Journal of Knot Theory and Its Ramifications

A NORMAL FORM FOR BRAIDS

XAVIER BRESSAUD

Institut de Mathématiques de Luminy, Case 907
163, avenue de Luminy, 13288 Marseille Cedex 9, France
bressaud@iml.univ-mrs.fr

ABSTRACT

We present a seemingly new normal form for braids, where every braid is expressed using a word in a regular language on some simple alphabet of elementary braids. This normal form stems from analysing the geometric action of braid groups on curves in a punctured disk.

Keywords: Braid groups. Normal form. Action of braid groups on free groups. Dehornoy ordering. Symbolic dynamics

Mathematics Subject Classification 2000: 57M25, 57M27

1. Introduction

For every $n \geq 2$, let B_n be the braid group with n strands. There exist several distinguished normal forms for braids, in particular the well-known greedy normal form(s) based on Garside's theory, which consists in expressing every braid as a fraction involving one positive braid (no σ_i^{-1}) and one negative braid. Here we propose a new normal form relying on a different principle, and, in particular, the positive and negative factors are not separated. This normal form relies on the geometric idea of relaxing curves in a punctured disk but it can be presented in purely combinatorial terms. Let us consider the elementary braids (see Figure 3) : $\delta_{i,j} = \sigma_i\sigma_{i+1}\cdots\sigma_{j-1}$, for $1 \leq i < j \leq n$, and $\delta_{i,j} = \delta_{j,i}^{-1} = \sigma_{i-1}^{-1}\sigma_{i-2}^{-1}\cdots\sigma_j^{-1}$, for $j < i$. For n fixed, we define the alphabet $\mathcal{A}_n = \{\delta_{i,n}, \delta_{i+1,1},\ i = 1,\ldots,n-1\}$, and the language of finite type $\mathcal{L}_n$ on $\mathcal{A}_n$ where the words $\delta_{i,j}\delta_{k,l}$ are allowed only if k is equal to i or between i and j (see Figures 4 and 5). Let us call $\mathcal{T}_n$ the set of braids having a representative in $\mathcal{L}_n$ (we shall see that such a representative is then unique). Not all braids can be written as words in $\mathcal{L}_n$, but we claim that every braid τ in B_n can be decomposed in a product $\tau = \tau_n\tau_{n-1}\cdots\tau_2$, where, for all $2 \leq p \leq n$, $\tau_p \in \mathcal{T}_p$. The resulting writing of τ is a normal form. It is the unique representative of τ in a regular language denoted $\mathcal{L}_{\leq n}$.

Examples. To give a rough idea of how things work, let us have a look at some examples. Consider for instance the braid $\tau = \sigma_1^2\sigma_2^{-2}\sigma_3^2\sigma_1^{-2}$. As it is easy to check

(see Figure 12), this braid can be written

$$\tau = \delta_{3,1}\delta_{2,4}\delta_{2,1}\delta_{2,4}\delta_{3,1}\delta_{3,4} \in \mathcal{L}_4,$$

showing it belongs to $\mathcal{T}_4$. Next, consider the braids $(\sigma_1\sigma_2^{-1})^N$ when N varies. As shown graphically on Figure 1, the normal expression of $\sigma_1\sigma_2^{-1} = \delta_{1,2}\delta_{3,2}$ is $\delta_{3,1}\delta_{2,3}.\delta_{1,2}$, with $\delta_{3,1}\delta_{2,3} \in \mathcal{L}_3$ and $\delta_{1,2} \in \mathcal{L}_2$. Thus, for all integers $N > 1$,

$$(\sigma_1\sigma_2^{-1})^N = \delta_{3,1}(\delta_{2,1}\delta_{2,3})^{N-1}\delta_{2,3}.\delta_{1,2}.$$

In this case, the normal form is the concatenation of $\delta_{3,1}(\delta_{2,1}\delta_{2,3})^{N-1}\delta_{2,3} \in \mathcal{L}_3$ and $\delta_{1,2} \in \mathcal{L}_2$.

Fig. 1. Graphical representation : $(\sigma_1\sigma_2^{-1})^3 = \delta_{3,1}(\delta_{2,1}\delta_{2,3})^2\delta_{2,3}.\delta_{1,2}$.

Geometrical presentation. This normal form is inspired by a more geometrical point of view. It is to be related to the strategy of relaxing the curve diagram developed in [4] and with the work [6] of Larue. To be more specific, let us consider the standard action of the braid group on the fundamental group $F_n = \langle u_1, \ldots, u_n \rangle$ of the punctured disk. A braid is completely determined by the images of a set of elementary loops. Obvious geometrical considerations show that images of elementary loops are loops with no self intersections. Basically we are going to deloop the images of these loops according to a specific procedure. The delooping strategy we propose is simple. Let w be a simple loop and write $w = u_{i_1}^{\epsilon_1} \cdots u_{i_N}^{\epsilon_N}$ in reduced form. The recursive step consists in applying either δ_{1,i_1} if $\epsilon_1 < 0$ or δ_{n,i_1} if $\epsilon_1 > 0$. This is natural from a geometric point of view (see for instance Figure 8). We claim that, on the one hand, repeated use of this step eventually yields a trivial loop and that, on the other hand, the sequence of inverses of the elementary delooping braids is in $\mathcal{L}_n$. This delooping procedure yields the normal form inductively. Indeed, consider a braid $\tau \in B_n$ and let $w = \tau(u_n)$ be the image of the elementary loop u_n. The procedure provides a delooping braid for w, that is a braid $\tau^{(n)}$ such that $\tau^{(n)}(w) = u_n$. Now, $\tau^{(n)} \circ \tau$ is in B_{n-1} because it leaves u_n invariant. Applying the same procedure in B_{n-1}, we can find the delooping braid of $\tau^{(n)} \circ \tau(u_{n-1})$, and so on and so forth. Finally, we obtain $n-1$ braids $\tau^{(n)}, \ldots, \tau^{(2)}$ (with $\tau^{(k)} \in B_k \subset B_n$)

such that $\tau^{(2)} \circ \tau^{(3)} \circ \cdots \circ \tau^{(n)} \circ \tau = Id$. The normal form of τ is obtained writing the inverses of these delooping braids in terms of elementary braids.

Idea of the proof. The proof we propose is based on a combinatorial presentation of this delooping procedure. To manipulate these objects in a combinatorial setting, the key point is to understand the action of elementary braids on words which code loops. In the computation of the image of a word, each letter is replaced with a word (substitution) ; then, the result is put in reduced form ; at this stage, some cancellations may occur. The technical point is hence to control where these cancellations occur : this is the object of Lemma 2. This analysis together with a few remarks on the coding of simple loops (Lemma 3) shows that, while delooping, the (combinatorial) length of the loop reduces (Lemma 4). It is then easy to prove recursively that the loop eventually becomes trivial and it is straightforward to see that the sequence (of the inverses of) the elementary braids needed to deloop w is in $\mathcal{L}_n$ (Proposition 1). The argument to prove uniqueness of the normal form of a braid relies on another consequence of Lemma 2 : we show (Proposition 2) that the beginning of the image of an elementary loop by a braid in normal form is determined by the first letter of the normal form.

Connection with the Dehornoy ordering. Our normal form seems to be adapted to tackling questions about the Dehornoy ordering of braids. It is known that every nontrivial braid admits an expression by a σ-word in which the generator with higher index appears only positively, or only negatively. The latter property, called Comparison Property in [4], is the key point in the construction of the above ordering of braids. Our normal form easily provides a variant (Theorem 2 and Corollary 1) of this property and hence yields a new effective proof for the existence of this ordering.

Algorithm. To compute the normal form of a given braid, one can compute the images of the elementary loops and compute algorithmically the delooping braids. But it is also possible to avoid the study of the action of the braid group on the free group. Indeed, we propose simple rewriting rules which allow one to compute the normal form of a braid from an expression for it in terms of the standard generators; this is based solely on the subshift point of view. We note that proving the correctness and termination of the algorithm yields another, purely combinatorial, proof of the existence of the normal form.

Perspectives. We conclude with some remarks about algorithmic properties of the normal form and several projects related to our initial motivations.

The picture. Most results are presented here in combinatorial terms. Since intuition is often geometric, we illustrate the statements with diagram representations.

In an appendix we give a more detailed description of the way things can be presented in a geometric setting. We conclude with an analysis of some geometric properties of the normal form and its relationship with an efficient coding of simple loops which are in the orbit of elementary loops.

2. Definitions, notations and statement of the main result

2.1. *Braid groups*

For all $n \geq 2$, let B_n be the braid group with n strands. The standard presentation using the set $G = \{\sigma_1, \ldots, \sigma_{n-1}\}$ of standard generators is

$$B_n = \langle \sigma_1, \ldots \sigma_{n-1}; \sigma_i\sigma_j = \sigma_j\sigma_i, |i-j| > 1; \sigma_i\sigma_{i+1}\sigma_i = \sigma_{i+1}\sigma_i\sigma_{i+1} \rangle .$$

This group has a classical interpretation in terms of homotopy classes of strands

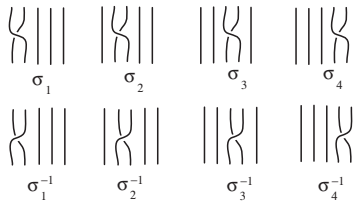

Fig. 2. The generators σ_i for B_5

between two disks. For our purposes, we will simply formally identify the generators with pictograms following Figure 2. Concatenation is compatible with this representation and the relations have intuitive graphical interpretation. The group has also an interpretation in terms of homotopy classes of homeomorphisms of punctured disks. In this context, the action of the group on the free group we will define and study in Section 3 is naturally interpreted as an action on (homotopy classes of) simple loops. The correspondence is intuitive and pictures are intended to help intuition while most results are stated and proved in combinatorial terms. We will make this interpretation more specific and prove more geometric properties of the normal form in Appendix A.

Remark 1. We stress that, we will represent $\sigma_i\sigma_j$ as σ_i above σ_j. With this convention, the left action of the group on the free group we are going to define will read graphically from bottom to top (see Figure 7).

As natural in the context of finitely presented groups, we will often identify words in $(G \cup G^{-1})^*$ and elements of the group. We must keep in mind that, in fact, an element of the group is an equivalence class of such words. We define the σ-length

of a braid to be the minimal (word) length of a word in the class. We note that for $p < n$, B_p is a subgroup of B_n.

The quotient of B_n by the relations $\{\sigma_i^2 = 1, i = 1, \ldots, n\}$ is the symmetric group $\mathfrak{S}_n$ (= group of permutations of $\{1, \ldots, n\}$). We will denote by $s_g \in \mathfrak{S}_n$ the equivalence class of $g \in B_n$. For instance, s_{σ_i} denotes the transposition $(i, i+1)$.

2.2. *Elementary braids*

For $1 \leq i < j \leq n$ integers, we will use the notation,

$$\delta_{i,j} = \sigma_i \sigma_{i+1} \cdots \sigma_{j-1},$$

and, for $j < i$,

$$\delta_{i,j} = \delta_{j,i}^{-1} = \sigma_{i-1}^{-1} \sigma_{i-2}^{-1} \cdots \sigma_j^{-1}.$$

The geometrical operation is to take strand i and put it (over the others) at place j, shifting everything in between, as shown on Figure 3. From the point of view of the left action we will define, the move is reversed. We will refer to these braids as *elementary braids* and denote by $\mathcal{E}_n$ the set of elementary braids of B_n.

Fig. 3. The $\delta_{i,j}$

For $W = \delta_{i,j}$, we write $\mathbf{i}(W) = i$, $\mathbf{j}(W) = j$ and $\xi(W)$ for the sign of $j - i$. We denote by $[i, j]$ the set of integers between i and j. That is if $i < j$ we denote by $[i, j]$ the set $\{i, i+1, \ldots, j\}$ and if $i > j$ the set $\{j, \ldots, i\}$. We put $[i, j[= [i, j] \setminus \{j\}$. We call *range* of $\delta_{i,j}$ and denote $[\delta_{i,j}]$ the set $[i, j]$. In the same spirit, the *strict range* of $\delta_{i,j}$ is the interval $[\delta_{i,j}[= [i, j[$.

We will consider words on the alphabet $\{\delta_{i,j}, 1 \leq i \neq j \leq n\}$. For such a word $W = W_1 \cdots W_K$, and $1 \leq k \leq K$, we write $\mathbf{i}_k(W) = \mathbf{i}(W_k)$, $\mathbf{j}_k(W) = \mathbf{j}(W_k)$ and $\xi_k(W) = \xi(W_k)$. We extend the notation for the range to words taking the range of the last letter : $[W_1 \cdots W_K] = [W_K]$.

2.3. *A language for each level*

Assume n is fixed. We consider the alphabet $\mathcal{A}_n = \{\delta_{i,n}, \delta_{i+1,1}, i = 1, \ldots, n-1\} \subset (G \cup G^{-1})^*$. We denote by $\mathcal{L}_n$ the language :

$$\mathcal{L}_n = \{W \in \mathcal{A}_n;\, \mathbf{i}(W_{k+1}) \in [\mathbf{i}(W_k), \mathbf{j}(W_k)[\}\,.$$

The restriction means that $\mathbf{i}(W_{k+1})$ lies between $\mathbf{i}(W_k)$ and $\mathbf{j}(W_k)$, being different from $\mathbf{j}(W_k)$. We note this by $\mathbf{i}(W_{k+1}) \in [W_k[$.

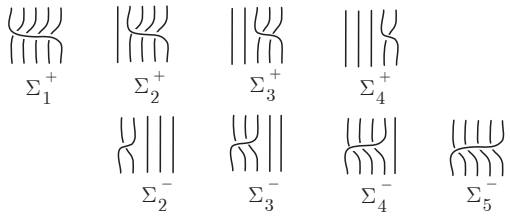

Fig. 4. The alphabet $\mathcal{A}_5$

Remark 2. A word $W = \delta_{i_1,j_1} \cdots \delta_{i_N,j_N}$ is in $\mathcal{L}_n$ if and only if for all $1 \leq k \leq N$, $j_k \in \{1, n\}$, $i_k \neq j_k$ (so that $\delta_{i_k,j_k} \in \mathcal{A}_n$) and, for all $1 \leq k < N$, $1 < i_{k+1} \leq i_k$ if $j_k = 1$ and $i_k \leq i_{k+1} < n$ if $j_k = n$, or equivalently, $i_{k+1} \in [i_k, j_k[$ (so that $W_k W_{k+1} \in \mathcal{L}_n$).

Remark 3. Considering the subset of set of $\mathcal{A}_n^2$,

$$L = \{W_1 W_2;\ \mathbf{i}(W_1) \leq \mathbf{i}(W_2) < n = \mathbf{j}(W_1) \text{ or } \mathbf{j}(W_1) = 1 < \mathbf{i}(W_2) \leq \mathbf{i}(W_1)\},$$

we see that $\mathcal{L}_n$ is a language of finite type : $\mathcal{L}_n = \{W \in \mathcal{A}_n;\ W_k W_{k+1} \in L\}$.

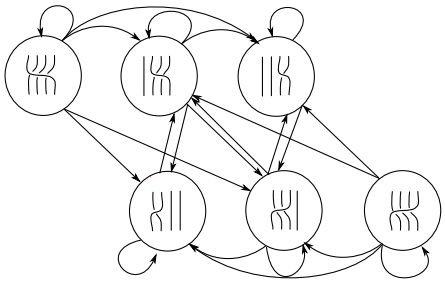

Fig. 5. The subshift for $\mathcal{L}_4$.

2.4. *The complete language*

For every integer n we define a language $\mathcal{L}_{\leq n}$ as follows. A word W is in $\mathcal{L}_{\leq n}$ if it is a concatenation of $n-1$ words $W^{(p)} \in \mathcal{L}_p$, $p = 2, \dots, n$:

$$W = W^{(n)} \cdots W^{(3)} W^{(2)}.$$

Note that $W^{(p)}$ may be empty for some p. Note that the decomposition may not be unique since some letters (and even words) could occur at the end of $W^{(p)}$ or at the beginning of $W^{(p-1)}$. We say that the splitting is *good* if for every p with $W^{(p)}$ non empty, p is in the range of $W^{(p)}$, or equivalently the last letter of $W^{(p)}$ does not belong to $\mathcal{A}_{p-1}$.

Remark 4. It turns out that the language $\mathcal{L}_{\leq n}$ is sofic. To see this properly, let us propose the construction of a language using a bigger alphabet which is of finite type and obviously projects on $\mathcal{L}_{\leq n}$. We consider the alphabet

$$A_n = \left\{(i,j,p) \in \{1,\dots,n\}^3 : i \in \{1,p\}; j \in \{1,\dots,p\} \setminus \{i\}\right\}.$$

For all $a = (i,j,p) \in A$, we set $\Sigma_a = \delta_{i,j}^{(p)} := \delta_{i,j} \in \mathcal{A}_p$. A word $(i_k, j_k, p_k)_{1 \leq k \leq N} \in A_n^*$ is in $\widehat{\mathcal{L}}_{\leq n}$ if and only if for all $1 \leq k < N$, either

$$\begin{cases} p_{k+1} = p_k \\ i_{k+1} \in [i_k, j_k[, \end{cases} \text{ or } \begin{cases} p_{k+1} < p_k \\ p_k \in \{i_k; j_k\} \end{cases}.$$

We say that a word W is in $\mathcal{L}_{\leq}$ if there is an integer n such that W is in $\mathcal{L}_{\leq n}$. For $W \in \mathcal{L}_{\leq}$, we denote $n^*(W) = \min\{m \geq 2 : W \in \mathcal{L}_{\leq m}\}$ and $|W|$ its word length.

2.5. *Normal form*

Our main result provides a normal form for all braids in B_n.

Theorem 1. *Let $\tau \in B_n$. There is a unique word $W = W_1 \cdots W_K \in \mathcal{L}_{\leq n}$ such that*

$$\tau = W_1 \cdots W_K.$$

The proof relies on the understanding of the action of τ on F_n. It is given in Section 4.

2.6. *Notation*

It will be convenient, when n is fixed, to write :

$$\Sigma_i^+ = \delta_{i,n}, \text{ and, } \Sigma_i^- = \delta_{i,1},$$

and, for their inverses,

$$\overline{\Sigma}_i^+ = \delta_{n,i}, \text{ and, } \overline{\Sigma}_i^- = \delta_{1,i}.$$

The alphabet becomes $\mathcal{A}_n = \{\Sigma_i^+, \Sigma_{i+1}^-,\ i = 1, \dots, n-1\}$. The letters are different but correspond to the same braids. The set L can be written as

$$L = \{\Sigma_i^+ \Sigma_j^\epsilon;\ i \leq j < n,\ \epsilon = \pm\} \cup \{\Sigma_i^- \Sigma_j^\epsilon;\ 1 < j \leq i,\ \epsilon = \pm\},$$

and the language $\mathcal{L}_n$ is still $\mathcal{L}_n = \{W \in \mathcal{A}_n;\ W_k W_{k+1} \in L\}$.

To stress the use of this notation, we will use I and Ξ to denote $I(\Sigma_i^\epsilon) = i$ and $\Xi(\Sigma_i^\epsilon) = \epsilon$; the correspondence is, if W and Σ are the same, $I(\Sigma) = \mathbf{i}(W)$ and $\Xi(\Sigma) = \xi(W)$. For the strict range, we notice that $[\Sigma_i^\epsilon[$ is the interval $[i, n[$ if $\epsilon > 0$ and $]1, i]$ if $\epsilon < 0$.

Remark 5. We immediately notice that Σ_1^+ and Σ_n^- play a particular role. They can only appear in the beginning of a word. It could be convenient to work with the alphabet $A = \{\Sigma_i^+, \Sigma_i^-,\ i = 2, \dots, n-1\}$ and allow occurrences of Σ_1^+ or Σ_n^- at the beginning of a word to describe the language.

3. Action of B_n on F_n

We denote F_n the free group with n generators. We consider the set of free generators $S^+ = \{u_1, u_2, \dots u_n\}$. We denote S^- the set of inverses $S^- = \{u_1^{-1}, \dots u_n^{-1}\}$ and $S = S^+ \cup S^-$. To a word in S^* we associate an element of F_n. Conversely, every element of F_n has a unique expression as a word in S^* in reduced form (i.e. a word in S^* with no factor $u_i^\epsilon u_i^{-\epsilon}$). For $w \in F_n$, we denote $|w|$ the word length of its representative in reduced form. If $w = u_{i_1}^{\epsilon_1} \cdots u_{i_{|w|}}^{\epsilon_{|w|}}$ is in reduced form, we write, for all $1 \leq k \leq |w|$, $i_k(w) = i_k$ and $\epsilon_k(w) = \epsilon_k$.

An action of B_n on the free group F_n can be expressed as follows. We define a morphism $\pi : B_n \to \text{Aut } F_n$ by the images of the generators. For every i, $\pi(\sigma_i)$ acts as,

$$\begin{cases} \pi(\sigma_i)(u_i) &= u_i u_{i+1} u_i^{-1}, \\ \pi(\sigma_i)(u_{i+1}) &= u_i, \\ \pi(\sigma_i)(u_j) &= u_j, \forall i \neq j, i+1 \neq j. \end{cases}$$

For $\tau \in B_n$, we will write, $\tau(u_k)$ instead of $\pi(\tau)(u_k)$ if it is not ambiguous. In fact, we will even often write $W(u)$ with $W \in \mathcal{A}_n^*$ to denote $\pi(\tau(u))$ where τ is the braid associated to the word W. For a geometric interpretation of this action, we will think of F_n as the fundamental group of the punctured disk (see Figure 6). For a formal explanation, we refer to Appendix A. For now, we stress that the graphical interpretation of a word must be written from top to bottom as can be seen in Figure 7, so the (left) action on loops reads from bottom to top. Notice that given the pictogram of an elementary braid, we also can read the (left) action of its inverse, from top to bottom (Figure 8). Finally, notice that on most pictures, loops are replaced by separatrices : we used the obvious correspondence formally stated in Lemma 11 to simplify the pictures.

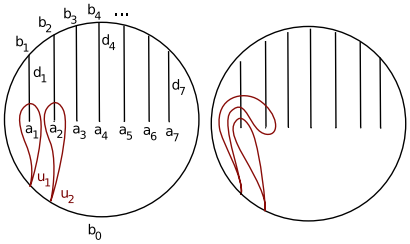

Fig. 6. Action of σ_1 on elementary loops.

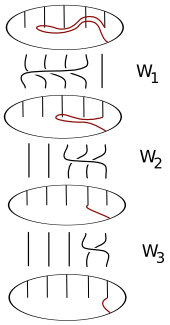

Fig. 7. Graphical representation of $W = W_1W_2W_3$ and left action on the elementary loop u_5 : $W_1W_2W_3(u_5) = u_4^{-1}u_2u_4u_5u_4^{-1}u_2^{-1}u_4$

We let $s \in \mathfrak{S}_n$ act naturally on F_n by $s(u_i) = u_{s(i)}$.

3.1. *The Artin representation*

We can associate to a braid $\tau \in B_n$ the n-tuple of images of the generators of the free group :

$$\psi : \tau \longrightarrow (\tau(u_1), \ldots, \tau(u_n)).$$

The point is that this object completely determines τ. Indeed, if two braids are different, then at least one of the generators have different image. This presentation of braid group is called *Artin presentation* of B_n. The next lemma states that the representation π is faithful.

Lemma 1. *The map ψ is an injection*

We do not give a proof of this standard result. We refer to [2] or to [4], Chapter 5. Notice that it is immediate in view of the definition of the braid group as homotopy classes of homeomorphisms of the punctured disk, since images of a base of the fundamental group characterizes image of all loops and hence the braid.

3.2. *Action of elementary braids*

We investigate the action of the elementary braids on the free group. It is straightforward to show that for $i < j$,

$$\delta_{i,j}(u_k) = \begin{cases} u_k & \text{if } k < i \\ u_i u_{k+1} u_i^{-1} & \text{if } i \leq k < j \\ u_i & \text{if } k = j \\ u_k & \text{if } k > j, \end{cases}$$

and for $j < i$,

$$\delta_{i,j}(u_k) = \begin{cases} u_k & \text{if } k < j \\ u_i & \text{if } k = j \\ u_i^{-1} u_{k-1} u_i & \text{if } j < k \leq i \\ u_k & \text{if } k > i. \end{cases}$$

Remark 6. We summarize, putting $\epsilon = \text{sign}(j - i)$,

$$\delta_{i,j}(u_k) = \begin{cases} u_k & \text{if } k \notin [i, j] \\ u_i^{\epsilon} u_{k+\epsilon} u_i^{-\epsilon} & \text{if } k \in [i, j[\\ u_i & \text{if } k = j. \end{cases}$$

Figure 8 shows a the geometrical interpretation of the action of such a braid (from top to bottom).

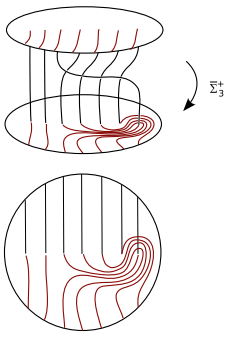

Fig. 8. Action of $\overline{\Sigma}_3^+$. This is a representation of Σ_3^+ so the action reads from top to bottom.

Now we investigate the action of the $\{\delta_{i,j}\}$ on F_n. The point is to understand where the cancellations occur. Fix $w \in F_n$ and write $w = u_{i_1}^{\epsilon_1} \cdots u_{i_N}^{\epsilon_N}$ in reduced form. We are going to give an expression of $\delta_{i,j}(w)$ in reduced form. We work with $i < j$ but the same result holds if $i > j$.

For this purpose, we decompose w in blocks according to $i_k \notin [i, j]$, $i_k = j$ and $i_k \in [i, j[$.

$$w = u_j^{\eta_1} V_1 u_j^{\eta_2} V_2 u_j^{\eta_3} V_3 \cdots u_j^{\eta_{R-1}} V_{R-1} u_j^{\eta_R}, \tag{3.1}$$

where either $V_p \in \langle u_1, \ldots, u_{i-1}, u_{j+1}, \ldots, u_n \rangle$ (block of type 0), or $V_p \in \langle u_i, \ldots, u_{j-1} \rangle$ (block of type 1) and $\eta_p \in \mathbb{Z}$. Notice that some η_p may be equal to 0. We denote $T(p)$ the type of the block V_p. We set $T(0) = T(R) = 0$. If the V_p are written in reduced form, then it is easy to recover the reduced form of w, erasing the $u_j^{\eta_p}$ whenever $\eta_p = 0$.

Lemma 2.

$$\delta_{i,j}(w) = u_i^{\zeta_1} \widetilde{V}_1 u_i^{\zeta_2} \widetilde{V}_2 \cdots \widetilde{V}_{p-1} u_i^{\zeta_p} \widetilde{V}_p \cdots \widetilde{V}_{R-1} u_i^{\zeta_R}$$

where $\widetilde{V} = s_{\delta_{i,j}}(V)$*, and, for all* $1 \leq p \leq R$*,*

$$\zeta_p = \eta_p + T(p+1) - T(p).$$

Proof. We put $\Sigma = \delta_{i,j}$, assuming $i < j$. We compute the image of each type of blocks.

- Let $V \in \langle u_i, \ldots, u_{j-1} \rangle$, say $V = u_{i_k}^{\epsilon_k} \cdots u_{i_l}^{\epsilon_l}$ in reduced form, be a block of type 1. The obvious cancellations of $u_i^{-1} u_i$ yield

$$\begin{aligned} \Sigma(V) &= \Sigma(u_{i_k}^{\epsilon_k} . \cdots . u_{i_l}^{\epsilon_l}) \\ &= \left(u_i u_{i_k+1}^{\epsilon_k} u_i^{-1}\right) \left(u_i u_{i_{k+1}+1}^{\epsilon_{k+1}} u_i^{-1}\right) \cdots \left(u_i u_{i_l+1}^{\epsilon_l} u_i^{-1}\right) \\ &= u_i u_{i_k+1}^{\epsilon_k} u_{i_{k+1}+1}^{\epsilon_{k+1}} \cdots u_{i_l+1}^{\epsilon_l} u_i^{-1} \\ &= u_i \widetilde{V} u_i^{-1}, \end{aligned}$$

  where $\widetilde{V} = u_{i_k+1}^{\epsilon_k} u_{i_{k+1}+1}^{\epsilon_{k+1}} \cdots u_{i_l+1}^{\epsilon_l}$, i.e., $\widetilde{V} = s_\Sigma(V)$. Notice that $|\widetilde{V}| = |V|$. We stress the fact that no other cancellation can occur since V was written in reduced form.
- If $V \in \langle u_1, \ldots, u_{i-1}, u_{j+1}, \ldots, u_n \rangle$ is of type 0, then $\Sigma(V) = V$, while $\widetilde{V} = s_\Sigma(V) = V$.
- We recall that $\Sigma(u_j^\eta) = u_i^\eta$.

Now we check what happens at the boundaries of blocks. Say U and U' are of type 0 and V and V' of type 1.

$$\begin{aligned}
\Sigma(Uu_j^\eta U') &= Uu_i^\eta U' \\
\Sigma(Vu_j^\eta V) &= (u_i\widetilde{V}u_i^{-1})u_i^\eta(u_i\widetilde{V}'u_i^{-1}) = u_i\widetilde{V}u_i^\eta\widetilde{V}'u_i^{-1} \\
\Sigma(Uu_j^\eta V') &= Uu_i^\eta(u_i\widetilde{V}'u_i^{-1}) = Uu_i^{\eta+1}\widetilde{V}'u_i^{-1} \\
\Sigma(Vu_jU') &= (u_i\widetilde{V}u_i^{-1})u_i^\eta U' = u_i\widetilde{V}u_i^{\eta-1}U'.
\end{aligned}$$

There are no further cancellations since $U, U' \in \langle u_1, \dots, u_{i-1}, u_{j+1}, \dots, u_n\rangle$ while $\widetilde{V}, \widetilde{V}' \in \langle u_{i+1}, \dots, u_j\rangle$. The proof is the same if $j < i$. □

3.3. *Action of B_n on elementary loops*

We investigate the action of a braid τ on one of the generators of the free group u. $\tau(u)$ is an element of the free group, with a particular form. We consider the map from B_n to F_n

$$\psi_u : \tau \mapsto \tau(u).$$

We consider the set $\mathcal{O}_n = \psi_u(B_n)$, i.e. the orbit of the generator $u \in S^+$. We notice that it does not depend on the choice of the generator. In other words, this is the set $\mathcal{O}_n \subset F_n$ that can be achieved as images of a generator :

$$\mathcal{O}_n = \{w \in F_n \ : \ \exists \tau \in B_n, \pi(\tau)(u_1) = w\}.$$

Geometrically, elements of $\mathcal{O}_n$ are (classes of) simple loops starting at a base point on the boundary with $s_\tau(u)$ inside. We will give a more specific geometrical description in Appendix A. For now, we summarize some elementary properties of these loops in combinatorial terms.

Lemma 3. *For all $w \in \mathcal{O}_n$, $w = u_{i_1}^{\epsilon_1} \cdots u_{i_N}^{\epsilon_N}$ in reduced form*

- *(i) there is $h \in F_n$ and $u \in S^+$ such that $w = huh^{-1}$, ; if $w = \tau(u_k)$ then, $u = s_\tau(u_k)$.*
- *(ii) there are no squares, i.e., $i_{k+1} \neq i_k$, $1 \leq k < N$.*
- *(iii) $\epsilon_1(i_2 - i_1) > 0$.*
- *(iv) For all $1 \leq k < N$, $(i_k - i_1)(i_{k+1} - i_1) \geq 0$.*

The first statement formalizes the fact we can represent a loop drawing only a simple curve from the boundary to the terminal point (staying inside the loop). The last statement means that if $(i_k - i_1)$ and $(i_{k+1} - i_1)$ are of different sign, one of i_k or i_{k+1} must be equal to i_1. In this case, ϵ_k is equal to the sign of $i_{k+1} - i_k$.

Proof. It is certainly possible but rather tedious (particularly (iv)) to give a combinatorial proof of these statements. We will rely on the geometrical point of view (in which the statements are obvious), see Lemma 11 and 12.

3.4. *Delooping one loop*

Fix $w \in \mathcal{O}_n$ and write $w = u_{i_1}^{\epsilon_1} \cdots u_{i_N}^{\epsilon_N}$ in reduced form. We are going to prove that $\overline{\Sigma}_{i_1}^{\epsilon_1}(w)$ is shorter than w itself.

Let us decompose w in blocks following decomposition (3.1), with $j \leftarrow i_1$ and $i \leftarrow 1$ (if $\epsilon_1 < 0$) or $i \leftarrow n$ otherwise,

$$w = u_{i_1}^{\eta_1} V_1 u_{i_1}^{\eta_2} V_2 u_{i_1}^{\eta_3} V_3 \cdots u_{i_1}^{\eta_{R-1}} V_{R-1} u_{i_1}^{\eta_R}, \tag{3.2}$$

where either $V_p \in \langle u_1, \ldots, u_{i_1-1} \rangle$ (block of type L), or $V_p \in \langle u_{i_1+1}, \ldots, u_n \rangle$ (block of type R). Note that the decomposition itself is independent of the sign of ϵ_1. But, if $\epsilon_1 > 0$, blocks of type L correspond to blocks of type 0, while if $\epsilon_1 < 0$, they correspond to blocks of type 1.

We use Lemma 3 to specify the structure of this decomposition. There is an obvious symmetry implying $\eta_R = -\eta_1$ and more generally $\eta_{R+1-k} = -\eta_k$ for all $k \leq R/2$ (Statement (i)). The first block, V_1 is always of type 1 (Statement (iii)). A priori, some η_k could be equal to zero, but Statement (iv) shows that there is always a $u_{i_1}^{\pm}$ between two blocks of different types. Together with statement (ii) this yields $\eta_k \in \{-1, 1\}$. A more subtle use of Statement (iv) shows that if a block of type 0 is followed by a block of type 1 then necessarily, $\eta_p = \eta_1$ (while between two blocks of type 1, sign can be $\pm$).

Remark 7. It could be that there are no blocks of type 0 ($i_1 = 1$ or $i_1 = n$). There are never successive blocks of type 0.

It follows from this analysis and from Lemma 2, that,

$$\overline{\Sigma}_{i_1}^{\epsilon_1}(w) = \widetilde{V}_1 u_*^{\zeta_2} \widetilde{V}_2 \cdots \widetilde{V}_{p-1} u_*^{\zeta_p} \widetilde{V}_p \cdots \widetilde{V}_{R-1}, \tag{3.3}$$

where $* = 1$ if $\epsilon_1 < 0$ and $* = n$ if $\epsilon_1 > 0$, $\widetilde{V} = s_{\overline{\Sigma}_{i_1}^{\epsilon_1}}(V)$, and, for all $1 < p < R$,

$$\zeta_p = \begin{cases} \eta_p \text{ if } V_{p-1} \text{ and } V_p \text{ are of the same type} \\ 0 \quad \text{otherwise.} \end{cases}$$

That is to say, if $\epsilon_1 < 0$,

$$\overline{\Sigma}_{i_1}^{-}(w) = \widetilde{V}_1 u_1^{\zeta_2} \widetilde{V}_2 \cdots \widetilde{V}_{p-1} u_1^{\zeta_p} \widetilde{V}_p \cdots \widetilde{V}_{R-1},$$

and if $\epsilon_1 > 0$,

$$\overline{\Sigma}_{i_1}^{+}(w) = \widetilde{V}_1 u_n^{\zeta_2} \widetilde{V}_2 \cdots \widetilde{V}_{p-1} u_n^{\zeta_p} \widetilde{V}_p \cdots \widetilde{V}_{R-1}.$$

Lemma 4. *If $w \in \mathcal{O}_n$, with $|w| > 1$ then $\left|\overline{\Sigma}_{i_1}^{\epsilon_1}(w)\right| < |w|$.*

Proof. Using the notations above, we compute

$$|w| = \sum_{i=1}^{R} |V_i| + |\eta_i|.$$

It is straightforward to compute the length of the image, since $|\widetilde{V}| = |V|$:

$$|\overline{\Sigma}_{i_1}^{\epsilon_1} w| = \sum_{i=1}^{R} |V_i| + |\zeta_i|.$$

It is enough to notice that since $|w| > 1$ there is at least one block (which must be of type 1). So that there are at least two cancellations. Indeed, $\zeta_0 = \zeta_R = 0$ while, for all i, $|\zeta_i| \leq |\eta_i|$. □

Remark 8. A slightly more precise analysis shows that the difference $\sum_{i=1}^{R}(|\zeta_i| - |\eta_i|)$ is equal to the number of changes of block types plus 2. Indeed, $\zeta_p < \eta_p$ if and only if V_{p-1} and V_p are of different types. See Figure 16 for a geometrical interpretation.

Lemma 5.

$$i_1(\overline{\Sigma}_{i_1(w)}^{\epsilon_1(w)} w) = i_2(w) + \epsilon_1(w).$$

Proof. It follows directly from (3.3). $i_2(w) + \epsilon_1(w)$ is a condensed way to say that the first letter of the image is the first letter of the block $\widetilde{V}_1$ which is of type 1. The block is permuted, and the permutation depends on the sign of ϵ_1. □

We are now in position to prove,

Proposition 1. *For every $w \in \mathcal{O}_n$, there is $W \in \mathcal{L}_n$ such that $|W^{-1}(w)| = 1$.*

We will refer to W^{-1} as the *delooping braid* of the loop w. See Figure 9 for an example. Notice that we obtain a (very) rough bound on the length of W : $|W| \leq |w|$.

Proof. Fix $w \in \mathcal{O}_n$. If $|w| = 1$, the result is obvious. Assume $|w| > 1$ and write $w = u_{i_1}^{\epsilon_1} \cdots u_{i_N}^{\epsilon_N}$ in reduced form. We define

$$L(w) = \overline{\Sigma}_{i_1}^{\epsilon_1}(w).$$

It follows from Lemma 4 that

$$|L(w)| < |w|.$$

Henceforth, the length of the loops $L^m(w)$ is strictly decreasing until $m = K$, the first index for which $|L^K(w)| = 1$. We consider the finite sequence $(I_k, \Xi_k)_{1 \leq k \leq K}$ defined by, for, $0 \leq m < K$,

$$I_{m+1} = i_1(L^m(w)), \text{ and, } \Xi_{m+1} = \epsilon_1(L^m(w)),$$

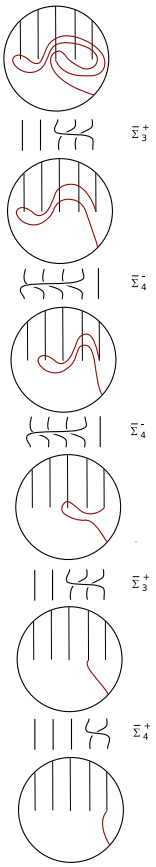

Fig. 9. Delooping using the $\overline{\Sigma}_i^\epsilon$ of $w = huh^{-1}$ with $h = u_3u_5^{-1}u_4^{-1}u_3^{-1}u_1u_3u_4u_5$ and $u = u_3$. Notice that $|L^4(w)| = 1$. The braid $\overline{\Sigma}_i^\epsilon$ act from top to bottom. Hence this is not a conformal graphical representation of the delooping braid, but of its inverse. The reason is that we will finally be interested in the inverse.

and the finite word :

$$W = (\Sigma_{I_k}^{\Xi_k})_{1\leq k\leq K}.$$

It follows from the definition of L that $W^{-1}(w) = L^K(w)$. For every $m < K$, we consider the loop $L^m(w)$; we set $i_1 = i_1(L^m(w))$ and $i_2 = i_2(L^m(w))$. Assume $\epsilon_1(L^m(w)) = -1$. By Statement (iii) of Lemma 3, $i_2 < i_1$. It follows from Lemma 5 that $\overline{\Sigma}_{i_1}^-(L^{m+1}(w))$ starts with u_{i_2+1}. Hence $I_{m+1} = i_2 + 1$, so that

$$2 \leq I_{m+1} \leq I_m.$$

Symmetrically when $\epsilon_1(L^m(w)) = 1$, $i_2 > i_1$ and $I_{m+1} = i_2 - 1$ yield $I_m \leq I_{m+1} < n-1$. Hence, $W \in \mathcal{L}_n$. □

3.5. *The beginning of the image*

Consider $w \in \mathcal{O}_n$. We are going to compute $\Sigma_j^\epsilon(w)$, for all $1 \leq j \leq n$ and all $\epsilon = \pm$. This will allow us to show that if $w = Y(u_n)$ and AY is in normal form then the first letter of $A(w)$ is determined by A, which is the key point to prove the uniqueness part of Theorem 1. More specifically, we prove :

Proposition 2. *Let $W \in \mathcal{L}_{\leq}$. Denote $n^* = n^*(W)$ the minimal index m such that $W \in \mathcal{L}_{\leq m}$. Then the first letter of $W(u_{n^*})$ is $u_{I_1(W)}^{\Xi_1(W)}$. That is $i_1(W(u_{n^*})) = I_1(W)$, $\epsilon_1(W(u_{n^*})) = \Xi_1(W)$.*

Proof. We proceed by recurrence on the length of W. Set $i = I_1(W)$, $\epsilon = \Xi_1(W)$ and $n^* = n^*(W)$. For convenience, we omit the *, i.e. we assume $n = n^*$. If $|W| = 1$ and $\epsilon = 1$, then $W = \delta_{i,n}$ and $W(u_n) = u_i$. If $|W| = 1$ and $\epsilon = -1$ then $W = \delta_{n,1}$, (meaning that $i = n$) and $W(u_n) = u_n^{-1}u_{n-1}u_n$.

Assume the result holds for all $Y \in \mathcal{L}_{\leq n}$, of length $|Y| = N$. Let $W \in \mathcal{L}_{\leq n}$ with $|W| = N + 1$. Again, set $i = I_1(W)$ and $\epsilon = \Xi_1(W)$. Write $W = AY$ with $A = \Sigma_i^\epsilon \in \mathcal{A}_n$ and $Y \in \mathcal{L}_{\leq n}$ of length $|Y| = N$. We must distinguish according to whether $n^*(Y) < n$ or $n^*(Y) = n$.

In the first case, $Y(u_n) = u_n$ so $W(u_n) = AY(u_n) = A(u_n)$ and the situation is the same as if $W = A$ was of length 1 (with respect to u_n). Hence the conclusion holds.

In the second case, we consider the image w of u_n by Y, $w = Y(u_n)$. The recurrence assumption guarantees that it starts with $u_{I_1(Y)}^{\Xi_1(Y)} = u_{I_2(W)}^{\Xi_2(W)}$. Firstly, we assume $\epsilon = 1$. To compute $A(w)$ using Lemma 2 we divide w in blocks following decomposition (3.1) (with $i \leftarrow n$ and $j \leftarrow i$) :

$$w = u_n^{\eta_1} V_1 u_n^{\eta_2} V_2 \cdots V_{p-1} u_n^{\eta_p} V_p \cdots V_{R-1} u_n^{\eta_R}, \tag{3.4}$$

where either $V_p \in \langle u_1, \ldots, u_{i-1} \rangle$ (block of type 0), either $V_p \in \langle u_i, \ldots, u_{n-1} \rangle$ (block of type 1) and $\eta_p \in \{-1, 0, +1\}$. We stress that, here, we have introduced symbols that may correspond to empty words to separate blocks of type 0 and 1. We put $T(0) = T(R) = 0$, and for $1 \leq p < R$, $T(p) = 1$ if V_p is of type 1. We can apply Lemma 2 to see that

$$W(u_n) = \Sigma_i^\epsilon(w) = u_i^{\zeta_1} \widetilde{V}_1 u_i^{\zeta_2} \widetilde{V}_2 \cdots \widetilde{V}_{p-1} u_i^{\zeta_p} \widetilde{V}_p \cdots \widetilde{V}_{R-1} u_i^{\zeta_R}$$

where $\widetilde{V} = s_{\Sigma_i^\epsilon}(V)$, and $\zeta_p = \eta_p + T(p+1) - T(p)$. So, it remains to show that $\zeta_1 = 1$.

Since $AY \in \mathcal{L}_n$, $I_2(W) = I_1(Y) = i_1(w)$ is in the strict range of A. Since $\epsilon = 1$, it implies $i \leq I_2(W) < n$. Hence the first letter of w is in $\langle u_i, \ldots, u_{n-1} \rangle$, showing that $\eta_1 = 0$ but that the first block is of type 1. We conclude that $\zeta_1 = 1$ so we are done. The case $\epsilon = -1$ implying $1 < I_2(W) \leq i$ is symmetric. □

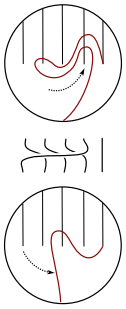

Fig. 10. Graphical interpretation of Proposition 2

The proof has a very simple geometric interpretation. The argument is that $\Sigma_i^\epsilon(w)$ moves an extremal point (a_1 or a_n) to position a_i, crossing the beginning of the loop w (see Figure 10).

4. Normal form

4.1. *Proof of Theorem 1*

Proof. Let $\tau \in B_n$. To prove existence, we build a word $W \in \mathcal{L}_{\leq n}$ such that

$$\tau = W_1 \cdots W_K.$$

The proof is by induction on the number n of strands. It is obvious in B_2 because $\mathcal{A}_2 = \{\delta_{1,2}, \delta_{2,1}\}$ and $\tau = \delta_{1,2}^K$ for some $K \in \mathbb{Z}$. Assume the statement holds in B_{n-1} and let $\tau \in B_n$. Consider the loop $\tau(u_n)$. Following Proposition 1, we construct a word $Y \in \mathcal{L}_n$ such that $|Y^{-1}(\tau(u_n))| = 1$. Say $Y^{-1}(\tau(u_n)) = u_k$. We notice that (since we followed the construction) k is in the strict range of Y, so that $W = Y\Sigma_k^+ \in \mathcal{L}_n$. But, $\overline{\Sigma}_k^+(Y^{-1}(\tau(u_n))) = \overline{\Sigma}_k^+(u_k) = u_n$. Hence $W = Y\Sigma_k^+$ satisfies

$$W^{-1}(\tau(u_n)) = u_n.$$

It implies that $W^{-1}\tau \in B_{n-1}$. We use the inductive assumption to find $V \in \mathcal{L}_{\leq n-1}$ such that $W^{-1}\tau = V$. Since $n \in [W]$, we conclude that, $WV \in \mathcal{L}_{\leq n}$ and

$$\tau = WV.$$

To prove it is unique, we show that two different words (U and V in $\mathcal{L}_{\leq}$) yield two different braids. Without loss of generality, we can assume that the first letter of the words are distinct (or else erase by taking the inverses). If $n^*(U) \neq n^*(V)$ (for instance, $n^*(U) < n^*(V) = n^*$), then, $U(u_{n^*}) = u_{n^*}$, while $V(u_{n^*}) \neq u_{n^*}$. If $n^*(U) = n^*(V) = n^*$, then we use Proposition 2 to see that the first letter of $U(u_{n^*})$ is $u_{I_1(U)}^{\Xi_1(U)}$ while the first letter of $V(u_{n^*})$ is $u_{I_1(V)}^{\Xi_1(V)}$, i.e. they are distinct. Since the free group is free, we use Lemma 1 to conclude. □

5. Connection with the Dehornoy order

We recall the definition of the Dehornoy ordering of braids [4] (here we consider the version in which σ_1 is smaller than σ_2).

Definition 1. (i) A σ-word is said to be *σ-positive* if the generator σ_m with maximal index occurring in w occurs positively only, i.e., σ_m occurs but σ_m^{-1} does not. (ii) Assume that x, y are braids. We say that $x < y$ holds if the quotient braid $x^{-1}y$ admits at least one expression by a σ-positive word.

Then the relation $<$ is a linear ordering on braids that is compatible with multiplication on the left. Our normal form seems to be adapted to tackling questions about this ordering of braids. Indeed, we prove that one of the normal forms of τ and τ^{-1}, when re-written in terms of σ is positive in the sense of Dehornoy.

For every braid τ, we denote $\varsigma(\tau)$ the σ-word obtained by rewriting all $\delta_{i,j}$ in terms of $\{\sigma_i^\epsilon\}$ in the normal form W of τ. Assume $n^*(W) = n$. In $\varsigma(\tau)$, the letter σ_{n-1} appears only in Σ_i^+ (positively) and in Σ_n^- (negatively). Hence τ is positive if W does not contain Σ_n^-, i.e. if it does not start with Σ_n^-.

Theorem 2. *Let $\tau \in B_n$. One of the two words $\varsigma(\tau)$ or $\varsigma(\tau^{-1})$ is positive.*

To prove this result we need the following lemma :

Lemma 6. *Assume $i_1(\tau(u_n)) = n$. Then $i_1(\tau^{-1}(u_n)) < n$*

Proof. Let τ be such that $i_1(\tau(u_n)) = n$. We are going to show that the first letter of $\tau^{-1}(u_n)$ can not be u_n. Let $W = W^{(n)} \cdots W^{(2)}$ be the normal form of τ and write $W = ZY$, with $Z = W^{(n)}$ and $Y = W^{(n-1)} \cdots W^{(2)} \in B_{n-1}$. First, we show that the first letter of $v = Z^{-1}(u_n)$ must be u_1 ; then, we check that no u_n can appear in the beginning of $Y^{-1}(v)$ since Y is in B_{n-1}.

First, we show that $i_1(Z^{-1}(u_n)) = 1$. Write $Z = W^{(n)} = W_1 \cdots W_K$. Its inverse is the delooping braid of $\tau(u_n)$. The assumption implies $W_1 = \Sigma_n^-$, and hence, $W_1^{-1}(u_n) = u_1$. It also implies that $W_k \neq \Sigma_1^+$ for all $2 \leq k \leq K$.

We proceed by induction on the length of $W^{(n)}$. For $k < K$, set $w = W_k^{-1} \cdots W_1^{-1}(u_n)$ and assume $i_1(w) = 1$. We apply Lemma 2 to compute $W_{k+1}^{-1}(w)$. We distinguish according to the sign $\xi(W_{k+1})$. If positive, (recalling that then, $W_{k+1} \neq \Sigma_1^+$), then the first block is of type 0 and hence left unchanged, so that and $i_1(W_{k+1}^{-1}(w)) = i_1(w) = 1$. If negative, then the first block is of type 1 and $W_{k+1}^{-1}(w)$ starts with $u_1 u_2 \cdots$. In both cases, $i_1(W_{k+1}^{-1}(w)) = 1$. We conclude by induction.

Next, set $v = Z^{-1}(u_n)$ and $Y = W^{(1)} \cdots W^{(n-1)} \in B_{n-1}$, so that $\tau^{-1}(u_n) = Y^{-1}(v)$. Write $v = V_1 u_n^{\eta_1} V_2 \cdots u_n^{\eta_r} V_{r+1}$ with $V_p \in \langle u_1, \ldots, u_{n-1} \rangle$ and $\eta_p \neq 0$. Notice that for every $1 \leq p \leq r+1$, $Y^{-1}(V_p)$ in reduced form is not empty. Hence, $Y^{-1}(v) = Y^{-1}(V_1) u_n^{\eta_1} Y^{-1}(V_2) \cdots u_n^{\eta_r} Y^{-1}(V_{r+1})$ is in reduced form. Since V_1 is not

empty, $Y^{-1}(V_1)$ is not empty.

It remains to show that $Y^{-1}(V_1)$ does not contain any u_n. We apply again Lemma 2 with the $\delta_{i,j}$ appearing in Y^{-1}, for which, $i < n$ and $j < n$. Here, the induction argument is the following : assume $i_1(w) < n$. Then, we consider $\delta_{i,j}(w)$ where $i < n$ and $j < n$. If $i_2(w) < n$, then either $i_1(\delta_{i,j}(w)) = i_1(w)$, or $i_1(\delta_{i,j}(w)) = s_{\delta_{i,j}}(i_2(w)) < n$ (because $s_{\delta_{i,j}}\{1,\dots,n-1\} = \{1,\dots,n-1\}$). If $i_2(w) = n$, then we have $T(1) = T(0) = 0$ and hence, $\xi_1 = \eta_1$, so that $i_1(\delta_{i,j}(w)) = i_1(w)$. Repeated use of this argument shows that $i_1(Y^{-1}(V_1)) < n$ and yields the result. □

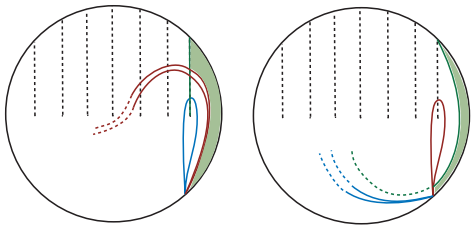

Fig. 11. The geometrical argument for the proof of Lemma 6. Consider the loop $\tau(u_n)$ (dark) and the segment d_n on the left hand side. The loop $\tau(u_n)$ is mapped onto u_n and d_n is mapped onto a separatrix $\tau^{-1}(d_n)$ starting on the top (on the right hand side), by τ^{-1}. The shadowed domain delimited by these curves and the boundary of the disk is mapped to the shadowed domain on the right hand side. Now, the point is that the separatrix $\tau^{-1}(d_n)$ after its (two) first intersections with the loop u_n must reach one of the d_i with $i < n$. But $\tau^{-1}(u_n)$ is on the left side of $\tau^{-1}(d_n)$ and they intersect only at the end. So $\tau^{-1}(u_n)$ must intersect some d_j, with $j \leq i < n$ before to reach d_n. Hence $i_1(\tau^{-1}(u_n)) < n$.

Proof of Theorem 2. As we already have mentioned, either $\varsigma(\tau)$ is positive or the normal form $V$of τ contains Σ_n^-. In the latter case, since $V \in \hat{\mathcal{L}}$, it means that $V_1 = \Sigma_n^-$. From Proposition 2, we deduce that $i_1(\tau^{-1}(u_n)) = i_1(V(u_n)) = n$. From Lemma 6, we deduce that $i_1(\tau(u_n)) < n$. We conclude that $\varsigma(\tau^{-1})$ is positive. □

So our approach gives a new proof of the following so-called Comparison Property of [4], σ-negative meaning that the generator with highest index appears negatively only.

Corollary 1. *Every nontrivial braid admits an expression by a word that is σ-positive, or σ-negative.*

6. Algorithm

The normal form of a braid is obviously computable looking at the images of the elementary loops and then applying the delooping procedure. Here we provide an

algorithm to recover it directly (providing as well a proof of existence). It is based on a Tetris-like computation of the normal form of $\sigma_i^\epsilon W$ given W in normal form. Roughly speaking, we show how σ_i^ϵ crosses W :

- Firstly, we will show how to cross one letter : the transition rules (6.5) shows how to write $\delta_{i,j}W_1$, with $W_1 \in \mathcal{A}_n$ in the form $\widetilde{W}_1\delta_{k,l}$, with $\widetilde{W}_1 \in \mathcal{L}_n$ (Lemma 7).
- Then, we will show that if we end up with $W_1\delta_{i,j}W_2W_3$ we can use the transition rules to write it as $W_1\widetilde{W}_2\delta_{k,l}W_3$, with $W_1\widetilde{W}_2$ in normal form (Lemma 8) ; so that $\delta_{i,j}$ can cross a whole word W in $\mathcal{L}_n$: $\delta_{i,j}W = \widetilde{W}\delta_{k,l}$, letting $\widetilde{W} \in \mathcal{L}_n$ (Lemma 9).
- Finally, we will analyse the three possible situations between $W^{(k)}$ and $W^{(k-1)}$ (Lemma 10). We can conclude that the procedure finishes.

Thus, given $\tau = \sigma_{i_1}^{\epsilon_1} \cdots \sigma_{i_K}^{\epsilon_K}$, we can write $\sigma_{i_K}^{\epsilon_K}$ in normal form and we proceed inductively : given the normal form of $\tau_{k+1} = \sigma_{i_{k+1}}^{\epsilon_{k+1}} \cdots \sigma_{i_K}^{\epsilon_K}$, we compute the normal form of $\tau_k = \sigma_{i_k}^{\epsilon_k}\tau_{k+1}$, and so on an so forth, until $k = 1$.

6.1. *Transitions*

A first lemma shows how to compute $\delta_{i,j}\Sigma_q^\epsilon$. We prove that

Lemma 7. *For all i, j, q and ϵ, we can write $\delta_{i,j}\Sigma_q^\epsilon = \Sigma\delta_{k,l}$, where $\Sigma \in \mathcal{L}_n$ and either $k \in [\Sigma[$ (situation 1) or $[k,l] \cap [\Sigma] = \emptyset$ (situation 2). More specifically,*

$$\delta_{i,j}\Sigma_q^\epsilon = \begin{cases} \Sigma_q^\epsilon\ \delta_{i,j} & \text{if } [i,j] \cap [q,*] = \emptyset \\ \Sigma_q^\epsilon\ \delta_{i-\epsilon,j-\epsilon} & \text{if } [i,j] \subset]q,*] \\ \delta_{i,*} & \text{if } j = q \\ \Sigma_i^\epsilon\Sigma_q^\epsilon\ \delta_{*-\epsilon,j-\epsilon} & \text{if } q \in [i,j[\text{ and } j \in]q,*] \\ \Sigma_i^{-\epsilon}\Sigma_q^\epsilon\ \delta_{n+1-*,j} & \text{if } q \in [i,j[\text{ and } j \notin [q,*]. \end{cases} \tag{6.5}$$

where $* = 1$ if $\epsilon < 0$ and $* = n$ if $\epsilon > 0$.

Proof. There are three main cases.

(1) If $[i,j] \cap [q,*] = \emptyset$, then $\delta_{i,j}\Sigma_q^\epsilon = \Sigma_q^\epsilon\delta_{i,j}$. We are obviously in situation (2).
(2) If $[i,j] \subset]q,*]$, then $\delta_{i,j}\Sigma_q^\epsilon = \Sigma_q^\epsilon\delta_{i-\epsilon,j-\epsilon}$. Here, since $j \in]q,*]$, $j - \epsilon \in [q,*[$ and we are in situation (1).
(3) If $q \in [i,j]$, then we distinguish
   (a) if $j = q$, then $\delta_{i,j}\Sigma_q^\epsilon = \delta_{i,*}$,
   (b) if $j \in]q,*]$ (which implies $q \in [i,*[$), then $\delta_{i,j}\Sigma_q^\epsilon = \delta_{i,*}\delta_{*,j}\Sigma_q^\epsilon = \Sigma_i^\epsilon\Sigma_q^\epsilon\delta_{*-\epsilon,j-\epsilon}$ (situation (2) because $* - \epsilon \in [q,*[$)
   (c) if $j \in [q,*]^c$ (which implies $q \in [i,*]^c$), then $\delta_{i,j}\Sigma_q^\epsilon = \delta_{i,n+1-*}\delta_{n+1-*,j}\Sigma_q^\epsilon = \Sigma_i^{-\epsilon}\Sigma_q^\epsilon\delta_{n+1-*,j}$ (situation (1) because $[j, n+1-*] \cap [q,*] = \emptyset$). □

We notice that $\delta_{k,l}$ is empty if $k = l$. This may happen only in case 3.a if $i = *$, in case 3.b if $j = *$ or in case 3.c if $j = n + 1 - *$.

6.2. *Along a word in normal form*

We recall that for $W \in \mathcal{L}_n$, $[W[$ denotes the interval $[i, j[$ corresponding to its last letter $\delta_{i,j}$.

Lemma 8. *Consider $W_1W_2W_3 \in \mathcal{L}_n$ and $\delta_{i,j}$ such that, either $i \in [W_1[$ or $[i, j] \cap [W_1] = \emptyset$. We can write $W_1\delta_{i,j}W_2W_3$ under the form,*

$$W_1\delta_{i,j}W_2W_3 = W_1\widetilde{W}_2\delta_{k,l}W_3$$

with $W_1\widetilde{W}_2W_3 \in \mathcal{L}_n$, $0 \leq |\widetilde{W}_2| \leq 2$, and, either $k \in [W_1\widetilde{W}_2[$, or $[k, l] \cap [W_1\widetilde{W}_2] = \emptyset$.

Notice that the decomposition $\widetilde{W}_2\delta_{k,l}$ may depend on W_3 (case 3.a).

Proof. First, notice that if we are in the case $[i, j] \cap [W_1] = \emptyset$, then either $[i, j] \cap [W_2] = \emptyset$ or $[i, j] \subset [W_2[$ because $I(W_2) \in [W_1[$. Following the line of Lemma 7, let us distinguish,

- $[i, j] \cap [W_2] = \emptyset$. Then $\delta_{i,j}W_2 = W_2\delta_{i,j}$ and the conclusion holds.
- $[i, j] \subset ([W_2] \setminus I(W_2))$. In this case, $\widetilde{W}_2 = W_2$ and $k = i - \Xi(W_2) \in [\widetilde{W}_2[$.
- $I(W_2) \in [i, j[$. Then $\widetilde{W}_2 = \delta_{i,*}W_2$. Since $i \in [W_1[$ and $I(W_2) \in [i, *[$, $W_1\delta_{i,*}W_2W_3 \in \mathcal{L}_n$. Moreover, either $k \in [W_2[$, either $[k, l] \cap [W_2] = \emptyset$ depending on the relative position of $[*, j]$ with respect to $[W_2]$ ($[*, j] \cap [W_2] = \emptyset$ or $[*, j] \subset [W_2]$). Notice that if $i = n$ then, $\delta_{k,l} = \emptyset$
- $I(W_2) = j$. In this case, $\delta_{i,j}W_2 = \delta_{i,*}$ (where $*$ depends on sign of $\Xi(W_2)$). We know that $i \in [W_1[$, so, either $I(W_3) \in [i, *[$, and then $W_1\delta_{i,*}W_3 \in \mathcal{L}_n$, or we are in situation where $[*, j] \cap [W_3] = \emptyset$ but, then, $W_1W_3 \in \mathcal{L}_n$. In the first case, we can take $\widetilde{W}_2 = \delta_{i,*}$, and $\delta_{k,l} = \emptyset$. In the latter case, $\widetilde{W}_2 = \emptyset$ and $\delta_{k,l} = \delta_{i,*}$. □

We consider the map

$$\begin{aligned} \Lambda : \mathcal{E}_n \times \mathcal{A}_n &\to \mathcal{L}_n \times \mathcal{E}_n \\ (\delta_{i,j}, W) &\mapsto (\widetilde{W}, \delta_{k,l}) \end{aligned}$$

defined by

$$(\widetilde{W}, \delta_{k,l}) = \begin{cases} (W,\, \delta_{i,j}) & \text{if } [i, j] \cap [W] = \emptyset \\ (W,\, \delta_{i-\epsilon,j-\epsilon}) & \text{if } [i, j] \subset]W] \\ (\emptyset,\, \delta_{i,*}) & \text{if } j = I(W) \\ (\Sigma_i^{\epsilon}W,\, \delta_{*-\epsilon,j-\epsilon}) & \text{if } I(W) \in [i, j] \text{ and } j \in]W] \\ (\Sigma_i^{-\epsilon}W,\, \delta_{n+1-*,i}) & \text{if } I(W) \in [i, j] \text{ and } j \notin [W]. \end{cases}$$

We are going to extend Λ recursively to a map on $\mathcal{E}_n \times \mathcal{L}_n$. Assume it is defined for words of length K. For a word $W \in \mathcal{L}_n$ of length K, a letter $A \in \mathcal{A}_n$ such that $WA \in \mathcal{L}_n$ and an elementary braid $\delta_{i,j}$, we denote $(\widetilde{W}, \delta_{r,s}) = \Lambda(\delta_{i,j}, W)$, $(\widetilde{A}, \delta_{k,l}) = \Lambda(\delta_{r,s}, A)$ and we set,

$$\Lambda(WA, \delta_{i,j}) = (\widetilde{W}\widetilde{A}, \delta_{k,l}).$$

Lemma 9. *The map Λ is well defined on $\mathcal{L}_n$. For every word $W \in \mathcal{L}_n$ and every elementary braid $\delta_{i,j}$, the pair $(\widetilde{W}, \delta_{k,l}) = \Lambda(\delta_{i,j}, W)$ satisfies $\widetilde{W} \in \mathcal{L}_n$ and $\delta_{i,j}W = \widetilde{W}\delta_{k,l}$.*

Notice that in general, $|W| \neq |\widetilde{W}|$.

Proof. The proof is a simple induction initiated with Lemma 7 and then using Lemma 8. The only detail to be careful with is the definition of $\widetilde{W_2}$ in the case $I(W_2) = i$. Indeed we can choose $\widetilde{W_2} = \emptyset$, even if W_1W_3 is not in $\mathcal{L}_n$ because in this case, $W_1\delta_{i,*}W_3$ is in $\mathcal{L}_n$. □

To conclude we must understand what happens at the transitions between words of different levels. We need the following technical lemma :

Lemma 10. *If $n \in [W]$, then three different cases may occur :*

(i) if $n \notin [k,l]$ then $n \in [\widetilde{W}]$,
(ii) if $l = n$, then $k \in [\widetilde{W}[$ and $\widetilde{W}\delta_{k,l} \in \mathcal{L}_n$,
(iii) if $k = n$, then $\widetilde{W} = \delta_{n,1}^q$ for some q and $\delta_{k,l} = \delta_{n,1}\delta_{1,l}$.

Notice that it could very well be that $\delta_{k,l}$ is empty, yielding case (i). Notice that if W is empty, we can have the three cases, replacing $[\widetilde{W}]$ with $[1,n]$.

Proof. Let $W \in \mathcal{L}_n$ with $n \in [W]$. If W is not empty then we call A the last letter of W and write $W = VA$, with $V \in \mathcal{L}_n$. Since $n \in [W]$, A is either $\delta_{n,1}$ or $\delta_{q,n}$, for some q.

- If $A = \delta_{n,1}$ then W must be a power of $\delta_{n,1}$. Then the analysis is straightforward since $\delta_{i,j}\delta_{n,1}^q$ is periodic of period n in q. Write $\delta_{i,j}\delta_{n,1}^q = \delta_{n,1}^{q'}\delta_{k,l}$, with $q' \in \{q-1, q, q+1\}$. If $n \notin [k,l]$, then we we are in case (i). If $l = n$, $\delta_{k,l} = \delta_{k,n}$ and we are in case (ii) since $k \in]1,n[$. If $k = n$, $\delta_{k,l} = \delta_{n,l}$ and we are in case (iii).
- If $A = \delta_{q,n}$ for some q, then, using Lemma 8 (or Lemma 7 if $|W| = 1$) we write $\delta_{i,j}W = \delta_{i,j}VA = \widetilde{V}\delta_{r,s}A$ with $\widetilde{V}A \in \mathcal{L}_n$ (and either $r \in [\widetilde{V}[$ or $[r,s] \cap [\widetilde{V}] = \emptyset$).
  - If $\delta_{r,s}$ is empty then we are obviously in case (i).
  - If $s = q$, we are in case 3.a. : $\delta_{r,s}A = \delta_{k,l} = \delta_{r,n}$. Since $\widetilde{V}A \in \mathcal{L}_n$, we have $q \in [\widetilde{V}[$ so $[q,r] \cap [\widetilde{V}] \neq \emptyset$. But $[r,s] \cap [\widetilde{V}] \neq \emptyset$ implies $r = k \in [\widetilde{V}[$. If $k \neq n$, then $k \in [\widetilde{V}[= [\widetilde{W}[$ so we are in case (ii). If $k = n$, $\delta_{k,n}$ is empty but $n \in [\widetilde{W}]$ and we are in case (i).
  - If $s \neq q$, then, we write $\delta_{r,s}A = \widetilde{A}\delta_{k,l}$ with $\widetilde{V}\widetilde{A} \in \mathcal{L}_n$ and $[\widetilde{A}] = [A]$. In this case, $n \notin [k,l]$ so we are in case (i). Indeed, we check, using Lemma 8 :
    - in case 1., $[r,s] \cap [q,n] = \emptyset$ so $n \notin [k,l] = [r,s]$,
    - in case 2., k and l are smaller than $n - \epsilon = n - 1$,
    - in case 3.b., k and l are smaller than $n - \epsilon = n - 1$,
    - In case 3.c., $l = s < q < n$ and $k = n + 1 - * = 1$. □

Let $W \in \mathcal{L}_{\leq n}$, $W = W^{(n)}W^{(n-1)}\cdots W^{(2)}$. Assume the splitting is nice, i.e. $p \in [W^{(p)}]$ for all p with $W^{(p)} \neq \emptyset$. Let $\delta_{i,j} \in B_n$. To put $\delta_{i,j}W$ in normal form, we propose the following inductive procedure. We explain how to complete one step. We set $(\widetilde{W}^{(n)}, \delta_{k,l}) = \Lambda(\delta_{i,j}, W^{(n)})$. Using Lemma 9, we write

$$\delta_{i,j}W^{(n)}W^{(n-1)}\cdots W^{(2)} = \widetilde{W}^{(n)}\delta_{k,l}W^{(n-1)}\cdots W^{(2)}.$$

If $W^{(n)}$ is empty then either $n \in [i,j]$ and $\delta_{i,j}W$ is in normal form, either $\delta_{i,j} \in B_{n-1}$ and we can proceed to step $n-1$. Otherwise we distinguish according to the three situations of Lemma 10.

- In case (i) we put $Y^{(n)} = \widetilde{W}^{(n)}$ and we proceed with $\delta_{k,l} \in B_{n-1}$.
- In case (ii), we put $Y^{(n)} = \widetilde{W}^{(n)}\delta_{k,l}$ and we proceed with $\emptyset$ (i.e. we stop).
- In case (iii), we write $\delta_{k,l} = \delta_{n,1}\delta_{1,l}$. We put $Y^{(n)} = \widetilde{W}^{(n)}\delta_{n,1}$ and we proceed with $\delta_{1,l} \in B_{n-1}$.

It follows from Lemma 10 that $\delta_{i,j}W^{(n)} = Y^{(n)}\delta$, with $Y^{(n)} \in \mathcal{L}_n$, $n \in [Y^{(n)}]$ and $\delta \in B_{n-1}$. To conclude, we stress that the case $n = 2$ is simpler since $\delta_{i,j}W^{(2)}$ must be a power of $\delta_{1,2}$ or $\delta_{2,1}$. Hence the procedure finishes.

6.3. *Examples*

This scheme obviously provides an algorithm to put any σ-word in normal form. It is as well interesting to understand the action of the group on infinite words in normal form (see further).

Normal form. Let us consider a simple example. Let τ be the braid in B_4

$$\tau = \sigma_1^{-1}\sigma_2\sigma_3^{-1} = \delta_{2,1}\delta_{2,3}\delta_{4,3}.$$

We are going to compute its normal form. Firstly, we write σ_3^{-1} in normal form,

$$\sigma_3^{-1} = \delta_{4,3} = (\delta_{4,1})(\delta_{1,3}).$$

Then, we reduce

$$\sigma_2\sigma_3^{-1} = \delta_{2,3}\delta_{4,3} = \delta_{2,3}\delta_{4,1}\delta_{1,3} = (\delta_{4,1}\delta_{3,4})(\delta_{1,3}).$$

We obtain,

$$\begin{aligned}
\tau &= \delta_{\mathbf{2},\mathbf{1}}(\delta_{4,1}\delta_{3,4})(\delta_{1,3}) \\
&= (\delta_{4,1}\delta_{\mathbf{3},\mathbf{2}}\delta_{3,4})(\delta_{1,3}) \\
&= (\delta_{4,1}\delta_{3,1}\delta_{\mathbf{1},\mathbf{2}}\delta_{3,4})(\delta_{1,3}) \\
&= (\delta_{4,1}\delta_{3,1}\delta_{3,4})\delta_{\mathbf{1},\mathbf{2}}(\delta_{1,3}) \\
&= (\delta_{4,1}\delta_{3,1}\delta_{3,4})(\delta_{1,3}\delta_{\mathbf{3},\mathbf{2}}\delta_{1,3}) \\
&= (\delta_{4,1}\delta_{3,1}\delta_{3,4})(\delta_{1,3}\delta_{1,3})(\delta_{\mathbf{2},\mathbf{1}}).
\end{aligned}$$

So finally,

$$\tau = (\delta_{4,1}\delta_{3,1}\delta_{3,4})\,(\delta_{1,3}\delta_{1,3})\,(\delta_{2,1}).$$

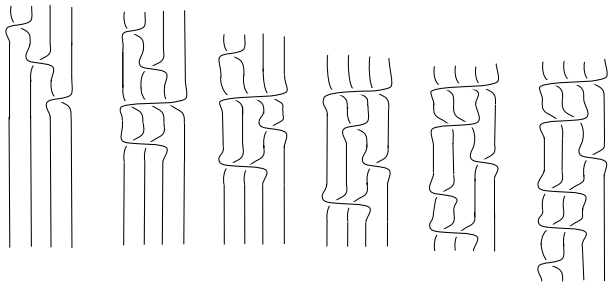

Fig. 12. Computation of the normal form of $\tau = \sigma_1^{-1}\sigma_2\sigma_3^{-1}$.

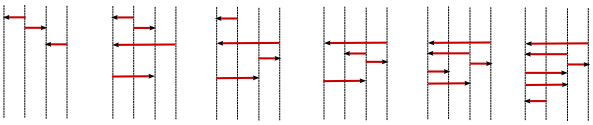

Fig. 13. Tetris-like presentation of Figure 12.

We represent the computation graphically on Figure 12 ; all braids are equal. We give another representation in condensed way (*Tetris-like*) on Figure 13 : an arrow from strand i to strand j stands for the elementary braid $\delta_{i,j}$.

Along a word in normal form. On Figure 14, we show graphically an example of computation of the normal form of $\delta_{i,j}W$ when W is given in normal form. The example was chosen so that most of the transitions types appear. Here, $\delta_{i,j} = \delta_{2,1}$ and

$$W = \delta_{4,1}\delta_{3,1}\delta_{2,1}\delta_{2,4}\delta_{2,4}\delta_{2,1}\delta_{2,4}\delta_{3,1}\delta_{2,1} \in \mathcal{L}_n.$$

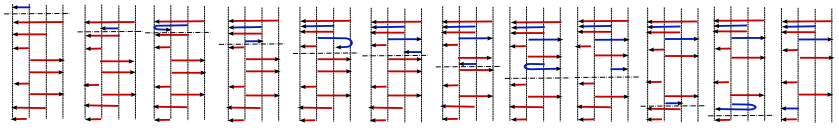

Fig. 14. Tetris-like computation of $\widetilde{W}$.

7. Complexity

We investigate the first properties of the normal form in terms of complexity. Let us consider a very simple example in B_4. We compare the braid $(\delta_{2,4}\delta_{3,1})^n$ written in normal form and the normal form of $\delta_{2,3}(\delta_{2,4}\delta_{3,1})^n$. Notice that

$$\delta_{2,3}\delta_{2,4}\delta_{3,1} = \delta_{2,4}\delta_{2,4}\delta_{3,2}\delta_{3,1} = \delta_{2,4}\delta_{2,4}\delta_{3,2}\delta_{3,2}\delta_{2,3}.$$

Hence, for every n, the word $w = (\delta_{2,4}\delta_{3,1})^n$ is in normal form while the normal form of $\sigma_2 w$ is :

$$\delta_{2,3}(\delta_{2,4}\delta_{3,1})^n = (\delta_{2,4}^2\delta_{3,1}^2)^n\delta_{2,3},$$

showing that adding only one σ-generator may change drastically the length of the normal form.

7.1. *Automatic structure*

We stress that the language $\mathcal{L}_n$ is of finite type while the language $\mathcal{L}_{\leq n}$ is regular as shown by Remark 4 ; it is recognized by an automaton $\mathbf{A}_n$. It is hence natural to ask if $(\mathbf{A}_n, \mathcal{L}_{\leq n})$ is an automatic structure for B_n (in the sense of [3]). It is not the case as shown by Lemma 2.3.9 (bounded length difference) in [3] and the example above : the length of the normal form of $\sigma_2 w$ is not bounded by the length of the normal form of w plus a constant while all elements in the group have a finite number (one) of representatives in $\mathcal{L}_{\leq n}$.

7.2. *Length*

The normal form is obviously not geodesic. It is crucial to decide if it is possible to bound the length of a representative in terms of its minimal σ-length. An exponential bound is obvious (it follows from the same arguments as in [6], i.e. the arguments of Lemma 4). My guess is that there is in fact a linear bound. The example above shows that such a bound can not be obtained by a naive recursion on the σ-length. Roughly speaking, my argument for the linear growth relies on the fact that the delooping strategy used to deloop one loop is not optimal but still efficient : the geometric point of view seems to show it would not be possible to deloop significantly faster using any other strategy. Notice that numerical experiments go in the same direction.

7.3. *Algorithmic complexity*

For the same kind of reasons, a naive computation only provides an exponential bound on the time needed to compute the normal form of the braid associated to a σ-word of length n. Indeed, each time we add a σ-generator, the length of the normal form may double so the bound is in $\sum_{k=1}^{n-1} 2^k \sim 2^n$. A linear bound on the length would immediately yield a quadratic bound for the speed of the algorithm. For the algorithm in its present form, it is certainly the best one can hope. Still, I conjecture that a statistical analysis of the algorithm would yield a linear expected speed (for reasonably randomly chosen σ-words).

8. Other perspectives

We raise a few points in link with our initial motivations.

8.1. *Poisson boundary*

Let μ be a probability measure on $G \cup G^{-1}$ with full support. We consider a random iid sequence $(g_n)_{n\geq 1}$ on $(\Omega, P) = ((G \cup G^{-1})^{\mathbb{N}}, \mu^{\otimes \mathbb{N}})$. We consider the random walk $(X_n)_{n\geq 0}$ defined by

$$\begin{cases} X_0 = e \\ X_n = g_n X_{n-1},\ n \geq 1 \end{cases}$$

Description of the possible asymptotic behaviors of such a random walk is the study of the so-called Poisson boundary of the group for μ. This boundary was identified by Kaimanovich and Masur ([5]) as the space of projective measured foliations of the punctured disk $\mathcal{D}_n$ endowed with a measure supported on uniquely ergodic foliations.

A combinatorial description of the boundary is missing. We hope that this work will give hints toward such a description. The language $\mathcal{L}_n$ generates a subshift of finite type in which infinite words may be interpreted as infinite braids. There is some hope to understand the Poisson boundary in terms of this subshift. This strategy works for B_3, but that is not a great deal since in this particular case, a combinatorial picture can be given by other means (see for instance [7]).

8.2. *Symbolic dynamics, induction and continued fractions*

Consider an infinite simple curve in the punctured disk starting on the boundary. Under reasonable conditions, this curve may define a separatrix of a minimal foliation. Our delooping scheme may be applied to such a curve, since the delooping strategy is guided by the beginning of the curve. This algorithm may be understood as a continued fraction algorithm describing approximations of the infinite curve.

The underlying dynamic may be seen as an interval exchange. Taking a section of the foliation and up to an orientation cover we can analyse the foliation as an

interval exchange. The coding of the separatrix would correspond to the coding of a particular orbit. Our scheme is indeed an induction scheme for this dynamical system, which seems to be distinct from the so called Rauzy induction, to be able to provide combinatorial information on the symbolic dynamics and to have nice duality properties.

Appendix A. The picture

The braid group B_n is the mapping class group of the punctured disk $\mathcal{D}_n = \mathcal{D} \setminus P$ with $P = \{a_1, \dots, a_n\}$ a set of n points in its interior. It naturally acts on homotopy classes of loops, hence on the fundamental group $F_n = \langle u_1, \dots, u_n \rangle$ of $\mathcal{D}_n$. Our normal form is deeply related to this action of the free group and most of the results we stated in combinatorial form have a nice geometric counterpart. Indeed, the proof of the existence of the normal form is based on an algorithm to deloop all elements in the orbit $\mathcal{O}_n$ of simple loops under the action of B_n.

Our goal here is to give a more formal description of the correspondence between the combinatorial and the geometrical points of view. It will allow us to give a rigorous basis to our statements on the combinatorial properties of elements of $\mathcal{O}_n$. Using the same language, we will give a complete description of the words coding loops in $\mathcal{O}_n$, not directly useful for the proofs of existence and uniqueness of the normal form, but which may have some interest in itself. Finally, we will show how to relate this description to the normal form.

Orbit of $\mathcal{O}_n$. It follows from a simple geometrical argument that representatives of loops in $\mathcal{O}_n$ delimit a disk containing exactly one point in P called the *terminal* point. Hence, the coding of a loop in $\mathcal{O}_n$ is always of the form hah^{-1}. The letter a corresponds to the terminal point. A loop is described by its *code*, (h, a), where $h \in F_n$ is written in normal form and $a \in S$. We will exhibit some basic properties of these words, used in the proof of the existence of the normal form. But, there is still too much information in the word h. To go further, we propose a strategy to get rid of the redundancies.

Efficient coding. Let us show how to build "recursively" all the elements of $\mathcal{O}_n$. Consider a separatrix with code h and terminal point a. Choose a terminal point $x \in P \setminus \{a\}$. We claim that there is a word u and $\epsilon = \pm$ such that the loop with code $(ha^\epsilon u, x)$ is in $\mathcal{O}_n$. We consider the shorter such u and we denote $S_x(h, a) = (ha^\epsilon u, x)$. Following this recursive step, we associate to every (finite) sequence of "terminal" points (without repetition) an element of $\mathcal{O}_n$. That is to $x_1 \cdots x_K$ we associate $S_{x_K} \circ \cdots \circ S_{x_2}(\emptyset, x_1)$. This construction has a nice converse. Given an element w of $\mathcal{O}_n$, it is easy to recover the sequence of terminal points. They correspond (up to exponents) to subwords of w characterized geometrically (*closest intersections*) or algebraically (*admissible prefixes*). Finally we have a bi-

jection between $\mathcal{O}_n$ and words in a subshift of finite type we call the *efficient coding*. This coding may be seen as a generalized continued fractions algorithm.

Delooping braid. The delooping braid of a loop can be understood directly on its efficient coding $(A_1, \dots, A_K)$. Basically we show that the element Σ of $\{\overline{\Sigma}_i^{\pm}\}$ we use to deloop is $\overline{\Sigma}_{A_1}^{\epsilon}$ with $\epsilon = \text{sign}(A_2 - A_1)$, and that it acts, up to permutation of the names of the points, as a shift on the efficient coding ; it erases the first letter and permutes the names of the terminal points :

$$\Sigma(A_1, \dots, A_K) = (s_\Sigma(A_2), \dots, s_\Sigma(A_K)).$$

A.1. *Mapping class group*

A geometric interpretation of the braid group B_n is to consider it as the group of homeomorphisms of the punched disk with n holes up to homotopy.

Let $\mathcal{D}$ be a disk. Fix n points $P = \{a_1, \dots, a_n\}$ in its interior (for instance think of them as regularly spaced on a diameter). Let $\mathcal{D}_n = \mathcal{D} \setminus P$ be the punched disk, $H^+(\mathcal{D}_n)$ be the group of orientation preserving homeomorphisms of $\mathcal{D}_n$ fixing the boundary $\partial\mathcal{D}_n$, $H_0^+(\mathcal{D}_n)$ the subgroup of homeomorphisms of $\mathcal{D}_n$ homotopic to the identity, and set $\Gamma_n = H^+(\mathcal{D}_n)/H_0^+(\mathcal{D}_n)$. It is well known that Γ_n and B_n are isomorphic (see [4]).

This point of view yields a nice interpretation of the action of B_n on F_n which indeed corresponds to the action of Γ_n on the fundamental group of $\mathcal{D}_n$.

A.2. *Loops and curves*

The fundamental group of $\mathcal{D}_n$ is the set of oriented loops up to homotopy. This group is the free group F_n with n generators.

For a representation, we fix a base point b_0 on the boundary $\partial\mathcal{D}$ of $\mathcal{D}$. A curve in $\mathcal{D}_n$ is a continuous map $\gamma : [0,1] \to \mathcal{D}$ with $\gamma(0) = b_0$ and $\gamma(]0,1[) \subset \mathcal{D}_n$. A curve is a loop if $\gamma(0) = \gamma(1) = b_0$. We will say it is a *separatrix* if $\gamma(0) = b_0$ and $\gamma(1) \in P$. We say that two curves γ_1 and γ_2 are homotopic if there is an homeomorphism f of $\mathcal{D}_n$ fixing the boundary such that $\gamma_1 = f \circ \gamma_2$. To draw pictures, we may take different base points on the boundary but this should not be misleading. We denote $[\gamma]$ the homotopy class of the curve γ. We do not want to enter more into technical details. We will use standard tools in this context.

A curve or a loop is simple if it has no self intersection, i.e. it is injective. A simple loop is the boundary of a topological disk immersed in $\mathcal{D}$. It has an interior and an exterior. Given two curves, or two loops, we may define their intersection. If this set is finite, then we can define their number of intersections. Given two curves γ_1 and γ_2, we define their intersection number$i(\gamma_1, \gamma_2)$ as the minimal number of intersections of two curves $\gamma_1' \in [\gamma_1]$ and $\gamma_2' \in [\gamma_2]$.

A.3. *Coding*

We will call $d_1, \ldots, d_n$ a family of (non intersecting) separatrices, say vertical segments, from b_i in the boundary $\partial\mathcal{D}_n$ to a_i (Figure 6). We assume they are oriented. We call $\mathcal{C}$ their union. We notice that $\mathcal{D}_n \setminus \mathcal{C}$ is a topological disk, i.e. it is simply connected.

As a set of generators for the fundamental group of $\mathcal{D}_n$, we can choose a family of non intersecting simple loops around each punched points (Figure 6). To be more specific, we will denote u_i the simple loop containing a_i having no intersection with $\cup_{j\neq i} d_j$. It is uniquely defined up to homotopy.

Using this basis, we are going to code a loop by the sequence of algebraic intersections of the loop with curves in $\mathcal{C}$. If this number of intersections is finite, the homotopy class of the loop will determine an element of F_n. It will always be possible to work with smooth loops whose coding correspond to the reduced writing of the element of the free group they represent. At least it is possible to guarantee that in the homotopy class, there is a representative with countably many intersections with $\mathcal{C}$. We say that a representative is *nice* if its number of intersections with $\mathcal{C}$ is minimal, in which case its coding is a reduced word.

More formally, let $(t_k)_{k\geq 1}$ be the sequence of times of intersections $\gamma(t_k) \in \mathcal{C}$. Call i_k the index i such that $\gamma(t_k) \in d_i$. The intersection is algebraic : call ϵ_k the sequence of orientations. We can set

$$\varphi : \gamma \mapsto (u_{i_k}^{\epsilon_k})_{k\geq 1}$$

The image can be interpreted as an element of the free group F_n ($\varphi(\gamma) = [\gamma]$) or as a word (in reduced form if γ is nice). If γ is a loop, this is a way to see the isomorphism between the fundamental group and the free group. If γ is a separatrix, then we need information on the terminal point : the pair $(\varphi(\gamma), \gamma(1))$ characterizes the homotopy class of a separatrix.

We notice that the action of Γ_n on the fundamental group corresponds to the action of the braid group on the free group. The point is that the image of a loop γ by an homeomorphism f is the loop $f \circ \gamma$. It is straightforward to see that the homotopy class of $f \circ \gamma$ is not changed if f is replaced by f' with $f' \circ f^{-1}$ homotopic to identity. Nor if we change γ with any $\gamma' = g \circ \gamma$ homotopic to γ. Hence Γ_n acts on homotopy classes. It is straightforward to check that this action corresponds to the action defined in Section 3. Figure 8 shows the relationships between the algebraic point of view and this picture.

A.4. *Simple loops*

There is a one-to-one correspondence between $\mathcal{O}_n$ and homotopy classes of simple separatrices. This is exactly the conjugacy : $h.a = hah^{-1}$. This correspondence justifies the use of the same letter to denote the terminal point a and the loop around a itself.

Lemma 11. *Let $w \in \mathcal{O}_n$. There is $h \in F_n$ and $a \in S^+$ such that $w = hah^{-1}$ and (h, a) codes a simple separatrix. Conversely, for every γ simple separatrix, there is $w \in \mathcal{O}_n$ with $w = hah^{-1}$, $\varphi(\gamma) = h$ and $\gamma(1) = a$.*

This point is illustrated by Figure 15.

Proof. There is a braid τ such that $\tau(u) = w$. Let $\gamma \in [u]$ and $f \in [\tau]$. Then $f(\gamma) \in [w]$ and it is a simple loop. Moreover, the interior of γ contains only one punched point, say $a_i \in P$. So does $f(\gamma)$. Consider a simple curve C in the interior (which is a disk minus one point) delimited by $f(\gamma)$, with $C(0) = b_0$ and $C(1) = f(a_i)$. C is a separatrix. Its homotopy class is determined by w.

To prove the converse statement, the main point is to notice that the image by a homeomorphism f of $\mathcal{D}_n$ of a simple separatrix γ is a simple separatrix $\gamma' = f \circ \gamma$. Indeed, $\partial\mathcal{D}_n$ is fixed and the endpoint, say a_k is mapped onto some $a_{k'} = f(a_k) \in P$, while $f \circ \gamma$ is injective (no multiple point can appear).

Now, let γ be a simple separatrix, $\gamma(0) = b_0, \gamma(1) = a_i$. There is a homeomorphism f of the disk $\mathcal{D}$ (not punctured) fixing the boundary and mapping an elementary separatrix ($\gamma'(0) = b_0, \gamma'(1) = a_1$) on γ. We compose it with another homeomorphism g fixing $\gamma \cup \partial\mathcal{D}$ and mapping the $f(a_k)$, $k \neq i$ onto the a_k, $k \neq i$. We can consider $g \circ f$ as a homeomorphism of the punctured disk $\mathcal{D}_n$. The class of $g \circ f$ is a braid τ satisfying $\tau(u_1) = hu_kh^{-1}$ with $(h, u_k) = [\gamma]$. □

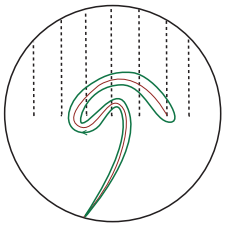

Fig. 15. Loop associated to a separatrix.

We summarize a few obvious properties of simple loops, in particular in relation with Lemma 3 and Decomposition (3.2).

Lemma 12. *Let $w \in \mathcal{O}_n$ and write $w = u_{i_1}^{\epsilon_1} u_{i_2}^{\epsilon_2} \cdots$ in reduced form. If $\epsilon_1 = 1$, then $i_2 > i_1$. Consider decomposition (3.2). There is always a $u_{i_1}^{\pm}$ between two blocks of different types ($\eta_p \neq 0$). Blocks are not empty (no occurrence of $u_{i_1} u_{i_1}$). If V_{p-1} is of type L (resp. R) and V_p of type R (resp. L) , then $\eta_p = 1$ (resp. $\eta_p = -1$). If $\epsilon_1 = 1$ (resp $\epsilon_1 = -1$) then there are no successive blocks of type L (resp. type R).*

Proof. The basic remark is the following. Consider a nice representative γ of w. Notice that $\gamma(t_1) \in d_{i_1}$. Hence $[\gamma(t_1), b_{i_1}] \subset d_{i_1}$. Consider the curve :

$$C = \gamma([0, t_1]) \cup [\gamma(t_1), b_{i_1}].$$

It cuts the disk $\mathcal{D}_n$ into two pieces (see Figure 16), one of them containing a_{i_1}. The $\cup_{i<i_1} d_i$ is in the left component, while $\cup_{i>i_1} d_i$ is in the right one. Depending on the sign of ϵ_1, the connected component containing a_{i_1} is to the left or to the right. Everything follows from basic geometrical considerations. □

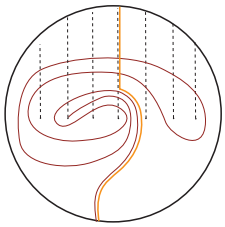

Fig. 16. Visualisation of the decomposition of Lemma 12. The disk is cut into two pieces. Here, $i_1 = 4$ and $\epsilon_1 = -1$ so that a_4 is in the left connected component. The code of the separatrix is : $(\mathbf{u_4}^{-1}u_3^{-1}u_2^{-1}u_3\mathbf{u_4}u_1u_2u_3\mathbf{u_4}u_7^{-1}u_6^{-1}u_5^{-1}\mathbf{u_4}^{-1}u_3^{-1}u_2^{-1}u_1^{-1}\mathbf{u_4}^{-1}u_3^{-1}, a_2)$

A.5. *Efficient coding*

We consider $w \in \mathcal{O}_n$. Let h and a be such that $w = hah^{-1}$ and consider the separatrix (h, a). Such curves can be coded keeping track only of "minimal" information. The idea is that we are going to pick only the closest intersections with the segments of $\mathcal{C}$ and show that this information is enough to recover the separatrix.

We consider a nice representative of a simple separatrix (h, a), denoted γ. To this separatrix we are going to associate a word $A = (A_i)_{1 \leq i \leq K}$ on the alphabet S_+. We call $\mathcal{C}_0 = \cup_{i=1}^n d_i$. Set $A_1 = u_{i_1}$ and $\mathcal{C}_1 = \mathcal{C}_0 \setminus [\gamma(t_1), b_{i_1}]$. It means that we keep only the segment $[a_{i_1}, \gamma(t_1)]$, so that the next time we intersect $\mathcal{C}_1$ we will be "closer" to the point u_{i_1}. Assume A_i and $\mathcal{C}_i$ defined for $1 \leq i \leq m$ for some m. Consider the first time t_{m+1} of intersection of $\gamma([t_m, 1])$ with $\mathcal{C}_m$. Let k be the unique index such that $\gamma(t_{m+1}) \in d_k$ and set $A_{m+1} = u_k$. Put $\mathcal{C}_{m+1} = \mathcal{C}_m \setminus [\gamma(t_{m+1}), b_k]$. Recursively, we define a word $A(\gamma)$. It is a finite word if the separatrix is finite. We end it with the name of the terminal point.

The word A is in fact a function of (h, a) and hence of $w = hah^{-1}$. Indeed nice representatives of (h, a) intersect the segments d_j in the same order. We denote it $\chi(w)$. Figure 19 illustrates this construction.

Denote $\mathcal{X}_n$ the set of (non empty) words on $S_+ = \{u_1, \dots, u_n\}$ with no repetitions, that is,

$$\mathcal{X}_n = \{x \in S_+^* \ : \ x_i \neq x_{i+1}, \forall 1 \leq i < |x|\}.$$

We show that the map χ is a bijection from $\mathcal{O}_n$ to $\mathcal{X}_n$.

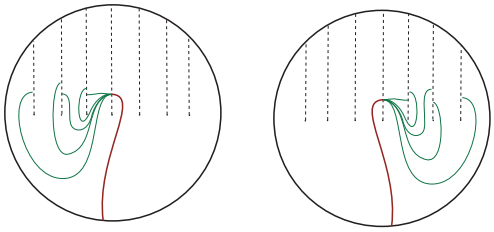

Fig. 17. How to continue a simple separatrix.

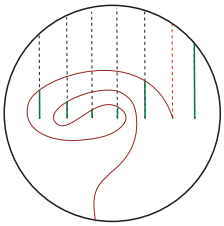

Fig. 18. Proof of Lemma 13.

Lemma 13. *For all $w \in \mathcal{O}_n$, $\chi(w) \in \mathcal{X}_n$. Conversely, let $A \in \mathcal{X}_n$. There is a unique $w \in \mathcal{O}_n$ such that $A = \chi(w)$.*

Proof. The first statement follows from the fact that repetition of a letter would correspond to a path from d_i to itself which is not possible if we choose a nice representative γ of w.

To prove the second statement, we proceed by induction on the length of the word $A \in \mathcal{X}_n$. If it is of length 1 then w is an elementary loop. Assume that for every word $U = U_1 \cdots U_K \in \mathcal{X}_n$ of length K, there is a unique $w \in \mathcal{O}_n$ such that

$U = \chi(w)$. Write $w = hU_K h^{-1}$. Choose a nice representative separatrix C of the homotopy class of (h, U_K). Call $U_K = u_i$ the terminal point. For all j, denote $\widehat{d}_j$ the connected component of $d_j \setminus C$ containing a_j (i.e. the segment between a_j and the closest intersection of C with d_j, or d_j itself if $C \cap d_j = \emptyset$). The interior of the surface $\widehat{\mathcal{D}}_n = \mathcal{D}_n \setminus (C \cup \bigcup_{j \neq i} \widehat{d}_j)$ is homeomorphic to a disk (see Figure 18).

Now choose $1 \leq k \leq n$, $k \neq i$ and set $U_{K+1} = u_k$. Since a_i and $\widehat{d}_k$ are on the boundary of the disk $\widehat{\mathcal{D}}_n$, all curves in this disk with $\gamma(0) = a_i$ and $\gamma(1) = a_k$ are homotopic. Choose a nice representative C_1 and denote g a reduced coding of this curve (as seen in the disk $\mathcal{D}_n$).

The segment $\widehat{d}_i$ cuts the disk $\widehat{\mathcal{D}}_n$ into two connected components. The point a_k may be in one or the other component. We choose a small enough neighborhood of a_i. It should now be clear that we can modify locally the curve $C \cup C_1$ in this neighborhood so that, for $\epsilon = \pm$ chosen according to the component, this new curve is a separatrix with coding $(hu_i^\epsilon g, a_k)$. Denote $\widehat{w} = hu_i^\epsilon g u_k (hu_i^\epsilon g)^{-1}$.

It follows from the construction that $\chi(\widehat{w}) = U_1 \cdots U_{K+1}$. It is unique because g is uniquely determined. □

Remark 9. We notice that the word $\chi(w)$ is a subword of w up to exponents. We could keep exponents but it is not useful since they are determined.

A.6. *Algebraic coding*

We give an alternative — more algebraic — presentation of this coding. Let $w \in \mathcal{O}_n$, $w = u_{i_1}^{\epsilon_1} \cdots u_{i_N}^{\epsilon_N}$. Let $k \leq m/2$. The prefix $v = u_{i_1}^{\epsilon_1} \cdots u_{i_k}^{\epsilon_k}$ of w is *admissible* if

$$v\, u_{i_{k+1}}\, v^{-1} \in \mathcal{O}_n.$$

Notice that $\emptyset$ is always admissible. Let $K = K(w)$ be the number of admissible prefixes of w and let $k_1, \ldots, k_K$ be the sequence of their lengths, in increasing order. We call $\kappa(w)$ the sequence of terminal points of the admissible prefixes. For all $1 \leq m < K$, let $U_m = u_{i_{k_m+1}}$, and set

$$\kappa(w) := (U_m)_{1 \leq m \leq K}.$$

See Figure 19 and Figure 20 for an illustration.

Lemma 14. $\chi(w) = \kappa(w)$.

Proof. Let $w \in \mathcal{O}_n$, $A = \chi(w)$, and γ a nice representative separatrix. For every m, we consider the curve $\gamma([0, t_m])$ and we notice that $\gamma([0, t_m]) \cup \widehat{d}_{A_m}$ is a simple curve since $\widehat{d}_{A_m}$ intersects $\gamma([0, t_m])$ only in $\gamma(t_m)$. Hence it is possible to construct a simple loop γ' (boundary of a neighborhood of $\gamma([0, t_m]) \cup \widehat{d}_{A_m}$, which is a disk) , containing $\gamma([0, t_m]) \cup \widehat{d}_{A_m}$ in its interior (see Figure 21, right) as close to $\gamma([0, t_m]) \cup \widehat{d}_{A_m}$ as needed. This simple loop contains only a_{A_m} in its interior. So the prefix $w_1 \cdots w_{m-1}$ corresponding to $\varphi([\gamma([0, t_m])] \cup \widehat{d}_{A_m})$ is admissible.

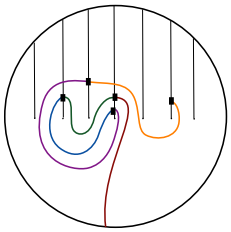

Fig. 19. Coding of a separatrix

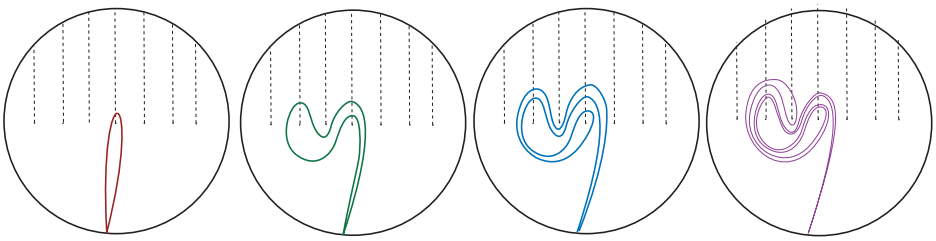

Fig. 20. Admissible prefixes

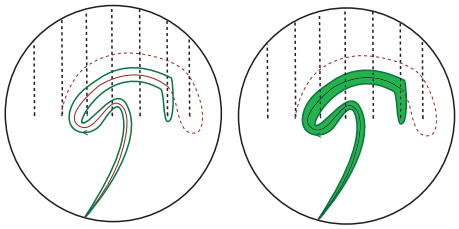

Fig. 21. Admissible prefix and closest intersection

Conversely, if a prefix $v = v_1 \cdots v_k$ is admissible, any nice representative γ of the corresponding loop $vw_{k+1}v^{-1}$ contains the "end point" a_i ($i = i_{k+1}(w)$) in its interior and also a piece of the segment d_i (see Figure 21, left). Let t be the time of the $k+1$st intersection of γ with $\mathcal{C}$. Necessarily, $\gamma(t) \in d_i$. Consider the curve $\gamma([0,t])$. Since the loop is simple, this curve does not intersect the segment $[a_i, \gamma(t)]$ which is in the interior of the loop. So the last intersection is the closest. The curve

$\gamma([0,t])$ is the beginning of a nice representative separatrix of w for which $\gamma(t)$ is a closest intersection with d_i. □

As a consequence we deduce that the map

$$\begin{aligned}\kappa : \mathcal{O}_n &\to \mathcal{X}_n \\ w &\mapsto (U_i)_{1\leq i\leq K}\end{aligned}$$

is bijective.

A.7. *Delooping and efficient coding*

We investigate the relationship between efficient coding and the so-called delooping braid of a loop. It will appear that they contain essentially the same information. For short, we will denote $A_1\cdots A_K$ the loop $\chi^{-1}(A_1\cdots A_K)$, using capital letters to avoid confusion.

Lemma 15. *Let $A\in\mathcal{X}_n$, $|A|>1$, $A=A_1\cdots A_K$. Set $\epsilon = sgn(A_2-A_1)$, $\Sigma=\overline{\Sigma}^{\epsilon}_{A_1}$ and $s=s_\Sigma$.*

$$\Sigma(A_1\cdots A_K) = s(A_2)\cdots s(A_K).$$

Proof. The result follows directly from the computation of the image of a loop in Section 3.4. Indeed, we saw that Σ acts as s on the loop, unless on the u_{i_1} separating blocks of type L and R (as well as on the first one) which may disappear. The first one disappears. The other ones (may disappear) but are not involved in the efficient coding. Indeed, those that disappear are those marking a change of type of block corresponding to an intersection with d_{i_1} between b_{i_1} and the first intersection, i.e., not corresponding to a closest intersection (see Remark 8, Figure 16 and proof of Lemma 12). □

Let $A\in\mathcal{X}_n$, $A=A_1\cdots A_K$. Set $\epsilon_1 = sgn(A_2-A_1)$, $\Sigma_1=\overline{\Sigma}^{\epsilon_1}_{A_1}$ and $s_1=s_{\Sigma_1}$. Recursively

$$\begin{cases}\epsilon_{k+1} &= sgn(s_k(A_{k+1})-s_k(A_k)),\\ \Sigma_{k+1} &= \overline{\Sigma}^{\epsilon_{k+1}}_{s_k(A_{k+1})},\\ s_{k+1} &= s_{\Sigma_{k+1}}\circ s_k.\end{cases}$$

We put $s_A=s_K$ and

$$\Sigma_A = \Sigma_{K-1}\cdots\Sigma_1.$$

We claim that Σ_A is indeed the delooping braid of the loop $A_1\cdots A_K$. More precisely

Lemma 16.

$$\Sigma_A(A_1\cdots A_K) = s_A(A_K).$$

Proof. This is an obvious induction based on Lemma 15. □

Acknowledgments

This work started with a question raised by Jean-Marc Gambaudo about random walks on braid groups. The normal form appeared after a discussion with Hamish Short about the action of braid groups on free groups. It seems to be closely related to an unpublished work by Anatoly Vershik and Andrei Malutin ; the discussions we had on the subject have been important for me. I wish to thank all of them. I also wish to thank Patrick Dehornoy for his interest in this work. I am indebted to the referee whose suggestions allowed me to improve the overall presentation of the paper.

References

[1] Artin, E. Theory of braids. *Ann. of Math.* (2) 48, (1947). 101–126.
[2] Birman, J. S. *Braids, links and mapping class groups.* Ann. Math. Studies 82, Princeton Univ. Press, 1974.
[3] Epstein, David B. A.; Cannon, James W.; Holt, Derek F.; Levy, Silvio V. F.; Paterson, Michael S.; Thurston, William P. *Word processing in groups.* Jones and Bartlett Publishers, Boston, MA, 1992.
[4] Dehornoy, Patrick; Dynnikov, Ivan; Rolfsen, Dale; Wiest, Bert *Why are braids orderable?* Panoramas et Synthèses, 14. Société Mathématique de France, Paris, 2002.
[5] Kaimanovich, Vadim A.; Masur, Howard The Poisson boundary of the mapping class group. *Invent. Math.* 125 (1996), no. 2, 221–264.
[6] Larue, David M. *Left-distributive and left-distributive idempotent algebras.* Ph.D. Thesis, University of Colorado, Boulder, CO, 1994.
[7] Mairesse, Jean ; Mathéus, Frederic Randomly Growing Braid on Three Strands and the Manta Ray. *Research Report LIAFA*, 2005-001, Univ. Paris 7, 2005. To appear in *Annals of Applied Probability.*