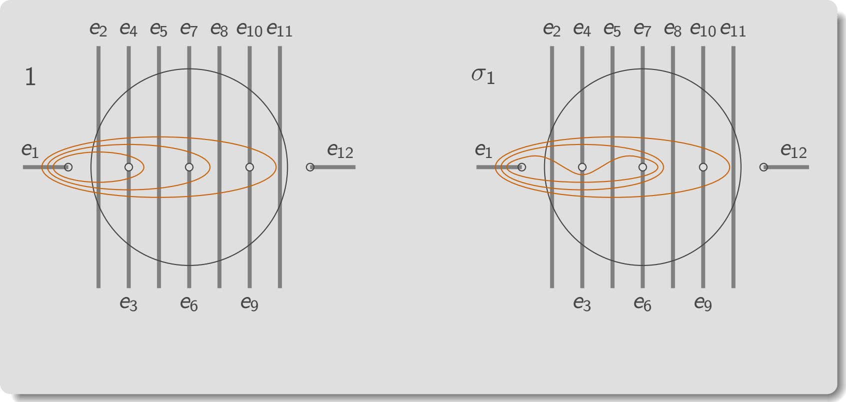
1



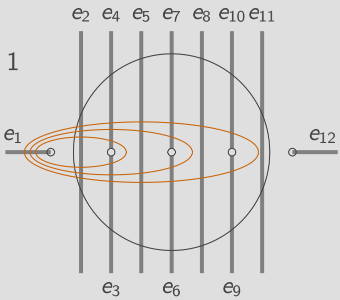
σ_1

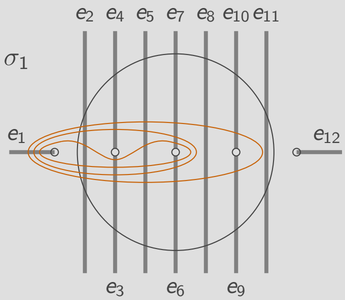


Dynnikov's coordinates



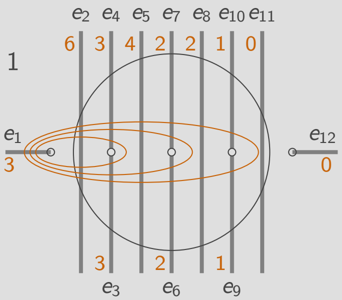
Dynnikov's coordinates

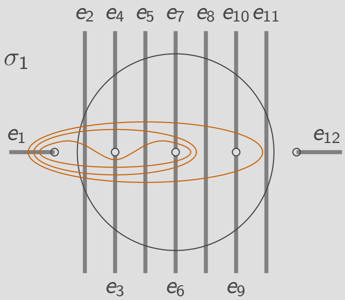




z_i = intersection number of the lamination with the edge e_i.

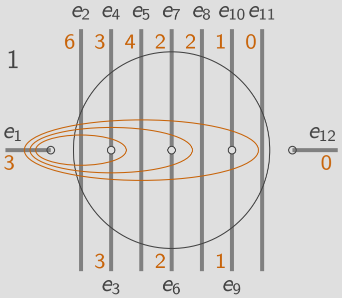
Dynnikov's coordinates

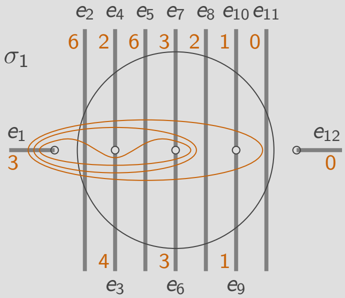




z_i = intersection number of the lamination with the edge e_i.

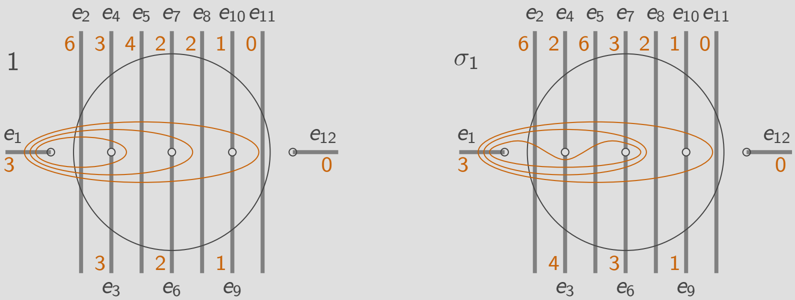
Dynnikov's coordinates





z_i = intersection number of the lamination with the edge e_i.

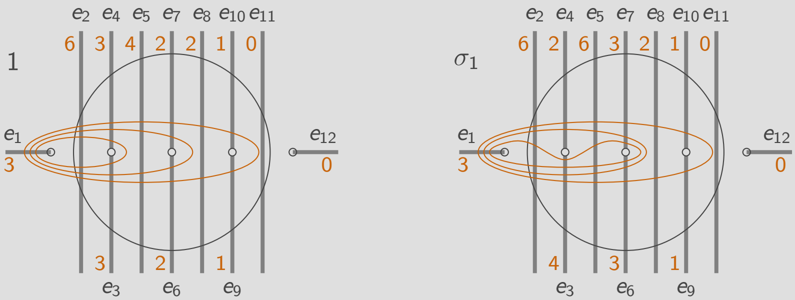
Dynnikov's coordinates



z_i = intersection number of the lamination with the edge e_i.

$x_i = (z_{3i} - z_{3i+1})/2$ and $y_i = (z_{3i-1} - z_{3i+2})/2$

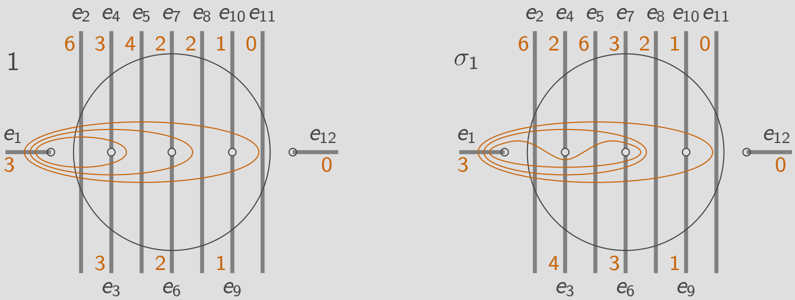
Dynnikov's coordinates



z_i = intersection number of the lamination with the edge e_i.
$x_i = (z_{3i} - z_{3i+1})/2$ and $y_i = (z_{3i-1} - z_{3i+2})/2$

- **Definition** : For $u \in S_n^*$, we put $\rho_D(u) = (x_1, y_1, \ldots, x_n, y_n) \in \mathbb{Z}^{2n}$.

Dynnikov's coordinates



z_i = intersection number of the lamination with the edge e_i.
$x_i = (z_{3i} - z_{3i+1})/2$ and $y_i = (z_{3i-1} - z_{3i+2})/2$

- **Definition** : For $u \in S_n^*$, we put $\rho_D(u) = (x_1, y_1, \ldots, x_n, y_n) \in \mathbb{Z}^{2n}$.

- **Examples** : $\rho_D(1) = (0, 1, 0, 1, 0, 1)$ and $\rho_D(\sigma_1) = (1, 0, 0, 2, 0, 1)$.

Dynnikov's coordinates – Properties

- Fact : Dynnikov's coordinates $\rho_D(u)$ can be easily computed.

- Fact : Dynnikov's coordinates $\rho_D(u)$ can be easily computed.
  - At most $O(|u|)$ elementary op. in the semiring $(\mathbb{Z} \cup \{+\infty\}, \max, +)$.

Dynnikov's coordinates – Properties

- Fact : Dynnikov's coordinates $\rho_D(u)$ can be easily computed.
  - At most $O(|u|)$ elementary op. in the semiring $(\mathbb{Z} \cup \{+\infty\}, \max, +)$.

- Theorem (I. Dynnikov 2002) For two S_n words u and v, we have

$$\rho_D(u) = \rho_D(v) \Leftrightarrow u \equiv v$$

Dynnikov's coordinates – Properties

- Fact : Dynnikov's coordinates $\rho_D(u)$ can be easily computed.
  - At most $O(|u|)$ elementary op. in the semiring $(\mathbb{Z} \cup \{+\infty\}, \max, +)$.

- Theorem (I. Dynnikov 2002) For two S_n words u and v, we have
$$\rho_D(u) = \rho_D(v) \Leftrightarrow u \equiv v$$
and Dehornoy ordering relation $\overline{u} < \overline{v}$ can be read on $\rho_D(u^{-1}v)$.

Dynnikov's coordinates – Properties

- Fact : Dynnikov's coordinates $\rho_D(u)$ can be easily computed.
  - At most $O(|u|)$ elementary op. in the semiring $(\mathbb{Z} \cup \{+\infty\}, \max, +)$.

- Theorem (I. Dynnikov 2002) For two S_n words u and v, we have
$$\rho_D(u) = \rho_D(v) \Leftrightarrow u \equiv v$$
and Dehornoy ordering relation $\overline{u} < \overline{v}$ can be read on $\rho_D(u^{-1}v)$.

- Definition : For $u \in S_4^*$ we define
$$h(u) = \sum_{i=0}^{7} \operatorname{rem}(c_i, 256)\ 256^i$$
where $(c_0, \ldots, c_7) = \rho_D(u)$ and $\operatorname{rem}(c, 256)$ is the reminder of $a \div 256$.

Dynnikov's coordinates – Properties

- Fact : Dynnikov's coordinates $\rho_D(u)$ can be easily computed.
  - At most $O(|u|)$ elementary op. in the semiring $(\mathbb{Z} \cup \{+\infty\}, \max, +)$.

- Theorem (I. Dynnikov 2002) For two S_n words u and v, we have
$$\rho_D(u) = \rho_D(v) \Leftrightarrow u \equiv v$$
and Dehornoy ordering relation $\overline{u} < \overline{v}$ can be read on $\rho_D(u^{-1}v)$.

- Definition : For $u \in S_4^*$ we define
$$h(u) = \sum_{i=0}^{7} \mathrm{rem}\,(c_i, 256)\; 256^i$$
where $(c_0, \ldots, c_7) = \rho_D(u)$ and $\mathrm{rem}\,(c, 256)$ is the reminder of $a \div 256$.
  - $h(u)$ is an integer of $[0, 2^{64} - 1]$,

Dynnikov's coordinates – Properties

- Fact : Dynnikov's coordinates $\rho_D(u)$ can be easily computed.
  - At most $O(|u|)$ elementary op. in the semiring $(\mathbb{Z} \cup \{+\infty\}, \max, +)$.

- Theorem (I. Dynnikov 2002) For two S_n words u and v, we have
$$\rho_D(u) = \rho_D(v) \Leftrightarrow u \equiv v$$
and Dehornoy ordering relation $\overline{u} < \overline{v}$ can be read on $\rho_D(u^{-1}v)$.

- Definition : For $u \in S_4^*$ we define
$$h(u) = \sum_{i=0}^{7} \operatorname{rem}(c_i, 256)\ 256^i$$
where $(c_0, \ldots, c_7) = \rho_D(u)$ and $\operatorname{rem}(c, 256)$ is the reminder of $a \div 256$.
  - $h(u)$ is an integer of $[0, 2^{64} - 1]$,
    - well-suited for 64-bits computers.

Bottleneck

Bottleneck

- Fact : With the current version of our algorirhm, the computation of $W_\ell \vdash B_n(S_n;\ell)$ requires to have two sets loaded in memory :
  - $W_{\ell-1} \vdash B_n(S_n;\ell-1)$,
  - $W_{\ell-2} \vdash B_n(S_n;\ell-2)$.

Bottleneck

- Fact : With the current version of our algorirhm, the computation of $W_\ell \vdash B_n(S_n; \ell)$ requires to have two sets loaded in memory :
  - $W_{\ell-1} \vdash B_n(S_n; \ell - 1)$,
  - $W_{\ell-2} \vdash B_n(S_n; \ell - 2)$.

  - As RAM capacity is not so big (some Go), we have a problem.

Bottleneck

- Fact : With the current version of our algorirhm, the computation of $W_\ell \vdash B_n(S_n; \ell)$ requires to have two sets loaded in memory :
  - $W_{\ell-1} \vdash B_n(S_n; \ell - 1)$,
  - $W_{\ell-2} \vdash B_n(S_n; \ell - 2)$.

  - As RAM capacity is not so big (some Go), we have a problem.
    - Massive use of swap memory, which impact performences.

Bottleneck

- Fact : With the current version of our algorirhm, the computation of $W_\ell \vdash B_n(S_n; \ell)$ requires to have two sets loaded in memory :
  - $W_{\ell-1} \vdash B_n(S_n; \ell - 1)$,
  - $W_{\ell-2} \vdash B_n(S_n; \ell - 2)$.

  - As RAM capacity is not so big (some Go), we have a problem.
    - Massive use of swap memory, which impact performences.

- Idea : Divides $B_n(S_n; \ell)$ on many independant pieces.

Bottleneck

- Fact : With the current version of our algorirhm, the computation of $W_\ell \vdash B_n(S_n; \ell)$ requires to have two sets loaded in memory :
  - $W_{\ell-1} \vdash B_n(S_n; \ell - 1)$,
  - $W_{\ell-2} \vdash B_n(S_n; \ell - 2)$.
  - As RAM capacity is not so big (some Go), we have a problem.
    - Massive use of swap memory, which impact performences.

- Idea : Divides $B_n(S_n; \ell)$ on many independant pieces.
  - Reduce RAM consumption.

Bottleneck

- Fact : With the current version of our algorirhm, the computation of $W_\ell \vdash B_n(S_n;\ell)$ requires to have two sets loaded in memory :
  - $W_{\ell-1} \vdash B_n(S_n;\ell-1)$,
  - $W_{\ell-2} \vdash B_n(S_n;\ell-2)$.

  - As RAM capacity is not so big (some Go), we have a problem.
    - Massive use of swap memory, which impact performences.

- Idea : Divides $B_n(S_n;\ell)$ on many independant pieces.

  - Reduce RAM consumption.
  - Allow parallelization.

Braid permutaion

- Definition : $\pi : B_n \to \mathfrak{S}_n$ is the morphism defined by $\pi(\sigma_i) = (i\ i{+}1)$.

Braid permutaion

- Definition : $\pi : B_n \to \mathfrak{S}_n$ is the morphism defined by $\pi(\sigma_i) = (i\ i{+}1)$.
  - $\pi(\beta)$ is the permutation of $\mathfrak{S}_n$ such that the strand ending at position i starts at position $\pi(\beta)(i)$.

Braid permutaion

- Definition : $\pi : B_n \to \mathfrak{S}_n$ is the morphism defined by $\pi(\sigma_i) = (i\ i{+}1)$.
  - $\pi(\beta)$ is the permutation of $\mathfrak{S}_n$ such that the strand ending at position i starts at position $\pi(\beta)(i)$.

- Example : $\beta = \sigma_1\sigma_2^{-1}\sigma_1\sigma_2$,

Braid permutaion

- Definition : $\pi : B_n \to \mathfrak{S}_n$ is the morphism defined by $\pi(\sigma_i) = (i\ i{+}1)$.
  - $\pi(\beta)$ is the permutation of $\mathfrak{S}_n$ such that the strand ending at position i starts at position $\pi(\beta)(i)$.

- Example : $\beta = \sigma_1 \sigma_2^{-1} \sigma_1 \sigma_2$,

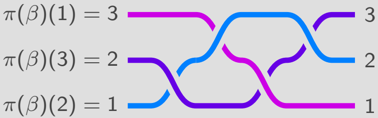
$$\pi(\beta) = (1\ 2)\,(2\ 3)\,(1\ 2)\,(2\ 3) = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \end{pmatrix}$$

Braid permutaion

- Definition : $\pi : B_n \to \mathfrak{S}_n$ is the morphism defined by $\pi(\sigma_i) = (i\ i{+}1)$.
  - $\pi(\beta)$ is the permutation of $\mathfrak{S}_n$ such that the strand ending at position i starts at position $\pi(\beta)(i)$.

- Example : $\beta = \sigma_1\sigma_2^{-1}\sigma_1\sigma_2$,

$$\pi(\beta) = (1\ 2)\,(2\ 3)\,(1\ 2)\,(2\ 3) = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \end{pmatrix}$$



Linking numbers

- Definition : Let $\beta \in B_n$ and $i, j \in \{1, \ldots, n\}$ with $i \neq j$.
  The linking number $\ell_{i,j}(\beta)$ is the algebraic number of crossings involving the strands i and j in β.

Linking numbers

- Definition : Let $\beta \in B_n$ and $i, j \in \{1, \ldots, n\}$ with $i \neq j$.
  The linking number $\ell_{i,j}(\beta)$ is the algebraic number of crossings involving the strands i and j in β.

- Example :

$\ell_{1,2} = 2$

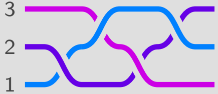
$\ell_{1,3} = -1$

$\ell_{1,3} = 1$

Linking numbers

- Definition : Let $\beta \in B_n$ and $i, j \in \{1, \dots, n\}$ with $i \neq j$.
  The linking number $\ell_{i,j}(\beta)$ is the algebraic number of crossings involving the strands i and j in β.

- Example :



$\ell_{1,2} = 2$

$\ell_{1,3} = -1$

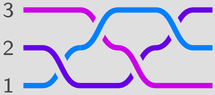
$\ell_{1,3} = 1$

- Lemma : For β, γ in B_n and $1 \leqslant i < j \leqslant n$ we have
$$\ell_{i,j}(\beta \cdot \gamma) = \ell_{i,j}(\beta) + \ell_{\pi(\beta)^{-1}(i),\pi(\beta)^{-1}(j)}(\gamma),$$
with the convention $\ell_{p,q} = \ell_{q,p}$ for $p > q$.

Linking numbers

- **Definition** : Let $\beta \in B_n$ and $i, j \in \{1, \dots, n\}$ with $i \neq j$.
  The linking number $\ell_{i,j}(\beta)$ is the algebraic number of crossings involving the strands i and j in β.

- **Example** :



$\ell_{1,2} = 2$
$\ell_{1,3} = -1$
$\ell_{1,3} = 1$

- **Lemma** : For β, γ in B_n and $1 \leqslant i < j \leqslant n$ we have
$$\ell_{i,j}(\beta \cdot \gamma) = \ell_{i,j}(\beta) + \ell_{\pi(\beta)^{-1}(i),\pi(\beta)^{-1}(j)}(\gamma),$$
  with the convention $\ell_{p,q} = \ell_{q,p}$ for $p > q$.

- **Corollary** : For $\alpha \in S_n$, $\ell_{i,j}(\beta \cdot \alpha)$ depends only of $\ell_{*,*}(\beta)$ and $\pi(\beta)$.

Braid template

- Definition : The template of $\beta \in B_n$ is

$$\tau(\beta) = (\pi(\beta), \ell_{1,2}(\beta), \ldots, \ell_{n-1,n}(\beta)) \in \mathfrak{S}_n \times \mathbb{Z}^{\frac{n(n+1)}{2}}$$

For $\ell \in \mathbb{N}$, we put $T_n(S_n;\ell) = \tau\left(B_n(S_n;\ell)\right)$.

Braid template

- **Definition** : The template of $\beta \in B_n$ is

$$\tau(\beta) = (\pi(\beta), \ell_{1,2}(\beta), \ldots, \ell_{n-1,n}(\beta)) \in \mathfrak{S}_n \times \mathbb{Z}^{\frac{n(n+1)}{2}}$$

For $\ell \in \mathbb{N}$, we put $T_n(S_n; \ell) = \tau\left(B_n(S_n; \ell)\right)$.

  - $u \equiv v$ implies $\tau(\overline{u}) = \tau(\overline{v})$.

Braid template

- Definition : The template of $\beta \in B_n$ is
$$\tau(\beta) = (\pi(\beta), \ell_{1,2}(\beta), \ldots, \ell_{n-1,n}(\beta)) \in \mathfrak{S}_n \times \mathbb{Z}^{\frac{n(n+1)}{2}}$$
For $\ell \in \mathbb{N}$, we put $T_n(S_n; \ell) = \tau\left(B_n(S_n; \ell)\right)$.
  - $u \equiv v$ implies $\tau(\overline{u}) = \tau(\overline{v})$.

- Fact : For $\beta \in B_n$ and $\alpha \in S_n$, the template $\tau(\beta) \odot \alpha = \tau(\beta\,\alpha)$ depends only of $\tau(\beta)$ and α.

Braid template

- Definition : The template of $\beta \in B_n$ is
$$\tau(\beta) = (\pi(\beta), \ell_{1,2}(\beta), \ldots, \ell_{n-1,n}(\beta)) \in \mathfrak{S}_n \times \mathbb{Z}^{\frac{n(n+1)}{2}}$$
For $\ell \in \mathbb{N}$, we put $T_n(S_n;\ell) = \tau\left(B_n(S_n;\ell)\right)$.

  - $u \equiv v$ implies $\tau(\overline{u}) = \tau(\overline{v})$.

- Fact : For $\beta \in B_n$ and $\alpha \in S_n$, the template $\tau(\beta) \odot \alpha = \tau(\beta\,\alpha)$ depends only of $\tau(\beta)$ and α.

- Definition : For $t \in T_n(S_n;\ell)$ we write
$$B_n(S_n;\ell,t) = \{\beta \in B_n(S_n;\ell) \text{ s.t. } \tau(\beta) = t\}.$$

Braid template

- Definition : The template of $\beta \in B_n$ is

$$\tau(\beta) = (\pi(\beta), \ell_{1,2}(\beta), \ldots, \ell_{n-1,n}(\beta)) \in \mathfrak{S}_n \times \mathbb{Z}^{\frac{n(n+1)}{2}}$$

For $\ell \in \mathbb{N}$, we put $T_n(S_n; \ell) = \tau\left(B_n(S_n; \ell)\right)$.

  - $u \equiv v$ implies $\tau(\overline{u}) = \tau(\overline{v})$.

- Fact : For $\beta \in B_n$ and $\alpha \in S_n$, the template $\tau(\beta) \odot \alpha = \tau(\beta\, \alpha)$ depends only of $\tau(\beta)$ and α.

- Definition : For $t \in T_n(S_n; \ell)$ we write

$$B_n(S_n; \ell, t) = \{\beta \in B_n(S_n; \ell) \text{ s.t. } \tau(\beta) = t\}.$$

  - The sets $B_n(S_n; \ell, t)$ provide a partition of $B_n(S_n; \ell)$.

Braid template

- **Definition** : The template of $\beta \in B_n$ is
$$\tau(\beta) = (\pi(\beta), \ell_{1,2}(\beta), \dots, \ell_{n-1,n}(\beta)) \in \mathfrak{S}_n \times \mathbb{Z}^{\frac{n(n+1)}{2}}$$
For $\ell \in \mathbb{N}$, we put $T_n(S_n;\ell) = \tau\left(B_n(S_n;\ell)\right)$.
  - $u \equiv v$ implies $\tau(\overline{u}) = \tau(\overline{v})$.

- **Fact** : For $\beta \in B_n$ and $\alpha \in S_n$, the template $\tau(\beta) \odot \alpha = \tau(\beta\,\alpha)$ depends only of $\tau(\beta)$ and α.

- **Definition** : For $t \in T_n(S_n;\ell)$ we write
$$B_n(S_n;\ell,t) = \{\beta \in B_n(S_n;\ell) \text{ s.t. } \tau(\beta) = t\}.$$
  - The sets $B_n(S_n;\ell,t)$ provide a partition of $B_n(S_n;\ell)$.

- **Fact** :
$$B_n(S_n;\ell,t) = \bigcup_{\alpha \in S_n} \{\beta \cdot \alpha \text{ with } \beta \in B_n(S_n;\ell-1, t \odot \alpha^{-1})\}$$

Parallelization

Parallelization

- Assume we have stored a representative sets of $B_n(S_n; \ell', t')$ for all $\ell' \leqslant \ell$ and all template $t' \in T_n(S_n; \ell')$.

Parallelization

- Assume we have stored a representative sets of $B_n(S_n; \ell', t')$ for all $\ell' \leqslant \ell$ and all template $t' \in T_n(S_n; \ell')$.
  - On a hard disk typically.

Parallelization

- Assume we have stored a representative sets of $B_n(S_n; \ell', t')$ for all $\ell' \leqslant \ell$ and all template $t' \in T_n(S_n; \ell')$.
  - On a hard disk typically.

- Algorithm (Storing a representative set of $B_n(S_n, \ell, t)$) :

$W_{\ell,t} \leftarrow \emptyset$
$W_{\ell-2,t} \leftarrow \mathtt{Load}(\ell-2, t)$
for $\alpha \in S_n$ **do**
  $t' \leftarrow t \odot \alpha^{-1}$
  $W_{\ell-1,t'} \leftarrow \mathtt{Load}(\ell-1, t')$
  for $u \in W_{\ell-1,t'}$ **do**
    $v \leftarrow u\,\alpha$
    if $v \not\triangleleft W_{\ell-2,t}$ and $v \not\triangleleft W_{\ell,t}$ **then**
      $W_{\ell,t} \leftarrow W_{\ell,t} \sqcup \{v\}$
    end if
  end for
end for
$\mathtt{Save}(W_{\ell,t}, \ell, t)$

Stable bijection

- Definition : A bijection μ of S_n^* is S_n-stable whenever :
  - μ preserves the word length,
  - for all u, v in S_n^* we have $\mu(u) \equiv \mu(v) \Leftrightarrow u \equiv v$
  - for all $u \in S_n^*$, the template $\tau(\overline{\mu(u)})$ depends only of $\tau(\overline{u})$

Stable bijection

- Definition : A bijection μ of S_n^* is S_n-stable whenever :
  - μ preserves the word length,
  - for all u, v in S_n^* we have $\mu(u) \equiv \mu(v) \Leftrightarrow u \equiv v$
  - for all $u \in S_n^*$, the template $\tau(\overline{\mu(u)})$ depends only of $\tau(\overline{u})$

- We have the following diagram

$$\begin{tikzcd}
S_n^\ell \arrow[r, "\mu"] \arrow[d, "\tau"'] & S_n^\ell \arrow[d, "\tau"] \\
T_n(S_n;\ell) \arrow[r, "\mu^T"'] & T_n(S_n;\ell)
\end{tikzcd}$$

Stable bijection

- Definition : A bijection μ of S_n^* is S_n-stable whenever :
  - μ preserves the word length,
  - for all u, v in S_n^* we have $\mu(u) \equiv \mu(v) \Leftrightarrow u \equiv v$
  - for all $u \in S_n^*$, the template $\tau(\overline{\mu(u)})$ depends only of $\tau(\overline{u})$

- We have the following diagram

$$\begin{array}{ccc} S_n^\ell & \xrightarrow{\mu} & S_n^\ell \\ \tau\downarrow & & \downarrow\tau \\ T_n(S_n;\ell) & \xrightarrow[\mu^T]{} & T_n(S_n;\ell) \end{array}$$

- Proposition : If $W \vdash B_n(S_n;\ell,t)$ then $\mu(W) \vdash B_n(S_n;\ell,\mu^T(t))$

Stable bijection

- Definition : A bijection μ of S_n^* is S_n-stable whenever :
  - μ preserves the word length,
  - for all u, v in S_n^* we have $\mu(u) \equiv \mu(v) \Leftrightarrow u \equiv v$
  - for all $u \in S_n^*$, the template $\tau(\overline{\mu(u)})$ depends only of $\tau(\overline{u})$

- We have the following diagram

$$\begin{tikzcd} S_n^\ell \arrow[r, "\mu"] \arrow[d, "\tau"'] & S_n^\ell \arrow[d, "\tau"] \\ T_n(S_n;\ell) \arrow[r, "\mu^T"'] & T_n(S_n;\ell) \end{tikzcd}$$

- Proposition : If $W \vdash B_n(S_n;\ell,t)$ then $\mu(W) \vdash B_n(S_n;\ell,\mu^T(t))$

  ▶ It is sufficient to compute only one of these sets.

Here $S_n = \Sigma_n$ and we fix $\ell \geqslant 2$.

- Definition :

Stable bijections – Artin case

Here $S_n = \Sigma_n$ and we fix $\ell \geqslant 2$.

- Definition :

$$\mathrm{inv}_{\Sigma_n}(x_1 \cdots x_\ell) = (x_\ell^{-1} \cdots x_1^{-1})$$

Stable bijections – Artin case

Here $S_n = \Sigma_n$ and we fix $\ell \geqslant 2$.

- Definition :

$$\mathrm{inv}_{\Sigma_n}(x_1 \cdots x_\ell) = (x_\ell^{-1} \cdots x_1^{-1})$$

$$\mathrm{mir}_{\Sigma_n}(x_1 \cdots x_\ell) = (x_\ell \cdots x_1)$$

Here $S_n = \Sigma_n$ and we fix $\ell \geqslant 2$.

- Definition :

$$\mathrm{inv}_{\Sigma_n}(x_1 \cdots x_\ell) = (x_\ell^{-1} \cdots x_1^{-1})$$

$$\mathrm{mir}_{\Sigma_n}(x_1 \cdots x_\ell) = (x_\ell \cdots x_1)$$

$$\Phi_{\Sigma_n}(x_1 \cdots x_\ell) = (\Phi_n(x_1) \cdots \Phi_m(x_\ell))$$

Stable bijections – Artin case

Here $S_n = \Sigma_n$ and we fix $\ell \geqslant 2$.

- Definition :

$$\mathrm{inv}_{\Sigma_n}(x_1 \cdots x_\ell) = (x_\ell^{-1} \cdots x_1^{-1})$$

$$\mathrm{mir}_{\Sigma_n}(x_1 \cdots x_\ell) = (x_\ell \cdots x_1)$$

$$\Phi_{\Sigma_n}(x_1 \cdots x_\ell) = (\Phi_n(x_1) \cdots \Phi_m(x_\ell)) \qquad \text{where } \Phi_n(\sigma_i^e) = \sigma_{n-i}^e$$

Stable bijections – Artin case

Here $S_n = \Sigma_n$ and we fix $\ell \geqslant 2$.

- Definition :

$$\mathrm{inv}_{\Sigma_n}(x_1 \cdots x_\ell) = (x_\ell^{-1} \cdots x_1^{-1})$$

$$\mathrm{mir}_{\Sigma_n}(x_1 \cdots x_\ell) = (x_\ell \cdots x_1)$$

$$\Phi_{\Sigma_n}(x_1 \cdots x_\ell) = (\Phi_n(x_1) \cdots \Phi_m(x_\ell)) \qquad \text{where } \Phi_n(\sigma_i^e) = \sigma_{n-i}^e$$

▶ They are all Σ_n-stable.

Stable bijections – Artin case

Here $S_n = \Sigma_n$ and we fix $\ell \geqslant 2$.

- Definition :

$$\mathrm{inv}_{\Sigma_n}(x_1 \cdots x_\ell) = (x_\ell^{-1} \cdots x_1^{-1})$$

$$\mathrm{mir}_{\Sigma_n}(x_1 \cdots x_\ell) = (x_\ell \cdots x_1)$$

$$\Phi_{\Sigma_n}(x_1 \cdots x_\ell) = (\Phi_n(x_1) \cdots \Phi_m(x_\ell)) \qquad \text{where } \Phi_n(\sigma_i^e) = \sigma_{n-i}^e$$

  - They are all Σ_n-stable.

- Lemma : The subgroups G_{Σ_n} of bijections of $T_n(\Sigma_n;\ell)$ generated by $\{\mathrm{inv}^T_{\Sigma_n}, \mathrm{mir}^T_{\Sigma_n}, \Phi^T_{\Sigma_n}\}$ is isomorphic to $(\mathbb{Z}/2\mathbb{Z})^3$.

Stable bijections – dual case

Here $S_n = \Sigma_n^*$ and we fix $\ell \geqslant 2$.

- Definition :

Stable bijections – dual case

Here $S_n = \Sigma_n^*$ and we fix $\ell \geqslant 2$.

- Definition :

$$\mathrm{inv}_{\Sigma_n^*}(x_1 \cdots x_\ell) = (x_\ell^{-1} \cdots x_1^{-1})$$

Stable bijections – dual case

Here $S_n = \Sigma_n^*$ and we fix $\ell \geqslant 2$.

- Definition :

$$\mathrm{inv}_{\Sigma_n^*}(x_1 \cdots x_\ell) = (x_\ell^{-1} \cdots x_1^{-1})$$

$$\varphi_{\Sigma_n^*}(x_1 \cdots x_\ell) = (\varphi_n(x_1) \cdots \varphi_n(x_\ell))$$

Stable bijections – dual case

Here $S_n = \Sigma_n^*$ and we fix $\ell \geqslant 2$.

- Definition :

$$\mathrm{inv}_{\Sigma_n^*}(x_1 \cdots x_\ell) = (x_\ell^{-1} \cdots x_1^{-1})$$

$$\varphi_{\Sigma_n^*}(x_1 \cdots x_\ell) = (\varphi_n(x_1) \cdots \varphi_n(x_\ell))$$

where

$$\varphi_n(a_{i,j}^e) = \begin{cases} a_{i+1,j+1}^e & \text{if } j < n, \\ a_{1,i+1}^e & \text{if } j = m. \end{cases}$$

Here $S_n = \Sigma_n^*$ and we fix $\ell \geqslant 2$.

- Definition :

$$\mathrm{inv}_{\Sigma_n^*}(x_1 \cdots x_\ell) = (x_\ell^{-1} \cdots x_1^{-1})$$

$$\varphi_{\Sigma_n^*}(x_1 \cdots x_\ell) = (\varphi_n(x_1) \cdots \varphi_n(x_\ell))$$

where

$$\varphi_n(a_{i,j}^e) = \begin{cases} a_{i+1,j+1}^e & \text{if } j < n, \\ a_{1,i+1}^e & \text{if } j = m. \end{cases}$$

- They are all Σ_n^*-stable.

Stable bijections – dual case

Here $S_n = \Sigma_n^*$ and we fix $\ell \geqslant 2$.

- Definition :

$$\mathrm{inv}_{\Sigma_n^*}(x_1 \cdots x_\ell) = (x_\ell^{-1} \cdots x_1^{-1})$$

$$\varphi_{\Sigma_n^*}(x_1 \cdots x_\ell) = (\varphi_n(x_1) \cdots \varphi_n(x_\ell))$$

  where

$$\varphi_n(a_{i,j}^e) = \begin{cases} a_{i+1,j+1}^e & \text{if } j < n, \\ a_{1,i+1}^e & \text{if } j = m. \end{cases}$$

  - They are all Σ_n^*-stable.
  - No counter part of mir_{Σ_n}.

Stable bijections – dual case

Here $S_n = \Sigma_n^*$ and we fix $\ell \geqslant 2$.

- Definition :

$$\mathrm{inv}_{\Sigma_n^*}(x_1 \cdots x_\ell) = (x_\ell^{-1} \cdots x_1^{-1})$$

$$\varphi_{\Sigma_n^*}(x_1 \cdots x_\ell) = (\varphi_n(x_1) \cdots \varphi_n(x_\ell))$$

where

$$\varphi_n(a_{i,j}^e) = \begin{cases} a_{i+1,j+1}^e & \text{if } j < n, \\ a_{1,i+1}^e & \text{if } j = m. \end{cases}$$

  - They are all Σ_n^*-stable.
  - No counter part of mir_{Σ_n}.

- Lemma : The subgroups $G_{\Sigma_n^*}$ of bijections of $T_n(\Sigma_n^*; \ell)$ generated by $\{\mathrm{inv}_{\Sigma_n^*}^T, \varphi_{\Sigma_n^*}^T\}$ is isomorphic to $\mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/n\mathbb{Z}$.

Template reduction

S_n denotes either Σ_n or Σ_n^*

Template reduction

S_n denotes either Σ_n or Σ_n^*

- Definition : A template $t \in T_n(S_n)$ is reduced if it is *minimal* in $G_{S_n} \star t$.

Template reduction

S_n denotes either Σ_n or Σ_n^*

- Definition : A template $t \in T_n(S_n)$ is reduced if it is *minimal* in $G_{S_n} \star t$.

- Proposition :

$$s(B_n, S_n; \ell) = \sum_{t \in T_n(S_n;\ell)} \mathrm{card}\left(B_n(S_n; \ell, t)\right)$$

$$= \sum_{\substack{t \in T_n(S_n;\ell) \\ t \text{ reduced}}} \mathrm{card}\left(B_n(S_n; \ell, t)\right) \times \mathrm{card}\left(G_{S_n} \star t\right)$$

Template reduction

S_n denotes either Σ_n or Σ_n^*

- Definition : A template $t \in T_n(S_n)$ is reduced if it is *minimal* in $G_{S_n} \star t$.

- Proposition :

$$s(B_n, S_n; \ell) = \sum_{t \in T_n(S_n;\ell)} \operatorname{card}\left(B_n(S_n; \ell, t)\right)$$

$$= \sum_{\substack{t \in T_n(S_n;\ell) \\ t \text{ reduced}}} \operatorname{card}\left(B_n(S_n; \ell, t)\right) \times \operatorname{card}\left(G_{S_n} \star t\right)$$

▶ Can be efficitively used from an algorithmic point of view.

Experimentation

- Implementation : Distribued `C++` code based on a clients / server model.

Experimentation

- Implementation : Distribued C++ code based on a clients / server model.

- Machine : A node of the computationnal plateform Calculco with
  - 256 Go of RAM memory
  - 2 processors with 64 cores each for a total of 128 cores
  - with an access to a distribued storage space of 30 To

Experimentation

- Implementation : Distribued `C++` code based on a clients / server model.

- Machine : A node of the computationnal plateform `Calculco` with
  - 256 `Go` of RAM memory
  - 2 processors with 64 cores each for a total of 128 cores
  - with an access to a distribued storage space of 30 `To`

- Validation of L. Sabalka formulas :

$$\mathcal{S}(B_3,\Sigma_3) = \frac{(t+1)(2t^3-t^2+t-1)}{(t-1)(2t-1)(t^2+t-1)},$$
$$\mathcal{G}(B_3,\Sigma_3) = \frac{t^4+3t^3+t+1}{(t^2+2t-1)(t^2+t-1)}.$$

Three strand – Dual case

Three strand – Dual case

ℓ	$s(B_3, \Sigma_3^*; \ell)$	$g(B_3, \Sigma_3^*; \ell)$	ℓ	$s(B_3, \Sigma_3^*; \ell)$	$g(B_3, \Sigma_3^*; \ell)$
0	1	1	11	38 910	6 639 606
1	6	6	12	83 966	26 216 418
2	20	30	13	180 222	103 827 366
3	54	126	14	385 022	412 169 970
4	134	498	15	819 198	1 639 212 246
5	318	1 926	16	1 736 702	6 528 347 778
6	734	7 410	17	3 670 014	26 027 690 886
7	1 662	28 566	18	7 733 246	103 853 269 650
8	3 710	110 658	19	16 252 926	414 639 810 486
9	8 190	431 046	20	34 078 718	1 656 237 864 738
10	17 918	1 687 890	21	71 303 166	6 617 984 181 606

- Conjecture :

$$\mathcal{S}(B_3, \Sigma_3^*) = \frac{(t+1)(2t^2-1)}{(t-1)(2t-1)^2}, \quad \mathcal{G}(B_3, \Sigma_3^*) = \frac{12t^3 - 2t^2 + 3t - 1}{(2t-1)(3t-1)(4t-1)}.$$

Three strand – Dual case

ℓ	$s(B_3, \Sigma_3^*; \ell)$	$g(B_3, \Sigma_3^*; \ell)$	ℓ	$s(B_3, \Sigma_3^*; \ell)$	$g(B_3, \Sigma_3^*; \ell)$
0	1	1	11	38 910	6 639 606
1	6	6	12	83 966	26 216 418
2	20	30	13	180 222	103 827 366
3	54	126	14	385 022	412 169 970
4	134	498	15	819 198	1 639 212 246
5	318	1 926	16	1 736 702	6 528 347 778
6	734	7 410	17	3 670 014	26 027 690 886
7	1 662	28 566	18	7 733 246	103 853 269 650
8	3 710	110 658	19	16 252 926	414 639 810 486
9	8 190	431 046	20	34 078 718	1 656 237 864 738
10	17 918	1 687 890	21	71 303 166	6 617 984 181 606

- Conjecture :

$$\mathcal{S}(B_3, \Sigma_3^*) = \frac{(t+1)(2t^2-1)}{(t-1)(2t-1)^2}, \quad \mathcal{G}(B_3, \Sigma_3^*) = \frac{12t^3 - 2t^2 + 3t - 1}{(2t-1)(3t-1)(4t-1)}.$$

Three strand – Dual case

ℓ	$s(B_3,\Sigma_3^*;\ell)$	$g(B_3,\Sigma_3^*;\ell)$	ℓ	$s(B_3,\Sigma_3^*;\ell)$	$g(B_3,\Sigma_3^*;\ell)$
0	1	1	11	38 910	6 639 606
1	6	6	12	83 966	26 216 418
2	20	30	13	180 222	103 827 366
3	54	126	14	385 022	412 169 970
4	134	498	15	819 198	1 639 212 246
5	318	1 926	16	1 736 702	6 528 347 778
6	734	7 410	17	3 670 014	26 027 690 886
7	1 662	28 566	18	7 733 246	103 853 269 650
8	3 710	110 658	19	16 252 926	414 639 810 486
9	8 190	431 046	20	34 078 718	1 656 237 864 738
10	17 918	1 687 890	21	71 303 166	6 617 984 181 606

- Conjecture :

$$\mathcal{S}(B_3,\Sigma_3^*) = \frac{(t+1)(2t^2-1)}{(t-1)(2t-1)^2}, \quad \mathcal{G}(B_3,\Sigma_3^*) = \frac{12t^3-2t^2+3t-1}{(2t-1)(3t-1)(4t-1)}.$$

  - With growth rates of 2 and 4 respectively.

Four strands - Artin case

Four strands - Artin case

ℓ	$s(B_4, \Sigma_4; \ell)$	$g(B_4, \Sigma_4; \ell)$	ℓ	$s(B_4, \Sigma_4; \ell)$	$g(B_4, \Sigma_4; \ell)$
0	1	1	13	9 007 466	281 799 158
1	6	6	14	27 218 486	1 153 638 466
2	26	30	15	82 133 734	4 710 108 514
3	98	142	16	247 557 852	19 186 676 438
4	338	646	17	745 421 660	78 004 083 510
5	1 110	2 870	18	2 242 595 598	316 591 341 866
6	3 542	12 558	19	6 741 618 346	1 283 041 428 650
7	11 098	54 026	20	20 252 254 058	5 193 053 664 554
8	34 362	229 338	21	60 800 088 680	20 994 893 965 398
9	105 546	963 570	22	182 422 321 452	84 795 261 908 498
10	322 400	4 016 674	23	547 032 036 564	342 173 680 884 002
11	980 904	16 641 454	24	1 639 548 505 920	1 379 691 672 165 334
12	2 975 728	68 614 150	25	4 911 638 066 620	5 559 241 797 216 166

Four strands - Artin case

ℓ	$s(B_4,\Sigma_4;\ell)$	$g(B_4,\Sigma_4;\ell)$	ℓ	$s(B_4,\Sigma_4;\ell)$	$g(B_4,\Sigma_4;\ell)$
0	1	1	13	9 007 466	281 799 158
1	6	6	14	27 218 486	1 153 638 466
2	26	30	15	82 133 734	4 710 108 514
3	98	142	16	247 557 852	19 186 676 438
4	338	646	17	745 421 660	78 004 083 510
5	1 110	2 870	18	2 242 595 598	316 591 341 866
6	3 542	12 558	19	6 741 618 346	1 283 041 428 650
7	11 098	54 026	20	20 252 254 058	5 193 053 664 554
8	34 362	229 338	21	60 800 088 680	20 994 893 965 398
9	105 546	963 570	22	182 422 321 452	84 795 261 908 498
10	322 400	4 016 674	23	547 032 036 564	342 173 680 884 002
11	980 904	16 641 454	24	1 639 548 505 920	1 379 691 672 165 334
12	2 975 728	68 614 150	25	4 911 638 066 620	5 559 241 797 216 166

- No good conjectures.

Four strands - Artin case

ℓ	$s(B_4,\Sigma_4;\ell)$	$g(B_4,\Sigma_4;\ell)$	ℓ	$s(B_4,\Sigma_4;\ell)$	$g(B_4,\Sigma_4;\ell)$
0	1	1	13	9 007 466	281 799 158
1	6	6	14	27 218 486	1 153 638 466
2	26	30	15	82 133 734	4 710 108 514
3	98	142	16	247 557 852	19 186 676 438
4	338	646	17	745 421 660	78 004 083 510
5	1 110	2 870	18	2 242 595 598	316 591 341 866
6	3 542	12 558	19	6 741 618 346	1 283 041 428 650
7	11 098	54 026	20	20 252 254 058	5 193 053 664 554
8	34 362	229 338	21	60 800 088 680	20 994 893 965 398
9	105 546	963 570	22	182 422 321 452	84 795 261 908 498
10	322 400	4 016 674	23	547 032 036 564	342 173 680 884 002
11	980 904	16 641 454	24	1 639 548 505 920	1 379 691 672 165 334
12	2 975 728	68 614 150	25	4 911 638 066 620	5 559 241 797 216 166

- No good conjectures.
- The storage of all braids of B_4 with geodesic Σ_4-length $\leqslant 25$

Four strands - Artin case

ℓ	$s(B_4, \Sigma_4; \ell)$	$g(B_4, \Sigma_4; \ell)$	ℓ	$s(B_4, \Sigma_4; \ell)$	$g(B_4, \Sigma_4; \ell)$
0	1	1	13	9 007 466	281 799 158
1	6	6	14	27 218 486	1 153 638 466
2	26	30	15	82 133 734	4 710 108 514
3	98	142	16	247 557 852	19 186 676 438
4	338	646	17	745 421 660	78 004 083 510
5	1 110	2 870	18	2 242 595 598	316 591 341 866
6	3 542	12 558	19	6 741 618 346	1 283 041 428 650
7	11 098	54 026	20	20 252 254 058	5 193 053 664 554
8	34 362	229 338	21	60 800 088 680	20 994 893 965 398
9	105 546	963 570	22	182 422 321 452	84 795 261 908 498
10	322 400	4 016 674	23	547 032 036 564	342 173 680 884 002
11	980 904	16 641 454	24	1 639 548 505 920	1 379 691 672 165 334
12	2 975 728	68 614 150	25	4 911 638 066 620	5 559 241 797 216 166

- No good conjectures.
- The storage of all braids of B_4 with geodesic Σ_4-length $\leqslant 25$ requires 26 To of disk space.

Four strands – dual case

Four strands – dual case

ℓ	$s(B_4, \Sigma_4^*; \ell)$	$g(B_4, \Sigma_4^*; \ell)$	ℓ	$S(B_4, \Sigma_4^*; \ell)$	$g(B_4, \Sigma_4^*; \ell)$
0	1	1	9	7 348 366	708 368 540
1	12	12	10	35 773 324	6 128 211 364
2	84	132	11	173 885 572	52 826 999 612
3	478	1 340	12	844 277 874	454 136 092 148
4	2 500	12 788	13	4 095 929 948	3 895 624 824 092
5	12 612	117 452	14	19 858 981 932	33 359 143 410 468
6	62 570	1 053 604	15	96 242 356 958	285 259 736 104 444
7	303 356	9 311 420	16	466 262 144 180	2 436 488 694 821 748
8	1 506 212	81 488 628	17	2 258 320 991 652	20 790 986 096 580 060

Four strands – dual case

ℓ	$s(B_4, \Sigma_4^*; \ell)$	$g(B_4, \Sigma_4^*; \ell)$	ℓ	$S(B_4, \Sigma_4^*; \ell)$	$g(B_4, \Sigma_4^*; \ell)$
0	1	1	9	7 348 366	708 368 540
1	12	12	10	35 773 324	6 128 211 364
2	84	132	11	173 885 572	52 826 999 612
3	478	1 340	12	844 277 874	454 136 092 148
4	2 500	12 788	13	4 095 929 948	3 895 624 824 092
5	12 612	117 452	14	19 858 981 932	33 359 143 410 468
6	62 570	1 053 604	15	96 242 356 958	285 259 736 104 444
7	303 356	9 311 420	16	466 262 144 180	2 436 488 694 821 748
8	1 506 212	81 488 628	17	2 258 320 991 652	20 790 986 096 580 060

- Conjecture :

$$\mathcal{S}(B_4, \Sigma_4^*) = -\frac{(t+1)(10t^6 - 10t^5 - 3t^4 + 11t^3 - 4t^2 - 3t + 1)}{(t-1)(5t^2 - 5t + 1)(10t^4 - 20t^3 + 19^2 - 8t + 1)}$$

Four strands – dual case

ℓ	$s(B_4,\Sigma_4^*;\ell)$	$g(B_4,\Sigma_4^*;\ell)$	ℓ	$S(B_4,\Sigma_4^*;\ell)$	$g(B_4,\Sigma_4^*;\ell)$
0	1	1	9	7 348 366	708 368 540
1	12	12	10	35 773 324	6 128 211 364
2	84	132	11	173 885 572	52 826 999 612
3	478	1 340	12	844 277 874	454 136 092 148
4	2 500	12 788	13	4 095 929 948	3 895 624 824 092
5	12 612	117 452	14	19 858 981 932	33 359 143 410 468
6	62 570	1 053 604	15	96 242 356 958	285 259 736 104 444
7	303 356	9 311 420	16	466 262 144 180	2 436 488 694 821 748
8	1 506 212	81 488 628	17	2 258 320 991 652	20 790 986 096 580 060

- Conjecture :

$$\mathcal{S}(B_4,\Sigma_4^*) = -\frac{(t+1)(10t^6-10t^5-3t^4+11t^3-4t^2-3t+1)}{(t-1)(5t^2-5t+1)(10t^4-20t^3+19^2-8t+1)}$$

- ▶ The growth rate of $\mathcal{S}(B_4,\Sigma_4^*)$ is $\simeq 4.8$.

Four strands – dual case

ℓ	$s(B_4, \Sigma_4^*; \ell)$	$g(B_4, \Sigma_4^*; \ell)$	ℓ	$S(B_4, \Sigma_4^*; \ell)$	$g(B_4, \Sigma_4^*; \ell)$
0	1	1	9	7 348 366	708 368 540
1	12	12	10	35 773 324	6 128 211 364
2	84	132	11	173 885 572	52 826 999 612
3	478	1 340	12	844 277 874	454 136 092 148
4	2 500	12 788	13	4 095 929 948	3 895 624 824 092
5	12 612	117 452	14	19 858 981 932	33 359 143 410 468
6	62 570	1 053 604	15	96 242 356 958	285 259 736 104 444
7	303 356	9 311 420	16	466 262 144 180	2 436 488 694 821 748
8	1 506 212	81 488 628	17	2 258 320 991 652	20 790 986 096 580 060

- Conjecture :

$$\mathcal{S}(B_4, \Sigma_4^*) = -\frac{(t+1)(10t^6 - 10t^5 - 3t^4 + 11t^3 - 4t^2 - 3t + 1)}{(t-1)(5t^2 - 5t + 1)(10t^4 - 20t^3 + 19^2 - 8t + 1)}$$

- ▶ The growth rate of $\mathcal{S}(B_4, \Sigma_4^*)$ is $\simeq 4.8$.
- ▶ No good conjecture for $\mathcal{G}(B_4, \Sigma_4^*)$.

Thank you !